PROCEEDINGS

The HKBU Third Computer Science Postgraduate Research Symposium

Jan 9, 2006

PG Day 2006

Department of Computer Science
Hong Kong Baptist University

The Third HKBU-CSD Postgraduate Research Symposium (PG Day) Program

January 9 Monday, 2006		
Time	**Sessions**	
09:15	**On-site registration**	
09:30-09:40	**Welcome**: Prof. YiuWing Leung, Acting Head of Computer Science Department (LMC 514)	
09:40-10:40	**Keynote Talk**: Prof. Weidong Kou ("IBM Service Oriented Architecture ") (LMC 514)	
10:40~10:50	**Tea Break**	
10:50-12:20	Session A: (Chair: ChuiYing Hui) *Pattern Recognition*	
	• *Enhancing security for face recognition systems* YiCheng Feng • *Writer identification using Hidden Markov Tree Model* ZhenYu He • *On The Solution to Singular Integral Equations with Logarithmic Kernel Based on Wavelet* LiMin Cui	
12:20-13:30	**Noon Break**	
13:30-15:00	Session B1: (Chair: ZhiLi Wu) (LMC 514) *Intelligent Informatics*	Session C1: (Chair: WingYan Chow) (LMC 509) *Networking*
	■ *Data Hiding on Polygonal Meshes for Authentication and Content Annotation* HaoTian Wu ■ *Compression Driven Partitioned POMDP Solving Via Belief Space* Xin Li ■ *Privacy Measure and Active Learning* XiaoFeng Zhang	■ *A New Approach to Mobile Location Estimation within a Radio Cellular Network* JunYang Zhou ■ *Processing Precision-Constrained Approximate Queries in Wireless Sensor Networks* MinJi Wu ■ *Algorithm to support Temporal Consistency of Sensor Data for Transaction Processing Applications in Broadcast Environments* ChuiYing Hui
15:00-15:10	**Tea Break**	
15:10-17:10	Session B2: (Chair: Xin Li) (LMC 514) *Intelligent Informatics*	Session C2: (Chair: MinJi Wu) (LMC 509) *Networking*
	■ *Regularized Selective Ensembles of Base Learners* ZhiLi Wu ■ *Dissimilarity Learning for Nominal Data* ChiWa Cheng ■ *Intelligent Agents in Resources Allocation* KaFung Ng	■ *Lightweight Piggybacking for Packet Loss Recovery in Internet Telephony* WingYan Chow ■ *The Client-Based Framework For Privacy-Preserving Location-Based Data Access* Jing Du ■ *An Adaptive Controller to Guarantee Proportional Delay Differentiation on Clustered Web Servers* KaHo Chan ■ *Studies on Location Estimation in Environments* ManChung Yeung
17:10–17:30	**Refreshment & End**	

TABLE OF CONTENTS

Session A: Pattern Recognition

Enhancing Security for Face Recognition Systems…...……………………………..………………………..……….1
Yicheng Feng

Writer Identification Using Hidden Markov Tree Model……………………………………………………………………7
Zhenyu He, Yuan Yan Tang

On the Solution to Singular Integral Equations with Logarithmic Kernel Based on Wavelet …………………………...17
L. M. Cui, Y. Y. Tang

Session B1: Intelligent Informatics

Data Hiding on Polygonal Meshes for Authentication and Content Annotation ……………………………………….26
Hao-tian Wu

Compression Driven Partitioned POMDP Solving Via Belief Space ………………………………………………….35
Xin Li

Privay Measure and Active Learning……………………………………………………………………………...41
Xiaofeng Zhang

Session C1: Networking

A New Approach to Mobile Location Estimation within a Radio Cellular Network……………………………………..48
Junyang Zhou

Processing Precision-Constrained Approximate Queries in Wireless Sensor Networks………………………………….59
Minji Wu, Jianliang Xu, Xueyan Tang

Algorithm to Support Temporal Consistency of Sensor Data for Transaction Processing Applications in Broadcast Environments……………………………………………………………………………………………………...67
Chui Ying Hui, Joseph Kee Yin Ng

Session B2: Intelligent Informatics

Regularized Selective Ensembles of Base Learners…………………………………………………………………...74
Zhili Wu, Chunhung Li, Jiming Liu

Dissimilarity Learning for Nominal Data……………………………………………………………………………...81
Victor Cheng, C. H. Li

Intelligent Agents in Resources Allocation……………………………………………………………………………86
Ng Ka Fung

Service-Oriented Implementation of Distributed Data Mining………………………………………………………….90
Ho-Fai Wong, Xiaofeng Zhang, William K. Cheung

Session C2: Networking

Lightweight Piggybacking for Packet Loss Recovery in Internet Telephony……………………………………………97
Wing Yan Chow, Yiu Wing Leung

The Client-based Framework For Privacy-Preserving Location-Based Data Access …………………………………..101
DU Jing

An Adaptive Controller to Guarantee Proportional Delay Differentiation on Clustered Web Servers………………….106
Chan Ka Ho

Studies on Location Estimation in Library Environments……………………………………………………………….113
Wilson M. Yeung

Enhancing security for face recognition systems

Yicheng Feng

Abstract

Biometric is a convenience and more secure method for identity authentication comparing with traditional methods. However, there are kinds of threats which aim at biometric authentication systems and the security of the biometric authentication system should be enhanced. In the latest 20 years the security of biometric system has been largely researched and kinds of methods are developed. However, researchers focus in face recognition have not paid enough attention to the security of face biometric data. In our research we want to find a scheme to protect the face recognition systems. We mainly concentrate on cryptography and the Reed-Solomon codes are used. A smartcard-based design is proposed which encrypts the face biometric data with Reed-Solomon codes and stores it in the card, in order to protect the transmission channel and database.

1 Introduction

There are many threats to biometric systems, such as Denial of service, circumvention, repudiation, contamination, convert acquisition, collusion, coercion. In Denial of Service (DoS), an attacker corrupts the authentication system so that legitimate users cannot use it. In circumvention, an attacker gains access to the system protected by the authentication application. In repudiation, the attacker denies accessing the system. For example, a corrupt bank clerk who modifies some financial records illegally may claim that her biometric data was ”stolen”, or she can argue that the False Accept Rate (FAR) phenomenon associated with any biometric may have been the cause of the problem. In contamination (covert acquisition), an attacker can surreptitiously obtain biometric data of legitimate users, and use it to access the system. In collusion, a legitimate user with wide access privileges is the attacker who illegally modifies the system. In coercion, attackers force the legitimate users to access the system.

Denial-of-service attacks can be particularly horrible in any system. The sensor is often fragile and sensitive, which is easily broken. Developments in contact-less and ultrasonic sensors promise design of more robust fingerprint sensors. On the other hand, forgetting a password or misplacing a token is a known problem in denying access to otherwise legitimate users.

Attacks against biometric systems could be divided into 8 types [1], referring to system structure: 1. Attack aim at sensor: A fake biometric pattern may be sent to the sensor to get promise to login. 2. Attack aim at the route between sensor and feature extractor: A previously stored/intercepted biometric data may be resubmitted. 3. Attack aim at feature extractor: Override the feature extraction process or compromise the tractor to produce feature model selected by attacker. 4. Attack aim at channel between feature extractor and matcher: Replace the geniue feature model with selected one. 5. The fifth type of attack: Corrupting the matcher, modify the matcher to output wanted score. 6. The sixth type of attack: Modify the template database, add new template or modify a recent template. 7. The seventh type of attack: Attack the channel between the stored templates and the matcher, intercept and modify it. 8. The last type: Override the final decision.

While designing a biometric system, the security of it should be considered. This could be considered in different aspects: biometric form, system structure, communication protocols and operation algorithm. For example, the challenge/response structure is designed considering the system structure, and cryptography is a common method for communication protocols. On the other hand, we could design schemes which aim at separate attacks against vulnerabilities of system. In our design, we

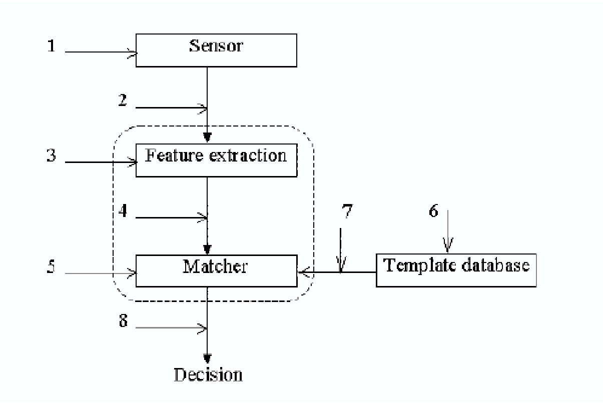


Figure 1: Figure for biometric attacks.

mainly enhance the security of transmission channel and database. A cryptography based scheme is proposed and applied to a face recognition system, which protects the data while authentication.

2 Related researches

G.I. Davida, et al. [3] has presented a scheme. It is based on an error-correcting codes. The error-correcting codes are described as follows: Assuming that A is a K bit signal which needs to be encoded, and B is the N bit codes which is encoded from A. Considering A and B as a K-dimensional vector,We have B(A-Q), in which Q is a KN matrix. The matrix could be presented as [I : C], in which I is a KK identity matrix, and C is a K(N-K) sub-matrix. When B is encoded from A, from the format of Q, we could see that B could be presented as [A : D], in which D=AC. D could be considered as the check digest of A. When in the decoding process, a bounded distance decoding is used, utilizing the check digest D. We call this error-correcting code (K, N, P) code, in which P is the minimum distance of the code. It could keep an error rate lower to (P-1)/2.

Assume that there is a strong link between the users and their biometric templates, and the biometric templates could not be fabricated. The private biometric scheme starts with a private key and public key generation. The generated public key is sent to the sender. When in user initialization, M K-bit biometric templates from the same user are extracted by the sender, and then transformed into one K-bit vector S through a majority decoder. The K-bit vector S is encoded with (K, N, Q) code scheme to a N-bit vector R, which could be presented as [S, C]. C is the check digest of S. The storage device is constructed with some information, as the user's name, which is denoted as NAME; the user's privileges such as access to the system, and other attributes, which is denoted as ATTR; the user's check digest C of his biometric template; and the authorization officer's signature of the user's message, which is denoted as SIGN(Hash(NAME, ATTR, T——C)). The hash function is used for data hiding which presents from data leakage.

When in authentication, the user present his card and his biometric templates(scan for M times to get M templates). These M templates are transformed into a K-bit vector T', and then corrected to T" with the check digest C, using the bounded distance decoding. Then the vector T" is used to verify the signature SIGN(Hash(NAME, ATTR, T——C)). If the user's template T" has less than P/2 errors, then he would be successfully recognized.

While we use this method, there are two conditions which must be satisfied. First, biometric template of user couldn't be reproduced by an attacker, and the storage device couldn't be generated by the attacker to pass a biometric verification. Second, a digital representation of biometric couldn't be reproduced by attackers to pass the authorization process. Also, in the paper it describes a method which uses biometrics as an enabler.

The base of fuzzy vault scheme is the fuzzy commitment scheme, which is proposed by Ari Juels, et al. [2]. The fuzzy commitment scheme aims at cryptographic algorithms which could tolerate some information variation or noises. In traditional commitment schemes, the sender who wants to send secret k to the receiver makes a concealing presentation y of k, and send y to the receiver. Later, he should open the commitment y to prove that y indeed represents b. Usually, y=f(k,x), in which x is a witness. These methods need precise input to achieve accurate cryptography. Small variation in original signal may lead to large difference in the encrypted one. And this disadvantage leads to an issue named fuzzy commitment scheme.

Error-correcting codes are used in the scheme, which are used for transmission in noisy channels. Assuming that the original message is a k-bit string s, it is extended to an n-bit string m before transmission, which makes the message has some redundant bits. So, when transmitted in the noisy channel, it would be possible that the useful bits are kept the same and only redundant ones are corrupted, or only few useful ones are corrupted. In a typical form, suppose the codewords of all original messages constitute a set C. In the decoding process, we should make a mapping function f from (0,1)n to CNULL. It means that all elements in (0,1)n should be mapped to the nearest element in C, otherwise mapped to NULL. A threshold t is used, if the distance (Such as hamming distance) from element a in (0,1)n to any element in C is larger than t, a would mapped to NULL.

When using error-correcting codes in the scheme, suppose the original data which needs to be encrypted is x, and the corresponding error-correcting code of x is c. Then encrypt x as follows: F(x) = (Hash(c), x-c), in which the hashing is an l bit sequence. In enrollment process, F(x) is stored in the database when enrolling. In verifying process, a data x' is presented. Decrypt c' using the stored data x-c as follows: c'=f(x'-(x-c))=f(c+(x'-x)). Compare Hash(c) with Hash(c'). If x'-x¡l, then c=c' and the verification succeeds.

The fuzzy commitment scheme has some disadvantages. It requires standard format input and lacks order variation, which limits its' usability. The fuzzy vault scheme which is proposed by Ari Juels, et al. [4] overcomes these advantages. Fuzzy vault is a novel cryptographic construction, which hides the secret k in a set A. The set A containing digital elements is constructed by the owner of the secret, and "locks" the secret with these digital elements using cryptography. If someone wants to "unlock" the secret, he would offer another set B which also contains digital elements. It is compared with A, if most elements in B are also belongs to A, the applicant could unlock the secret.

The fuzzy vault scheme could be considered as an error-correcting cryptography. The secret keeper offers a set A which contains some authentication digital elements, and encrypts the secret with A as the encryption key. In the encryption, the secret k is embedded into a polynomial in some way such as relating to the coefficients of the polynomial. The polynomial has a single variable x and is concerned about k. When the secret k is encrypted in A, the elements in A could be treated as x-coordinate values and separately computed the corresponding values of polynomial. This could be considered as a mapping from elements in A to polynomial values of these elements. After this process, some chaff values which are not concerned about the polynomial are generated, and mixed with the computed values. The all values, containing chaff values and values computed from elements in A through the polynomial, are collected to a set C.

When someone wants to decrypt the secret k, he offers a digital set B. Each element in B is computed through the polynomial and then a set of polynomial values is generated from B, which is denoted as D. If B is close to A, then the computed set D would be close to C, but may contain a few error values. Then the applicant could reveal the polynomial from D using error-correcting methods, and get the secret. If B is not close to A, he would fail to reveal the polynomial. Because of the chaff values, data leakage is limited and attackers could not know the real message of all values.

The nuclear problem of this scheme is that how to recover the polynomial using the set D which may contain some error values irrelative to the polynomial.

A secure smartcard-based vault scheme is proposed by T. C. Clancy et al. [5] This scheme utilizes smartcard methods and the fuzzy vault scheme. It could be seen as an incorporation of these two kinds of methods.

There are two kinds of attacks which mostly occur on smartcard systems: Invasive attacks and noninvasive attacks. Invasive attacks means that the attacker could somehow fetch the smartcard, decompounds it and analyze its' stored content. The smartcard could prevent from these attacks only by encryption on the stored message. The noninvasive attacks are less stronger, and could be counteracted by clever algorithms or other decisions. When design the system, the private key of user should be encrypted in the smartcard, which is the primary thinking of this scheme.

The scheme uses a fuzzy vault. The secret l is hided in the coefficients of a degree k polynomial f, and Q denotes the finite field of the polynomial. Then f become the secret message which needs to be encrypted. The locking set C is a set of t values ai which belongs to Q. All the values (ai , f(ai)) and some chaff pairs (bj , cj) in which f(bj)cj constitute the locked vault U. The number of elements in U is r. Then k¡t¡r. To decode the fuzzy vault, the scheme uses a Reed-Solomon decoder [6], which is named Berlekamp-Massey algorithm.

In order to get the locking set C, the fingerprint should be scanned for N times. Then N features of the fingerprint are gotten. Suppose that aij is the j-th minutia of the i-th feature Ai, and A is the average set with multiplicity. For each aij in each Ai, try to find a minutia aef in A which is nearest to aij and — aef - aij —¡T. (T is the threshold.) If a desired minutia exists, then aij is added to average and the multiplicity increases 1. If there is no suitable minutia existing, then aij is put in A with multiplicity 1. After all minutiae in every features are processed, elements in A would be chosen if their multiplicities are larger than S. These minutiae constitute the locking set U.

Chaff points are generated, and put into the locking set. Because the user should be able to distinguish the chaff points and real ones, the chaff points could not be placed too close to the real ones. A threshold d is chosen and all chaff points should be put with at least a distance d with all the real ones. Also, because if the chaff points are too close to each other the attacker may be easy to distinguish them from real ones, chaff points should keep at least a distance d with each other.

While the user want to decode the vault, an unlocking set V is presented, which is the set of minutiae extracted by the sensor. If the system has no noise, V would be included in C. However, as the system channels are noisy and may cause some corruption, minutiae in V may be different from real ones in C or even closer to the chaff points. The Reed-Solomon codes are used for this condition.

3 Design for face biometric system

Our proposal is based on the schemes presented above, mainly on the secure smartcard-based scheme. In the smartcard-based scheme, the smartcard should contain messages as follows: 1. The user's name, which is denoted as NAME. 2. The user's privilege and other attributes, denoted as ATTR. 3. The locked vault U, containing both useful points and chaff ones. 4. The Signature of the authorization officer, denoted as SIGN(Hash(NAME, ATTR, l)), in which l is the secret.

In the authentication process, the applicant presents a face template and the sensor extracts its' feature. The feature is represented as an unlocking set, and combined to U to compute the polynomial f'. While f' is computed, the secret l' would be extracted from coefficients in f'. Then l' is verified with SIGN(Hash(NAME, ATTR, l)).

To extract the polynomial f, the unlocking set V is compared with locked vault U and an unlocked vault W is extracted. If there is no noise, W should contain all the valid t points, maybe contain some extra ones. At this condition the brute-force research could be implemented to recover the polynomial. However, the system is noisy, and there may be some lacks of points or variation, or even noisy points which is close to the chaff ones. This makes we use the Reed-Solomon decoder.

Comparing with the two kinds of Reed-Solomon decoding algorithms: the Berlekamp-Massey algorithm and the Guruswami-Sudan algorithm, the first one needs more real points in the unlocking set. However, the Guruswami-Sudan algorithm leads to significant computation. Actually, these two algorithms produce nearly the same decoding complexities in a wide range. The Berlekamp-Massey algorithm is then chosen.

To improve the error-correcting ability, it may be valuable for us to try the Guruswami-Sudan algorithm.

When the Reed-Solomon decoding algorithm is chosen, another parameter should be decided for the decoding process. This parameter is chosen to decide how many points should be chosen for decoding. Originally, because there should be t real points in the unlocked vault, we should find a degree k polynomial f which contains t points in the vault. However, there is a parameter , which satisfies that k¡¡t and spurious k degree polynomials which contains points is less than one. This parameter could be computed with parameters in the scheme. The formula is . So we only have to search for the degree k polynomial f which contains points in the vault. When the degree k polynomial f is found, the secret is extracted and verified with the signature.

In my proposal I hope to find a scheme which has improved error-correcting ability and a more simple computation. In fact, the scheme presented above has the error-correcting ability of recover the secret embedded in a set of t elements from discretional elements in the set, using the Reed-Solomon decoder. One way to this approach is to use the Guruswami-Sudan algorithm, and manage to decrease the computation of this method.

Another approach to improve the error-correcting ability is to link secret l with elements in the locking vault.

Suppose the elements in the locking vault are a1, a2, a3, a4, a5, an These values with l constitute of a vector Y=(a1a2a3a4a5 l).(The vector means that all strings are combined together.) The vector is then multiplied with an Lm matrix X. L represents the length of the vector. P=YX Modify the smartcard-based scheme. When the unlocking vault V is constructed, it is compared with the locked vault U. Give a definition of distance from an element a to a set B: it means that the distance of the nearest element from a in B. With this definition, n closest elements (their distances to U are the most little) in V are chosen, denoted as b1, b2, b3, b4, b5, These n closest elements constitute the unlocked vault instead of the one given by the original scheme. When the secret l is extracted from the vault, if l is successfully verified, the authentication is then successful. If l is not successfully verified with the signature, it is then checked with the check matrix P. Assume Z=(b1, b2, b3, b4, b5, l). The check matrix P is then used to correct Z, and a modified b1, b2, b3, b4, b5, bn are got. Using these modified values to extract the secret l' from locked vault and then verify it with the signature again. These processes are repeated until the secret is successfully verified or the repeating time has exceeded a threshold d.

In this modified scheme, we should note that when we don't choose the elements in V which have a distance less than d to U but choose n most close elements in V, it may increase the error rate because maybe elements which have distances larger than d are chosen. Second, the computation complexity may be increased because we should repeat the polynomial reconstruction process. Another problem we should take into account is that whether the modified values would be convergent to the ones which I need. These three main problems are what the proposed scheme should overcome. Also, with changes in the scheme, parameters of the scheme should be reconsidered. Some parameters should be decided, such as n, k, the threshold d, and a. This is also the mission of my research.

4 Experiment results

In our experiment, each face image will play the role of applicant and claim to be any person recorded in database. So the system will be tested for 400*40=16000 times. The eigenface[7] recognition system is used for test and we apply the tranditional Reed-Solomon code. The result is shown in figure 2,3.

d	k	FAR	FRR	Computation
20	11	34.96%	1.00%	-
20	12	13.01%	4.50%	-
20	13	12.96%	3.25%	-
20	14	29.66%	2.75%	-
50	36	2.88%	2.75%	29532.731000
50	36	2.97%	2.50%	29417.234000
50	37	1.08%	4.50%	30400.031000
50	37	1.21%	6.50%	30013.536000
50	38	1.08%	5.00%	29485.093000
50	38	1.09%	5.75%	30035.797000
50	38	1.15%	6.00%	29913.318000
50	40	0.33%	12.00%	29920.766000
50	40	0.35%	12.00%	-
50	42	0.05%	23.00%	-

Figure 2: The experiment result of algorithm 1.

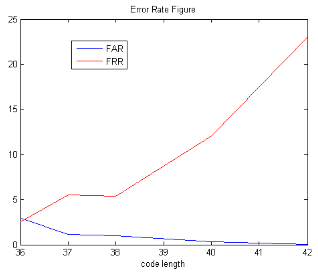


Figure 3: Figure for algorithm1.

The experiment result in figure 2 and 3 shows that the algorithm 1 works nicely. It can achieve a low error rate of about 2.8% in both FAR and FRR when the parameters of the algorithm is carefully chosen. The computation time of this system is nearly 2 seconds each time. The performance of authentication is not weakened much.

5 Conclusion

In our study it it shown that the proposed method is efficient. It highly protects the security of the system with little degeneration of error rate. However, this is only a basic result and lot of work needs to do yet. Actually, the computation time of this design is a little high. In our future work, we will mainly aim at the improve of the computation complexity. Also, we will try the new approach proposed above.

References

[1] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *End-to-End Security*, vol.40, 2001.

[2] A. Juels and M. Wattenberg, "A Fuzzy Commitment Scheme," *Sixth ACM Conference on Computer and Communications Security*, pages 28-36, ACM Press. 1999.

[3] G.I. Davida, Y. Frankel, and B.J. Matt, "On enabling secure applications through off-line biometric identification," *IEEE Symposium on Privacy and Security*, pp. 148-157, 1998.

[4] A. Juels and M. Sudan, "A Fuzzy Vault Scheme," *Proceedings of IEEE Internation Symposium on Information Theory*, p.408, 2002.

[5] T. C. Clancy, N. Kiyavash, and D. J. Lin, "Secure smartcard-based fingerprint authentication," *Proceedings of IEEE Internation Symposium on Information Theoryin Proc. ACMSIGMM 2003 Multimedia, Biometrics Methods and Applications Workshop*, pp.45-52, 2003.

[6] J.I. Hall, "Generalized Reed-Solomon Codes," *Notes on Coding Theory*, 2003.

[7] M. Turk and A. Pentald, "Eigenfaces for recognition," *Journal of Cognitive Neuro-science*, Match 1991.

Writer identification using Hidden Markov Tree Model

Zhenyu He
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong, Hong Kong
zyhe@comp.hkbu.edu.hk

Yuan Yan Tang
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong, Hong Kong
yytang@comp.hkbu.edu.hk

Abstract

Handwriting-based writer identification is a hot research topic in pattern recognition field. Despite continuous effort, off-line handwriting-based writer identification still remains as a challenging problem because writing features only can be extracted from the handwriting image. As a result, plenty of dynamic writing information, which is very valuable for writer identification, is lost in the case of offline. At present, 2-D Gabor filter model is widely acknowledged as a good method for off-line handwriting identification. In this paper, we present a novel method based on Hidden Markov Tree(HMT) model to replace the traditional 2-D Gabor filter. Our experiments show HMT method, compared with 2-D Gabor method, not only achieves better experiment results but also greatly reduces the elapsed time on calculation.

1 Introduction

Even in such a highly developed time, handwriting are still very important and widely used in human society. Handwriting generally is regarded as a sign of the writer. A long history before, people have realized the importance of finding out the true writer of one handwriting document. In fact, writer identification of handwriting (for example, there are signature, letter, notes) has a wide application field: to confirm the document authenticity in the financial sphere, to solve the expert problems in criminology, etc.

Because of the importance of handwriting in human's transactions, automated writer identification of handwriting has a practical significance in many real-world applications. We can classify handwriting-based writer identification in several ways. However, the most straightforward one is to classify it into on-line and off-line writer identifications. The former assumes that a transducer device is connected to the computer, which can convert writing movement into a sequence of signals and then transmit the information to computer. Since dynamic information captured by the transducer device contains many valuable writing features of writer, on-line handwriting-based writer identification, compared with off-line handwriting-based writer identification, is comparatively easier to achieve a high accuracy. Unfortunately, on-line writer identification systems are inapplicable in many cases where transducer device can not be used, thus developing techniques on off-line writer identification is an urgent task.

Further, off-line writer identification can also be divided into two parts: text-dependent and text-independent writer identification [1]. Text-dependent methods match the same character and consequently require the writer to write the same fixed text. Contrastively, given that the handwritings of different people are often visually distinctive, text-independent methods do not require fixed characters but consider the global style of handwriting text. Generally speaking, text-dependent methods have a better identification result, however as mentioned above, they are inapplicable in many practical applications because of their requirement on same character. Our research work focuses on the off-line text-independent writer identification.

2 Relative work

Nowadays, writer identification is an active research field, and more and more researchers have touched on this field and some attempts have been presented [2]. For text-independent writer identification, Duverony has reported that the most important variation of the writers transfer is reflected in the low-frequency band of Fourier spectrum of the handwriting images [3]. Similarly, Kuckuck has used Fourier transform techniques to process handwritten text as texture [4]. Inspired by the idea of multichannel spatial filtering technique, Said, Tan and Baker propose a texture analysis approach [1] . In this method, they regard the handwriting as an image containing some special textures and apply a well-established 2-D Gabor filtering technique to extract feature of such textures. Besides the methods based on frequency-domain analysis, other type approaches are also presented on the text-independent writer identification. In 2000, Schrihari and Cha extracted twelve shape features from the handwriting text lines to represent personal handwriting style. The features mainly contain visible characteristics of the handwriting, such as width, slant and height of the main writing zones [5]. Some other papers also adopte multiple features integration to writer identification [5] [6].

3 Pre-processing

The origin image contains characters of different sizes, spaces between text lines and even noises. So before feature extraction, origin image should be processed firstly. Commonly, the steps of pre-processing are as follows: firstly, removing the noises in the handwriting image; secondly, locating the text line and separating the single character using projection; thirdly, normalizing each character into a same size; finally, creating the texture image by text padding. In our application, we design a pre-processing method which produces texture image for writer identification from original handwriting image. Since some papers have discussed pre-processing [1][2], and this problem is not our focus in this paper, we do not introduce our pre-processing method in detail. An example of image pre-processing is shown in fig 1 and fig 2.

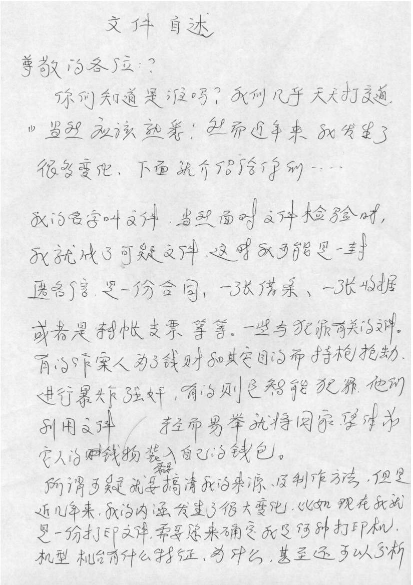


Figure 1. Original handwriting image.

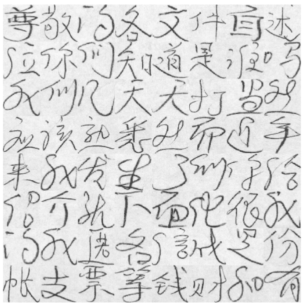


Figure 2. Texture image.

4 A classic method for writer identification: Gabor method

In [1], a well-designed 2-D Gabor filter is proposed for text-independent writer identification. Besides this paper, [2] also applies the same Gabor filter on Chinese text-independent writer identification. Both of the two papers say good results are achieved in their experiments. And the academia also widely acknowledges the Gabor method is an effective method on text-independent writer identification. In this paper, to display the advantage of our new algorithm, we will compare it with the traditional Gabor method. While at first, we will introduce the Gabor model briefly.

The Gabor function is the name given to a Gaussian weighted sinusoid. The function is named after Dennis Gabor who used this function in the 1940s. Later, Daugman proposed the function to describe the spatial response of cells in visual stimuli experiments [7]. The pre-processing of images by Gabor function is chosen for its biological relevance and technical properties. The Gabor function is of similar shape as the receptive fields of simple cells in the primary visual cortex. It is localized in both space and frequency domains and has the shape of plane waves restricted by a Gaussian function.

The computational model of the 2D Gabor filters proposed in reference [1] [2] is given as follows:

$$h_e(x,y) = g(x,y)\cos[2\pi f(x\cos\theta + y\sin\theta)] \quad (1)$$

$$h_o(x,y) = g(x,y)\sin[2\pi f(x\cos\theta + y\sin\theta)] \quad (2)$$

where h_e and h_o denote the so-called even- and odd- symetric Gabor filters, and $g(x,y)$ is an isotropic Gaussian function.

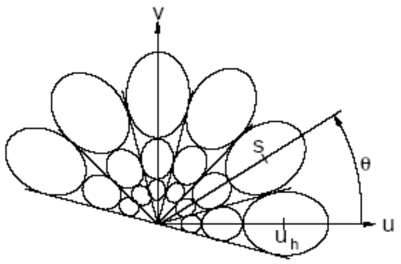


Figure 3. Tiling of the frequency plane by 2-D Gabor filter.

The spatial frequency responses of the Gabor functions are

$$H_e(u,v) = \frac{[H_1(u,v) + H_2(u,v)]}{2} \tag{3}$$

$$H_o(u,v) = \frac{[H_1(u,v) - H_2(u,v)]}{2j} \tag{4}$$

where $j = \sqrt{-1}$ and

$$H_1(u,v) = \exp\{-2\pi^2\sigma^2[(u - f\cos\theta)^2 + (v - f\sin\theta)^2]\}$$

$$H_2(u,v) = \exp\{-2\pi^2\sigma^2[(u + f\cos\theta)^2 + (v + f\cos\theta)^2]\}$$

Here, f, θ, σ are the spatial frequency, orientation, and space constant of the Gabor envelope, separately. Frequency responses of the used Gabor filters are shown in fig 3. For a given input image, $h_e(x,y)$ and $h_o(x,y)$ will combine to provide different channel outputs of the input image with different f, θ and σ. A example of multichannel output of Gabor filters are show in fig **??**.

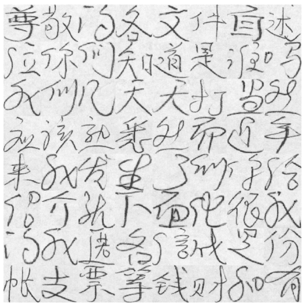


Figure 4. Input handwriting image.

The mean (M) and variance (σ) of the channel outputs are selected as features to represent writer global feature for writer identification.

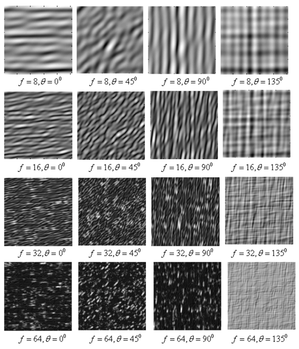


Figure 5. Multichannel outputs of 2-D Gabor filter at different orientations and frequencies.

After extracting the writing features, Weighted Eucliden Distance(WED) is applied for feature matching.

$$WED(k) = \sum_{i=1}^{N} \frac{(M_i - M_i^k)^2}{\sigma_i^k} \tag{5}$$

where M_i denotes the ith mean value of the testing handwriting, M_i^k and σ_i^k denote the ith mean and variance of the training handwriting of writer K separately, and N denotes the total number of mean values.

5 Our method based on Hidden Makovel Tree

Though 2-D Gabor filter is effective in handwriting-based writer identification, this method still suffers from some inherent disadvantages, which greatly limit its practicability. One of the most serious disadvantages is its intensively computational cost, because the 2-D Gabor filter has to convolute the whole image for each orientation and each frequency. As a comparatively new multichannel analysis tool, wavelet transform has a more power to decompose the signal and image, and we can only deal with certain subbands we are interested in. Another disadvantage of the Gabor method mentioned above is that it regards the Gabor coefficients in each subband are independent, ignoring the relation between them. To well capture the relation between wavelet coefficients, Hidden Markovel Tree (HMT) is a ideal model. To give the reader a clear concept on HMT model in wavelet domain, we would like to briefly introduce wavelet transform and HMT separately.

5.1 A simple introduction of wavelet transform

Wavelet is close to the human's visual system to obtain the image information, as is supported by the biologists' study that human's visual cortex can be modelled as a a set of independence channels, each with a particular orientation and spatial frequency tuning.

Wavelet transform is a tool that cuts up data or functions or operators into different frequency components, and then studies each component with a resolution matched to its scale. The basic introduction of wavelet theory has been published in a series of classic papers and books [8][9]. Here, we only introduce basic concepts of wavelet.

A function $\psi \in L^2(R)$ is called an admissible or basic wavelet if it satisfies the following "admissibility" condition.

$$C_\psi := \int_R \frac{|\widehat{\psi}(w)|^2}{|w|} dw < \infty \qquad (6)$$

With translation and dilation of the basic wavelet $\psi(x)$, a 1-D continuous or integrable wavelet transform is defined by

$$(W_\psi f)(a,b) = \int_{-\infty}^{\infty} f(t)\overline{\psi_{a,b}(t)}dt \qquad (7)$$

Where $\psi_{a,b}(t) = a^{-\frac{1}{2}}\psi(\frac{x-b}{a})$. If only discrete value of a and b are used, the eq 7 represents the 1-D discrete wavelet transform.

Choosing some special wavelet functions $\psi(t)$ and scaling functions $\phi(t)$, one signal can be represented as [8][9]:

$$f(t) = \sum_b U_b \phi_{a_0,b}(t) + \sum_{a=-\infty}^{a_0} \sum_b (W_\psi f)(a,b)\psi_{a,b}(t) \qquad (8)$$

where $U_b = \int_{-\infty}^{\infty} f(t)\overline{\phi_{a_0,b}(t)}dt$, the dilation and translation of scaling function are same as that of wavelet function.

In this representation, a means the scale of the resolution of wavelet analysis: smaller a corresponds to higher resolution; b means the spatial location of wavelet analysis. Further, 1-D wavelet transform can be easily extended to 2-D wavelet by using separable tension product.

5.2 Hidden Markov Tree

In [10][11][12], researchers find the wavelet coefficients satisfy two properties–clustering, persistence across scale. Clustering means if the value of one wavelet coefficient is large/samll, those wavelet coefficients nearby this coefficient have a large possibility to be also large/small. Persistence across scale means the large/small values of wavelet coefficients tend to propagate across scales.

To match the nonGaussian nature of the wavelet coefficients, M-state mixture density is used to as probabilistic models for an individual wavelet coefficient.

Generally, an M-state Gaussian mixture model for a individual wavelet coefficient W consists of:

1. A pmf of state variable S with value s, $P_S(s)$, where $s \subset 1, 2, ..., M$.

2. The Gaussian conditional pdfs (probability density function) $f_{W|S}(w|s)$.

3. Finally, the pdf of W is given as follows.

$$f_W(w) = \sum_{s=1}^{M} P_S(s) f_{W|S}(w|s) \qquad (9)$$

The value of wavelet coefficient W is observed, but the value of the sate variable S is not, therefore S is called hidden.

The multisresolution property of wavelet transform suggest a inter-scale dependency between a wavelet coefficient ω_i at a coarse resolution and its corresponding coefficients at the next resolution, which are also called the children of ω_i. By linking the hidden states between the ω_i and its children, we can well capture the inter-scale dependency [13]. Since this model can be drawn as a tree, it is called Hidden Markov Tree (HMT) model.

Fig 6 is a vivid example graph to show the hidden markov tree in wavelet domain for one dimensional signal.

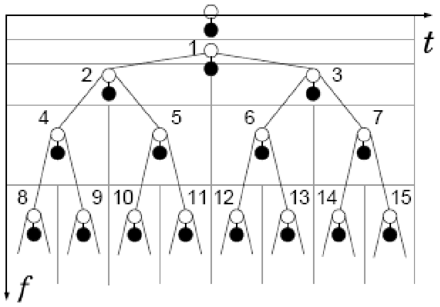


Figure 6. A example reveal the structure of hidden markov tree model

In fig 6, black node represents a wavelet coefficient and white node represents its hidden state. In this figure, the signal is fully decomposed, so the number of tree is one. If the signal is not fully composed, the number of tree is

$N/2^L$, L is level of wavelet decomposition, N is the length of the signal.

Using an M-state Gaussian mixture model for each wavelet coefficient W_i, the Hidden Makove Tree model can be completely defined by the following parameters.

1. $P_{S_1}(m)$, the pmf of state value of the root node 1.

2. $\epsilon_{mr}^{i,p(i)} = P_{S_i|S_{P(i)}}(m|r)$, the $parent- > children$ link between hidden states.

3. μ_{im}, σ_{im}, the mean and variance of Gaussian pdf of wavelet coefficient W_i given state $S_i = m$.

We usually assume $M = 2$ (that is the state only can be in the states "small" and "large"), because in this case the state value has a clear physical meaning. Wavelet coefficients with large value contain significant contributions of signal energy, wavelet coefficients with small value contain little signal energy. And since the wavelet coefficients are generated by the wavelet filters with zero sum, they can be considered to be zero-mean.

5.3 Our method based on HMT model for writer identification

Our method based on HMT model for writer identification is basically consists of 3 steps – preprocessing, HMT training for a given handwriting image, and similarity measurement between two HMT model.

As we mentioned in section 3, the original scanned handwriting image should be firstly processed before extracting the features of it. The function of preprocessing is to remove the noise, normalize the size of characters, and form the texture handwriting image, etc.

Since we have decide to use the HMT model to describe the feature of a handwriting image, the most important work is to train the HMT model according to the input texture handwriting image.

In subsection 5.2, we have said that a HMT model can be completely represented by a set of parameters, $\theta = (P_{S_1}(m), \epsilon_{i,p(i)}^{mr}, \sigma_{im})$

Here, S_1 is the root node of the HMT model; $m, r = 1, 2$ is the state value, $i = 2, ..., J$, J is the level of wavelet decomposition. We simplify the HMT model to assume the wavelet coefficients at the same level of HMT model share the same statistical parameters.

In training HMT model, we look for the parameters that best fit a given set of wavelet coefficients. Maximum Likelihood Estimation (MLE) is a effective principle for parameter estimation. That is, we choose the model parameters that maximize the probability of the observed wavelet data. Therefore, we adopt Baum-Welch algorithm (an iterative Expectation Maximization(EM) algorithm) for training. For details for this part, please refer to [13].

After obtaining the HMT model parameters, we can regard a handwriting image is completely represented by its corresponding HMT model. In other words, the similarity between two HMT models can be considered as the similarity between the two corresponding handwriting images.

From other researchers's works on statistical framework, we know the Kullback-Leibler distance(KLD) should be used to measure the similarity between two HMT models [14].

KLD between two pmfs (probability massive function)is given as:

$$D(\omega||\bar{\omega}) = \sum_i \omega_i \log \frac{\omega_i}{\bar{\omega_i}} \tag{10}$$

KLD between two pdfs is defined as follows:

$$D(p(X;\theta_i)||p(X;\theta_j)) = \int p(x;\theta_i) log \frac{p(x;\theta_i)}{p(x;\theta_j)} \tag{11}$$

In HMT model, the probability function is very complex which can be viewed as a mixture of large number of pdfs, and we do not have a simple, direct expression for the KLD. Monte-Carlo method is a traditional approximation of the KLD, while it need high computational cost [15]. To save the computational cost, we adopt the method to computes the upper bound for the KLD [16].

Lemma 1: The KLD between two mixture densities $\Sigma_i \omega_i f_i$ and $\Sigma_i \bar{\omega_i} \bar{f_i}$ is upper bounded by

$$D(\Sigma_i \omega_i f_i || \Sigma_i \bar{\omega_i} \bar{f_i}) \leq D(\omega||\bar{\omega}) + \Sigma_i \omega_i D(f_i||\bar{f_i}) \tag{12}$$

with equality if and only if $\frac{\omega_i f_i}{\bar{\omega_i} \bar{f_i}} = const$. For the proof of this lemma, please refer to [14].

For a tree node i (please refer to fig 6), its conditional probability density of its observation value given its state is m is defined as:

$$P(O_i = o|S_i = m) = b_m^i(o) \tag{13}$$

Generally, $b_m^i(o)$ is a gaussian function.

Then for a tree node i, define β_m^i is the conditional probability density of the observation value of the subtree of the node i given its state is m, $p(i)$ is the parent of the tree node i, $C(i)$ is the set of children node of tree node i, T_i is the subtree of all nodes with root at i and O_{T_i} is the wavelet values(observed value) of this subtree. Based on the hidden markov chain rule, we can get

$$\beta_m^i(O_{T_i}) = b_m^i(o_i) \prod_{c \subset C(i)} \sum_{m=1}^{2} \epsilon_{mr}^{i,p(i)} \beta_m^c(O_{T_c}) \tag{14}$$

For a leaf node i with no children node,

$$\beta_m^i(O_{T_i}) = b_m^i(o_i) \tag{15}$$

For the root node 1, the probability of the whole observation tree is defined as

$$P(O_{T_1} = o_{T_1}|\theta) = \sum_{m=1}^{2} P_{S_1}(m)\beta_m^1(o_{T_1}) \tag{16}$$

Then, based on lemma 1, KLD between two HMTs is upper bounded as

$$\begin{aligned} D(P(O_{T_1} = o_{T_1}|\theta)|P(\bar{O}_{T_1} = \bar{o}_{T_1}|\bar{\theta})) &\leq D(P_{S_1}||\bar{P}_{S_1}) \\ &+ \sum_{m=1}^{2} P_{S_1}(m)D(\beta_m^1||\bar{\beta}_m^1) \end{aligned} \tag{17}$$

$D(\beta_m^1||\bar{\beta}_m^1)$ can be calculated in the following way.

$$\begin{aligned} &D(\beta_m^i||\bar{\beta}_m^i) = \\ &D(b_m^i||\bar{b}_m^i) + \sum_{c\subset C(1)} D(\sum_{n=1}^{2} a_{mn}^{ic}\beta_n^c|| \sum_{n=1}^{2} \bar{a}_{mn}^{ic}\bar{\beta}_n^c) \\ &\leq D(b_m^i||\bar{b}_m^i) + \sum_{c\subset C(1)} (D(a_m^i||\bar{a}_m^i) + \sum_{n=1}^{2} a_{mn}^{ic} D(\beta_n^c||\bar{\beta}_n^c) \end{aligned} \tag{18}$$

This induction can be iteratively operated till the leaf node. And for leaf node i of the HMT, the KLD can be wrote as

$$D(\beta_m^i||\bar{\beta}_m^i) = D(b_m^i||\bar{b}_m^i) \tag{19}$$

$D(b_m^i||\bar{b}_m^i)$ is a KLD between two Gaussian pdfs. And the following expression for KLD between two d-dimensional Gaussian pdfs is used [17].

$$\begin{aligned} D(N(.;\mu,C)||D(N(.;\bar{\mu},\bar{C}) &= \frac{1}{2}[\log\frac{|\bar{C}|}{C}] - d \\ &+ trace(\bar{C}^{-1}C) + (\mu-\bar{\mu})^T\bar{C}^{-1}(\mu-\bar{\mu})) \end{aligned} \tag{20}$$

As we mentioned above, the pdf of wavelet coefficients is a Gaussian with zero-mean, so this equation can be simplified as

$$D(N(.;\mu,C)||D(N(.;\bar{\mu},\bar{C}) = \frac{1}{2}[\log\frac{|\bar{C}|}{C}] - d + trace(\bar{C}^{-1}C)) \tag{21}$$

In sum, the KLD of two HMT models can be express by the eq 17, and $D(\beta_m^1||\bar{\beta}_m^1)$ in eq 17 can be calculated out by using eq 18 iteratively till leaf node. The KLD between two leaf nodes is expressed by eq 19.

6 Experiments

In this experiment, we compare the results of HMT method and Gabor method on text-independent writer identification. All handwritings are scanned into computer with a resolution of 300 dpi. Then via the pre-processing procedure mentioned above, we produce the handwriting texture images from the scanned handwriting images. Experiments display the size of handwriting texture image should be suitable, since large size image leads to high computational cost and small size decreases the identification accuracy. In our experiments, we select size of texture image as 512×512 pixels.

Since there is not a free,available, large handwriting database for writer identification , we have to create such a database ourselves. 20 Chinese handwritings written by 10 persons have been carried out in experiments, with one training handwriting and one testing handwriting for each person. We produce one handwriting texture image from each handwriting, and thus a total of 20 handwriting texture images are obtained. The training and testing texture images both consist of 64 Chinese characters with size 64×64 pixels, as is shown in fig 7.

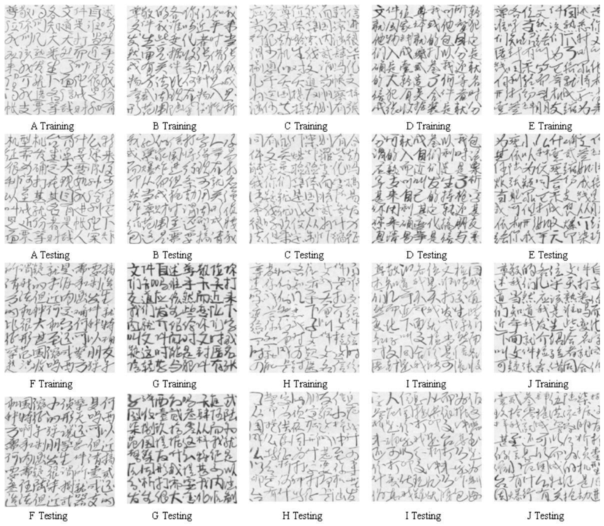


Figure 7. Handwriting samples carried out in our experiment

6.1 Experimental results of 2-D Gabor filters

In this section, we show the effect of Gabor filtering technique on writer identification, not only including the identification accuracy, but also including the calculational time.

In Gabor method, several combinations of different spatial frequencies are used, ranging from 8 to 256. Though [18] shows that for a image of size $N \times N$, the most significant frequency components are equal to or smaller than N/4, we still use the 256 as one frequency value to test this conclusion. And our experiment result also prove the correctness of this conclusion, because the experimental result of combination of $f = 32, 64, 128, 256$ is worse than that of the combination of $f = 16, 32, 64, 128$ and the combination of $f = 8, 16, 32, 64$. For each spatial frequency, we select 0, 45, 90 and 135 degree as orientations.

A testing handwriting texture is matched with all training handwriting texture. Then we sort the matching results in ascending order to produce a list. The position of writer of the testing handwriting in this list is regarded as the experiment result to evaluate algorithm accuracy. (For example, if the matching result between the training handwriting and testing handwriting, both of which are written by the same writer, is minimum in the list and consequently occupies the position 1, we say the position of real writer is top 1; in other words, the topper the position of one writer is, the more possibility of being the real writer of the testing handwriting the writer has).

Table 1 shows the experimental accuracy of different samples based on varied combinations of Gabor frequency. And the table 2 provides a summary of experimental performance of one certain combination of Gabor frequency. The highest accuracy Top $1 = 90\%$ is achieved when $f = 16, 32, 64, 128$. we can find out that the more frequencies are used, the high accuracy is achieved. This effect is clearly demonstrated by both table 1 and table 2

We not only note the identification accuracy, but also record the average elapsed time as a measurement of the computational cost. We run and record the average elapsed time of Gabor method in our PC computer. The software environment of our computer is Windows XP and Matlab 7.0, the hardware configuration of our computer is Intel Pentium IV 3.0G CPU, 512 MB RAM. The experiment result is in the table 3. Though more Gabor frequencies improve the identical result, it also obviously increase the elapsed time.

6.2 Experimental results of HMT method

In HMT method, we decompose the handwriting image via different wavelet filters. The employed wavelet transform are the traditional discrete wavelet transform(DWT), the used wavelets are Daubechies orthogonal wavelets. The wavelet decomposition $J = \log_2(N)$, N is the size of image. That is, we make a full decomposition.

Table 4 show the experimental accuracy using different wavelet filters decompositions. And the table 5 provide summaries of experimental performance of different wavelet filters. The average elapsed time of GGD methods are offered by the tabel 6. The meaning of the experiment results in these tables are same as that of Gabor method. From these tables, we can find out these wavelets make a little difference, though the db2 always be somewhat superior to other db wavelets.

Compared with the results of Gabor method, it is apparent that the HMT model outperform the Gabor model. The reason is very simple, the HMT model is able to well capture the statistical properties of wavelet coefficients at each wavelet sub-band, while the mean and variance used in Gabor method can hardly fully represent the statistical features of Gabor coefficients. What is more, the HMT method also cost fewer time for calculation than Gabor method. For example, the best result (Top1 $= 80\%$, Top $2 = 20\%$ using db2 wavelet) only averagely cost 27.70 seconds, on the other hand, the best result of Gabor ($f = 32, 64, 128, 256$) cost 91.344 seconds. This advantage is very valuable when we need to deal with a large number of handwriting image. And in fact, most practical applications involve in thousands of handwritings or more.

6.3 Experiment remarks

In summary, the following observations can be made based on the experimental result presented above.

1. The HMT model is superior to the Gabor model, because it can better capture the statistical properties of the coefficients in each subband.

2. The average elapsed time of HMT model is much shorter than that of Gabor model.

3. Generally, the length of the wavelet filter does make a difference, and the db2 wavelet is better than other db wavelets.

7 Conclusion

In this paper, we presented a new method based on HMT model in wavelet domain for off-line text-independent handwriting identification. Compared with Gabor method, this method achieves a higher accuracy and significantly reduces the computational time. Because text-independent methods do not care about the writing content, the methods discussed in this paper are also available for English, Korean, Japanese and Latin Language, etc.

8 Acknowledgment

This research was partially supported by a grant (60403011) from National Natural Science Foundation of China. This research was also supported by the grants (RGC and FRG) from Hong Kong Baptist University.

References

[1] H.E.S. Said, T.N. Tan, K.D. Baker, Personal identification based on handwriting. Pattern Recognition. vol.33, no.1, pp.149–160, 2000.

[2] Y. Zhu, Y. Wang, T. Tan, Biometric Personal Identification Based on Handwriting. 15th International Conference on Pattern Recognition. vol.2, pp.801-804, Sep. 2000, Barcelona, Spain.

[3] J. Duvernoy, Handwriting synthesis and classification by means of space-variant transform and Karhunen-Loeve analysis. J.Opt.Soc.Am. pp.1331–1336, 1975.

[4] W. Kuckuck, Writer identifcation by spectra analysis. Proceedings. of the international Conference On Security Through Science Engineering, pp.1–3, Aug. 1980, West Berling, Gerany.

[5] S.Cha, S.N.Srihari, *Multiple Feature Integration for writer Verification* the Precddings of 7th IWFHR2000, pp.333-342, Sep. 2000, amstredam, Netherland.

[6] E.N.Zois, V.anastassopousls, *Fusion of Correlated Decisions for Writer Verification* Pattern Recognition, vol.33, no.10, pp.1821-1829, 1999.

[7] J.G.Daugman, *Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters* J.Opt.Soc.Am. A2, pp.1160-1169, 1985.

[8] I.Daubechies, Ten lectures on wavelet. New York:SIAM, 1992.

[9] M.Vetterli, Wavelets and subband coding, Prentice Hall, 1995.

[10] M.T.Orchard, K.Ramchandran, An investigation of wavelet-based image coding using an entropy constrained quantization framwork. proceeding of Data Compression Conference ’94 , pp.341–350, 1994, Snowbird, Utah.

[11] S.Mallat, S.Zhong, Characterization of signals from multiscale edges. IEEE Trans. on Pattern Analysis and Machine Intelligence vol. 14, pp.710–732, Apr. 1992.

[12] S.Mallat, W.Hwang, Singularity detection and processing with wavelets. IEEE Trans. Inform.Theory, vol.38, no.2, pp.617–643, Jul. 1992.

[13] M.Crouse, R.D.Nowak, R.G.Baraniuk, Wavelet-based signal processing using hidden Markov model. IEEE Trans.Signal proc.(Special Issue on Wavelets and Filterbanks) pp.886–902, 1998.

[14] T.M.Cover, J.A.Thomas, Elements of Information Theory, Wiley Interscience, New York, NY, 1991.

[15] B.H.Juang, L.R.Rabiner, A probabilistic distance measure for hidden Markov models, AT&T Tech. J., vol.64, no.2, pp.391-408, Feb.1985.

[16] M. N. Do, Fast approximation of Kullback-Leibler distance for dependence trees and hidden Markov models. IEEE Signal Processing Letters, vol. 10, pp.115–118, Apr. 2003

[17] Y.Singer, M.K.Warmuth, Batch and on-line parameter estimation of Gaussian mixtures based on the joint entropy. Advances in Neural Information Processing Systems 11 (NIPS’98), pp.578–584, 1998.

[18] T.N.Tan, Texture feature extraction via visual cortical channel modelling. Proceedings of ICPR1992, pp.607–610, Aug. 1992, Hague, The Netherlands.

Table 1. EXPERIMENTAL RESULTS OF 2-D GABOR FILTERS

Features	f = 8,16,32,64	f = 16,32,64,128	f = 32,64,128,256	f = 32,64	f = 64,128	f = 16	f = 32	f = 64	f = 128
A testing	Top 1	Top 1	Top 1	Top 2	Top 2	Top 3	Top 4	Top 2	Top 1
B testing	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1
C testing	Top 2	Top 1	Top 1	Top 2	Top 1	Top 2	Top 2	Top 2	Top 1
D testing	Top 1	Top 1	Top 2	Top 1	Top 1	Top 1	Top 2	Top 1	Top 2
E testing	Top 1	Top 1	Top 2	Top 2	Top 3	Top 1	Top 1	Top 3	Top 2
F testing	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 2	Top 1
G testing	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1
H testing	Top 1	Top 1	Top 1	Top 1	Top 1	Top 2	Top 1	Top 1	Top 1
I testing	Top 1	Top 1	Top 1	Top 1	Top 1	Top 1	Top 2	Top 1	Top 1
J testing	Top 3	Top 4	Top 2	Top 4	Top 3	Top 4	Top 5	Top 2	Top 3

Table 2. IDENTIFICATION ACCUARCY OF 2-D GABOR FILTERS

Features	f = 8,16,32,64	f = 16,32,64,128	f = 32,64,128,256	f = 32,64	f = 64,128	f = 16	f = 32	f = 64	f = 128
Top 1	80%	90%	70%	60%	70%	60%	50%	50%	70%
Top 2	10%	0%	30%	30%	20%	20%	30%	40%	20%
Top 3	10%	0%	0%	0%	10%	10%	0%	10%	10%
Top 4	0%	10%	0%	10%	0%	10%	10%	0%	0%
Top 5	0%	0%	0%	0%	0%	0%	10%	0%	0%

Table 3. AVERAGE ELAPSED TIME OF 2-D GABOR FILTERS

Features	f = 8,16,32,64	f = 16,32,64,128	f = 32,64,128,256	f = 32,64	f = 64,128	f = 16	f = 32	f = 64	f = 128
Average elapsed time(second)	90.875	91.344	92.266	46.188	45.860	22.875	23.141	23.063	23.047

Table 4. EXPERIMENTAL RESULTS OF HMT-based METHOD

Features	db2	db4	db6	db8	db10
A testing	Top 1	Top 1	Top 2	Top 2	Top 2
B testing	Top 1	Top 1	Top 1	Top 1	Top 1
C testing	Top 1	Top 2	Top 1	Top 3	Top 1
D testing	Top 1	Top 1	Top 1	Top 1	Top 1
E testing	Top 1	Top 1	Top 1	Top 1	Top 2
F testing	Top 1	Top 1	Top 1	Top 1	Top 1
G testing	Top 1	Top 1	Top 1	Top 1	Top 1
H testing	Top 1	Top 1	Top 1	Top 3	Top 3
I testing	Top 1	Top 1	Top 1	Top 1	Top 1
J testing	Top 2	Top3	Top 5	Top 3	Top 3

Table 5. IDENTIFICATION ACCUARCY OF WAVELET-BASED GGD METHOD, THREE-LEVEL DWT DECOMPOSITION

Features	db2	db4	db6	db8	db10
Top 1	90%	80%	80%	60%	60%
Top 2	10%	10%	10%	10%	20%
Top 3	0%	10%	0%	30%	20%
Top 4	0%	0%	0%	0%	0%
Top 5	0%	0%	10%	0%	0%

Table 6. AVERAGE ELAPSED TIME OF WAVELET-BASED GGD METHOD, THREE-LEVEL DWT DECOMPOSITION

Features	db2	db4	db6	db8	db10
Average elapsed time(second)	27.70	32.91	38.23	42.23	51.78

On The Solution to Singular Integral Equations with Logarithmic Kernel Based on Wavelet

L. M. Cui, and Y. Y. Tang
Department of Computer Science
Hong Kong Baptist University
lmcui@comp.hkbu.edu.hk

Abstract

In this paper, Wavelet-Galerkin algorithm for solving the first kind of weak singular integral equations with the logarithmic kernel is presented. Because of the singularity of logarithmic kernel we use Tikhonov regularization method to solve the system of stiff equation. And at last the convergence and numerical result of approximate solutions are discussed.

1 Introduction

There are a lot of advantages to solve integral equations with Wavelet-Galerkin algorithm, such as the numerical sparsity of stiff matrix, the diagonal precondition procedure etc. A great deal of difficulties arises if we only use Wavelet-Galerkin algorithm to solve singular integral equations. This paper studies the solution to weak singular integral equations with logarithmic kernel based on wavelet. For the purpose of illuminating this question, Laplace's equation is, firstly, presented.

To the first kind weak singular integral equation as follows:

$$-2\int_0^1 [\log|\gamma(t)-\gamma(s)|]ds = g(t). \tag{1}$$

It will occur when we use the single potential representation.

The interior question about Laplace's equation can be described as follows:

$$\begin{cases} \Delta\omega(x) = 0, & x=(x_1,x_2)\in\Omega \\ \omega(x) = q(x), & x=(x_1,x_2)\in\Gamma \end{cases}, \tag{2}$$

where $q(x)$ is the given boundary function and $q(x) \in C(\Gamma)$, Γ is the smooth boundary of Ω,and $\omega(x)$ is the function to be found. Through the single potential representation, we know that solving (2) can be transformed into solving the following integral equations

$$\omega(x) = -\int_\Gamma \rho(y)[\log|x-y|]dS_y, \quad x=(x_1,x_2)\in\Omega.$$

We must solve $\rho(x)$ by the following boundary integral equation

$$-\int_\Gamma \rho(y)[\log|x-y|]dS_y = q(x), \quad x=(x_1,x_2)\in\Gamma. \tag{3}$$

We can write $x=\gamma(t),\ t\in[0,1]$, as the representation of Γ because Γ is smooth. Then

$$-\int_0^1 \rho(\gamma(s))\gamma'(s)[\log|\gamma(t)-\gamma(s)|]ds = q(\gamma(t)). \tag{4}$$

Let $f(s)=\frac{\rho(\gamma(s))\gamma'(s)}{2}$, then (4) can be transformed into (5)

$$-2\int_0^1 f(s)[\log|\gamma(t)-\gamma(s)|]ds = q(\gamma(t)). \tag{5}$$

Let

$$Mf(t) = -2\int_0^1 f(s)[\log|\gamma(t)-\gamma(s)|]ds$$

and

$$g(t)=q(\gamma(t)),$$

then (5) can be transformed into (1), and we denote

$$Mf = g. \tag{6}$$

This paper solves (1) using the characteristic of periodic wavelet on $L^2[0,1]$. At first, we decompose the unknown function on scaling space not only to decrease computational complexity but also to get a sparse matrix. The solutions to the discrete system of equations which are obtained above are ill-posed problems, because of the characteristic of periodic wavelet on $L^2[0,1]$ and logarithmic kernel. This paper, firstly, solves this ill-posed system of equations with Tikhonov regularization method. Next first by adding some valve value according to requirements, a quick and simple numerical implementation to solve (1) is formed. In the end, an example of calculation is presented for the purpose of accounting for the algorithm's validity. Periodic wavelet and multiresolution analysis is introduced in Sec.2. Wavelet-Galerkin algorithm for solving integral equations is introduced in Sec.3. Sec.4 analyses the properties of operator M. The deduction and characteristic of stiff matrix is introduced in Sec.5. Sec.6 solves the system of equations which is ill-posed after discretization with Tikhonov regularization method. An algorithm to solve the first kind of singular integral equation with the logarithmic kernel and the verification convergence of this algorithm are presented in Sec.7, and an example of computation is given in Sec.8.

2 Periodic Wavelet

Let $\varphi(x)$ and $\psi(x)$ be the normal orthogonal scaling function and the compactly supported wavelet function. Suppose $\varphi(x)$ and $\psi(x)$ adapt to multiresolution analysis respectively. Meanwhile $\varphi(x)$ and $\psi(x)$ satisfy the double-scaling equations as follows:

$$\varphi(x)=\sqrt{2}\sum_n h_n\varphi(2x-n) \ \ and \ \ \psi(x)=\sqrt{2}\sum_n g_n\psi(2x-n)$$

where $\{\varphi_{j,k}\}_k$ and $\{\psi_{j,k}\}_k$ are orthonormal basis of V_j and W_j respectively. We periodize scaling function $\varphi(x)$ to $\tilde{\varphi}(x)$ and wavelet function $\psi(x)$ to $\tilde{\psi}(x)$ on $L^2[0,1]$ [1], namely

$$\tilde{\varphi}(x):=\sum_l \varphi(x+l), \quad \tilde{\psi}(x):=\sum_l \psi(x+l).$$

Periodic scaling function space and periodic wavelet function space are marked with

$$\tilde{V}_J := closedspan\{\tilde{\varphi}_{J,k}, k=0,1,\cdots,2^J-1\},$$

$$\tilde{W}_J := closedspan\{\tilde{\psi}_{J,k}, k=0,1,\cdots,2^J-1\}.$$

where $\{\tilde{\varphi}_{J,k}\}_k$ and $\{\tilde{\psi}_{J,k}\}_k$ are orthonormal basis of $\tilde{V}_J$ and $\tilde{W}_J$ respectively. We get some equations as the following[2]

$$\begin{cases} \tilde{\varphi}(x)=\sqrt{2}\sum h_k\tilde{\varphi}(2x-k), & \{h_k\}\in l^2, \\ \tilde{\psi}(x)=\sqrt{2}\sum g_l\tilde{\varphi}(2x-l), & \{g_l\}\in l^2, \\ \tilde{V}_j=\tilde{W}_{j-1}\bigoplus\tilde{W}_{j-2}\bigoplus\cdots\bigoplus\tilde{W}_0\bigoplus\tilde{V}_0, & j\in Z^+, \\ \tilde{V}_j\perp\tilde{W}_l, & j\neq l, \\ \bigcup_{j\in Z}\tilde{W}_j=L^2[0,1]. & \end{cases} \tag{7}$$

where $\{\tilde{\varphi}_{0,0}\}\bigcup\bigcup_{0\le j\le J-1}\{\tilde{\psi}_{j,k}\}_k$ is a group of orthonormal basis of $\tilde{V}_J$ too, because $\tilde{V}_J = \tilde{W}_{J-1}\bigoplus\tilde{W}_{J-2}\bigoplus\cdots\bigoplus\tilde{W}_0\bigoplus\tilde{V}_0, \ J \in Z^+$.We know a transition matrix $W_{\tilde{\varphi},\tilde{\psi}}$ exits between $\{\tilde{\varphi}_{J,k}\}_k$ and $\{\tilde{\varphi}_{0,0}\}\bigcup\bigcup_{0\le j\le J-1}\{\tilde{\psi}_{j,l}\}_l$ and $W_{\tilde{\varphi},\tilde{\psi}}$ is positive definite. In fact $\tilde{\varphi}_{0,0}=0$.

3 Wavelet-Galerkin Algorithms for Solving Integral Equation

The Wavelet-Galerkin Method is a standard Galerkin method with wavelet $\bigcup_{-1\le j\le J-1}\{\tilde{\psi}_{j,k}\}_k$ as the trail basis and the test basis, where

$$\{\tilde{\psi}_{-1,k}\}_k=\{\tilde{\psi}_{0,0}\}. \tag{8}$$

Denoting f with (3.2) under scale J. Then we have

$$f_J=\sum_{j=-1}^{J-1}\sum_k x_{j,k}\tilde{\psi}_{j,k}. \tag{9}$$

Replacing f in (6) with (9), we obtain

$$M\sum_{j=-1}^{J-1}\sum_k x_{j,k}\tilde{\psi}_{j,k}=g. \tag{10}$$

Taking the inner product with $\tilde{\psi}_{j',k'}$ on either hand of (10), we have

$$\langle M\sum_{j=-1}^{J-1}\sum_k x_{j,k}\tilde{\psi}_{j,k},\tilde{\psi}_{j',k'}\rangle=\langle g,\tilde{\psi}_{j',k'}\rangle.$$

So

$$\sum_{j=-1}^{J-1}\sum_k x_{j,k}\langle M\tilde{\psi}_{j,k},\tilde{\psi}_{j',k'}\rangle=\langle g,\tilde{\psi}_{j',k'}\rangle.$$

Then we solve (6) to obtain the approximate solution to (11):

$$\mathbf{M}_{\tilde{\psi}}\mathbf{x}=\mathbf{g}, \tag{11}$$

where $\mathbf{M}_{\tilde{\psi}}$ is the moment matrix with entries elements $\langle M\tilde{\psi}_{j,k},\tilde{\psi}_{j',k'}\rangle$, $\mathbf{g}$, $\mathbf{x}$ are vectors with elements $\langle g,\tilde{\psi}_{j',k'}\rangle$ and $x_{j,k}$ respectively, and $\langle\cdot,\cdot\rangle$ is the inner product on $L^2[0,1]$. But the method is hindered by the computers of the entries of $\mathbf{M}_{\tilde{\psi}}$ and $\mathbf{g}$, which are the integrals of the product of the wavelet basis $\tilde{\psi}$ with the known kernel function in the integral equation (6) and the known right hand side $\mathbf{g}$. One method to deal with it is to use the fast transform $\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}$. In which, the moment matrix $\mathbf{M}_{\tilde{\varphi}}=[\langle M\tilde{\varphi}_{J,k},\tilde{\varphi}_{J,k'}\rangle_{k,k'}]$ and the right hand side vector $\mathbf{g}'=\{\langle g,\tilde{\varphi}_{J,k}\rangle\}_k$ with scaling function basis $\{\tilde{\varphi}_{J,k}\}_k$ are calculated first, and then from relationship, we can find $\mathbf{M}_{\tilde{\psi}}$ and $\mathbf{g}$ by

$$\mathbf{M}_{\tilde{\psi}}=\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}\mathbf{M}_{\tilde{\varphi}}\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}^T$$

$$\mathbf{g} = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}\mathbf{g'}.$$

Solving the equation (11) becomes:

finding $f_J = \sum_{j=-1}^{J-1}\sum_k x_{j,k}\tilde{\psi}_{j,k}$ such that

$$\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}\mathbf{M}_{\tilde{\varphi}}\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}}\mathbf{x} = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}\mathbf{g'}, \qquad (12)$$

where $\mathbf{M}_{\tilde{\varphi}}$ is the moment matrix with entries $\langle M\tilde{\varphi}_{J,k}, \tilde{\varphi}_{J,k'}\rangle$ and **g'** is a vector with elements $\langle g, \tilde{\varphi}_{J,k}\rangle$.

4 Properties of Operator $\mathbb{M}$

Let Γ be a circle whose radius is α, then we have

$$x = (x_1, x_2) = \alpha(\cos(2\pi s), \sin(2\pi s)).$$

Therefore

$$Mf(t) = -2\int_0^1 f(s)[\log|2\alpha\sin\pi(t-s)|]ds.$$

Using the formula in the theory of generalized function, we have

$$\frac{1}{\pi}\log|2\sin(\frac{\theta}{2})| = -\frac{1}{2\pi}\sum_{\substack{-\infty \\ n\neq 0}}^{+\infty}\frac{1}{|n|}e^{in\theta} = -\frac{1}{\pi}\sum_{n=1}^{+\infty}\frac{1}{n}\cos n\theta.$$

We get the following equalities:

$$-2\log|2\alpha\sin\pi(t-s)| = -2\log\alpha - 2\log|2\sin\pi t|$$

$$= -2\log\alpha + \sum_{\substack{-\infty \\ n\neq 0}}^{+\infty}\frac{1}{|n|}e^{2\pi in(t-s)}.$$

Without loss of generality, let v to be a function with 1-period, then the Fourier series expression of v is

$$v(t) = \sqrt{2\pi}\sum_{n=-\infty}^{+\infty}\hat{v}(n)e^{2\pi int}.$$

Therefore

$$Mv(t) = -2\sqrt{2\pi}(\log\alpha)\hat{v}(0) + \sqrt{2\pi}\sum_{\substack{-\infty \\ n\neq 0}}^{+\infty}\frac{1}{|n|}\hat{v}(n)e^{2\pi int} \qquad (13)$$

Assume $\tilde{v}(t)$ is the periodic version of function $v(t)$, namely $\tilde{v}(t) = \sum_{-\infty}^{+\infty}v(t+l)$. We have

$$\tilde{v}(t) = \sqrt{2\pi}\sum_{-\infty}^{+\infty}\hat{v}(2\pi l)e^{2\pi int}.$$

So the periodic scaling function is given by

$$\tilde{\varphi}_{J,k}(t) = \sqrt{2\pi}\sum_{-\infty}^{+\infty}\hat{\varphi}_{J,k}(2\pi l)e^{2\pi int}. \qquad (14)$$

Then

$$\hat{\tilde{\varphi}}_{J,k}(l) = \hat{\varphi}(2\pi l). \qquad (15)$$

Using (13), we have

$$M\tilde{\varphi}_{J,k}(t) = -2\sqrt{2\pi}(\log\alpha)\hat{\tilde{\varphi}}_{J,k}(0) + \sqrt{2\pi}\sum_{\substack{-\infty \\ n\neq 0}}^{+\infty}\frac{1}{|n|}\hat{\tilde{\varphi}}_{J,k}(n)e^{2\pi int} \qquad (16)$$

5 The Computation and Properties of The Moment Matrix $\mathbb{M}_{\tilde{\varphi}}$

Using (15) and (16), the elements of $\mathbf{M}_{\tilde{\varphi}}$ can be gained by the following deduction:

$$\langle M\tilde{\varphi}_{J,k}, \tilde{\varphi}_{J,k'}\rangle = -4\pi(\log\alpha)\hat{\tilde{\varphi}}_{J,k}(0)\overline{\hat{\tilde{\varphi}}}_{J,k'}(0)$$

$$+2\pi sum^{+\infty}_{\substack{-\infty \\ n\neq 0}}\frac{1}{|n|}\hat{\tilde{\varphi}}_{J,k}(n)e^{2\pi int}\hat{\tilde{\varphi}}_{J,k'}(n')e^{2\pi in't}$$

By the orthogonality of $e^{2\pi int}$ and $\hat{\varphi}_{J,k}(n) = 2^{-\frac{J}{2}}e^{-in\frac{k}{2^J}}\hat{\varphi}(\frac{n}{2^J})$, we have

$$\langle M\tilde{\varphi}_{J,k}, \tilde{\varphi}_{J,k'}\rangle = -4\pi(\log\alpha)\cdot 2^{-J}$$

$$+2\pi\cdot 2^{-J}\sum_{\substack{-\infty \\ n\neq 0}}^{+\infty}\frac{1}{|n|}\exp^{-\frac{i(k-k')2\pi n}{2^J}}|\hat{\varphi}(\frac{2\pi n}{2^J})|^2.$$

Namely

$$\langle M\tilde{\varphi}_{J,k}, \tilde{\varphi}_{J,k'}\rangle = -4\pi(\log\alpha)\cdot 2^{-J}$$

$$+4\pi\cdot 2^{-J}\sum_{n=1}^{+\infty}\frac{1}{|n|}\cos\frac{i(k-k')2\pi n}{2^J}|\hat{\varphi}(\frac{2\pi n}{2^J})|^2. \qquad (17)$$

Theorem 1 *Equation (17) satisfies properties as follows*
(1)Symmetry, namely $m_{k,k'} = m_{k',k}$*;*
(2)Circulation, namely $m_{k,k'} = m_{k+1,k'+1}$ *;*
(3)Repetitiveness, namely $m_{k,k'} = m_{k,2^J-k'+2}$ *.*

proof: We only prove (3).

$$m_{k,2^J-k'+2} = \langle M\tilde{\varphi}_{J,k}, \tilde{\varphi}_{J,2^J-k'+2}\rangle$$

$$= -4\pi(\log\alpha)\cdot 2^{-J}$$

$$+4\pi\cdot 2^{-J}\cdot\sum_{n=1}^{+\infty}\frac{1}{n}\cos\frac{[k-(2^J-k'+2)2\pi n]}{2^J}|\hat{\varphi}\frac{2\pi n}{2^J}|^2$$

$$= -4\pi(\log\alpha)\cdot 2^{-J} + 4\pi\cdot 2^{-J}\cdot\sum_{n=1}^{+\infty}\frac{1}{n}\cos\frac{(k-k')2\pi n}{2^J}|\hat{\varphi}\frac{2\pi n}{2^J}|^2$$

$$= \langle M\tilde{\varphi}_{J,k}, \tilde{\varphi}_{J,k'}\rangle$$

$$= m_{k,k'}$$

We get from theorem 5.1 that $\mathbf{M}_{\tilde{\varphi}}$ is a circular symmetrical matrix with rank 2^J, so only $2^{J-1}+1$ elements are need to calculate. In this way the computational complexity and the storage are decreased sharply. The computation of compactly supported scaling function $\varphi(x)$ is as same as the counterpart in Sec.4.

When g is smooth, the computation of $\langle g, \tilde{\varphi}_{J,k}\rangle$ is relatively easy, which results are only normal numerical solutions on the functional integral based on wavelet [3]. For example, Gauss-type quadrature

$$\int_R f(t)\varphi_{J,k}(t)dt \approx 2^{-\frac{J}{2}} f(\frac{m_1+k}{2^J}). \qquad (18)$$

where $m_n = \int_R t^n\varphi(t)dt$, and m_n can be calculated by equations as below [4]:

$$\begin{cases} m_0 = 1 \\ m_l = \frac{1}{2^l-1}\frac{\sqrt{2}}{2}\sum_{j=0}^{l-1}\binom{l}{j}m_j(\sum_n h_n n^{l-j}) \end{cases}$$

The Fourier transform of compactly supported scaling function $\varphi(x)$ can be calculated by

$$\hat{\varphi}(x) = \frac{1}{\sqrt{2\pi}}\prod_{j=1}^{\infty}H(2^{-j}\omega), \quad H(\omega) = \frac{1}{\sqrt{2}}\sum_{k\in Z}h_k e(-ik\omega).$$

6 Wavelet Regularization Method

Literature [5] concretely describes a Schwarz iteration algorithm which is used to rapidly solve linear ill-posed problems, in order to dispose these ill-posed problems. This algorithm adopts the Tikhonov regularization method. It employs spline function or Daubechies wavelet and applies them to the first kind integral equation. On the basis of literature [5], this paper further decomposes functions on wavelet space, so it can be organically linked up with Wavelet-Galerkin method which is mentioned in Sec.5 and takes wavelet function as basis.

6.1 Regulation Method Using Additive Schwarz iteration for Ill-posed Problems

We mainly study how to apply periodic wavelet in the problem of solving the first kind operator equation as follows

$$Mf = g. \qquad (19)$$

where $M : X \to Y$ is a compact operator, so (19) is ill-posed.

Considering problem (19), we now assume that noisy data $g^\varepsilon \in Y$ are available satisfying $\|g - g^\varepsilon\|_Y \le \varepsilon$ for a known error bound $\varepsilon > 0$. According to Tikhonov regularization method, a computable approximation to f^* is then provided by the unique solution $f_l^{\varepsilon,\alpha}$ of the finite dimensional normal equation

$$(M_l^* M_l + \alpha I)f_l = M_l^* g^\varepsilon, \quad \alpha > 0. \qquad (20)$$

In (20), $M_l = MP_l$, where $P_l : X \to \tilde{V}_l$ is the orthogonal projection on the meaning of inner product, and space $\tilde{V}_l$ is the scaling function space of periodic wavelet, and $\tilde{V}_l \subset X$. Under these assumptions the quantity

$$\gamma_l := \|M - M_l\| = \|M(I - P_l)\|. \qquad (21)$$

Based on the characteristic of multiresolution analysis of periodic wavelet, we have

$$\gamma_l \le \gamma_{l+1}, \quad and \quad \gamma_l \to 0\,(l \to 0) \Leftrightarrow Miscompactoperator[6].$$

In addition, on the basis of the characteristic of orthogonal decomposition of space $\tilde{V}$, we have

$$\tilde{V}_l = \tilde{V}_{l_{min}} \bigoplus \bigoplus_{j=l_{min}}^{l-1} \tilde{W}_j, \quad l_{min} \le l-1. \qquad (22)$$

Accordingly, the following decompose of projection operator can be got:

$$P_l = P_{l_{min}} + \sum_{j=l_{min}}^{l-1} Q_j,$$

where Q_j is orthogonal projection from X to space $\tilde{W}_j$. Let $\tilde{V}_j$ and $\tilde{W}_j$ be the scaling function space and the wavelet space of periodic wavelet respectively, and let operator $M : X \to Y$ be a compact linear operator. Then

$$\|MQ_l\| \le \gamma_l \to 0 \quad (l \to \infty),$$

where γ_l is defined in (21).

Taking the inner product with v_l on either hand of (20):

$$\langle (M_l^* M_l + \alpha I)f_l, v_l\rangle = \langle M_l^* g^\varepsilon, v_l\rangle$$

we have

$$\langle M_l^* M_l f_l, v_l\rangle + \langle f_l, v_l\rangle = \langle M_l^* g^\varepsilon, v_l\rangle$$

namely

$$\langle M_l f_l, M_l v_l\rangle + \langle f_l, v_l\rangle = \langle M_l^* g^\varepsilon, v_l\rangle$$

Define the bilinear form $a : X \times X \to L^2[0,1]$ as follow:

$$a(u,v) := \langle Mu, Mv\rangle_Y + \alpha\langle u,v\rangle_X. \qquad (23)$$

Clearly, α is symmetric positive definite.

Therefore, the problem of solving (20) can be transformed into solving the problem described as below:

finding $f_l^{\varepsilon,\alpha} \in V_l$, such that $f_l^{\varepsilon,\alpha}$ satisfies

$$a(f_l^{\varepsilon,\alpha}, v_l) := \langle K_l^* g^\varepsilon, v_l\rangle_X, \quad \forall v_l \in \tilde{V}_l. \qquad (24)$$

Define the following operators:

$$A_l = M_l^* M_l + \alpha P_l$$

and

$$B_l = Q_l M_l^* M_l Q_l + \alpha Q_l.$$

Obviously,

$$a(u_l, v_l) = \langle A_l u_l, v_l\rangle_X, \quad \forall u_l, v_l \in \tilde{V}_l \quad and \quad a(w_l, z_l)$$

$$= \langle B_l w_l, z_l\rangle_X, \quad \forall w_l, z_l \in \tilde{W}_l.$$

Considering the definition of A_l, (20) can be transformed into

$$A_l f_l = K_l^* g^\varepsilon.$$

Another bilinear form $b_j : \tilde{W}_j \times \tilde{W}_j \to L^2[0,1]$ is defined by

$$b_j(w_j, u_j) := \alpha\langle w_j, u_j\rangle_X.$$

We know from lemma 6.1, b_j is approximating α defined by (23) on space $\tilde{W}_j$ when j is increased enough.

By introducing an operator $F_j = \alpha^{-1} Q_j A_l$ for $v_l \in \tilde{V}_l$ and $w_j \in \tilde{W}_j$, we have

$$b_j(F_j v_j, w_j) = \alpha(v_l, w_j).$$

Let $\tilde{V}_{l_{min}}$ be the coarsest approximate space, for $v_l \in \tilde{V}_l$ and $v_{l_{min}} \in \tilde{V}_{l_{min}}$, and we introduce another operator:

$$R_{l_{min}} = P_{l_{min}} A_{l_{min}}^{-1} P_{l_{min}} A_l.$$

Then

$$R_{l_{min}} : \tilde{V}_l \to \tilde{V}_{l_{min}} \quad and \quad a(R_{l_{min}} v_l, v_{l_{min}}) = a(v_l, v_{l_{min}})$$

Next adding all F_j to $R_{l_{min}}$, we have

$$R_{l_{min}} + \sum_{j=l_{min}}^{l-1} F_j = P_{l_{min}} A_{l_{min}}^{-1}(P_{l_{min}} + \alpha^{-1}\sum_{j=l_{min}}^{l-1} Q_j) A_l.$$

Through the formula about decomposition in space (22), we know $C_{l,l_{min}} := P_{l_{min}} A_{l_{min}}^{-1} P_{l_{min}} + \alpha^{-1}\sum_{j=l_{min}}^{l-1} Q_j$ is the approximation of A_l^{-1}.

For (24), we can use additive Schwarz iteration to solve this problem rapidly:

$$u_l^{\mu+1} = u_l^\mu - \omega C_{l,l_{min}}(A_l u_l^\mu - K_l^* g^\varepsilon), \quad \mu = 0,1,2,\cdots, \qquad (25)$$

where an initial guess $\mu_l^0 \in \tilde{V}_l$ and a damping parameter $\omega \in R$.

6.2 Deduction of Algebraic Equations

In this section we present a matrix version of the iteration (25) given suitable bases in $\tilde{V}_l$ and $\tilde{W}_l$.

Let space $X = L^2[0,1]$, $\tilde{\varphi}$ and $\tilde{\psi}$ are scaling function and wavelet function of periodic wavelet separately. They satisfy multiresolution analysis of periodic wavelet.

Since function $f_l = \sum_k c_k^l \tilde{\varphi}_{l,k} \in V_l$ and function $g_l = \sum_k d_k^l \tilde{\psi}_{l,k} \in W_l$, we have an equation as follows

$$f_l + g_l = \sum_k c_k^{l+1} \tilde{\varphi}_{l+1,k}.$$

Also

$$c_k^{l+1} = \sum_i h_{k-2i} c_k^l + \sum_j h_{k-2j} d_j^l.$$

which we write in matrix notation as

$$c_k^{l+1} = H_{l+1}^T c^l + G_{l+1}^T d^l.$$

Clearly, $H_{l+1} : R^{n_l+1} \to R^{n_l}$ and $G_{l+1} : R^{n_l+1} \to R^{n_l}$.

The solution $f^{\varepsilon,\alpha}$ can be expanded in the basis of $\tilde{W}_l$ as

$$f_l^{\varepsilon,\alpha} = \sum_k (\xi_l)_k \tilde{\varphi}_{l,k},$$

where the vector ξ of the expansion coefficients is the unique solution of the linear system

$$A_l \xi_l = \beta_l.$$

The concrete forms of the matrix A_l and the vector β_l are given in [5] as follows

$$(A_l)_{i,j} = \langle M\tilde{\varphi}_{l,i}, M\tilde{\varphi}_{l,j}\rangle_Y + \alpha\langle\tilde{\varphi}_{l,i}, \tilde{\varphi}_{l,j}\rangle_X,$$

$$(\beta_l)_j = \langle g^\varepsilon, M\tilde{\varphi}_{l,j}\rangle_Y.$$

$H_{l,j}$ and$G_{l,j}$ are defined by

$$\mathbf{H}_{l,j} := H_{j+1} H_j \cdots H_{l-1} H_l : R^{n_l} \to R^{n_j}.$$

and

$$\mathbf{G}_{l,j} := G_{j+1} H_j \cdots H_{l-1} H_l : R^{n_l} \to R^{n_j}.$$

where $j \leq l-2$. Let $\mathbf{H}_{l,l-1} := H_l$ and $\mathbf{G}_{l,l-1} := G_l$, then the above iteration can be transformed into

$$z_l^{k+1} = z_l^k - \omega C_{l,l_{min}}(A_l z_l^k) - \beta_l, \quad k = 0, 1, 2, \cdots, \tag{26}$$

where

$$C_{l,l_{min}} = H^T_{l,l_{min}} A^{-1}_{l,l_{min}} H_{l,l_{min}} + \alpha^{-1} \sum_{j=l_{min}}^{l-1} G^T_{l,j} B_j^{-1} G_{l,j} \tag{27}$$

$$z_l^0 = H^T_{l,l_{min}} A^{-1}_{l,l_{min}} H_{l,l_{min}}. \tag{28}$$

Different from [5], this paper defines a matrix A_l and a vector β_l as follows

$$(A_l)_{i,j} = \langle M\tilde{\psi}_{l,i}, M\tilde{\psi}_{l,j}\rangle_Y + \alpha\langle \tilde{\psi}_{l,i}, \tilde{\psi}_{l,j}\rangle_X,$$

$$(\beta_l)_j = \langle g^\varepsilon, M\tilde{\psi}_{l,j}\rangle_Y.$$

Then we use the iteration format as similar to (26) , but (27) and (28) respectively are changed to

$$C_{l,l_{min}} = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}(H^T_{l,l_{min}} A^{-1}_{l,l_{min}} H_{l,l_{min}}$$

$$+\alpha^{-1} \sum_{j=l_{min}}^{l-1} G^T_{l,j} B_j^{-1} G_{l,j})\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}}$$

$$z_l^0 = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}(H^T_{l,l_{min}} A^{-1}_{l,l_{min}} H_{l,l_{min}})\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}}\beta_l.$$

7 Algorithm and Convergence

On the basis of the above analysis, we conclude the Wavelet-Galerkin algorithm for solving (1) as follows:

- **Step-1** Compute $g_l, m_{k,l},\ l = 0, 1, \cdots, 2^J - 1$ by (18), (17). We get the matrix $\mathbf{M}^J_{\tilde{\varphi}}$ and the vector **g**.
- **Step-2** Calculating $W_{\tilde{\varphi},\tilde{\psi}}$ and we get the equation (12).
- **Step-3** Solving regularization equation (20) by (26) and (27), we get $x_k,\ k = 0, 1, \cdots, 2^J - 1$.

Theorem 2 *Let $\tilde{m}_{\tilde{\psi}kl}$, $m_{\tilde{\psi}kl}$ be the elements of $\tilde{\mathbf{M}}_{\tilde{\psi}}$ and $\mathbf{M}_{\tilde{\psi}}$ respectively. Then we get f_J by (9). In the Wavelet-Galerkin algorithm,$\tilde{m}_{\tilde{\psi}kl}$ uniformly converges to $m_{\tilde{\psi}kl}$ as $n \to \infty$, and f_J uniformly converges to the solution $f(x)$ of function (1) as $J \to \infty$ and $n \to \infty$.*

proof: Since $\tilde{\mathbf{M}}_{\tilde{\psi}} = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}\tilde{\mathbf{M}}_{\tilde{\varphi}}\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}}$, $\mathbf{M}_{\tilde{\psi}} = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}\mathbf{M}_{\tilde{\varphi}}\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}}$,we known from [7] that uniformly converges to m_{kl} when $n \to \infty$. So $\|\tilde{\mathbf{M}}_{\tilde{\varphi}} - \mathbf{M}_{\tilde{\varphi}}\| \to 0 \quad (n \to \infty)$. Since $\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}$ is orthogonal matrix, we have

$$\|\tilde{\mathbf{M}}_{\tilde{\psi}} - \mathbf{M}_{\tilde{\psi}}\| = \|\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}(\tilde{\mathbf{M}}_{\tilde{\varphi}} - \mathbf{M}_{\tilde{\varphi}})\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}}\|$$

$$= \|\tilde{\mathbf{M}}_{\tilde{\varphi}} - \mathbf{M}_{\tilde{\varphi}}\| \to 0 \quad (n \to \infty).$$

Then $\tilde{m}_{\tilde{\psi}kl}$ uniformly converges to $m_{\tilde{\psi}kl}$ as $n \to \infty$.

Since $f_J = \mathbf{W}_{\tilde{\varphi},\tilde{\psi}}F_n P_J f(x)$, and known from [7],$F_n P_J f(x)$ uniformly converges to $\mathbf{W}^T_{\tilde{\varphi},\tilde{\psi}} f(x)$ as $J \to \infty, n \to \infty$.Similarly since $\mathbf{W}_{\tilde{\varphi},\tilde{\psi}}$ is orthogonal, f_J uniformly converges to the solution $f(x)$ of equation (1.1).

We can prove the convergence of wavelet regularization by [5].

8 Numerical Examples

Let $g(x) = e^{x_1}\cos x_2$ in the Laplace's interior problem (2), and assume $\alpha = 0.5$. Then we solve the boundary integral equation (1) by the above algorithm.

8.1 Computation of Stiff Matrix

We employ $J = 4$ and $J = 7$ as two kinds of scaling discretization for $[0, 1]$, namely dividing $[0, 1]$ into 16 and 128 equal parts. We assign Daubechies scaling function with $N = 6$ as $\varphi_{4,0}(x)$ and $\varphi_{7,0}(x)$, then transform them into periodic function on space $L^2[0, 1]$.

Since elements in $\mathbf{M}^J_{\tilde{\varphi}}$ are circular symmetry, only $2^{J-1} + 1$ of these elements are needed to compute and to be stored. Let $n = 100$ in (16). For scaling discrete $J = 4$, the matrix $\mathbf{M}^J_{\tilde{\varphi}}$ is 16×16, and for scaling discrete $J = 7$, the matrix $\mathbf{M}^J_{\tilde{\varphi}}$ is 128×128.

Similarly, we know from Figure 1-4,$\mathbf{M}^J_{\tilde{\psi}}$ is sparser than $\mathbf{M}^J_{\tilde{\varphi}}$ and increasing J is the key to improve effect.

8.2 Analysis of Regularization

Analysis and Explanation:

(1) Figure 5 and Figure 6 are situation when $J = 4$. Error is no more than $1.677093352110495e^{-004}$ with Tikhonov regularization using Schwartz iteration, while error is no more than $5.863185146877874e^{-005}$ with Tikhonov regularization using Newton iteration. The number of times conduct both iterations is large.

(2) Figure 7 and Figure 8 are situation when $J = 7$. Error is no more than $5.183141329873582e^{-007}$ with Tikhonov regularization using Schwartz iteration, while error is no more than $5.959144566129969e^{-005}$ with Tikhonov regularization using Newton iteration. The times of Schwartz iteration are 1, while the times of Newton iteration are 11. It is obvious that Tikhonov regularization using Schwartz iteration can deal with the situation when J is relatively

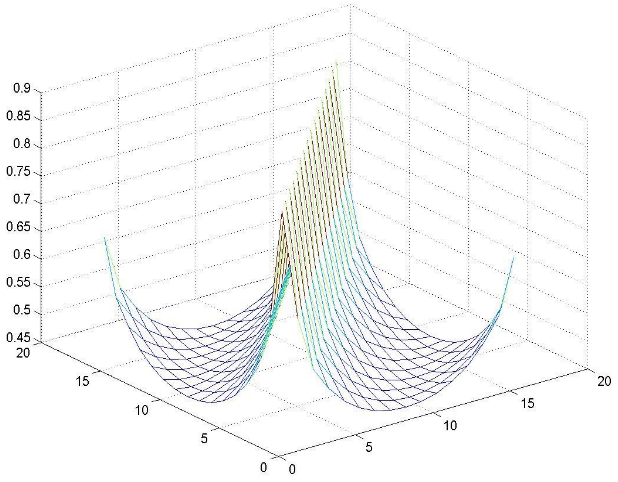


Figure 1. Matrix $\mathrm{M}^J_{\tilde{\varphi}}$ when $J = 4$

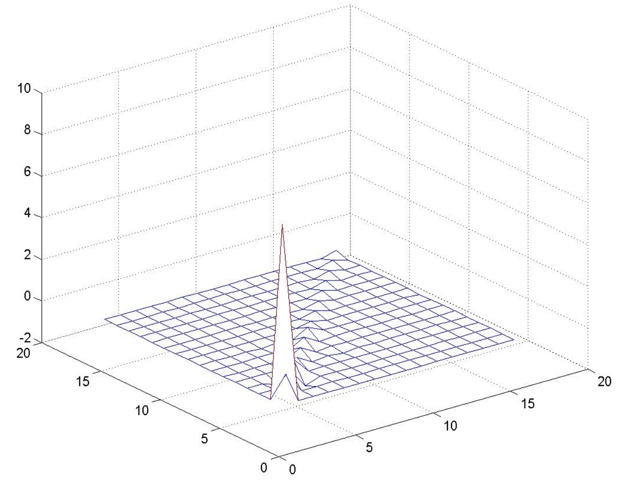


Figure 2. Matrix $\mathrm{M}^J_{\tilde{\psi}}$ when $J = 4$

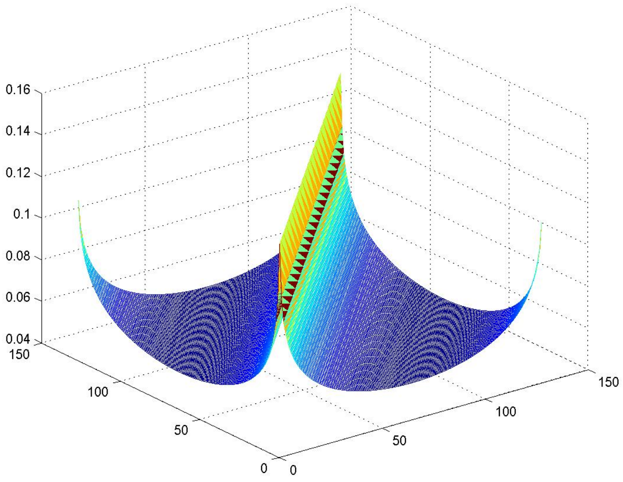


Figure 3. Matrix $\mathrm{M}^J_{\tilde{\varphi}}$ when $J = 7$

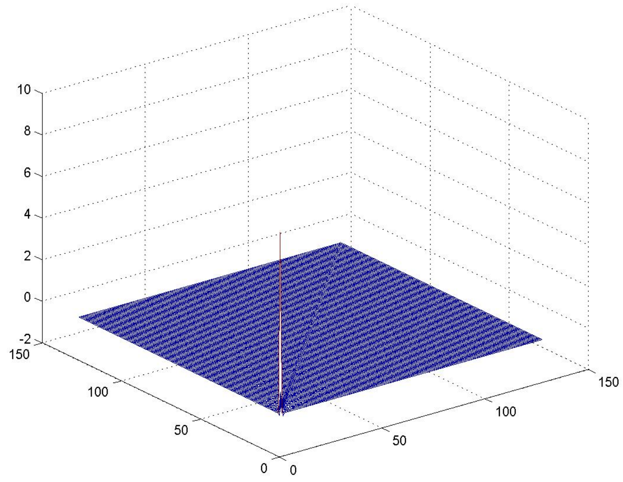


Figure 4. Matrix $\mathrm{M}^J_{\tilde{\psi}}$ when $J = 7$

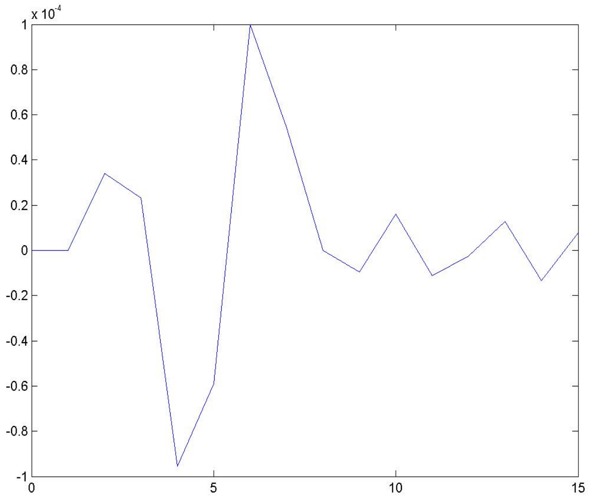


Figure 5. Error of Tikhonov Regularization Using Schwartz iteration When $J = 4$

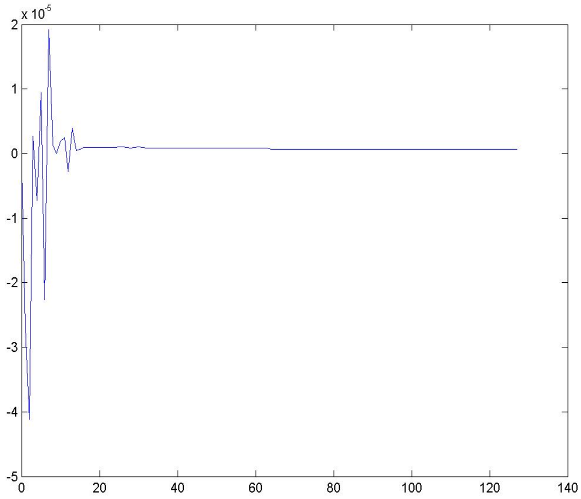


Figure 7. Error of Tikhonov Regularization Using Schwartz iteration When $J = 7$

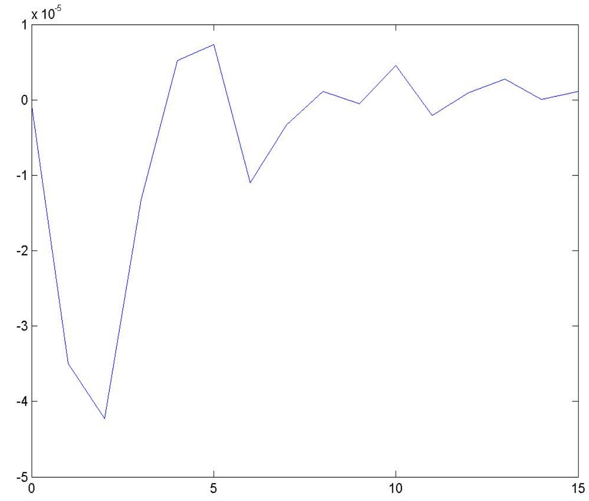


Figure 6. Error of Tikhonov Regularization Using Newton iteration When $J = 4$

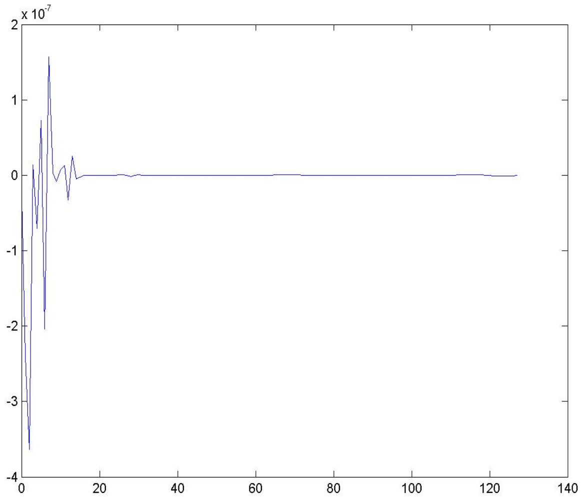


Figure 8. Error of Tikhonov Regularization Using Newton iteration When $J = 7$

large.
(3) We make decomposition only one time in the regularization method using Schwartz iteration when $J = 4$ and $J = 7$. Solutions become more precise when times of decomposition is smaller, and solution become coarser when times of decomposition is bigger, though regularization method using Schwartz iteration is better than regularization method using Newton iteration in precision, regularization method using Schwartz iteration is better than regularization method using Newton iteration in precision in iterations when J is large.
(4) We make decomposition only once in the additive Schwarz iteration Tikhonov regularization method in Figure 3, Figure 4. It is known that fewer times of decomposition are made, more precise solutions is obtained and more times of decomposition are made, coarser solution is obtained.
(5) The errors in Figure 5-8 are obtained by comparing the double ends of $\mathbf{M}_{\tilde{\psi}}^{J}\mathbf{x} = \mathbf{g}$ after substituting results for the counterpart of the double ends of this equation.

The difficulty of solving singular integral equation with wavelet method is corresponding to different singular kernels, we should use different numerical method to gain satisfied effectiveness on stable and rapid algorithms of calculation of singular integral. This paper uses the character of periodic wavelet on $L^2[0,1]$ and logarithmic kernel to make stiff matrix sparser and uses Tikhonov regularization method to remove singularity for the purpose of approximate calculating this kind of weak singular integral equation rapidly. The method discussed in this paper can be used to deal with other kinds of singular integral problems with trigonometric function singular kernel.

References

[1] I. Daubechies, “The orthonomal bases of compactly supported wavelets,” Comm.Pure Appl. Math , vol. 41, pp 906C996, 1988.

[2] S. L. PengD. F. Li and Q. H. Sheng, Theory and Applications of the periodic wavelets,Beijing,Science Press, 2003.

[3] G. Strong,and T. Nguyen, Wavelets and Filter Banks,Wellesley-Cambridge Press, 1966.

[4] A. Cohen,I. Daubechies and P. Vial,“Wavelet on the interval and fast wavelet transforms,”Appl. Comp. Harm. Anal., Vol. 1, No. 1, pp 54-81, 1993.

[5] A Rieder,“A wavelet multilevel method for ill-posed problems stabilized by Tikhonov regularization,”Numer.Math.,Vol. 75, No. 4, pp 501-522, 1997.

[6] C. W. Groetsch, The theory of Tikhonov regularization for Fredholm equations of the first kind,Boston,Pitman,1984.

[7] C. F. Xu and Y. F. Yao,“The wavelet method for solving singular integral equations with the Hilbert kernel,”,Numerical Mathematics A Journal of Chinese Universities, No. 1, pp 128-35, 2000.

Data Hiding on Polygonal Meshes for Authentication and Content Annotation

Hao-tian Wu

Abstract

In this paper, two methods are proposed to embed data within polygonal meshes for different applications. The first one provide a solution to public authentication of 3D mesh models by employing digital signature scheme. By generating the signature from a mesh piece and using it to replace the LSBs of vertex coordinates, the authenticity of the signed mesh can be publicly validated. The possible tampering can be localized by using the mesh partitioning technique. In the second method, a string of bit values is embedded into the meshes by modulating the distance from a vertex to its traversed neighbors. The embedded data is invariant to translation, rotation and uniformly scaling of the cover mesh, as well as the mantissa truncation of vertex coordinates within a range. Since a mesh can have different scale, orientation and position in the 3D space without affecting its integrity, even different precision level after compression, the embedded data can be used for content annotation. To show and compare the efficiency of the proposed methods, the numerical experiments are conducted. Finally, some concluding remarks are drawn and the future work is outlined.

1 Introduction

With the development of digital modeling and visualizing techniques for 3D objects, 3D models have been widely created and used for geometry representation, such as cultural heritage recording like Digital Michelangelo Project [1], CAD models, and structural data of biological macromolecules [2]. Polygonal meshes are considered as the common representation of 3D shapes and it's easy to convert other types of 3D models into meshes. As more and more meshes appear on the Internet, how to hide information within them has received much attention for a variety of purposes, ranging from copyright enforcement (e.g. [3], [6]) to content verification (e.g. [5]). Compared with digital images, video and audio streams, there exists no grid for meshes, i.e., each vertex in a mesh is connected with variable neighboring vertices at different distances. This flexibility of mesh data makes it an attractive cover object for data hiding.

In the literature, quite a few data hiding methods (e.g.[3]-[17]) have been proposed to meet different requirements. For instance, one application of hiding data within mesh is for copyright protection so that the robustness of embedded data is emphasized while the original mesh can be used in the retrieval process [6]. In contrast, for authentication applications, the embedded data should be sensitive to illegal modifications made to the mesh and the embedded data must be blindly retrieved [17]. In some cases, only one bit of information is enough to perform the task, while the capacity of other schemes is expected to be as high as possible. Nevertheless, there are some common requirements, such as security and fidelity. A data hiding scheme is considered secure if it is almost impossible to remove, detect or change the embedded data [18] without the additional information that can be kept from public access. It should be assumed that the algorithms are publicly known so that the system security can only rely on the secrecy of the private information and the difficulty to compute it. Fidelity means that the embedded data is invisible (except the case that it's intentionally visible), i.e., the embedding process should not introduce noticeable distortion into the cover object. It is preferred that the introduced error can be numerically analyzed and bounded.

In this paper, two methods are proposed to embed data within 3D polygonal meshes for different applications. The first one aims to publicly authenticate 3D mesh models using digital signature scheme. By generating the signature from a mesh piece and replacing the LSBs of vertex coordinates with it, the authenticity of the signed mesh can be publicly validated. The possible tampering can be localized by using the mesh partitioning technique. The second method is proposed for content annotation of polygonal meshes. Since a mesh model can have different scale, orientation and position in the 3D space without affecting its integrity, even different precision level after geometry compression, the embedded data should be invariant to translation, rotation and uniformly scaling of the cover mesh, as well as truncation of vertex coordinates in a range. To embed the annotation data, the distance from a vertex to the centroid of its traversed neighbors is modulated using dithered modulation technique [20]. In the modulation, a quantization step independent from the cover mesh is used to balance the introduced distortion and the allowable range of coordinate truncation. Compared with 0.5 bit/vertex in the formerly reported method [14], the capacity of this method is nearly 1 bit/vertex. The numerical results has been given to show the efficiency of the proposed methods. At last, some concluding remarks are drawn and our future work is outlined.

2 A Method for Public Authentication of 3D Mesh Models

In this section, we propose a public-key scheme for mesh authentication. Firstly, the mesh is partitioned into several pieces with the same number of vertices. Then the corresponding signature of each piece is generated and used to replace the LSBs of vertex coordinates within it, respectively. In the authentication process, the knowledge of the cryptographic algorithms and the public key is used to decrypt the value from the retrieved signature and produce the hash value from the content to be authenticated. At last, the two produced values are compared with each other to authenticate the mesh model and localize the possible tampering.

2.1 Mesh Partitioning

Traditional digital signature schemes such as RSA and DSA [21][22] can detect the tampering, but cannot localize the tampering position. It is partially due to the generated signature is separate from the content to be authenticated. If we can divide the content into several blocks and embed the corresponding signature of each block within it as in [23], the modification made to one block will not affect the others so that the tampering can be localized. Since the signatures generated from every block are in the same size, the mesh needs to be partitioned into pieces with a fixed amount of vertices so that the same number of LSBs can be replaced by the signature. The following mesh partitioning algorithm can be used to perform this task.

We order the process of mesh partitioning with the vertex indices in the mesh file. At first, the first indexed vertex is selected as the starting vertex of the first piece and marked as visited. Then the neighboring vertices that are unvisited and connected with the visited vertices are found out. At each time, only the first indexed one among those neighboring vertices is traversed and marked as visited. After a new vertex is traversed, the neighboring vertices need to be updated to find out the next vertex to be visited. A piece is formed when the required number of vertices are traversed. Consequently, the first indexed vertex among the unvisited vertices is selected as the starting vertex of the next piece, and so on until all the vertices are traversed. Finally, the last piece may have less vertices than the others, but the other pieces have the same number of vertices. In this case, we combine the last two pieces to form one larger piece.

If there are multiple meshes in one model, those meshes will be traversed one by one. In case that all the vertices within one mesh have been visited but the vertex number in the current piece is less than required, the first indexed vertex in the next mesh will be picked and added into the current piece. After that, the neighboring vertices will be found out in the newly traversed mesh until the required number is reached. As a result, vertices from different meshes may be contained in the same piece.

2.2 The Generation of Signature

After dividing the mesh model into pieces with the required number of vertices, the next step is to generate the corresponding signature from each of them. The geometrical and topological information of each piece, as well as other properties, such as the color, texture and material properties, should be taken to produce the hash value. Before putting those values as the input of hash function, the LSBs of vertex coordinates need to be set as zeros since they will be replaced by the generated signature. The mesh topology (i.e. the connectivity of vertices) can be represented by the IndexedFaceSet as in VRML [19] format that indicates which vertices a face consists of. For example, the coordinates of three vertices in one triangle face can be listed one by one to indicate those three vertices constitute a triangle face. The faces in each piece are ordered by the face indices in the mesh file so that the sequence of faces is ordered. Each face is represented by the coordinates of vertices within it and the sequence of vertices is ordered by the vertex indices. For each vertex, its coordinates are in the order of X, Y and Z axes. Therefore, the geometry and topology information can be represented by the ordered sequence of vertex coordinates. Other properties of each piece can also be added after the sequence of vertex coordinates as the input of the hash functions.

Some faces consist of vertices contained by multiple pieces, so they cover the boundaries between the neighboring pieces. Since the full topology information within each piece should be taken into account, the hash value will be produced from all faces that the vertices within the piece belong to, including those boundary faces. The produced hash value will be independent from those vertices in other pieces if their index values instead of their coordinates are taken in the sequence of vertex coordinates. In the later experiments, we will use the SHA-1 hash function [24] to produce a hash value of 160 bits from the sequence of vertex coordinates and other properties of each piece. After the hash value is produced, the corresponding signature of each piece is generated by encrypting the produced hash value using the private key. The RSA algorithm [25] is used to generate a signature with 1024 bits. The DSA algorithm [21][22] can also be used to generate a signature of 320 bits. The same private key is used for each piece so that the corresponding public key can be used to authenticate the whole mesh model.

2.3 The Embedding of Signature

After the corresponding signature of each piece is generated, the next step is to used the generated signature to replace the LSBs of vertex coordinates within it. To address the LSB replacement of vertex coordinates, the format of floating-point number [26] should be introduced first. We take a single-precision binary floating-point number for instance, which is stored in a 32-bit word and consists of three fields, named as the Sign with the most significant bit (with index 31), the Exponent with the middle 8 bits (with index from 23 to 30), and the Mantissa with the last 23 bits (with index from 0 to 22), as shown in Figure 1.

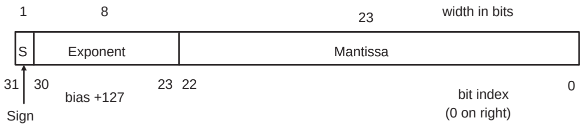


Figure 1. A single-precision binary floating-point number is stored in 32-bit word.

The stored value of the Sign is 0 or 1, representing for the sign value of 1 and -1, respectively. The stored value of Exponent is in the range from 0 to 255, with 0 used for zeroes and denormalised numbers, 255 for infinities and NaN (short for Not a Number), and the range from 1 to 254 for normalised numbers. Since the vertex coordinates do not have the values of infinities and NaN, we will not discuss these two cases. In order to be able to represent both tiny and huge values, exponents have to be signed value. So the stored value of Exponent is offset (by 127 for a single-precision floating-point number) from the actual value. For a single-precision number, the actual exponent for normalised numbers is between -126 and 127. For normalised numbers, the stored value of Mantissa is the fractional part of the significand, i.e., the significand is the binary number 1 followed by the radix point followed by the binary bits of Mantissa. The stored value of Mantissa is 0 for zeroes and non zero for denormalised numbers. If we use s, e and m to represent the stored value of Sign, Exponent and Mantissa, respectively, the value v of a floating-point normalised number can be calculated by

$$v = (-1)^s \times 2^{e-127} \times 1.m, \tag{1}$$

where the denormalised numbers are the same except the exponent is not $e - 127$ but $e - 126$ and the significand is not $1.m$ but $0.m$. From the format of floating-point number, it is not hard to conclude that the least significant bit is the last bit of Mantissa, i.e., the bit with the index 0. If the least significant bit is replaced, the introduced relative error is no more than 2^{-23}.

Since each vertex has three coordinates on the X, Y and Z axes, respectively, no less than 342 vertices are required in each piece to carry the signature of 1024 bits if only the least significant bit is replaced. Else if we use the last four bits in Mantissa to carry the signature, each piece should consist of at least 86 vertices. The LSB replacement process is ordered by the vertex indices, i.e., the smaller the index value of the vertex is, the earlier the LSB of the vertex coordinate are replaced. If the vertex number is more than required, such as 342 if only the least significant bit of vertex coordinate is replaced, the LSBs of vertices out of the required number need not to be set as zeroes before the signature generation, for they will not be replaced by the signature. To localize the tampering, the size of piece should be as small as possible, given the signature has enough length for security.

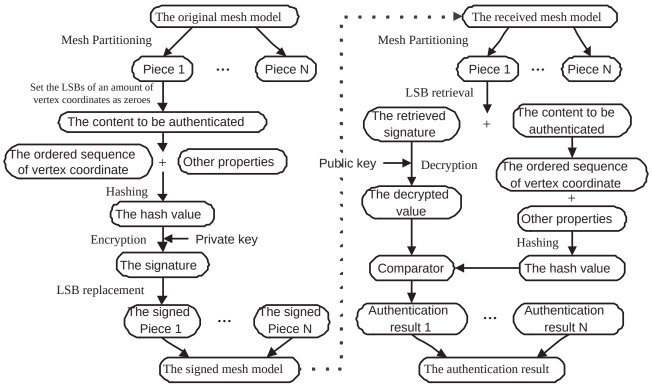


Figure 2. The procedure of using the proposed public-key scheme to publicly authenticate 3D mesh model.

2.4 The Authentication Process

In the authentication process, the same mesh partitioning technique as in Subsection 2.1 is used to divide the mesh into pieces with a certain amount of vertices. For each piece, the signature is retrieved from the LSBs of vertex coordinates, ordered by the vertex indices. Using the public key corresponding to the private key used in the generation of signature, the value is decrypted from the retrieved signature. After that, the LSBs of vertex coordinates are set as zeroes and a new hash value is produced from the ordered sequence of vertex coordinates and other properties, as in Subsection 2.2. The decrypted value is compared with the produced hash value to authenticate each piece. A piece is considered as intact only if the two values generated from it are identical to each other. After all pieces have been authenticated one by one, the authenticity of the mesh model is verified if all of them are intact. Otherwise, the tampering will be localized within those pieces in which the two values do not match with each other. The procedure of using the proposed scheme to authenticate 3D mesh models can be outlined in Figure 2.

3 Data Hiding on Polygonal Meshes for Content Annotation

In this section, we propose a new method to embed the annotation information in polygonal meshes. There are two parts of information contained in the mesh data, i.e. the mesh geometry and topology. The mesh geometry can be represented by the set of vertex positions $V = \{v_1, \cdots, v_m\}$, which defines the shape of the mesh in R^3, given m vertices in a mesh. The mesh topology, i.e., the connectivity between vertices, is described using IndexedFaceSet as in VRML [19] format, which specifies the n vertices $\{v_1, \cdot\cdot\cdot, v_n\}$ within each polygonal face. Compared with the method proposed for public authentication purpose, this method uses a secret key instead of a key pair, which consists of data embedding and message retrieval processes.

3.1. Data Embedding

Given a string of data bits $B = (w_i)_{i=1}^N$, where N is the length of message, the task of data embedding is to embed the value of each bit w_i into the mesh geometry. Since we aim to embed the data invariant to translation, rotation and uniformly scaling of the cover mesh, the ratio between the distances defined within the cover mesh serves as a good candidate to carry information. In our method, a special case of quantization index modulation (QIM) called dither modulation [20] is employed and the distance d_j from a vertex to the centroid of its traversed neighbors is chosen as the embedding primitive. The quantization step S used in the modulation is made proportional to the distance from the last traversed vertex to the mesh centroid so that the ratio between d_j and S is independent from the scale, orientation and position of the cover mesh in the 3D space.

The mesh topology is taken into account during the embedding process. However, if we modulate the distance from a vertex to the centroid of all its neighbors by adjust its position, the distance from the formerly traversed vertex to the centroid of its neighbors will be simultaneously changed so that the information formerly embedded in it might be lost. To solve the problem of causality, among the neighbors of each vertex, only the traversed ones are numerated to calculate their centroid. By adjusting the position of the newly traversed vertex, the distance from it to the centroid of its traversed neighbors is modulated. After the adjustment, the position of the newly traversed vertex will not be changed any more.

The detailed process to embed the data bits B is as follows: Initially, we use a secret key K as the seed of pseudo-random generator to permute the vertex indices I. The generated indices I' will order the following mesh traversal. At first, the vertex first indexed by I' is firstly traversed without adjusting its position since all of its neighboring vertices have not been traversed. After that, by referring to the connectivity information of the cover mesh, all the neighboring vertices of the traversed vertices are numerated and the one first indexed by I' is selected. For a newly traversed vertex v_j, suppose N_j of its neighboring vertices have been traversed and denoted as $(v_i^j)_{i=1}^{N_j}$. Then the centroid of the traversed neighbors can be obtained by

$$v_{jc} = \frac{1}{N_j} \sum_{i=1}^{N_j} v_i^j. \quad (2)$$

The distance from v_{jc} to v_j is defined as

$$d_j = \sqrt{(v_{jcx} - v_{jx})^2 + (v_{jcy} - v_{jy})^2 + (v_{jcz} - v_{jz})^2}, \quad (3)$$

where $\{v_{jcx}, v_{jcy}, v_{jcz}\}$ and $\{v_{jx}, v_{jy}, v_{jz}\}$ are the coordinates of v_{jc} and v_j in R^3, respectively. To modulate the distance d_j, an appropriate value is assigned to the quantization step S to calculate the integer quotient Q_j and the remainder R_j by

$$Q_j = \lfloor d_j/S \rfloor, \quad (4)$$

$$R_j = d_j \% S. \quad (5)$$

To embed one bit value w_i, we modulate d_j by

$$d'_j = e_j + \begin{cases} Q_j \times S + S/2 & \text{if } Q_j\%2 = w_i \\ Q_j \times S - S/2 & \text{if } Q_j\%2 = \overline{w_i} \ \&\ R_j < S/2 \\ Q_j \times S + 3S/2 & \text{if } Q_j\%2 = \overline{w_i} \ \&\ R_j \geq S/2 \end{cases}, \quad (6)$$

where d'_j is the modulated distance and $\overline{w_i} = 1 - w_i$. The distance d_j is modulated in this way so that $Q'_j\%2 = w_j$ given $e_j \in (-\frac{S}{2}, \frac{S}{2})$. To make the modulation step statistically undetectable, the component e_j should be randomly distributed within $(-\frac{S}{2}, \frac{S}{2})$. As a result, the change of d_j is within $(-S, S)$ as $Q_i\%2 = w_i$, while $(-\frac{3S}{2}, \frac{3S}{2})$ as $Q_i\%2 = \overline{w_i}$ so that the modulation is bounded. Consequently, the resulting d'_j is used to adjust the position of v_j by

$$v'_j = v_{jc} + (v_j - v_{jc}) \times \frac{d'_j}{d_j}, \quad (7)$$

where v'_j is the adjusted vertex position. At each iteration, to embed one bit value, the position of the newly traversed vertex is adjusted to modulate the distance from it to the centroid of its traversed neighbors. So the number of the embedded bits is equal to the number of the adjusted vertices. Given m vertices in the cover mesh, there will be $m-1$ bit values embedded after the position of the last traversed vertex is adjusted. Then the centroid of the whole mesh is obtained by

$$v_c = \frac{1}{m}\sum_{i=1}^{m} v_i, \tag{8}$$

and the distance from the last traversed vertex v_l to the mesh centroid is calculated by

$$D = \sqrt{(v_{lx}-v_{cx})^2 + (v_{ly}-v_{cy})^2 + (v_{lz}-v_{cz})^2}. \tag{9}$$

The ratio R between D and S is obtained by

$$R = D/S, \tag{10}$$

which will be used in the retrieval process to calculate the modulation step S.

3.2. Message Retrieval

To retrieve the message hidden in the cover mesh, the quantization step S used in the modulation is required. To obtain S, the distance D from the last traversed vertex to the mesh centroid and the parameter R need to be provided. Since the mesh traversal is ordered by the permuted vertex indices I', the secret key K is required to generate I' from the vertex indices I. Therefore, the secret key K and the parameter R are required in the message retrieval.

The detailed process of message retrieval is as follows: At first, the vertex indices I in the cover mesh is permuted by using K as the seed of pseudo-random generator to generate I'. Then the last traversed vertex v_l is found out by performing the mesh traversal ordered by I'. The mesh centroid is calculated by Eq.(8) and the distance D from the last traversed vertex to it is obtained by Eq.(9) as in the embedding process. With the provided parameter R, the quantization step S is obtained by

$$S = D/R. \tag{11}$$

To retrieval the embedded $m-1$ bit values, the mesh traversal is performed again to traverse the vertices one by one. For a vertex v_j, the centroid v_{jc} of its traversed neighbors is calculated by Eq.(2) and the modulated distance d'_j from v_{jc} to v_j is obtained by Eq.(3). With the obtained S, the modulated integer quotient Q'_j is calculated by

$$Q'_j = \lfloor d'_j/S \rfloor, \tag{12}$$

and the bit value w'_j is retrieved by

$$w'_j = Q'_j\%2. \tag{13}$$

The whole message string $B = (w_i)_{i=1}^{m-1}$ will be retrieved after the last bit is retrieved from the last traversed vertex.

3.3. The Properties of The Embedded Data

As in the aforementioned discussion, the ratio between the distance from a vertex to the centroid of its traversed neighbors and the quantization step remains the same after the cover mesh is translated, rotated, or uniformly scaled, as well as the embedded data. If the mesh topology of the cover mesh is changed, the neighboring information of each vertex will be changed so that the mesh traversal will not be performed correctly. So the embedded data is sensitive to any topological modification made to the cover mesh.

As for the mantissa truncation of vertex coordinate, which is stored as the single-precision floating-point number, we suppose the truncation error is distributed within $(-T, T)$, then the errors introduced to the coordinates of the mesh centroid in Eq.(8) and the centroid of a vertex's neighboring vertices in Eq.(2) are also distributed within $(-T, T)$. The error introduced to d'_j and D will be distributed within $(-2\sqrt{3}T, 2\sqrt{3}T)$, as seen from Eq.(3) and Eq.(9). Based on Eq.(11), we know the error introduced to S is within $(-\frac{2\sqrt{3}T}{R}, \frac{2\sqrt{3}T}{R})$ so that Eq.(12) can be rewritten as

$$Q'_j = \lfloor \frac{d'_j + \Delta d}{S + \Delta S} \rfloor, \tag{14}$$

where Δd and ΔS are the errors introduced to d'_j and S, respectively, with $\Delta d \in (-2\sqrt{3}T, 2\sqrt{3}T)$ and $\Delta S = \frac{\Delta d}{R}$. It can be seen the integer quotient Q'_j will be different from $\lfloor d'_j/S \rfloor$ if $d'_j\%S + \Delta d - \lfloor d'_j/S \rfloor \times \frac{\Delta d}{R} \notin (0, S)$. Note a component e_j, which is randomly distributed within $(-\frac{S}{2}, \frac{S}{2})$, is added in the modulation of d_j in Eq.(6) so that $d'_j\%S \in (0, S)$. In this way, the quantization step S is statistically undetectable and the formula $Q'_j\%2 = w_j$ is not affected by the adding of e_j. To allow the truncation of the vertex coordinates, e_j should be distributed within a smaller interval, e.g. $(-\frac{S}{3}, \frac{S}{3})$, so that $d'_j\%S \in (\frac{S}{6}, \frac{5S}{6})$ instead of $(0, S)$. As a result, Q'_j in Eq.(14) will be identical to $\lfloor d'_j/S \rfloor$ if $|\frac{R-\lfloor d'_j/S \rfloor}{R} \times \Delta d| < \frac{S}{6}$, i.e., $T < \frac{RS}{12\sqrt{3}|R-\lfloor d'_j/S \rfloor|}$. Hence, the mantissa truncation of vertex coordinates is allowed if

$$T < \frac{S}{12\sqrt{3}|1 - \frac{\lfloor d'_j/S \rfloor}{R}|}. \tag{15}$$

Otherwise, the embedded data will probably be changed.

For the geometrical modifications other than translation, rotation and uniformly scaling, we take for instance the case that one vertex is modified with v'_j and v'_{jm} as its positions before and after the modification. From Eq.(3), we can see the modulated distance d'_j from the modified vertex to its traversed neighbors will be changed by the modification so

Table 1. The mesh models used in the experiments

Models	meshes	vertices	faces	pieces[a]	Capacity(bits)[b]
fish	1	742	1408	8	741
teapot	5	1631	3080	18	1627
dog	48	7616	13176	88	7568
rhino	90	8762	16031	101	8086
horse	31	10316	18359	119	10285

[a] For the method for public authentication.
[b] For the method for content annotation.

we can denote it as $d'_j + \Delta d$ with Δd as the introduced error. Suppose the quantization step S obtained from Eq.(11) is not affected by the modification, the integer quotient Q'_j obtained from Eq.(12) will be changed if $|\Delta d| > \frac{S}{6}$ given $d'_j \% S \in (\frac{S}{6}, \frac{5S}{6})$. For the untraversed neighbors of the modified vertex, i.e., those vertices regarding the modified vertex as their traversed neighbor, the modulated distances to the centroids of their traversed neighbors will also be changed by the modification. In summary, the data embedded by adjusting the positions of the modified vertex and its untraversed neighbors will probably be altered by the modification.

4 Experimental Results and Comparison

4.1 The Method for Public Authentication

We implemented the method for public authentication on several mesh models listed in Table 1. In the experiments, the SHA-1 hash function and the RSA algorithm were used to produce the hash value and generate the signature, respectively. For those models that consist of only a few hundred of vertices, the last four bits of Mantissa field, i.e. the bits with index from 0 to 3 within the floating-point number, were replaced by the signature, so that the mesh model could be divided into pieces with at least 86 vertices. In this case, the introduced relative error is no more than 2^{-19}. For large meshes that consist of much more vertices, we can also use the last bit of Mantissa field to carry the signature, then each piece should consist of at least 342 vertices. The original and the signed mesh models “dog” are shown in Figure 3. It can be seen that the embedding of signature within the mesh model has introduced no perceptible effect.

4.1.1 Tampering Detection

We modified the geometry, topology and other properties of the signed mesh model, respectively. To authenticate the mesh model, the embedded signature was retrieved from the LSBs of vertex coordinates and the public key was used to decrypt the value from the retrieved signature. A new hash value was also produced by setting the LSBs of vertex coordinates as zeroes. The two values were compared with each other within each piece. The authentication results obtained from the experiments show that all those pieces affected by the tampering can be identified, for the two values generated from them do not match with each other. Since it is almost unfeasible to compute the private key from the public key, the security of the public-key authentication scheme is guaranteed.

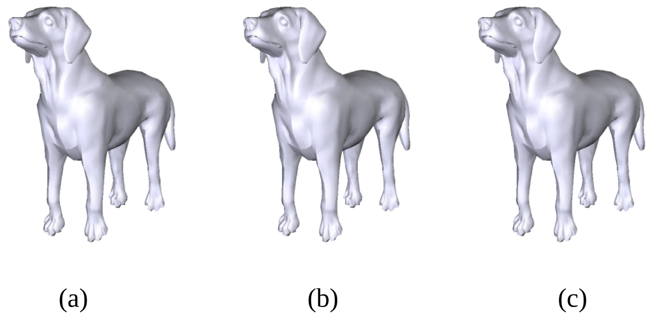
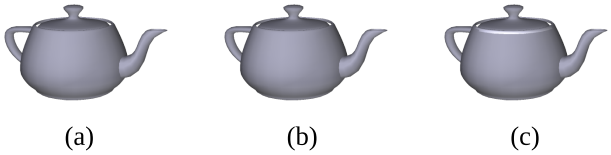


Figure 3. (a). The original mesh model “dog"; (b). The signed dog model with only the least significant bits (index 0) of vertex coordinates are replaced by the signature and (c). The signed dog model with the four bits (index from 0 to 3) of vertex coordinates are replaced by the signature.

4.1.2 Tampering Localization

In the authentication process, the mesh model was partitioned into pieces to authenticate them individually. If there is tampering made to the mesh model, those pieces that have been modified will be found out so that the tampering is localized. If only the geometry, i.e., the positions of vertices, has been modified, only those pieces containing the modified vertices are affected, as the signature of each piece is independent from the geometrical information of other pieces. It is more complex to deal with topology and other properties of the mesh model. Since the pieces share some topological information, i.e., vertices in one piece may be connected with vertices in the neighboring pieces, if the topology between the pieces are modified, all those neighboring pieces will be affected.

In Figure 4, the original mesh model “teapot”, as shown in Figure 4(a), was signed using the proposed scheme by replacing the four bits (with index from 0 to 3) of vertex coordinates with the generated signature within each piece.



(a) (b) (c)

Figure 4. (a). The original mesh model “teapot"; (b). The signed teapot model with the four bits (index from 0 to 3) of vertex coordinates are replaced by the signature; (c). After one vertex position of the signed teapot model was slightly modified, the piece containing the modified vertex is identified and highlighted.

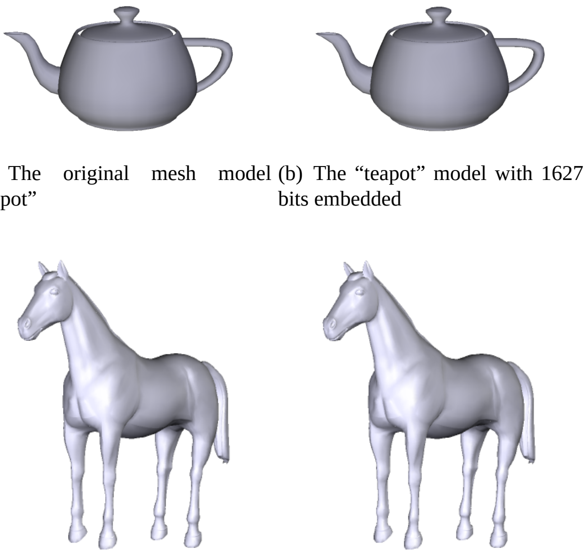
The signed “teapot” model, as shown in Figure 4(b), was slightly tampered by modifying one vertex within it. After authenticating the modified model with the public information, the piece containing the modified vertex was identified and highlighted, as shown in Figure 4(c).

4.2 The Method for Content Annotation

We also performed the method for content annotation on the models in Table 1. The embedded data was unaffected by the translation, rotation and uniformly scaling of the cover mesh. To allow the truncation of vertex coordinates in a range, the component e_j in the modulation was chosen to be distributed within $(-\frac{S}{3}, \frac{S}{3})$ without disclosing the quantization step S. If $1/10000$ of the largest dimension D_m of the mesh model, which is defined as the distance from the furthest vertex to the mesh centroid, was assigned to S, truncating of 8 least significant bits (LSB) of each vertex coordinate was allowed. If $1/100000$ of D_m was assigned to S, only 5 LSBs of each vertex coordinate could be truncated without changing the embedded data.

4.2.1 Distortion of the Cover Mesh

In the experiments, the quantization step S should be carefully chosen to satisfy the fidelity requirement. Since the modulation of the distance from a vertex to its traversed neighbors is bounded by $(-\frac{4S}{3}, \frac{4S}{3})$ given $e_j \in (-\frac{S}{3}, \frac{S}{3})$ in Eq.(6), the adjustment of each vertex position can be numerically bounded. From Eq.(7), we can conclude that $|v'_j - v_j| = |d'_j - d_j| < \frac{4S}{3}$, which means the adjustment of each vertex is in the sphere with $\frac{4S}{3}$ as its radius. It is not hard to conclude that the distortion is proportional to the quantization step S used in the modulation. Upon the fact that the mesh topology has not been changed, the distance from the adjusted vertex to its former position is used to represent the distortion introduced to the cover mesh. In the experiments, if $1/10000$ of D_m is assigned to S, the largest error, which is defined as the longest distance among all the adjusted vertices normalized by D_m, never exceeded 1.4×10^{-4}. If $1/100000$ of D_m is assigned to S, the largest error never exceeded 1.4×10^{-5}. The mesh models “teapot” and “horse” before and after the embedding process are shown in Figure 5.



(a) The original mesh model “teapot”

(b) The “teapot” model with 1627 bits embedded

(c) The original mesh model “horse”

(d) The “horse” model with 10285 bits embedded

Figure 5. Totally 1627 and 10285 bits were hidden within the mesh model “teapot” and “horse”, respectively, by choosing $1/10000$ of the largest dimension of the mesh model as the quantization step S and randomly choosing e_j from $(-\frac{S}{3}, \frac{S}{3})$.

4.2.2 Capacity

Given m vertices in the cover mesh, the capacity of our method will be $m - 1$ bits, which is much higher than 0.5 bit/vertex in [14]. If a mesh model consists of M separate meshes as in Table 1, the capacity will be $m - M$ bits since the first indexed vertex within each mesh is traversed without adjusting its position.

4.2.3 Security

The security of the method for content annotation relies on the secrecy of the secret key K and the undetectable quantization step S. Given there are m vertices in a mesh model, the number of the possibly permuted vertex indices is $m!$.

Without the secret key K, the mesh traversal must be performed $m!$ times to guarantee the embedded data can be retrieved given the accurate quantization step S.

4.2.4 Complexity

The method is performed in the spatial domain and applicable to all polygonal meshes. The vertex indices permutation, the mesh traversal and the embedding/extraction operations are all straightforward. The runtime of the embedding and retrieval processes for the "teapot" model were only 0.485 seconds and 0.578 seconds in a 2.66G Pentium 4 PC with 512MB RAM, while those for the "horse" model were 18.609 seconds and 25.078 seconds, respectively.

5 Concluding Remarks and Future Work

Two methods have been proposed for public authentication and content annotation of 3D mesh models, respectively. The basic idea of the first one is to embed the signature within the mesh model so that only the algorithms and the public key irrelative to the specific mesh model are required for authentication application. Once the authenticity of the public key is confirmed, the authenticity of the mesh content can be verified. Additionally, the mesh partitioning algorithm is implemented to divide the mesh into pieces with a fixed amount of vertices. Since the generation, embedding and retrieval of signature are performed within each piece individually, the possible tampering made to the mesh model can be localized into the certain piece.

The second method is high-capacity to embed data in polygonal meshes for content annotation. By modulating the distance from a vertex to its traversed neighbors, the embedded data is invariant to translation, rotation and uniformly scaling of the cover mesh, as well as the mantissa truncation of vertex coordinates within a range. A tradeoff between the allowable range of coordinate truncation and the distortion of the cover mesh can be achieved by assigning an appropriate value to the quantization step used in the modulation. By keep the secret key K and the parameter R from public access, the proposed method is considerably secure to carry annotation information within 3D meshes.

However, there are some unclear aspects in the second method, whether it can be used for authentication of 3D mesh models with the secret key and the parameter R for instance. As a matter of fact, the tampering on one vertex affects the data embedded in the positions of itself and those regarding it as their traversed neighbors so that tampering localization can be easily achieved. Therefore, the security of using dithered modulation for fragile watermarking needs to be investigated, i.e., can it be made as secure as cryptography? If it cannot, what is the reason; and if it can, how can we achieve it? From a general point of view, the security of fragile watermarking should be systematically studied.

References

[1] M. Levoy, K, Pulli, B, Curless, S, Rusinkiewicz, D, Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk, "The Digital Michelangelo Project: 3D Scanning of Large Statues," *Proc. ACM SIGGRAPH,* pp. 131-144, 2000.

[2] The protein data bank, http://www.rcsb.org/pdb/.

[3] O. Benedens, "Geometry-based watermarking of 3-D models," *IEEE Comput. Graph., Special Issue on Image Security,* pp. 46-55, Jan./Feb. 1999.

[4] O. Benedens and C. Busch, "Toward blind detection of robust watermarks in polygonal models," *Proc. EUROGRAPHICS Comput. Graph. Forum,* vol. 19, pp. C199-C208, 2000.

[5] B. L. Yeo and M. M. Yeung, "Watermarking 3-D objects for verification," *IEEE Comput. Graph. Applicat.,* pp. 36-45, Jan./Feb. 1999.

[6] E. Praun, H. Hoppe and A. Finkelstein, "Robust mesh watermarking," *Proc. ACM SIGGRAPH,* pp. 69-76, 1999.

[7] R. Ohbuchi, H. Masuda and M. Aono, "Watermarking Three-Dimensional Polygonal Models Through Geometric and Topological Modifications," *IEEE J. Select. Areas Commun.,* vol. 16, pp. 551-560, Apr. 1998.

[8] R. Ohbuchi, S. Takahashi, T. Miyasawa and A. Mukaiyama, "Watermarking 3-D polygonal meshes in the mesh spectral domain," *Proc. Graphics Interface,* pp. 9-17, Ottawa, ON, Canada, June 2001.

[9] H. Date, S. Kanai and T. Kishinami, "Digital watermarking for 3-D polygonal model based on wavelet transform," *Proc. ASME Des. Eng. Techn. Conf.,* Sept. 12-15, 1999.

[10] M. G. Wagner, "Robust Watermarking of Polygonal Meshes," *Proc. Geometric Modeling & Processing 2000,* pp. 201-208, Hong Kong, April, 2000.

[11] K. Yin, Z. Pan, J. Shi and D. Zhang, "Robust mesh watermarking based on multiresolution processing," *Computers & Graphics,* vol. 25, pp. 409-420, 2001.

[12] F. Cayre and B. Macq, "Data hiding on 3-D triangle meshes," *IEEE Trans. Signal Processing,* vol. 51, pp. 939-949 (4), 2003.

[13] A. Kalivas, A. Tefas and I. Pitas. “Watermarking of 3D Models using Principal Component Analysis,” *Proc. ICASSP*, vol. 5, pp.676-679, 2003.

[14] Y. Maret and T. Ebrahimi, “Data Hiding on 3D Polygonal Meshes,” *Proc. ACM Multimedia & Security Workshop*, pp. 68-74, Magdeburg, Germany, 2004.

[15] F. Uccheddu, M. Corsini, and M. Barni, “Wavelet-based blind watermarking of 3d models,” *Proc. ACM Multimedia*, pp. 143-154, New York, 2004.

[16] Wu H.T. and Cheung Y.M., “A New Fragile Mesh Watermarking Algorithm for Authentication,” *IFIP 20th International Information Security Conference*, pp. 509-523, Chiba, Japan, 2005.

[17] H. T. Wu and Y. M. Cheung, “A Fragile Watermarking Scheme for 3D Meshes,” *Proc. ACM Multimedia & Security Workshop*, pp. 117-123, New York, 2005.

[18] T. Kalker, “Considerations on watermarking security,” *IEEE Int. Workshop on Multimedia Signal Processing*, pp. 201-206, Cannes, France, October, 2001

[19] The Web3D Consortium, http://www.vrml.org/

[20] B. Chen and G.W. Wornell, “Dither modulation: a new approach to digital watermarking and information embedding,” *Proc. SPIE: Security and Watermarking of Multimedia Contents*, vol. 3657, pp. 342-353, San Jose, January 1999.

[21] B. Schneier, “Applied Cryptography: Protocols, Algorithms, and Source Code in C,” *2nd ed. John Wiley & Sons, Inc.*, 1996.

[22] A. J. Menezes, P. C. Van Oorschot and S. A. Vanstone, “Handbook of Applied Cryptography,” *CRC Press, Inc.*, Boca Raton, 1997.

[23] P. W. Wong, N. Memon N., “Secret and public key image watermarking schemes for image authentication and ownership verification,” *IEEE Trans on Image Processing*, Vol.10, pp. 1593-1601, 2001.

[24] U.S. Dept. of Commerce/NIST, “Secure Hash Standard,” *FIPS publication*, 180-1, 1995.

[25] R. L. Rivest, A. Shamir and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Commun. ACM*, Vol.21, pp.120C126, Feb. 1978.

[26] ANSI/IEEE Standard 754-1985, “Standard for Binary Floating Point Arithmetic.”

[27] R. L. Rivest, “RFC1321: The md5 message-digest algorithm,” *Internet Activities Board*, 1992.

[28] S. Ichikawa, H. Chiyama and K. Akabane, “Redundancy in 3D Polygon Models and Its Application to Digital Signature,” *Journal of WSCG (Special Issue WSCG’2002 - FULL papers - 10th Int’l Conf. Central Europe on Computer Graphics, Visualization and Computer Vision 2002)*, Vol. 10, No. 1, pp. 225-232, 2002.

Compression driven Partitioned POMDP Solving Via Belief Space

Xin Li

Abstract

While partially observable Markov decision process (POMDP) is a powerful tool for supporting optimal decision making in stochastic environments, solving large-scale POMDPs is known to be computationally intractable. Value Directed Compression (VDC) by reducing the scale of POMDP which is directly based on the problem's structure has recently been shown to be effective in tackling the scale-up issue. Inspired by the fact that further reduction in the belief state dimension can then result in a more efficient POMDP solver, this paper proposes a VDC method in sampled beliefs to enhance compression's efficiency as well as speed up the compression process via a dimension reduction oriented clustering by exploring the sparsity of belief space. With the conjecture that the temporally close belief states should possess a low degree of freedom due to the problem's intrinsic regularity, a minimum spatio-temporal criterion function that measures belief states' spatial and temporal discrepancies is adopted to control the belief clustering.

1 Introduction

Building intelligent agents which can make optimal decisions in a stochastic environment is known to be important and have found useful applications in robot navigation [14], moving target search [5], etc. The problem is equivalent to computing the optimal policy for an agent to decide its next action to perform based on some feedback observed from the environment so as to maximize a long-term reward and at the same time complete a given mission.[1] The stochastic nature of the problem can be due to the uncertainty of the environment and that of the agent's performed actions. The agent policy is a mapping of the observed information to the action to be performed. The ideal scenario of the problem is that the stochastic environment can be accurately modelled by a set of abstracted states with stochastic transitions and the states are fully observed. However, in many real situations, we cannot expect the agent to have full observation of the current world state. In other words, the environment is only partially observable to the agent. This kind of situations makes optimal decision making much more challenging as the agent needs to memorize its inflatable observation history for its action decision.

Markov decision process (MDP) and partially observable Markov decision process (POMDP) are the two known mathematical models representing the problem with fully observable and partially observable environments respectively. MDP defines a model with a set of discrete states and a corresponding stochastic state transition model which embodies the possible state transitions given an action. For POMDP, due to the partial observability of the world state, it is common to represent an observation history into a belief state which is essentially the probability distribution over the unobservable real states of the history. The belief state is continuous and can be interpreted as a form of internal state summarizing all the information regarding the past. A discrete state POMDP is thus transformed into a continuous state MDP. In the literature, most of the existing algorithms solving the POMDP problems operate on the belief states.

The POMDP's policy is defined over the continuous belief state and thus is a mapping from a continuous state to an action. For large-scale POMDP problems, it is computationally infeasible even though there exist some computational shortcuts (e.g., the witness algorithm [1]) which make use of the piecewise linear and convex (PWLC) property of the POMDP's value function [1]. In the literature, there exist a number of methods proposed to solve large-scale POMDP problems efficiently. The grid-based algorithm [6] approximates the value function by selecting a finite set of grid points and solves the corresponding MDP. The point-based value iteration (PBVI) [6; 7], algorithm optimizes the value function for a set of specific beliefs instead of over the entire belief space. The Bounded Finite State Controllers algorithm (BFSC) searches though the space of bounded-size, stochastic finite state controllers, combining gradient ascent and policy iteration [13], etc.

In this paper, we are inspired by the recently proposed belief compression approach [10] and VDC algorithm[12] argue that compressing a POMDP problem via its sampled beliefs instead of structure itself, and analyzing a sample of be-

[1] Note that the problem can be extended to situations with multi-agents [3] which are decentralized with multiple sources of feedback. This is however not the main focus of this paper.

lief states could in fact provide us a lot of hints for reducing the problem complexity in a problem-specific manner. Belief compression uses the fact that belief states of POMDP can typically be characterized by a much lower dimensional state space, and adopts dimension reduction techniques to reduce the problem complexity. With the conjecture that temporally close belief states could form clusters with further reduced intrinsic dimensions, we propose to cluster belief states based on their spatial (in the belief space) and temporal differences. The resulting belief state clusters can then induce a much more compressed POMDP than using VDC directly.

The rest of the paper is organized as follows. Section 2 reviews the mathematical frameworks of MDP and POMDP. Section 3 analyze some related works inspiring our current work. Section 4 proposes the VDC method in Belief space and corresponding deduction. Section 5 describes the proposed method combining with Perseus. Section 6 provides the intuition and techniques of clustering belief states. Policy computation and application are reported in Section 7 with possible extensions included in Section 8. Section 8 concludes the paper.

2 Partially observable Markov decision process

An MDP model is characterized by a finite set of states $\mathcal{S}$, a finite set of actions $\mathcal{A}$, a set of corresponding state transition probabilities $T : \mathcal{S} \times \mathcal{A} \rightarrow \Pi(\mathcal{S})$, and a reward function $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$. It is a Markov process as the state transition probabilities $T(s_i, a, s_j) = P(s_j|s_i, a)$ depend only on the current state but not the history of predecessor states. Solving an MDP problem means finding an optimal policy $\pi : S \rightarrow A$ which maps each state to an action so as to achieve the best long-term reward, mostly characterized by a value function $V : \mathcal{S} \rightarrow \mathbb{R}$ which computes the expected total reward over time. The optimal policy can be obtained via iterations using Eq.(1) and (2).

$$V(s_i) = max_a[R(s_i, a) + \gamma \sum_{s_j} T(s_i, a, s_j)V(s_j)] \quad (1)$$

$$\pi^*(s_i) = argmax_a[R(s_i, a)+\gamma \sum_{s_j} T(s_i, a, s_j)V(s_j)] \quad (2)$$

One of the most important assumptions in MDP is that the state of the environment is fully observable. This, however, is unfit to a lot of real-world problems. Partially observable Markov decision process (POMDP) generalizes MDP in which the decision process is based on incomplete information observed about the state. A POMDP model is essentially equivalent to that of MDP with the addition of a finite set of observations $\mathcal{Z}$ and a set of corresponding observation probabilities $O : \mathcal{S} \times \mathcal{A} \rightarrow \Pi(\mathcal{Z})$, where $O(s_j, a, z) = P(z|s_j, a)$ is the probability of getting observation z given that the agent took action a and landed in state s_j. Solving a POMDP problem typically makes use of the belief state concept. A belief state is defined as a probability mass function of the current state, given as $b = (b(s_1), b(s_2), ...b(s_{|S|}))$, where $s_i \in \mathcal{S}, b(s_i) \geq 0$, $\sum_{s_i \in \mathcal{S}} b(s_i) = 1$. Thus, a standard POMDP model is described by a tuple $< \mathcal{S}, \mathcal{A}, \mathcal{Z}, T, O, R >$, usually companied with two more parameters the discount factor $\gamma \in [0, 1]$ and the initial belief state b_0. Using the belief state, one can transfer a POMDP model to a belief state MDP.

Assume that the current belief state is b_t. The next belief state b_{t+1} after taking an action a and receiving the observation z can be computed as

$$\begin{aligned} b_{t+1}(s_j) &= P(s_j|z, a, b_t) \\ &= \frac{P(z|s_j, a, b_t)P(s_j|a, b_t)}{P(z|a, b_t)} \\ &= \frac{P(z|s_j, a) \sum_{s_i \in \mathcal{S}} P(s_j|s_i, a, b_t)P(s_i|a, b_t)}{P(z|a, b_t)} \\ &= \frac{P(z|s_j, a) \sum_{s_i \in \mathcal{S}} P(s_j|s_i, a)b_t(s_i)}{P(z|a, b_t)} \\ &= \frac{O(s_j, a, z) \sum_{s_i \in \mathcal{S}} T(s_i, a, s_j)b_t(s_i)}{P(z|a, b_t)} \quad (3) \end{aligned}$$

$$P(z|a, b_t) = \sum_{s_j \in \mathcal{S}} O(s_j, a, z) \sum_{s_i \in \mathcal{S}} T(s_i, a, s_j)b_t(s_i) \quad (4)$$

The reward function can then be computed as $\rho(b_j, a) = \sum_{s_i \in \mathcal{S}} b_j(s_i)R(s_i, a)$. Also, the transition function over the belief states becomes $\tau(b_i, a, b_j) = P(b_j|b_i, a) = \Sigma_{z \in \mathcal{O}} P(b_j|b_i, a, z)P(z|b_i, a) = \Sigma_{z \in \mathcal{O}, b_j = SE(b_i, a, z)} P(z|b_i, a)$, where $b_j = SE(b_i, a, z)$ is defined using Eq.(10) and (4). Naturally we can get the value iteration step Eq.(5) and Eq.(6) analogous to the counterparts of MDP, given as

$$V(b_i) = max_a[\rho(b_i, a) + \gamma \sum_{b_j} \tau(b_i, a, b_j)V(b_j)] \quad (5)$$

$$\pi^*(b_i) = argmax_a[\rho(b_i, a) + \gamma \sum_{b_j} \tau(b_i, a, b_j)V(b_j)]. \quad (6)$$

The value iteration on the belief states is not as straightforward as that of MDP for the number of the belief states is infinite. Fortunately, the value function over the belief space has been proven piecewise linear and convex [1]. This changes the problem into computing the finite intersections of the value hyperplanes. However, the problem is still intractable where the number of states is large.

3 Compression Driven POMDP Solving

There exist some approximation methods which engender the factored POMDP via reducing the problem's dimension using some techniques such as state aggregation, projection to tackle the large-scale POMDP problems with numerous real states or continuous states. In the literature, both linear projection and nonlinear projection have been used in the belief space compression. Generally speaking, the nonlinear projection is much more suitable to dig out the structures of high-dimensional data set than the traditional linear projection, however the induced value functions PWLC properties loss makes the nonlinearly factored POMDP's optimal policy computation more challenging. Some detailed analysis about recently proposed compression driven methods via linear projection and nonlinear projection which inspire our work are discussed as follow.

Belief compression is a recently proposed paradigm [10], which reduces the sparse high-dimensional belief space to a low-dimensional one via projection. The principle behind is to explore the redundancy in computing the optimal policy for the entire belief space which is typically sparse. Using a sample of belief states computed based on observations of a specific problem, data analysis techniques like exponential principal component analysis (EPCA) can be adopted for characterizing the originally high-dimensional belief state space using a compact set of belief state bases. This paradigm has been found to be effective in making some POMDP problems tractable. However, the transformation between high-dimensional space and low-dimensional space is a non-linear one, which makes the value function of the projected belief states no longer piecewise linear. The consequence is that many existing algorithms taking the advantage of the piecewise-linear value function become not applicable together with belief compression. As suggested in [10], those sampled belief states in the projected space can be used as the states of a correspondingly formed MDP. One can then compute the policy for the associated MDP. However the high performance policy achieving is quite sensitive to the intrinsic property of problems, which is the bottleneck for the optimal or near optimal policy computing though the nonlinear projection could achieve much better low-dimensional representation. Tending to decompose the original problem with belief clustering is another direction to speed up the problem solving, which use a more accurate and much lower intrinsic dimension representation [17] than non-clustering case to compensate the value loss of associated MDP.

The value directed compression (VDC) algorithm [12] is a linear compression algorithm which uses the *Krylov* iteration to explore the POMDP problem's structure achieving the lossless value directed reduced POMDP and corresponding parameters such as compressed reward function and transition functions. The linearly reduced POMDP can be handled by any existing favorite algorithms, which speeds up the policy solving as well as enhances the policy performance. However the *Krylov* iteration could not guarantee the compression efficiency, that is to say, for some POMDP problems, the computed "reduced" POMDP has the same size with the original problem after the *Krylov* iteration. Therefore, the truncated *Krylov* iteration and alternating optimization are introduced by [12] to acquire a forcibly compressed POMDP. These approximations focus on minimizing the errors between high dimensional functions and low dimensional ones instead of dealing with the sparsity of belief space. The compression quality is still limited. Because VDC is a kind of compression over the whole belief space via *krylov* space analysis on the reward function and transition functions, the clustering driven decomposition can not be used directly to combine with it like the belief compression case mentioned before, even we have the same conjecture that the clustered beliefs owns the much lower intrinsic dimension representation.

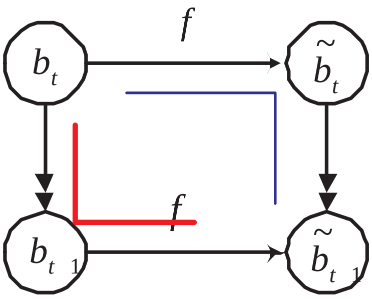


Figure 1: Transformations between beliefs

4 Value Directed Compression Over Sampled Belief States

In this section, we propose a VDC method in sampled belief states. The intuition is that we want to keep the PWLC property to make compressed POMDP fit to any our favorite algorithms as well as to achieve the low-dimensional representation as compact as possible which benefit from the belief space sparsity analysis. Instead of using the *Krylov* iteration to compress the problem, we execute the divergence based nonnegative matrix factorization (NMF)[4] on the sampled belief set to get the reduced dimension belief space and deduct the corresponding functions in this space. Recalling the NMF method, $V \approx WH$, each column vector of the nonnegative matrix V can be explained as the weighted sum of the vectors in nonnegative matrix W, here the matrix H is the weight matrix. To compare with some standard dimension reduction techniques like PCA, NMF can guarantee all the elements of reduced and reconstructed belief state to be positive, which is important as each belief state is a probability distribution by itself.

Let S denote the set of true states, $\mathbb{B}$ denote the belief state space of dimension $|S|$, $b \in \mathbb{B}$ denote the belief state where its j^{th} element $b_i(j) \geq 0$ and $\sum_{j=0}^{|S|} b_i(j) = 1$, B denote a $n \times |S|$ matrix defined as $[b_1|b_2|...|b_n]^T$ where n is the number of belief states in the training sample.

one can apply NMF and obtain a $|S| \times l$ transformation matrix F which factors B into the matrices F and $\widetilde{B}$ such that

$$B \approx \widetilde{B}F \tag{7}$$

where each row of B equals $b \approx b^r = \widetilde{b}F$ and the dimension of $\widetilde{B}$ is $n \times l$. As the main objective of F is for dimension reduction, it is typical that $l << |S|$. The deductive functions in low dimensional space is given as below

$$\begin{aligned} V(b) &= \sum_{s_i} b(s_i) \cdot R(s_i, a) \\ &= bR(:, a) \\ &= \widetilde{b}FR(:, a) \end{aligned} \tag{8}$$

$$
\begin{aligned}
V(\widetilde{b}) &= \widetilde{b}\widetilde{R}(:,a) && (9)\\
\widetilde{R} &= FR && (10)
\end{aligned}
$$

The deduction of low dimensional reward function is due to the fact that we keep the expected value unchanged over the two spaces during the transformation. The deduction of the low dimensional transition function is based on the supposition that the two path (red path and blue path see Figure 1) would reach the same belief.

$$
\begin{aligned}
\widetilde{b}^T_{t+1} &= \widetilde{\widetilde{SE}}(\widetilde{b}_t, a, z) && (11)\\
&= \widetilde{\alpha}\widetilde{G}^{<a,z>}\widetilde{b}^T_t\\
b^T_{t+1} &= F^T\widetilde{b}^T_{t+1} && (12)\\
&= SE(b_t, a, z)\\
&= \alpha G^{<a,z>} b^T_t\\
&= \alpha G^{<a,z>} F^T \widetilde{b}^T_t\\
F^T\widetilde{\alpha}\widetilde{G}^{<a,z>}\widetilde{b}^T_t &= \alpha G^{<a,z>} F^T \widetilde{b}^T_t && (13)\\
\widetilde{G}^{<a,z>} &= pinv(F^T)G^{<a,z>}F^T && (14)
\end{aligned}
$$

5 Point Based Value Iteration

As mentioned before, the linear reduction preserves the PWLC properties so that the reduced POMDP is suitable for any existing POMDP solving algorithm. Here, we choose Perseus an efficiently randomized point-based approximate value iteration algorithm [15] to solve the reduced POMDP combining with the VDC in Belief Space. The combination is straightforward because of the complete low dimensional function achievement. Note that our belief set is sampled following some specific policy induced trajectory. Meanwhile, Perseus is also a trajectory-based approach. Therefore, we suggest to take the sampled beliefs as the backup belief set in Perseus to execute the point-based value iteration. Intuitively, it will make the belief space analysis (see Section 6) consistent with the backup beliefs which effect the value iteration.

6 Clustering Belief States for Belief Space Decomposition

6.1 Dimension Reduction Oriented Clustering

For the compression driven POMDP solving method, the policy's performance depends on the efficiency of belief compression greatly. To further exploit the dimension reduction paradigm, we propose to decompose the belief space by analyzing the manifold of a set of sampled belief states for clustering. We anticipate that in those cases, there should exist some clusterings which could result in more substantial dimension reduction per cluster when compared with that of the overall belief states. This idea can be intuitively interpreted as exploitation of the the statistical properties of belief state.

6.2 A Minimum Spatio-Temporal Criterion Function for Clustering

In this paper, we propose to cluster the belief states based on both their euclidean distance as well as their temporal difference, with the conjecture that regularities should be easier to identify for temporally close belief states. The temporal information is considered since in some real problems especially the navigation problems, agent would be wandering in some regions (over the real states) in most cases, meanwhile the belief states occur mass in these regions. Due to the coding of states, the physically near states might be far away in the relative positions of belief states representation. For example, a highly assured navigating agent (the current belief states occurs one mode) evolve its belief states after taking an action (e.g. move forward, turn right), now the agent must locate in the real state near the last step state, however, the mode of the updated belief may switch to an area far away from the former mode under the given specific coding. Under this situation, one can not guarantee to use spatial criterion to cluster these two beliefs together without the help of temporal information. Among all the clustering algorithms, the k-means algorithm [9] is here chosen just for the simplicity reason. It bases on a function defined for measuring the distance between the cluster means and each data item. Data found to be closest to one of the cluster means will contribute to the update of that mean in the next iteration. The whole process will repeat until it converges. For clustering belief states with the dimension-reduction objective, we define a minimum spatio-temporal distance function between two belief states, given as

$$
\begin{aligned}
dist(b_i, b_j) &= Min(dist_{spatial}(b_i, b_j), dist_{temporal}(b_i, b_j))\\
&= Min(\|b_i - b_j\|, \|\frac{i-j}{\lambda}\|) && (15)
\end{aligned}
$$

where λ is a window's threshold parameter for controlling which kind of distance will dominate the clustering.

As each belief state is a probability distribution, Kullback-Leibler (KL) divergence could be used for evaluating the discrepancy between the original belief states and the reconstructed belief states, as given in Eq.(16).

$$
\overline{KL}(B) = \frac{\sum_{i=1}^{n} KL(b_i \| b_i^r)}{n} \quad (16)
$$

$$
KL(b_i \| b_i^r) = \sum_{j=1}^{|S|} b_i(j) \ln\left(\frac{b_i(j)}{b_i^r(j)}\right). \quad (17)
$$

Now the belief state sample be partitioned into P clusters $\{C_1, C_2, ..., C_P\}$ and the weight matrix of the p^{th} cluster C_p to be F_p.

7 Policy Computation and Application

Now, the policy computation will operate independently on each the clustered belief sets $\{C_1, C_2, ..., C_P\}$ and get the corresponding functions, then the policies on each low dimensional $\widetilde{C}_p$ are achieved by Perseus. Note that Perseus computes the α vectors (optimal value function) over the belief space, so for the un-partitioned compressed POMDP's policy application, the current belief should be compressed directly with its compression matrix F then be sent to those α vectors for the best action selection. In our partitioned case, one can use KNN method to determine the new high-dimensional belief's belonged cluster, or vote the best one for the all actions computed from the all clusters.

8 Discussion and Future Works

This paper mainly demonstrates the possibility of clustering the belief states in a spatio-temporal manner to achieve further belief state compression and good policy performance. We are currently working on several extensions of this work as depicted as follows.

8.1 Online POMDP

What being described so far assumes that the whole model of the decision process is known. That is, we have the perfect knowledge about the reward function, transition function and observation function. Solving the corresponding POMDP problems is an off-line process. It is also interested to see how the multi-agent approach can be extend to support online learning (e.g., Q-learning [16]) for POMDP under partial observation scenarios.

8.2 The Multi-Agent Consideration

As the decomposition based on the proposed belief clustering may not result in a set of sub-POMDP problems which are equivalent to the original POMDP problems, interaction between those agents for achieving the overall optimal policy is an important research issue. Nash equilibrium is an important concept commonly used in multi-agent learning [8] for solving decentralized MDP [2] and POMDP problems [11]. Our research agenda also includes how to apply this paradigm to the our decomposition scheme. One possibility is that every agent would conjecture other agents' behaviors and give the best response to other agents from its local view. A Nash equilibrium usually would not deduce the optimal policy, but a not-too-bad sub-optimal solution could still be guaranteed in most of the cases.

What being described so far assumes that the whole model of the decision process is known. That is, we have the perfect knowledge about the reward function, the state transition function and the observation function. Solving the corresponding POMDP problems is an off-line process. It is also interested to see how the proposed method can be extended to support online learning (e.g., Q-learning [16]) of POMDP under the partial observation scenario.

8.3 Towards Optimal Spatio-Temporal Clustering

While the criterion function used in this paper has shown to be effective empirically, it is by no means an optimal choice. In addition, we still lack automatic mechanisms (other than exhaustive search) for setting the parameters to govern the clustering. We believe that this is an immediate and important research direction to be pursued in the future.

8.4 Towards accurately Nonlinear compressed POMDP Solving

The non-linear compression for POMDP is another branch for POMDP solving. For some problems, non-linear compression achieves more accurate and lower dimensions representation than the traditional linear compression [**?**], however the more persuasive and powerful solving methods on the non-linear reduced structure are still lacking. Exploring the more accurately nonlinear solving is a hopefully achievable direction.

9 Conclusion

This paper extends the recently proposed value directed compression by applying it on the sparse belief space to achieve much more efficient and effective reduction on the original problem and introduce a minimum spatio-temporal belief state clustering to address large-scale POMDP problems. Future research directions include further enhancement of accurately nonlinear compressed POMDP solving, online POMDP solving and optimal Spatio-Temporal Clustering.

References

[1] A.Cassandra. *Exact and approximate algorithms for partially observable Markov decision processes*. U.Brown, 1998.

[2] R. Becker, S. Zilberstein, V. Lesser, and C. V. Goldman. Transition-Independent Decentralized Markov Decision Processes. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 41–48, Melbourne, Australia, July 2003. ACM Press.

[3] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of markov decision processes. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence (UAI-00)*, pages 32–37, San Mateo, CA, 2000. Morgan Kaufmann Publishers.

[4] D. D.Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 1999.

[5] J. N. Eagle. The optimal search for a moving target when the search path is constrained. *Operations Research*, 32:1107–1115, Sept.-Oct 1984.

[6] M. Hauskrecht. Incremental methods for computing bounds in partially observable markov decision processes. In *Proceedings of the 14th National Conference on Artificial Intelligence, AAAI'97, Providence, Rhode Island, USA*, pages 734–739. AAAI Press, 27–31 1997.

[7] M. Hauskrecht. Value-function approximations for partially observable Markov decision processes. *Journal of AI Research*, 13:33–94, 2000.

[8] M. P. W. Junling Hu. Nash q-learning for general-sum stochastic games. *Journal of Machine Learning Research*, 4:1039–1069, 2003.

[9] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *5th Berkley Symposium on Mathematics and Probability*, pages 281–297, 1967.

[10] N. Roy, G. Gordon and S. Thrun. Finding approximate POMDP solutions through belief compressions. *Journal of Artificial Intelligence Research*, 23:1–40, 2005.

[11] R. Nair, M. Tambe, M. Yokoo, D. Pynadath, and S. Marsella. Taming decentralized POMDPS: Towards efficient policy computation for multiagent settings. 2003.

[12] P. Poupart and C. Boutilier. Value-directed compression of POMDPS. In S. T. S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 1547–1554. MIT Press, Cambridge, MA, 2003.

[13] P. Poupart and C. Boutilier. Bounded finite state controllers. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.

[14] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. *Artificial Intelligence Journal*, 1997.

[15] M. T. J. Spaan and N. Vlassis. Perseus: Randomized point-based value iteration for POMDPs. *Journal of Artificial Intelligence Research*, 24:195–220, 2005.

[16] C. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge Univ., Cambridge England, 1989.

[17] X.Li, W. K.Cheung, and J. Liu. Towards solving large-scale pomdp problems via spatio-temporal brief state clustering. *Proceedings of IJCAI-05 Workshop on Reasoning with Uncertainty in Robotics (RUR-05)*, 2005.

Privacy Measure and Active Learning

Xiaofeng Zhang

Abstract

Privacy Preserving Data Mining attracts more and more research efforts. In a distributed environment, how to protect local data information whereas to carry on global data analysis becomes the topic of this paper. Based on the previous learning-from-abstraction paradigm, the paper first abstractizes distributed local data sources as Gaussian mixture models and then aggregated the abstractions for global analysis. Then, we investigated the use of the normalized negative log likelihood to quantify the degree of privacy protection and proved that the privacy of the local data can still be maintained after going through the abstraction and the aggregation. Furthermore, a greedy-based active search method is given to improve global analysis results. Experiments are evaluated on a synthetic data set of 2000 items evenly distributed on 3 local sources. Results observed show that the proposed privacy measure can protect data privacy in a distributed environment and the active learning way is an effective one.

Keywords: distributed data mining, privacy measure, active learning

1. Introduction

With the proliferation of the Web, Grid and other latest ubiquitous computing platforms, user data are more widely distributed and accessible. This poses a new opportunity and at the same time a great challenge for the deployment of knowledge discovery and data mining (KDD) techniques. One possible methodology is to pool together the data from the distributed sources, and then apply the existing KDD techniques. However, the ever-increasing volume of the daily-generated data and the privacy concern in many application domains, like business and healthcare, sometimes make direct data sharing for global analysis infeasible.

1.1 Privacy-Preserved Data Mining

Recently, KDD with privacy preserving capability has been an area gaining a lot of researchers' attention. For example, secure multiparty computation [9] was proposed for situations where the number of distributed sources is relatively small (due to its high computational requirement) and the global analysis to be supported can be derived from a given set of secure "primitives". Random perturbation [4] is the another approach needed for situations where accessing the original form of the data attributes is mandatory. Also, the generalization approach [6] is ideal for situations where accessing individual data attributes is not needed, but only some global statistical characteristics are the expected outcomes instead. This paper focuses on research issues which are under the category of the generalization approach for privacy preserving KDD.

1.2 Distributed Model-based Data Mining

A common methodology for distributed KDD is to first apply KDD techniques locally and then combine the local results for global analysis. Among the large variety of KDD approaches, we here focus on distributed KDD techniques that are of model-based type. In the literature, examples of distributed model-based data mining (DMDM) include the use of meta-learning for distributed decision tree classifiers [8], combining Bayesian networks, each as an orthogonal basis for modelling an exclusive sub-set of the data attributes (i.e., the data are vertically partitioned) [5, 2], etc.

Recently, the use of the generalization approach to distributed KDD has been proposed [7] where probabilistic abstractions of local data are aggregated and then resampled so as to support the subsequent global model-based clustering or classification. The resampling step involved, however, could be computationally expensive. In our previous work [10], a novel learning-from-abstraction paradigm was first proposed, which does not require data resampling. Instead, a hierarchy of Gaussian mixture models with different numbers of components at each level of the hierarchy was computed locally based on some hierarchical clustering algorithms. Then, local GMM parameters at an agreed level of details were sent to a global server for model-based clustering directly from the local GMM parameters. In [11], this learning-from-abstraction paradigm was shown to be also applicable to high dimensional data manifold unfolding. In principle, the global model can be of any generative model type.

1.3 Paper Contribution and Organization

This paper builds on top of the previous work on learning-from-abstraction DMDM with an in-depth study of the use of a normalized log likelihood for measuring the degree of privacy protection brought by the probabilistic local data abstraction. In particular, the theoretical properties of the privacy measure change 1) when one traverses along the local data abstraction hierarchy, and more importantly, 2) when the local data abstractions leave the local sources and are aggregated at the global server. We managed to show that the privacy protection capability will at least remain unchanged after the abstractions are aggregated. In addition, with the quantification of privacy measure an empirically active learning mechanism can be proposed to improve the global model performance in a sufficient way. We performed detailed performance evaluation of the global models with fixed different local privacy levels and with dynamically changed local privacy levels according to active request of global server.

The remaining of this paper is organized as follow. Section 2 provides an overview of the learning-from-abstraction paradigm for DMDM. Section 3 describes a particular privacy measure adopted as well as its theoretical properties when applied in the DMDM setting. Empirical study of the active learning of global model is showed in Section 4. Experimental results can be found in Section 5 and Section 6 concludes the paper.

2. Learning-from-abstraction - An Overview

Assume that there are totally L local sources. Let $t_i \in D_l$ denote a data item of dimension d at the l^{th} local source, Θ_l denote the set of parameters of the local Gaussian mixture model (GMM) with K_l components for abstracting the l^{th} source. The corresponding probability density function can be written as,

$$p_{local}(t_i|\Theta_l) = \sum_{j=1}^{K_l} \alpha_{jl} p_j(t_i|\theta_{lj}) \tag{1}$$

where $\sum_{j=1}^{K_l} \alpha_{jl} = 1$, and

$$p_j(t_i|\theta_{lj}) = \frac{1}{(2\pi|\Sigma_j|)^{d/2}} \exp\{-\frac{(t_i-\mu_j)^T\Sigma_j^{-1}(t_i-\mu_j)}{2}\} \tag{2}$$

with μ_j and Σ_j being the mean and covariance matrix.

For the purpose of providing efficient way for global analysis, at each local source, Agglomerative hierarchical clustering algorithm (AGH) is applied to build up a hierarchical clustering tree. The basic idea of AGH is to start with one data item as one cluster first. Then, the Euclidean distances between the clusters are computed. The nearest pair of clusters are then merged together to form a new cluster and the between-cluster distances have to be updated. This merging process repeats until one single cluster is left. Given the hierarchical clustering, clusters at each level can be treated as the GMM components. Then the local GMM parameter estimates Θ_l can be aggregated to form an aggregated GMM at the global server. In the following, l is abused and used to be the reindexing of the aggregated local GMMs components, and Eq.(1) also represent the aggregated version of the local GMMs.

The global model, in principle, can be any type of generative model. For instance, GMM can be used as the global model for clustering application [10] and generative topographic mapping (GTM) [1] can be used instead for high-dimension data manifold unfolding [11]. In this paper, details with GMM being the global model are repeated to make it self-contained. Given what one needs to perform at the global server is model-based clustering using a global GMM with M components, the probability density function can be written as,

$$p_{global}(t_i|\Theta_g) = \sum_{k=1}^{M} \alpha_k p_k(t_i|\theta_k).$$

The global GMM's parameters are to be learnt using a modified EM algorithm. Instead of considering the posterior probability that one local data item is generated by a global GMM component as in the E-Step of the conventional EM algorithm, we consider the posterior probability that one local GMM component is generated by a global GMM component instead, given as

$$R_{lk} = p(p_k|p_l) = \frac{p(p_l|p_k)\alpha_k}{\sum_k^M p(p_l|p_k)\alpha_k}$$

where p_k and p_l correspond to the global and local components respectively.

To estimate $p(p_l|p_k)$ for computing R_{lk}, the Kullback Leibner Divergence [3] can be used, resulting in

$$R_{lk} = \frac{\alpha_k \exp\{-\zeta D(p_{global}(t|\theta_k)||p_{local}(t|\theta_l))\}}{\sum_{k=1}^M \alpha_k \exp\{-\zeta D(p_{global}(t|\theta_k)||p_{local}(t|\theta_l))\}} \tag{3}$$

where $D(P||Q)$ denotes the distance between two probabilistic models P and Q. If P and Q are identical, then $D(P||Q) = 0$. $D(p_{global}(t|\theta_k)||p_{local}(t|\theta_l))$ can be derived as

$$\ln \frac{|\Sigma_l|^{\frac{1}{2}}}{|\Sigma_k|^{\frac{1}{2}}} + \frac{1}{2}(trace(\Sigma_l^{-1}\Sigma_k) + (\mu_l-\mu_k)^T\Sigma_l^{-1}(\mu_l-\mu_k) - d).$$

The new M-step can then be given as

$$\mu_k \quad = \quad \frac{\sum_{l=1}^L R_{lk}\mu_l}{\sum_{l=1}^L R_{lk}} \tag{4}$$

$$\alpha_k = \frac{1}{L}\sum_{l=1}^{L} R_{lk} \quad (5)$$

$$\Sigma_k = \frac{\sum_{l=1}^{L} R_{lk}(\Sigma_l + \mu_l\mu_l^T)}{\sum_{l=1}^{L} R_{lk}} - \mu_k\mu_k^T. \quad (6)$$

where μ_l and Σ_l are the local component's mean and covariance matrix, and μ_k are Σ_k are those for the global one.

The E-step and M-step iterate alternatively until the parameter estimates converge.

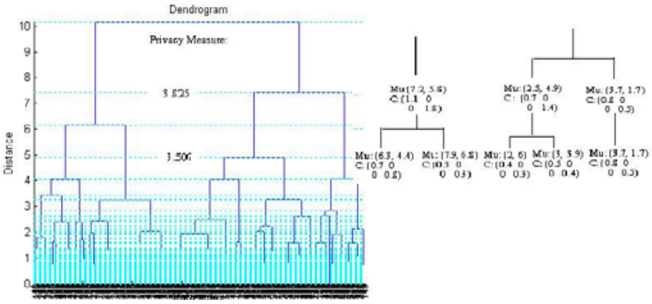


Figure 1. A hierarchy of data abstractions, where a higher level of abstraction is acquired by merging two nearest data sub-groups at the next level and of finer data details.

3. Privacy Control on Local Data Abstraction - A Theoretical Study

In the literature, it is common to discuss privacy in some microscopic sense, say, for instance, whether a particular data record is made known to be public or not (e.g., k-anonymity, l-diversity). For our abstraction-based privacy protection, it makes more sense to understand privacy control in some macroscopic sense which is here referred to as the mechanism for controlling the data details, e.g., in the form of data statistics, to be released. In this section, the emphasis is on macroscopic data privacy controls for application to the hierarchical GMM local data abstraction.

To effectively enforce abstraction-based data privacy policy at the local data sources, a privacy measure that can be objectively interpreted is essential.

Definition 1 (privacy measure)
For a given data set $L := \{t_i\}$, the degree of privacy protection Q_L brought by $P_L(t_i|\Theta^{ML})$ as its abstraction is defined as the normalized negative log likelihood value, given as:

$$Q_L = -\frac{1}{|L|}\sum_{i}^{|L|}\log P_L(t_i|\Theta^{ML}) \quad (7)$$

where Θ^{ML} is the corresponding maximum likelihood (ML) parameter estimates given P_L and L.

A similar privacy measure has been proposed in [7]. Note that it is a normalized one. That is, its value remains the same for data sets of different sizes but following the same distribution. As it is the negative log likelihood, its value will be lower if the probability that the data are generated by $P_L(t_i|\Theta)$ is high. One can interpret the proposed measure as how likely a particular data item (which could be a personal record) can be revealed by presenting the pdf as the data abstraction. So, the lower the privacy measure, the more likely some particular data items can be deduced on average.

As our proposed local data abstraction is a hierarchy of Gaussian mixture models, GMMs with more mixing components should correspond to the cases with more details about the data being revealed and a correct privacy measure should be able to associate them with some lower values of privacy measures. In the following, we first derive some essential properties of the proposed privacy measure on data sets with multivariate Gaussian and then GMM as the abstractions. We then show that the derived properties can help explain how the local data abstraction will behave at different data privacy levels.

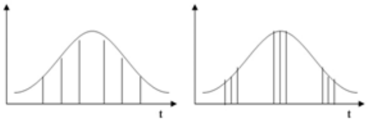


Figure 2. The privacy measures of two different data sets but identical GMM parameter estimates Θ^{ML}.

Lemma 1 (Gaussian-based privacy measure)
Let L be a data set of dimension d and assume that it is abstractized by a multivariate Gaussian with the ML parameter estimates Θ^{ML}, containing the mean μ and the covariance matrix Σ. The privacy measure Q_L can be formulated as

$$Q_L = \log(2\pi)^{d/2}|\Sigma|^{d/2} + \frac{1}{2}. \quad (8)$$

Proof.

$$\begin{aligned} Q_L &= -\frac{1}{|L|}\sum_{i}^{|L|}\log P_L(t_i|\Theta^{ML}) \\ &= -\frac{1}{|L|}\sum_{i}^{|L|}\log\left(\frac{\exp -\frac{1}{2}(t_i-\mu)^T\Sigma^{-1}(t_i-\mu)}{(2\pi)^{d/2}|\Sigma|^{d/2}}\right) \\ &= \frac{1}{|L|}\sum_{i}^{|L|}(\log(2\pi)^{d/2}|\Sigma|^{d/2} + \frac{1}{2}(t_i-\mu)^T\Sigma^{-1}(t_i-\mu)) \end{aligned}$$

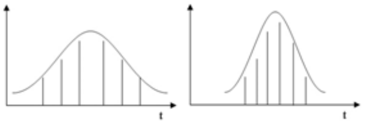


Figure 3. The privacy measures is higher when the data set is more dispersed.

$$= \log{(2\pi)^{d/2}}|\Sigma|^{d/2} + \frac{1}{2}$$

This completes the proof.

Corollary 1 *Data sets abstracted by multivariate Guassians with identical ML parameter estimate will have equal privacy measures.*

Proof. *As the privacy measure Q_L is independent of the data set $\{t_i\}$ but only the ML parameter estimates, it is straight-forward to prove that this corollary is true.*

Corollary 1 states that the privacy measure of a data set abstracted by a multivariate Gaussian depends only on the data sets' statistics (the mean and covariance matrix), but not directly on the exact data items. This can be interpreted as a formal way to state that the privacy measure is a macroscopic one.

Corollary 2 *The privacy measure of a data set L with multivariate Gaussian as its abstraction is higher (lower) when the data set is more (less) dispersed.*

Proof. *As shown in Lemma 1, the privacy measure Q_L is directly proportional to the log of the covariance matrix's determinant (except for a constant $1/2$). As the log function is monotonic increasing, it is obvious to see that Q_L will increase (decrease) when the data sets' covariance matrix determinant increases (decreases), which in turn implies that the data set is more (less) dispersed. This completes the proof.*

Corollary 2 formally states our common intuition that a sharper Gaussian would reveal more information of its representing data items than a flatter one. In other words, it means that there is a higher chance to make a right guess for the existence of a particular data item if the data set can be modelled by a sharper Gaussian.

Lemma 2 (GMM-based privacy measure)
Let L be a data set of dimension d and assume that it is modelled as a Gaussian mixture model of K components with the ML parameter estimates $\Theta^{ML} = \{\Theta_k^{ML}\}$ where Θ_k^{ML} corresponds to the ML parameters for the k^{th} Gaussian component. The lower bound and upper bound of the privacy measures Q_L can be formulated as

$$\sum_k^K \alpha_k Q_k \leq Q_L \leq \sum_k^K \alpha_k(-\log\alpha_k + Q_k) \qquad (9)$$

where K is the number of components in the GMM.

Proof.

$$\begin{aligned} Q_L &= -\frac{1}{|L|}\sum_i^{|L|}\log\sum_k^K \alpha_k P_k(t_i|\Theta_k^{ML}) \\ &\leq -\frac{1}{|L|}\sum_k^K\sum_i^{|A_k|}\log\alpha_k P_k(t_i|\Theta_k^{ML}) \\ &= -\frac{1}{|L|}\sum_k^K(\sum_i^{|A_k|}\log\alpha_k + \sum_i^{|A_k|}\log P_k(t_i|\Theta_k^{ML})) \\ &= -\sum_k^K\frac{|A_k|}{|L|}(\log\alpha_k + Q_k) = -\sum_k^K\alpha_k(\log\alpha_k + Q_k) \end{aligned}$$

and

$$\begin{aligned} Q_L &\geq -\frac{1}{|L|}\sum_i^{|L|}\log P_k(t_i|\Theta_k^{ML}) \\ &= -\frac{1}{|L|}\sum_k^K\sum_i^{|A_k|}\log P_k(t_i|\Theta_k^{ML}) \\ &= \frac{1}{|L|}\sum_k^K|A_k|Q_k = \sum_k^K\alpha_k Q_k \end{aligned}$$

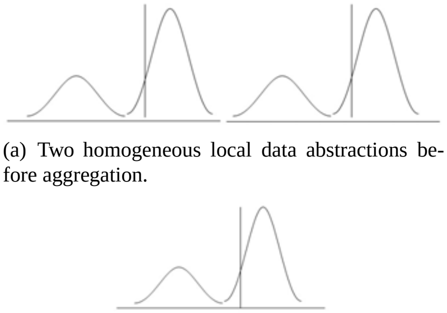
where Q_k is privacy measure of the k^{th} component. This completes the proof.

In particular, when the GMM components are equally weighted with identical covariance matrices, it becomes $Q_k \leq Q_L \leq Q_k + \log K$. *Lemma 2* states that different data sets which can be abstracted by GMMs of the same ML parameters possess the same lower and upper bounds for its privacy measure, and those bounds are independent of the exact sampling of the data. In other words, this implies that it is reliable to use the Q_L to evaluate privacy measure which can be estimated simply based on the abstraction. This is important as the privacy measure of the hierarchical local data abstraction is based on GMM.

Lemma 3 (Lower bound of aggregated privacy) *If L is randomly partitioned into n local data sets $\{A_1, A_2, ..., A_n\}$ in such a way that they all share the same ML GMM abstraction Θ_A^{ML}, the privacy measure of the corresponding abstraction aggregation will reach its lower bound and equal to that of the local abstraction.*

Proof. *1. When $n = 1$, the original data set is not partitioned, the conclusion follows immediately.*
2.When $n > 1$, the privacy measure of L based on the aggregated GMM abstraction, denoted as Q_L, is given as:

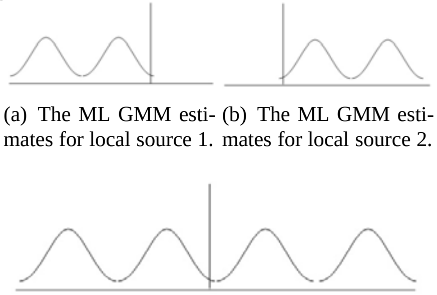
$$Q_L = -\frac{1}{|L|}\sum_i^{|L|}\log\sum_j^n\frac{|A_j|}{|L|}P_{A_j}(t_i|\Theta_{A_j}^{ML})$$



(a) Two homogeneous local data abstractions before aggregation.

(b) The global GMM abstraction after aggregation.

Figure 4. A pictorial illustration of Lemma 3.



(a) The ML GMM estimates for local source 1.

(b) The ML GMM estimates for local source 2.

(c) The global GMM data abstraction after aggregation.

Figure 5. The pictorial illustration of Lemma 4.

$$\geq \quad -\frac{1}{|L|}\sum_{j}^{n}\sum_{i}^{|A_j|}\log P_{A_j}(t_i|\Theta_{A_j}^{ML})$$

The above step holds as the probability density value of a mixture model can only at best be the same as that of its mixing components. The equality holds only when all the local abstractions as mixing components of the aggregated one are identical, where

$$\begin{aligned} Q_L &= -\frac{1}{|L|}\sum_{i}^{|L|}\log\sum_{j}^{n}\frac{|A_j|}{|L|}P_A(t_i|\Theta_A^{ML}) \\ &= -\frac{1}{|L|}\sum_{i}^{|L|}\log P_A(t_i|\Theta_A^{ML}) \\ &= -\frac{1}{|L|}\sum_{j}^{n}\sum_{i}^{|A_j|}\log P_A(t_i|\Theta_A^{ML}) = Q_A \end{aligned}$$

This completes the proof.

Lemma 3 formally states that the privacy measure based on the aggregated GMM local data abstraction is at least the same as that of the homogeneous sources.

Lemma 4 (Upper bound of aggregated privacy) *If L is randomly partitioned into $\{A_1, A_2, ..., A_n\}$, each modelled by their own ML GMM abstractions $\{\Theta_{A_j}^{ML}\}$, the privacy measure after the abstraction aggregation will have a upper bound of equal to:*

$$Q_L = -\frac{1}{|L|}\sum_{j}^{n}(|A_j|\log\frac{|A_j|}{|L|} - |A_j|Q_{A_j}) \qquad (10)$$

Proof. *With the assumption that the GMM abstractions of $\{A_1, A_2, ..., A_n\}$ are different, the privacy measure of L based on the aggregated GMM abstraction, Q_L, is given as:*

$$\begin{aligned} Q_L &= -\frac{1}{|L|}\sum_{i}^{|L|}\log\sum_{j}^{n}\frac{|A_j|}{|L|}P_{A_j}(t_i|\Theta_{A_j}^{ML}) \\ &\leq -\frac{1}{|L|}\sum_{j}^{n}\sum_{i}^{|A_j|}\log\frac{|A_j|}{|L|}P_{A_j}(t_i|\Theta_{A_j}^{ML}) \\ &= -\frac{1}{|L|}\sum_{j}^{n}(\sum_{i}^{|A_j|}\log\frac{|A_j|}{|L|} + \sum_{i}^{|A_j|}\log P_{A_j}(t_i|\Theta_{A_j}^{ML})) \\ &= -\sum_{j}^{n}(\frac{|A_j|}{|L|}\log\frac{|A_j|}{|L|} + \frac{|A_j|}{|L|}Q_{A_j}) \end{aligned}$$

where the equality holds when the local GMM abstractions are infinitely far from each others. This completes the proof. If all the local GMM abstractions are identical, it is interesting to see that Q_L will become $\log n + Q_A$. As n increase (i.e., the data are more distributed), the privacy measure after the abstraction aggregation will increase.

To summarize, the above lemmas show that the degree of privacy protection which has been agreed and evaluated at the local source by the data owner will *not* decrease. This kind of guarantee is important as one will lose control of the data abstractions after it is being shared.

4. Active learning of Global Model - A Empirical Study

With the privacy measure defined before, each local source can be seen to possess same privacy level if the same privacy measure value is given. If local privacy requirements of local sources are predefined, the local abstractions can easily be selected out from the local hierarchy trees for global analysis. Generally, it is high possible that this way of selecting local abstractions can result in bad performance of global model due to the reason that each local source try to maximally protect its privacy. When the global model learned is not accurate enough for global data analysis, there should exist a way to allow it call for more detailed data levels from local sources to improve the global model accuracy. Which local sources should provide more accurate local data abstractions whereas keeping the minimum overall privacy loss at the same time becomes a research issue. This issue can be interpreted as how a global model can largely improve its performance by purposely selecting the least local abstractions increment from some certain sources. The problem differentiate it from the previous local abstractions selection problem by its prominent feature of dynamical property. To solve this problem, we empirically designed some search methods in a greedy way.

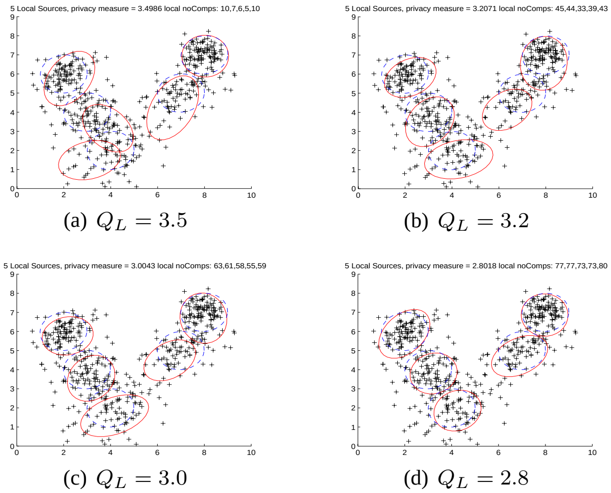
In each step of the basic greedy search (BGS) search, BGS always lower the privacy measure value for the source with the current highest privacy level. The threshold δ denote the value privacy measure changed and is empirically set. The modified greedy search (MGS) firstly computes the likelihood changed of all global models virtually supposing only one local source changes its privacy level each time. The local source with the maximum likelihood change of global model is selected as real candidate to take action to lower its privacy requirement. The two methods immediately stop searching when the global model accuracy is satisfying or a pre-defined iteration of searching is reached.

5. Experiments

The purpose of the experiments is to evaluate the performance of the global model with fixed local privacy levels and with actively selected local privacy levels. The original data set consist of 2000 2-D data points generated by a GMM with 5components. To simplify the problem, 3 local sources were assumed and each with equal size of data items which was randomly assigned. At local sites, hierarchy trees of GMMs were first built with each level associated with a privacy measure value. The abstractions then will be chosen based on this value. Global models learned with different local privacy measures were to be compared with the true data model.

5.1 Clustering Performance vs. Privacy Preservation

Experiments were first performed with different local privacy requirements predefined. Then, we compared the global GMM model learned based on local data abstractions and the true GMM model used to generate the local data sets by computing their KL divergence. The results obtained, as shown in Figure , were found to be consistent to our understanding that higher privacy should lead to less detailed information, and thus not that good clustering results. Figure shows the corresponding clustering results which are again consistent to what we expected, showing that the privacy measure adopted is an effective one.



(a) $Q_L = 3.5$ (b) $Q_L = 3.2$

(c) $Q_L = 3.0$ (d) $Q_L = 2.8$

Figure 6. Comparison between the true GMM (dotted lines) and the GMM learned under different privacy requirements.

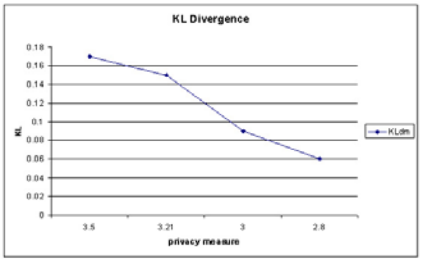
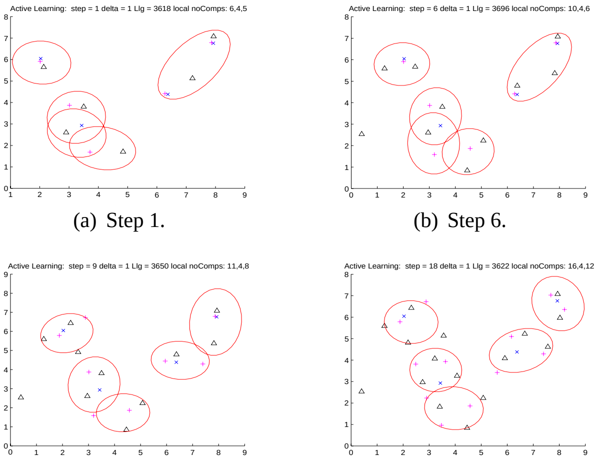


Figure 7. KL-divergence computed between global model acquired based on local abstractions and the true model under different privacy requirements.

5.2 Active Way to Improve Clustering Performance

Experiments were carried on with a set of local abstractions of higher privacy. Correspondingly, the accuracy of global model learned based on this set of abstractions could be poor compared with true model. A simplified version of greedy search was used here instead of SGS and MGS which first select the local source having the highest privacy measure value as candidate. Then the next data level with lower privacy measure in its hierarchy tree was chosen to form a new local abstractions set with original abstractions from other sources. A new global model was learned based on this changed abstractions set. At each step, only one source changes its abstractions. The clustering results of step 1, 6, 9 and 18 are showed in Figure 8. At step 1 shown in Figure 8(a), 3 clusters of global model overlapped together with its local components in each source were $\{6,4,5\}$. Figure 8(b)-(d) showed the global model performance gracefully improved by increasing its local abstractions setting to $\{10,4,6\}$, $\{11,4,8\}$ and $\{16,4,12\}$. Source 2 was seldom selected because the original data set was horizontally partitioned and thus source 2 is under the same distribution with source 1 and 3. Therefore, the effect on global model by changing source 2 could be smaller than that of source 1 and source 3. The empirical results shows it to be an effective active learning way.



(a) Step 1.

(b) Step 6.

(c) Step 9.

(d) Step 18.

Figure 8. The GMM learned by actively selecting lower privacy levels.

6. Conclusion

In this paper, we mainly proposed the way to define privacy measure for distributed privacy preserving data mining and discussed the important properties introduced by this measure in depth. AGH was adopted to build up a hierarchy of GMMs as local abstractions for each source. The normalized negative log likelihood value was computed as the privacy measure. Given a particular privacy measure value, a set of local abstractions can be selected out from the local sources and aggregated for global model learning. Gracefully degrading global clustering results were obtained as the local privacy measure increases. An empirical way is given to actively improve the global model performance. Theoretical study of the active learning needs our future efforts.

References

[1] C. M. Bishop, M. Svensén, and C. K. I. Williams. GTM: The generative topographic mapping. *Neural Computation*, 10(1):215–235, 1998.

[2] R. Chen and S. Krishnamoorthy. A New Algorithm for Learning Parameters of a Bayesian Network from Distributed Data. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002)*, pages 585–588, Maebashi City, Japan, December 2002. IEEE Computer Society.

[3] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New York, 1991.

[4] W. Du and Z. Zhan. Using Randomized Response Techniques for Privacy-Preserving Data Mining. In *In Proceedings of The 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD 2003)*, pages 505–510, Washington, DC, August 2003.

[5] H. Kargupta, B. Park, D. Hershberger, and E. Johnson. Collective Data Mining: A New Perspective Towards Distributed Data Mining. In H. Kargupta and P. Chan, editors, *Advances in Distributed and Parallel Knowledge Discovery*, pages 133–184. MIT/AAAI Press, 2000.

[6] M. Klusch, S. Lodi, and G. L. Moro. Distributed Clustering Based on Sampling Local Density Estimates. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 2003)*, pages 485–490, Mexico, August 2003.

[7] S. Merugu and J. Ghosh. Privacy-preserving Distributed Clustering using Generative Models. In *The Third IEEE International Conference on Data Mining (ICDM'03)*, Melbourne, FL, November 2003.

[8] A. Prodromidis and P. Chan. Meta-learning in Distributed Data Mining Systems: Issues and Approaches. In H. Kargupta and P. Chan, editors, *Advances of Distributed Data Mining*. MIT/AAAI Press, 2000.

[9] J. Vaidya and C. Clifton. Privacy-Preserving K-Means Clustering over Vertically Partitioned Data. In *The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, August 2003.

[10] X. Zhang and W. K. Cheung. Learning Global Models Based on Distributed Data Abstractions. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 2005)*, pages 1645–1646, Edinburgh, August 2005.

[11] X. Zhang and W. K. Cheung. Visualizing Global Manifold Based on Distributed Local Data Abstraction. In *Proceedings of Fifth IEEE International Conference on Data Mining (ICDM 2005)*, pages 821–824, Houston, November 2005.

A New Approach to Mobile Location Estimation within a Radio Cellular Network

Junyang Zhou

***Abstract*— Mobile location estimation or mobile positioning is becoming an important service for a mobile phone network. It is well-known that GPS can provide accurate location estimation. But it is also a known fact that GPS does not perform well in urban areas like downtown New York and cities like Hong Kong. Then many mobile location estimation approaches based on radio cellular networks have been proposed to compensate the problem of the lost of GPS signals in providing location services to mobile users in metropolitan areas. Among different kinds of mobile location estimation technologies, only the class of signal strength based algorithm which estimates the location of mobile stations by signal strength can be applied to different kinds of cellular networks, and therefore, is more general. In this paper, we have designed a directional propagation model – the Modified Directional Propagation Model (MDPM), which makes use of a common signal propagation model to perform location estimation service, and we present a new iterative approach, not the EM algorithm, to provide an estimation of the model parameter. And we also present a formula of the uniform signal propagation model, which includes all the signal propagation models we mentioned in this paper. Then we use a Bayes method to provide the estimation of the MS location based on our statistical model. We have tested MDPM with real data taken in Hong Kong and it is proven that MDPM outperforms other existing location estimation algorithms among different kinds of terrains.**

I. Introduction

Recently, mobile location estimation is receiving considerable attention in the field of wireless communications due to its great potential in different kinds of applications such as logistics, tourism and entertainment. Many positioning technologies have been developed. The Global Positioning System (GPS) is one of the location systems that is mature enough and commercially available [1], [2]. Although GPS is widely used for location estimation, it may not provide accurate result in metropolitan area, like Kong Kong. This is because satellite signals are often reflected, deflected or blocked by high buildings and thus causing inaccurate estimations or no estimation at all. On the other hand, it is also in these populated areas that the radio cellular network is providing an excellent coverage. Hence using a radio cellular network for location estimation could be an alternative approach for mobile positioning. Furthermore, the radio cellular network has a good coverage in most of the populated areas and therefore using an existing radio cellular network for location estimation can be an alternative method for mobile location estimation and is a more economical solution. Moreover, the U.S. Federal Communication Commission (FCC) requires all cellular operators be able to estimate the mobile device locations for Enhanced 911 services in 1996 [3]. Thus, using a radio cellular network for location services has become a popular research topic.

Other proposed location estimation methods including Time-Of-Arrival (TOA), Time Difference Of Arrival (TDOA), Enhanced Observed Time Difference (E-OTD) and Angle-Of-Arrival (AOA) have been proposed for applications [4]. These positioning technologies are based on timing information or angular information. Time based methods, such as, TOA, TDOA and E-OTD, calculate the distance between the Mobile Station (MS) and the Base Station (BS) by measuring the propagation time of the signal and multiply it by the speed of light. By using trilateration, the position of the MS can be estimated. On the other hand, angular approaches, like AOA, measure the angle between the MS and the BS and then estimate the location of the MS by using triangulation. Although these positioning technologies are simple, these approaches are only applicable to CDMA system since it can provide the timing or angular information. These approaches are either not available in other radio cellular network like GSM network or required additional hardware and hence increase the implementation cost. However, quite a number of countries have adopted GSM network instead of CDMA network. And GSM network can only provide the loss of signal strength due to signal attenuation [5]. Since the loss of signal strength is the common attribute of all radio cellular networks, thus location estimation algorithms proposed here are applicable to all radio cellular networks for ubiquitous/pervasive computing.

Many signal strength-based positioning algorithms have been proposed over the past few years [6]–[18]. There are two major threads about the location-aware positioning algorithms. One is the geometric algorithm based on the received signal strength (RSS). Our research group has proposed several location estimation approaches based on the RSS [14]–[17], [19], these methods focus on the model about the received signal strength and the distance between the BS and the MS, then provide an estimation of the MS location based on the geometric method. The Center of Gravity (CG) algorithm is a weighted mean of the locations of the BSs, so the estimation of CG is always within a convex hull formed by the locations of BSs regardless the actual position of the MS is outside or inside this convex hull [15], [16]. While the Circular Trilateration (CT) algorithm uses the intersection of three circles formed by three BSs [17]. CT algorithm has improved the defect of CG algorithm since the estimation of CT can be outside or inside the convex hull formed by the locations of the BSs, but it has its defect, because these three circles may not always intersect in only one point due to signal fading. Moreover, both CG algorithm and CT algorithm are just the simple geometric algorithms based on the free space signal propagation model [4]. These algorithms have not considered the directional transmission property of antenna. In view of that, we present a directional transmission propagation model, the Ellipse Propagation Model (EPM), to improve the defects of CG and CT algorithms , and we also propose two algorithms, the Geometric Algorithm and the Iterative Algorithm, to provide a more accurate estimation based on our directional signal propagation model in applications [14], [19]. These geometric algorithms we proposed are simple and efficient, and the experiment results in Hong Kong area have shown that they are useful.

On the other hand, many probabilistic approaches based on the RSS have been proposed to provide an estimation of the MS location in order to handle the signal attenuation problem [10], [11], [13], [18], [20]. The Teemu's research group has presented a statistical propagation model (SPM) to provide an estimation of the MS location [10]. They assume that the RSS obeys a normal distribution, then use an EM algorithm to provide an estimation of the model parameter which is a Maximum Likelihood Estimate (MLE), and then provide an estimation of the MS location based on SPM. While the Michael's research group has proposed another thread to provide an estimation of the MS location in the view of filter method to handle the signal attenuation problem [11], [13], [21]. They use a filter method to handle the signal attenuation problem and provide an estimation based on the survey data and the estimate is a Minimum Variance Unbiased Estimate (MVUE). Our group followed the above research threads and presented another statistical propagation model considered the antenna directional transmission property—the Directional Propagation model (DPM), then used a modified EM algorithm to provide the estimation of the model parameter, then provide a MLE as the MS location based on DPM [18]. Moreover, our research group has proposed a modified algorithm in order to provide a more accurate estimation of the MS location based on DPM [20].

These statistical approaches have given us a new viewpoint for positioning and the experiment results are good. However, all these statistical methods mentioned above just consider the survey data to provide the estimation of the statistical propagation model parameter (MLE, MUVE), and then provide an estimation of the MS based on the statistical propagation model. The information of survey data is the only considered factor to provide an estimation of the model parameter, while some useful information about the model parameter has been discarded. However, our experiments show that an accurate estimation of the MS location depends on an accurate directional signal propagation model. So if we want to provide an accurate estimation of the MS location, we need to build up an accurate directional signal propagation model to describe the relationship between the RRS and the surroundings factors. Only the information of survey data is not enough, the useful prior information about model parameter needs to be considered. However, the methods mentioned above, neither the Teemu's estimate (MLE) nor the Michael's estimate (MVUE), have not considered the useful prior information about model parameter. In view of this, we provide a Bayes Estimate (BE) to provide an estimation of our directional signal propagation model, then provide a more accurate estimation of the MS location based on this directional signal propagation model. In the Bayes statistics, we assume that we have some prior information (the experiment knowledge) of the model parameter before we estimate it, after we have taken some sample data, we then have more information about the model parameter, so we will revise our old viewpoint and have a new viewpoint about the model parameter, which we call it as a posterior knowledge about the model parameter. That is, we use the information of the sample data and our prior knowledge not just the information of the sample data to provide an estimation of the model parameter. So the effect of Bayes Estimate (BE) is not worse than that of MLE (or MVUE) just bases on the survey data in the view of the information using.

In this paper, we present a directional signal propagation model—the Modified Directional Propagation Model (MDPM) and provide a more accurate estimation of the MS location based on MDPM. MDPM derives from DPM, one directional propagation model had previously been proposed by our research group, and combines the merits of EPM and SPM. Furthermore, we propose an iterative method to provide the estimation of our model parameters besides the EM algorithm in order to reach the **global maximum** of its likelihood function, then we provide an estimation of the MS location based on our model with Bayes method. And we also present a formula of the uniform signal propagation model, which includes all the signal propagation models we mentioned in this paper. Our approach can be applicable to all signal propagation models.

This paper is divided into seven sections. In the following section, we will present our directional signal propagation model (MDPM). Then we describe the estimation method of model parameter in section 3. And we will present an iterative method to provide the Bayes Estimate of model parameter under normal distribution in section 4. Then we describe our estimation method of the MS location section 5. In section 6, we will show the simulation results of our model. And lastly in section 7, we present a summary of our research and discuss about our future work.

II. Signal Propagation Model

Antennas with energy focusing in a direction is called directional antenna. Due to the directional gain of the antenna, the transmitting power in different directions could be different. In general, antenna does not amplify the transmitting power. Instead, the antenna gain gives the ratio of the antenna radiated power density at a distinct point and the total antenna input power radiated isotropically, hence is a measurement of energy distribution.

Teemu Roos, Myllymäki and Tirri had proposed a statistical propagation model (SPM) that added an additional propagation parameter to the signal propagation model [10]. This parameter is associated with the direction of transmission. In brief, SPM is defined as follows.

$$\mu(d,\delta,p,\theta) = p + \beta_0 + (\beta_1 + \beta_2\delta)ln(d) \qquad \text{(II.1)}$$

where $\mu(d, \delta, p, \theta)$ is the mean of the received power, δ is the deviation between the direction of the receiver and the direction of transmission which is between zero and 180 degrees, d is the distance between the BS and the MS, p is the transmitted power in decibels, θ denotes the set of propagation parameters, and $\beta = (\beta_0, \beta_1, \beta_2)^T$ are the regression coefficients.

Despite SPM has included the directional coefficient δ, the model has not been fully tested in a real environment. Furthermore, it may not be appropriate to take the directional coefficient as the exponent of T-R separation. The Log-Distance Path Loss Model has shown that the exponent of T-R separation (Path Loss exponent) is related to the surrounding environment instead of the direction of the transmission [23].

Form our observations, the directive gain of the directional antenna affects the RSS to a certain extent. The environmental factor is another attributes which should be considered. Unfortunately, most of the existing propagation models have not dealt with these two factors. In view of this, we design a Directional Propagation Model (DPM) which will take these factors into account. DPM is defined as,

$$\overline{pl} = \beta_0 + \beta_1 g + \beta_2 log(h) + (\beta_3 + \beta_4 log(h) + \beta_5 e) ln(d) \quad \text{(II.2)}$$

We have also presented another directional propagation model, EPM, in our previous work. EPM considers the directional transmission property and assumes the contour line of signal strength is an ellipse. EPM is defined as,

$$d = k(s_0/s)^{1/\alpha} \frac{1-e}{1-e\cos(\delta)} \quad \text{(II.3)}$$

EPM describes the relationship between the RRS and the MS-BS distance and focuses on the distance between the MS and the BS. Furthermore, EPM is a directional propagation model. In order to look insight into EPM, we rewrite it as the form of SPM or DPM.

Rewrite the formula of EPM, we have

$$-ln(s) = -ln(s_0) - \alpha ln(k(1-e)) + \alpha ln(1 - e\cos(\delta)) + \alpha ln(d)$$

It is similar with the signal propagation model as follows,

$$\mu(p, d, \delta, \theta) = p + \beta_0 + \beta_1 \cos(\delta) + \beta_2 ln(d) \quad \text{(II.4)}$$

where p is the transmitted power in decibels, δ is the deviation between the direction of the receiver and the direction of transmission, d is the distance between the BS and the MS, $\theta = (\beta_0, \beta_1, \beta_2)$ are parameters of the signal propagation model.

Although EPM considers the directional transmission property, the path loss exponent β_2 (α) is a constant. We must take the environment effect into account for the path loss exponent.

We can draw a uniform formula of a directional signal propagation model from analyzing the structures of SPM, EPM and DPM. As we can see, these three directional signal propagation models have a uniform formula,

$$\mu = f + gln(d) \quad \text{(II.5)}$$

where μ is the mean received signal strength in decibels, d is the distance between the MS and the BS, f and g are two functions about the surrounding factors, and g is also called the path loss exponent.

In SPM, f is a constant and g is a function of the deviation (δ), and in DPM, f is a function of the deviation (δ) and the height of antenna (h), while in EPM, f is a function of the deviation (δ) and g is a constant. Thus, a signal propagation model can derive from the uniform signal propagation model (II.5). We call the model (II.5) as the **Uniform Signal Propagation Model** (USPM), which includes all the signal propagation models mentioned in this paper.

So we combine the merits of SPM, DPM and EPM to build up a new signal propagation model to provide an accurate estimation for implementation. Our new model will take the surrounding environment factor and the direction factor into account.

The mean of the received signal strength (RSS) is given by

$$\begin{aligned}\mu(p, d, h, e, \delta, \beta) = \quad & p + \beta_0 + \beta_1 cos(\delta) + \beta_2 log(h) \\ & +(\beta_3 + \beta_4 e + \beta_5 log(h)) ln(d) + \beta_6 h\end{aligned} \quad \text{(II.6)}$$

where $\mu(p, d, h, e, \delta, \beta)$ is the mean received signal strength in decibels; p is the transmitted signal strength in decibels; d is the distance in meter between the MS and the BS; e is the environment index and δ is the deviation between the direction of transmission and the direction of the receiver as measured from the transmitter, the values of δ are clearly between zero and 180 degrees. $\beta = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6)^T$ are the parameters of the signal propagation model. and $\alpha = \beta_3 + \beta_4 e + \beta_5 log(h)$ is called the path loss exponent, h is the height in meter of the antenna.

We compare the formula of our model with the uniform signal propagation model (II.5), then we can get $f = p + \beta_0 + \beta_1 cos(\delta) + \beta_2 log(h) + \beta_6 h$ and $g = \beta_3 + \beta_4 e + \beta_5 log(h)$.

Our model (II.6) derives from DPM and combines the merits of SPM and EPM. For our research, the height of antenna has taken an important part in the signal propagation model, we add a more term ($\beta_6 h$) about the height of antenna into the new model in order to build up a signal propagation model to provide a more accurate estimation. On the other hand, we have modified a term about the directional gain, g, into $cos(\delta)$ from DPM. We call our model as the Modified Directional Propagation Model (MDPM).

The received signal strength, s, has a relationship with the mean received signal strength, $\mu(p, d, h, e, \delta, \beta)$,

$$s = \mu(p, d, h, e, \delta, \beta) + \varepsilon \quad \text{(II.7)}$$

where ε follows a normal distribution with the mean zero and variance σ^2.

So the distribution s is Gaussian with the following p.d.f.

$$p(s|p, d, h, e, \delta, \beta, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(s - \mu(p, d, h, e, \delta, \beta))^2}{2\sigma^2}) \quad \text{(II.8)}$$

III. Estimation Method of Model Parameter

A. Structure

The received signal strength (RSS), s, follows a Gaussian distribution and its probability density function(p.d.f) is as follows,

$$p(s|\beta,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(s-\mu(\beta))^2}{2\sigma^2}) \tag{III.1}$$

where β is the model parameter, $\mu(\beta)$ and σ^2 are the mean and deviation respectively. And $\mu(\beta)$ is a signal propagation model.

We have the prior distribution of β, denoted by $\pi(\beta)$. We assume it follows a distribution. Then we can derive the posterior p.d.f of β as follows:

$$p(\beta|s,\sigma^2) = p(s|\beta,\sigma^2)\pi(\beta)/p(s|\sigma^2) \tag{III.2}$$

where $p(s|\sigma^2) = \int p(s|\beta,\sigma^2)\pi(\beta)d\beta$, which is called the predictive distribution of s under σ^2, and the integration is performed over the domain of β.

So the posterior pdf of the parameter β can be rewritten by:

$$p(\beta|s,\sigma^2) = \frac{p(s|\beta,\sigma^2)\pi(\beta)}{\int p(s|\beta,\sigma^2)\pi(\beta)d\beta} \tag{III.3}$$

Consider a Bayes statistical model, the distribution of observation s depends on the parameter β, $\beta \in \Theta$, where Θ is called the parameter space. Let A be an possible action space. For an action $a \in A$ and a parameter value $\beta \in \Theta$, we need a loss function $l(\beta, a)$ to provide a criteria to value an action. For example, for a function of parameter $q(\beta)$ and an action a, a common loss function is the square loss function $l(\beta, a) = (q(\beta) - a)^2$, of course, we also can choose an absolute loss function $l(\beta, a) = |q(\beta) - a|$. For an estimation $q(\beta)$, if we choose a square loss function, the Bayes Estimate is the expectation of the posterior pdf $E^{p(\beta|s,\sigma^2)}(q(\beta))$ [24], [25].

B. Data Collection

According to the GSM specification, the range of the received signal strength is between -29dBm and -114dBm only.

1) Sufficient Data Collection: We classify the observed RSS and actual RSS according to the range of received signals into three types which are defined as follows.

$$\begin{cases} o_i^{(t)} - \epsilon_l \le s_i^{(t)} < o_i^{(t)} + \frac{\epsilon_n}{2} & (C1 : s_i^{(t)} \le -110) \\ o_i^{(t)} - \frac{\epsilon_n}{2} \le s_i^{(t)} < o_i^{(t)} + \frac{\epsilon_n}{2} & (C2 : -110 < s_i^{(t)} < -48) \\ o_i^{(t)} - \frac{\epsilon_n}{2} \le s_i^{(t)} < o_i^{(t)} + \epsilon_u & (C3 : s_i^{(t)} \ge -48) \end{cases} \tag{III.4}$$

where $o_i^{(t)}$ is the i-th observed RSS from a BS of antenna type t at a location. $s_i^{(t)}$ is the i-th actual RSS from a BS of antenna type t at a location. While ϵ_i, ϵ_n and ϵ_u are the lower boundary error, normal error and upper boundary error respectively, their values are shown as follows.

$$\begin{cases} \epsilon_l = 4dBm \\ \epsilon_n = 1dBm \\ \epsilon_u = 19dBm \end{cases} \tag{III.5}$$

The mean observed RSS $\overline{o}$ and the mean actual RSS $\overline{s}$ are defined as follow.

$$\overline{o} \overset{def}{=} E(o_i^{(t)}) \tag{III.6}$$

$$\overline{s} \overset{def}{=} E(s_i^{(t)}|o_i^{(t)}) \tag{III.7}$$

So, $\overline{o}$ and $\overline{s}$ can be represented as follows,

$$\overline{o} - l \le \overline{s} < \overline{o} + u \tag{III.8}$$

where $l = \frac{2C_1\epsilon_l + C_2\epsilon_n}{2(C_1+C_2)}$, $u = \frac{C_2\epsilon_n + 2C_3\epsilon_u}{2(C_2+C_3)}$,

C_1 is the total no. of observations from a BS of antenna type t at a location satisfies C_1 in Eq.(III.4)

C_2 is the total no. of observations from a BS of antenna type t at a location satisfies C_2 in Eq. (III.4)

C_3 is the total no. of observations from a BS of antenna type t at a location satisfies C_3 in Eq. (III.4)

So the likelihood function of $\overline{s}$ is:

$$\begin{aligned} p(\overline{s}|\beta,\sigma^2) &= \int_{\overline{o}-l}^{\overline{o}+u} p(s|\beta,\sigma^2)ds \\ &= \Phi(\tfrac{\overline{o}+u-\mu(\beta)}{\sqrt{\sigma^2}}) - \Phi(\tfrac{\overline{o}-l-\mu(\beta)}{\sqrt{\sigma^2}}) \end{aligned} \tag{III.9}$$

where $\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} exp(-\frac{t^2}{2})dt$.

2) Insufficient Data Collection: In general, the received power values cannot be directly observed because of physical restrictions. First, the received power values have to be truncated, i.e., rounded to finite accuracy. Second, because of sensitivity limitations, the received power on only some channels—those with the strongest signal—is reported. The only information about the other channels is that their received power does not exceed the power on any of the reported channels. In such cases, we say that the received power variable is truncated at a point given by the smallest of the known values.

To solve this problem, we would like to introduce the hidden RSS. Generally, a MS can received signals from 9 BSs only. However, when there a lot of surrounding BSs, the MS may receive more than 9 BSs. These RSSs can not be collected directly but they should not be greater than the minimum RSS measured by the MS at a location. We called these RSSs as hidden RSS. Thus, in addition to the observed RSS, we could used the hidden RSS to increase the training sample size.

In brief, the hidden RSS is given by

$$h_i^{(t)} \le min(o_i^{(t)}) + \frac{\epsilon_n}{2} \tag{III.10}$$

where $o_i^{(t)}$ and ϵ_n are defined above, $h_i^{(t)}$ is the i-th hidden RSS from a BS of antenna type t at a location.

The mean hidden RSS $\overline{h}$ is defined as:

$$\overline{h} \overset{def}{=} E(h_i^{(t)}|min(o_i^{(t)})) \tag{III.11}$$

and

$$\overline{h} \le o_{min} + \frac{\epsilon_n}{2} \tag{III.12}$$

where $o_{min} = E(min(o_i^{(t)}))$.

So the likelihood function of $\overline{h}$ is:

$$p(\overline{h}|\beta,\sigma^2) = \int_{-\infty}^{o_{min}+\frac{\epsilon_n}{2}} p(s|\beta,\sigma^2)ds = \Phi(\frac{o_{min} + \frac{\epsilon_n}{2} - \mu(\beta)}{\sqrt{\sigma^2}}) \tag{III.13}$$

where $\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} exp(-\frac{t^2}{2})dt$.

For an antenna type, Its likelihood function is constructed by the effect of the sufficient data and the insufficient data. So for each antenna type, we can obtain the likelihood function $p(\mathbf{s}|\beta,\sigma^2)$:

$$\begin{aligned} p(\mathbf{s}|\beta,\sigma^2) = \quad & \prod_{s_i\in suf.}[\Phi(\tfrac{o_i+u-\mu(\beta)}{\sqrt{\sigma^2}})-\Phi(\tfrac{o_i-l-\mu(\beta)}{\sqrt{\sigma^2}})] \\ & \cdot\prod_{s_i\in insuf.}\Phi(\tfrac{o_{min}+\frac{\epsilon_n}{2}-\mu(\beta)}{\sqrt{\sigma^2}}) \end{aligned} \tag{III.14}$$

C. Parameter Estimation

When a mobile phone is connected, in addition to receiving signal from the serving BS, it can receive signals from the neighboring BSs as well. We find the posterior p.d.f. of each location with respect to each receiving signal. We derive the posterior p.d.f. of θ based on the information we have. For the Bayes rule, we have the following relationship:

$$p(\beta|\mathbf{s},\sigma^2)=\pi(\beta)p(\mathbf{s}|\beta,\sigma^2)/p(x|\sigma^2) \tag{III.15}$$

where

$p(\mathbf{s}|\sigma^2)=\int_\Theta p(\mathbf{s}|\beta,\sigma^2)\pi(\beta)d\beta$,

$p(\beta|\mathbf{s},\sigma^2)$ is the posterior pdf of β,

$\pi(\beta)$ is the prior pdf of β,

$p(\mathbf{s}|\beta,\sigma^2)$ is the likelihood function of $\mathbf{s}$, which depends on β and σ^2,

$p(\mathbf{s}|\sigma^2)$ is the likelihood function of the predictive distribution of $\mathbf{s}$ under σ^2.

Since we have done some research work about the model $\mu(\beta)$, we have accumulated some prior information about the parameter β. We assume that $\pi(\beta)$ obeys a distribution.

So the pdf of the predictive distribution of the sample data $\mathbf{s}$ under σ^2 is:

$$p(\mathbf{s}|\sigma^2) \quad = \quad \int_\Theta p(\mathbf{s}|\beta,\sigma^2)\pi(\beta)d\beta \tag{III.16}$$

We can obtain the posterior p.d.f. of θ:

$$p(\beta|\mathbf{s},\sigma^2)=\frac{\pi(\beta)p(\mathbf{s}|\beta,\sigma^2)}{\int_\Theta\pi(\beta)p(\mathbf{s}|\beta,\sigma^2)d\beta} \tag{III.17}$$

We can provide a Bayes Estimate of the parameter θ based on a loss function we choose. For example, if we choose a square loss function, the estimation is $\widehat{\beta}=E^{p(\beta|\mathbf{s},\sigma^2)}(\beta)$, while we choose an absolute loss function, the estimation of β is $\widehat{\beta}=med(p(\beta|\mathbf{s},\sigma^2))$, where $med(.)$ is the median function.

We choose the one that maximizes the posterior pdf (III.17) as the estimation of the model parameters. If the prior distribution $\pi(\beta)$ obeys a uniform distribution, we choose the estimate which maximizes the posterior pdf as the Bayes Estimate, then this estimate is the same with the local **MLE**, namely, $\hat{\beta}=argmax_{\{\beta\in\Theta\}}\{p(\beta|\mathbf{s},\sigma^2)\}=\widehat{\beta}_{\mathbf{MLE}}$, and if the definition field of this uniform distribution is Θ, namely, we have no prior information about the β, the estimate which maximizes the posterior pdf is same with the maximum likelihood estimate (MLE). We denote the Bayes Estimate as $\hat{\beta}_{\mathbf{BE}}$.

Define

$$\mathbf{SSE}\overset{def}{=}\Sigma_{i=1}^n(s_i-\mu_i(\hat{\beta}_{BE}))^2$$

So the estimation of σ is: $\hat{\sigma}=\sqrt{\frac{1}{n-r}\mathbf{SSE}}$, where r is the number of the parameters.

But if we compute the value of $\hat{\beta}_{\mathbf{BE}}$, we need to know the value of $\hat{\sigma}$, and if we want to provide the value of $\hat{\sigma}$, we need to know the value of $\hat{\beta}_{\mathbf{BE}}$ too. The deadlock occurs when we compute $\hat{\beta}_{\mathbf{BE}}$ and $\hat{\sigma}$ at the same time. So we can not provide estimation of $\hat{\beta}_{\mathbf{BE}}$ and $\hat{\sigma}$ at the same time.

We present an algorithm to solve this problem based on the idea of **EM** algorithm [10], [26]. Since we can not provide the estimation of $\hat{\beta}_{\mathbf{BE}}$ and $\hat{\sigma}$ at the same time, we need to divide it into two steps to compute these value: fixed $\hat{\sigma}$ then compute $\hat{\beta}_{\mathbf{BE}}$ and then compute $\hat{\sigma}$ under this $\hat{\beta}_{\mathbf{BE}}$. We present a method to provide the value of $\hat{\beta}_{\mathbf{BE}}$ in the the following section (IV).

We describe our algorithm details as follows:

Step1: Given a value $\hat{\sigma}_0$,

Step2: Provide the Bayes Estimate $\hat{\beta}_{\mathbf{BE}}$ using the posterior p.d.f. (III.17) under $\hat{\sigma}_0$,

Step3: Calculate the **SSE** value, $\mathbf{SSE}=\Sigma_{i=1}^n(s_i-\mu_i(\hat{\beta}_{\mathbf{BE}}))^2$,

Step4: Compute the value of $\sigma=\sqrt{\frac{1}{n-r}\mathbf{SSE}}$, denote it $\hat{\sigma}_1$,

Step5: If $\hat{\sigma}_1<\hat{\sigma}_0$, then update the $\hat{\sigma}_0$ value, assign $\hat{\sigma}_0=\hat{\sigma}_1$, go to Step 2; else stop the computation and output the answer.

Choose a suitable initial value of $\hat{\sigma}_0$, and we can get a feasible solution based on the algorithm mentioned above, since this algorithm will stop after a finite step computation or converge.

Set $S=\{\hat{\sigma}_i\}$ is a set generated by our algorithm. And we assert this set is no-empty, since there is at least one $\hat{\sigma}_0\in S$. If S is only a finite set, namely, it has finite elements, we choose the least $\hat{\sigma}_i$ and its corresponding $\hat{\beta}_{BE}$ as our solution. While S has infinite elements, this series $\{\hat{\sigma}_i\}$ will converge, we choose the convergence value and its corresponding $\hat{\beta}_{BE}$ as our solution. Since under the definition of S, we have $\hat{\sigma}_i<\hat{\sigma}_j$, $i>j$, and $\hat{\sigma}_i,\hat{\sigma}_j\in S$, and we know the restriction of $\hat{\sigma}$, $\hat{\sigma}>0$. So this series converges.

Under the assumption of our liner regression model, we can provide $\hat{\beta}_{\mathbf{LSE}}$ ($\hat{\beta}_{\mathbf{LSE}}$ means the Least Squares Estimate) and $\widehat{\sigma^2}_{\mathbf{LSE}}=\mathbf{SEE}/(n-r)$, where n is the number size of the data, and r is the rank of the data matrix [27]. These estimations only use the data information and have not include the prior information of the parameters. We choose $\widehat{\sigma^2}_{\mathbf{LSE}}$ as $\widehat{\sigma^2}_0$ and use our algorithm to provide the $\hat{\beta}_{\mathbf{BE}}$.

IV. Bayes Estimate with a normal prior distribution

Consider a linear regression model

$$Y=X\beta+\varepsilon \tag{IV.1}$$

where Y is a $1\times n$ vector, β is a $1\times r$ vector, and X is a $n\times r$ matrix and ε obeys a multi-dimensional normal distribution,with the mean $\mathbf{0}$ and variance matrix Σ.

Since we have done some research about this linear regression model, we have some information about the parameter β. We assume that the parameter β follows a multi-dimensional normal distribution with the mean $\hat{\beta}$ and the variance matrix $\widehat{\sigma^2}(X'X)^{-1}$, denote it as $\beta\sim N(\hat{\beta},\widehat{\sigma^2}(X'X)^{-1})$.

So the prior pdf of β is :

$$\begin{aligned} \pi(\beta)= \quad & (\tfrac{1}{\sqrt{2\pi}})^r|\widehat{\sigma^2}(X'X)^{-1}|^{-\frac{1}{2}} \\ & .exp(-\tfrac{(\beta-\hat{\beta})'(\widehat{\sigma^2}(X'X)^{-1})^{-1}(\beta-\hat{\beta})}{2}) \end{aligned} \tag{IV.2}$$

Since the estimation of β which maximizes its posterior pdf of β is the same with the one that maximizes its logarithm function.

Set $l(\beta) = ln(p(\beta|\mathbf{s}, \hat{\sigma^2}))$, where $p(\beta|\mathbf{s}, \hat{\sigma^2})$ is defined in (III.17). Then we have

$$\begin{aligned} l(\beta) &= c_0 + c_1 - (\beta - \hat{\beta})^{'}(\hat{\sigma^2}(X'X)^{-1})^{-1}(\beta - \hat{\beta})/2 \\ &\quad + \textstyle\sum_{suf} ln[\Phi(\frac{o_i - p_i + u - X_i\beta}{\hat{\sigma}}) - \Phi(\frac{o_i - p_i - l - X_i\beta}{\hat{\sigma}})] \\ &\quad + \textstyle\sum_{insuf} ln(\Phi(\frac{o_{min} - p_i + \epsilon_n/2 - X_i\beta}{\hat{\sigma}})) \end{aligned} \tag{IV.3}$$

where c_0 and c_1 are two constants which are independent with β. That we find a β which maximizes $l(\beta)$ is just required to consider these last three parts in the above formula (IV.3).

Define

$$\begin{aligned} A_1 &= -(\beta - \hat{\beta})^{'}(\hat{\sigma^2}(X'X)^{-1})^{-1}(\beta - \hat{\beta})/2 \\ A_2 &= \textstyle\sum_{suf} ln[\Phi(\frac{o_i - p_i + u - X_i\beta}{\hat{\sigma}}) - \Phi(\frac{o_i - p_i - l - X_i\beta}{\hat{\sigma}})] \\ A_3 &= \textstyle\sum_{insuf} ln(\Phi(\frac{o_{min} - p_i + \epsilon_n/2 - X_i\beta}{\hat{\sigma}})) \end{aligned}$$

So $l(\beta) = c_0 + c_1 + A_1 + A_2 + A_3$.

In order to provide a β which maximizes $l(\beta)$, we need to derive the first order derivative and the second order derivative of $l(\beta)$ about β. The solution which maximizes $l(\beta)$ satisfies that its first order derivative is 0 and its second order derivative is a negative definite matrix.

We differentiate the first order derivative and the second order derivative for A_1, A_2 and A_2, respectively.

$$\begin{aligned} \frac{\partial A_1}{\partial \beta} &= -(\hat{\sigma^2}(X'X)^{-1})^{-1}(\beta - \hat{\beta}) \\ \frac{\partial A_2}{\partial \beta} &= -\frac{1}{\hat{\sigma}} \textstyle\sum_{suf} \frac{p(x_{1i}) - p(x_{2i})}{\Phi(x_{1i}) - \Phi(x_{2i})} X_i^{'} \\ \frac{\partial A_3}{\partial \beta} &= -\frac{1}{\hat{\sigma}} \textstyle\sum_{insuf} \frac{p(x_{3i})}{\Phi(x_{3i})} X_i^{'} \end{aligned}$$

and

$$\begin{aligned} \frac{\partial^2 A_1}{\partial \beta \partial \beta'} &= -\frac{(X'X)}{\hat{\sigma^2}} \\ \frac{\partial^2 A_2}{\partial \beta \partial \beta'} &= -\frac{1}{\hat{\sigma^2}} \textstyle\sum_{suf} f_1(x_{1i}, x_{2i}) X_i^{'} X_i \\ \frac{\partial^2 A_3}{\partial \beta \partial \beta'} &= -\frac{1}{\hat{\sigma^2}} \textstyle\sum_{insuf} f_2(x_{3i}) X_i^{'} X_i \end{aligned}$$

where

X_i is a row of the data matrix, X, it is a $r \times 1$ vector,
$p(x) = \frac{1}{\sqrt{2\pi}} exp(-\frac{x^2}{2})$, $\Phi(x) = \int_{-\infty}^{x} p(t)dt$,
$x_{1i} = \frac{o_i - p_i + u - X_i\beta}{\hat{\sigma}}$, $x_{2i} = \frac{o_i - p_i - l - X_i\beta}{\hat{\sigma}}$,
$x_{3i} = \frac{o_{min} - p_i + \epsilon_n/2 - X_i\beta}{\hat{\sigma}}$,
$x_{1i} - x_{2i} = (u + l)/\hat{\sigma} = \epsilon_i$,
$f_1(x_{1i}, x_{2i}) = \frac{p(x_{1i})x_{1i} - p(x_{2i})x_{2i}}{\Phi(x_{1i}) - \Phi(x_{2i})} + (\frac{p(x_{1i}) - p(x_{2i})}{\Phi(x_{1i}) - \Phi(x_{2i})})^2$,
$f_2(x_{3i}) = \frac{p(x_{3i})x_{3i}}{\Phi(x_{3i})} + (\frac{p(x_{3i})}{\Phi(x_{3i})})^2$.

So the first order derivative of $l(\beta)$ about β as following:

$$\begin{aligned} \frac{\partial l(\beta)}{\partial \beta} &= \frac{\partial A_1}{\partial \beta} + \frac{\partial A_2}{\partial \beta} + \frac{\partial A_3}{\partial \beta} \\ &= -\frac{(X'X)}{\hat{\sigma^2}}(\beta - \hat{\beta}) \\ &\quad -\frac{1}{\hat{\sigma}} \textstyle\sum_{suf} \frac{p(x_{1i}) - p(x_{2i})}{\Phi(x_{1i}) - \Phi(x_{2i})} X_i^{'} - \frac{1}{\hat{\sigma}} \sum_{insuf} \frac{p(x_{3i})}{\Phi(x_{3i})} X_i^{'} \end{aligned}$$

and the second order derivative of $l(\beta)$ about β is:

$$\begin{aligned} \frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'} &= \frac{\partial^2 A_1}{\partial \beta \partial \beta'} + \frac{\partial^2 A_2}{\partial \beta \partial \beta'} + \frac{\partial^2 A_3}{\partial \beta \partial \beta'} \\ &= -\frac{1}{\hat{\sigma^2}}\{(X'X) \\ &\quad + \textstyle\sum_{suf} f_1(x_{1i}, x_{2i}) X_i^{'} X_i \\ &\quad + \textstyle\sum_{insuf} f_2(x_{3i}) X_i^{'} X_i\} \end{aligned}$$

Set $g_1(x) = \frac{p(x+\epsilon) - p(x)}{\Phi(x+\epsilon) - \Phi(x+\epsilon)}$, $g_2(x) = \frac{p(x)}{\Phi(x)}$, $f_1(x) = \frac{p(x+\epsilon)(x+\epsilon) - xp(x)}{\Phi(x+\epsilon) - \Phi(x)} + (\frac{p(x+\epsilon) - p(x)}{\Phi(x+\epsilon) - \Phi(x)})^2$, $f_2(x) = \frac{p(x)x}{\Phi(x)} + (\frac{p(x)}{\Phi(x)})^2$, then we have $g_1^{'}(x) = -f_1(x)$, $g_2^{'}(x) = -f_2(x)$.

So

$$\frac{\partial l(\beta)}{\partial \beta} = -\frac{(X'X)}{\hat{\sigma^2}}(\beta - \hat{\beta}) - \frac{1}{\hat{\sigma}} \sum_{suf} g_1(x_{2i}) X_i^{'} - \frac{1}{\hat{\sigma}} \sum_{insuf} g_2(x_{3i}) X_i^{'}$$

$$\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'} = -\frac{(X'X)}{\hat{\sigma^2}} - \frac{1}{\hat{\sigma^2}} \sum_{suf} f_1(x_{2i}) X_i^{'} X_i - \frac{1}{\hat{\sigma^2}} \sum_{insuf} f_2(x_{3i}) X_i^{'} X_i$$

So the second order derivative matrix $\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'}$ is a negative definite matrix. The one which satisfies $\frac{\partial l(\beta)}{\partial \beta} = 0$ is the solution which maximizes the posterior p.d.f. (III.17).

Now we consider the first order derivative of $l(\beta)$ about β. $\frac{\partial l(\beta)}{\partial \beta} = 0$ can be described as:

$$\beta = \hat{\beta} - \sum_{suf} g_{1i}(\beta)\hat{\sigma}(X'X)^{-1} X_i^{'} - \sum_{insuf} g_{2i}(\beta)\hat{\sigma}(X'X)^{-1} X_i^{'}$$

Since β appears in both side of the above equation, we choose an iterative method to provide its solution.

Thus, we derive an iterative formula to provide an estimation of β. The iterative formula is :

$$\left\{ \begin{aligned} \beta_n &= \hat{\beta} - \textstyle\sum_{suf} g_{1i}(\beta_{n-1})\hat{\sigma}(X'X)^{-1} X_i^{'} \\ &\qquad - \textstyle\sum_{insuf} g_{2i}(\beta_{n-1})\hat{\sigma}(X'X)^{-1} X_i^{'} \\ \beta_0 &= \hat{\beta} \end{aligned} \right. \tag{IV.4}$$

Theorem 4.1: And this iterative formula (IV.4) converges. Furthermore, the above iterative formula (IV.4) converges in only one solution.

We have known that $\frac{l(\beta)}{\partial \beta} = 0$ has only one convergence solution, and $\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'} < \mathbf{0}$. So the solution of $\frac{l(\beta)}{\partial \beta} = 0$ is the one that maximizes $l(\beta)$. Furthermore, it reaches the **global maximum** of $l(\beta)$.

V. Location Estimation

The true location of the mobile station (MS) is denoted $\mathbf{l} = (x, y)^T$, where (x, y) is the location of the MS in two-dimensional space. In this paper, we just consider a two-dimensional location, but our approaches are easily extended into a three dimension. The RSS path loss measurements are given in decibels.

When a mobile phone is connected, in addition to receiving signal from the serving BS, it can receive signals from the neighboring BSs as well. We find the posterior p.d.f. of each location with respect to each receiving signal. Then we provide the Bayes Estimate as an estimation of the MS location.

We provide the posterior p.d.f. of the location $\mathbf{l}$ based on the information we have. For the Bayes rule, we have the relationship:

$$p(\mathbf{l}|\mathbf{s}, \hat{\beta}, \hat{\sigma^2}) = \pi(\mathbf{l}) p(\mathbf{s}|\mathbf{l}, \hat{\beta}, \hat{\sigma^2}) / p(\mathbf{s}|\hat{\beta}, \hat{\sigma^2}) \tag{V.1}$$

where

$p(\mathbf{s}|\hat{\beta}, \hat{\sigma^2}) = \int_{\Theta} p(\mathbf{s}|\mathbf{l}, \hat{\beta}, \hat{\sigma^2}) \pi(\mathbf{l}) d\mathbf{l}$,
$p(\mathbf{l}|\mathbf{s}, \hat{\beta}, \hat{\sigma^2})$ is the posterior pdf of the location $\mathbf{l}$,

$\pi(\mathbf{l})$ is the prior pdf of **l**,

$p(\mathbf{s}|\mathbf{l},\hat{\beta},\hat{\sigma^2})$ is the likelihood function of **s**, which depends on **l** and $\hat{\beta}$, $\hat{\sigma^2}$,

$p(\mathbf{s}|\hat{\beta},\hat{\sigma^2})$ is the likelihood function of the predictive distribution of s.

We assume that $\pi(\mathbf{l})$ obeys a uniform distribution, that is to say,

$$\pi(\mathbf{l}) = \begin{cases} \frac{1}{L} & \mathbf{l}\in\Theta_L \\ 0 & otherwise \end{cases} \tag{V.2}$$

where L is the area of Θ_L, and $\Theta_L \subset \Theta$.

So the pdf of the predictive distribution of the sample data **s** is:

$$p(\mathbf{s}|\hat{\beta},\hat{\sigma^2}) = \int_{\Theta} p(\mathbf{s}|\mathbf{l},\hat{\beta},\hat{\sigma^2})\pi(\mathbf{l})d\mathbf{l} = \frac{1}{L}\int_{\Theta_L} p(\mathbf{s}|\mathbf{l},\hat{\beta},\hat{\sigma^2})d\mathbf{l} \tag{V.3}$$

We can obtain the posterior p.d.f. of location **l**:

$$p(\mathbf{l}|\mathbf{s},\hat{\beta},\hat{\sigma^2}) = \frac{exp(-\frac{1}{2}\sum_{i=1}^{n}\frac{(s_i-\mu_i(\mathbf{l},\hat{\beta}))^2}{\hat{\sigma_i^2}})}{\int_{\Theta_L} exp(-\frac{1}{2}\sum_{i=1}^{n}\frac{(s_i-\mu_i(\mathbf{l},\hat{\beta}))^2}{\hat{\sigma_i^2}})d\mathbf{l}} \tag{V.4}$$

where $\mathbf{s} = (s_1, s_2, ..., s_n)^T$, it is the received signal information, and $\mathbf{l} \in \Theta_L$.

So we can provide the Bayes Estimate based on different loss functions. That is, we can provide the MS location. For example, if we choose a square loss function, the estimation of the MS location is $\hat{\mathbf{l}} = E^{p(\mathbf{l}|\mathbf{s},\hat{\beta},\hat{\sigma^2})}(\mathbf{l})$, while we choose an absolute loss function, the estimation of **l** is : $\hat{\mathbf{l}} = med(p(\mathbf{l}|\mathbf{s},\hat{\beta},\hat{\sigma^2}))$, where $med(.)$ is the median function, and if we choose the estimation which maximizes the posterior p.d.f., it is the local maximum likelihood estimate, $\hat{\mathbf{l}} = argmax_{\{\mathbf{l}\in\Theta_L\}}\{p(\mathbf{l}|\mathbf{s},\hat{\beta},\hat{\sigma^2})\} = \hat{\mathbf{l}}_{\mathbf{MLE}}$.

VI. Simulation Results

With the technical support of two mobile operators in Hong Kong, we have conducted an intensive field test in many regions in Hong Kong in order to validate our model. We have divided the data into two parts: 30% of the data for training the models, and 70% of the data for estimating the location of the MS. We choose the distance between the exact location and the estimation location as the performance criteria to describe the estimation accuracy.

We divided our experiment in two phases: the first phase is to train the model using 30% of the field test data, and the second phase is to estimate the location of the MS with the rest of the 70% of the field test data. Since we have received the information of RSS and TA from a serving cell, we use the information of TA zone to select our solution.

We used the real data collected from different regions in Hong Kong to validate MDPM. These regions contain different kinds of environment including the coastal, rural, suburban, urban and metropolitan area in Hong Kong. We used these data to provide a BE of the parameters of MDPM at first. In the testing phase, we provide a BE of the MS location based on our directional signal propagation model. These results are then used to compare with those of CG, CT algorithms, the Geometric Algorithm and the Iterative Algorithm under EPM, SPM and DPM with MLE.

A. Importance Test on the Added Parameter

The Modified Directional Propagation Model (MDPM) has combined the merits of SPM, DPM and EPM. MDPM derives from DPM, which modifies a term of DPM $g(\delta)$ as $cos(\delta)$, and then adds one more term h into our model.

From our study, we find that the height of antenna pays an important role on building up a signal propagation model. Although DPM pays an important role about the height of the antenna, which has two terms about the height of antenna:$log(h)$ and $log(h)ln(d)$, it is necessary to add more term h in our signal propagation model in order to build up a more accurate directional signal propagation model. So MDPM has seven parameters, which is one more parameter than that of DPM.

And we choose 13 Antenna Types to build up our model, which include about 94.56% of the collected data. And the parameter of term h in MDPM about these Antenna Types are all significant. The results are shown in Table(I), which means that it is reasonable to add one more term h in our model. Since we have presented a new algorithm to provide the estimation of the model parameters, even the number of parameters of MDPM is one more parameter than that of DPM, the computational cost of MDPM is the same level with that of DPM.

B. Estimating the MDPM parameters

Based on our study, there are almost 30 types of antennas to provide the mobile phone services in Hong Kong. And the first 13 types of antennas include almost 94.56% of the collected data. In order to save the computational cost, we just build up a MDPM for each antenna type of these first 13 types of antennas using 30% of the collected data, while the rest of the collected data is used to calculate the estimation of the MS location. Since we have previously done some research about DPM, we have some prior information about the parameters of the directional signal propagation model, we provide a Bayes Estimate as an estimation of the parameters of our directional signal propagation model. In order to provide this Bayes Estimate, we derive the least squares estimate (LSE) of our model first, then use an iterative method mentioned in the appendix to provide an Bayes Estimate of our model based on this least square estimate.

C. Results of the MDPM

After we have a MDPM for each antenna type, we use the rest of the field test data to calculate the MS location. Since we also receive the information of TA (timing advance) besides RSS from a serving cell, we use a TA zone formed by the information of TA to help us provide a Bayes Estimate of the MS location. Since a mobile phone can receive the TA and RSS from a serving cell in the GSM network, the information of TA can help us provide a more accurate estimation for providing location services.

In the estimating MS location phrase, we choose two methods to calculate the rest of the field test data. Since the Bayes Estimate depends on a loss function we choose, we provide two Bayes Estimates: one is based on a quadratic loss

Antenna Type	F-Value	F-Test	Antenna Type	F-Value	F-Test
1	174.2406	0.00000	8	262.0608	0.00000
2	392.7927	0.00000	9	56.9367	0.00000
3	79.1050	0.00000	10	203.0342	0.00000
4	346.4012	0.00000	11	320.3630	0.00000
5	653.2067	0.00000	12	50.3453	0.00000
6	644.0162	0.00000	13	23.9623	0.00000
7	441.2741	0.00000			

TABLE I

F-VALUE AND F-TEST

Region	Ave.	Imp.	Std.	67%	95%	Region	Ave.	Imp.	Std.	67%	95%
Aberdeen	312.67m	6.86%	147.58	399.38	556.43	SKipMei	240.91m	9.33%	131.32	271.77	532.94
BU	290.47m	2.65%	94.54	332.10	334.12	SShui	338.35m	-12.37%	257.58	399.71	888.10
CWBay	146.74m	2.32%	103.14	176.82	336.39	SWan	157.79m	27.88%	81.25	194.83	296.37
Central	158.62m	0.49%	110.58	184.10	339.23	SSWan	272.08m	12.12 %	137.56	347.49	512.83
CShaWan	146.97m	26.73%	96.32	164.76	306.83	SKWan	329.14m	-16.53%	173.87	490.95	532.15
FoTan	402.18m	4.84 %	169.79	456.40	701.46	TShing	948.90m	-10.69%	1111.49	565.38	2974.00
GCoast	383.13m	22.22%	277.83	503.88	1037.12	TaiO	712.81m	25.94%	184.59	740.93	1092.10
HValley	317.58m	8.76%	230.96	427.02	709.16	TPIndust	517.38m	30.51%	308.73	605.11	1101.29
HungHom	546.70m	4.98%	356.84	651.37	1229.21	TaiWai	164.07m	14.85%	93.23	180.00	354.58
KLBay	197.05m	38.47%	168.80	207.51	616.46	TWoHau	181.62m	19.71%	97.32	212.52	360.58
KLCity	147.73m	19.37%	103.32	173.15	333.67	ThePeak	148.77m	71.27%	260.81	85.42	322.72
KLTong	226.40 m	21.44%	157.49	277.03	518.40	TSWai	502.11m	19.85%	370.92	456.93	1371.52
KFong	145.63m	15.36%	90.25	176.10	324.97	ToLoHway	643.90m	-1.91%	434.01	704.17	1626.60
KTong	187.53m	24.36%	153.66	185.28	586.84	TKwanO	199.26m	4.69%	122.12	212.40	458.49
LKok	147.10m	11.18%	83.00	182.75	304.22	TSTusi	436.32m	5.79%	435.27	547.62	1439.17
LKing	523.22m	44.36%	354.15	748.03	1019.08	TsingYi	315.75m	20.55%	302.93	320.89	853.85
MShan	333.68m	25.56%	287.05	333.19	1057.57	TsuenWan	125.09m	-4.13%	78.24	144.52	291.53
MeiFoo	237.26m	50.04%	160.56	289.97	502.95	TWanShan	290.82m	30.51%	223.56	327.30	733.38
Mongkok	127.63m	34.30%	91.27	147.48	299.32	TuenMun	224.34m	15.66%	199.75	224.58	795.62
NP-QB	236.94m	14.47 %	244.11	224.29	552.63	WanChai	146.74m	17.55%	155.45	144.47	359.20
MuiWo	781.50m	27.28%	373.70	906.82	1562.04	Western	224.28m	17.92%	158.22	248.96	590.85
PChung	644.92m	28.57%	373.68	657.11	1515.97	WTaiSin	418.66m	22.41%	319.66	478.38	1115.56
PE-MK	155.90m	35.59%	97.01	192.93	329.49	YauTong	387.16 m	26.42%	371.44	395.72	1200.14
PEdward	147.72m	35.43%	90.66	180.63	302.70	YYChuen	401.35m	36.41%	323.73	425.02	867.40
SaiKung	193.87m	49.23%	109.23	242.17	399.56	YuenLong	490.65m	4.51%	729.18	175.38	1985.78
SShuiPo	140.84m	37.96%	70.47	165.18	256.69	CheungChau	326.09m	20.99%	259.02	407.42	653.05
ShamTseng	936.52m	-4.57%	664.71	1475.99	1839.85	NgongPing	442.45m	3.89%	91.33	497.00	588.81
ShaTin	325.09m	15.01%	130.74	387.64	549.86						

TABLE II

RESULT OF MDPM WITH BE AND IMPROVEMENT OVER THAT OF MDPM WITH MLE

function, the other is same with the local MLE. And we denote these two estimates as **BE** and **MLE** respectively. And the results of MDPM have shown in the Table (II).

The result of MDPM with **BE** is based on a quadratic loss function, and it is the expectation value of the MS location within a TA zone, while the result of DMPM with **MLE** is the one that maximizes the poster p.d.f. within a TA zone, since we assume that the prior information of the MS location follows a uniform distribution with a TA zone, then the one that maximizes the poster p.d.f. is same with the **local MLE** which reaches a local maximum of its likelihood function.

Based on the results of Table (II), the result of MDPM with **BE** is better than that of MDPM with **MLE** in most regions in Hong Kong. And the result in some regions has great improvement between MDPM with **BE** and MDPM with **MLE**, such as MeiFoo and The Peak, the improvements between these two methods in these two regions are 50.04% and 71.27% respectively. And MDPM also has a good performance in the most seashores and hilly terrains.

However, there remains some regions that the result of MDPM with **BE** is worst than that of MDPM with **MLE**, such as SheungShui, SokKuWan. And in some regions, both MDPM with **BE** and MDPM with **MLE** do not perform well, such as MuiWo , ShamTseng and TaiKooShing. We need to do a further research to analyze these special cases.

D. Compare among others algorithms

In this section, we compare the results of different algorithms and models using real data.

We have done some research about the geometric algorithms, and we have presented some algorithms to provide an estimation of the MS location: CG algorithm, CT algorithm and the Geometric Algorithm and the Iterative algorithm based on EPM. On the other hand, we also proposed a directional signal propagation model and used a modified EM algorithm to provide a MLE as the estimation of the parameters of a directional signal propagation model, then provide a MLE

Model	**Average Error**	**Improvement %**	**Std.**	Sample size	**success ratio %**
CG	482.99m	32.87 %	781.62	116284	93.22%
CT	470.20m	31.05 %	904.24	116284	78.58 %
EPM	359.47m	9.81 %	518.00	116284	79.13%
SPM with MLE	443.73m	26.93 %	461.28	116284	99.15%
SPM with BE	338.50m	4.22 %	375.67	116284	99.21%
DPM with MLE	434.00m	25.29 %	458.02	116284	99.04%
DPM with BE	334.80m	3.16 %	369.61	116284	99.06%
MDPM with MLE	429.86m	24.58 %	454.71	116284	99.07 %
MDPM with BE	324.22m	0%	362.38	116284	99.07%

TABLE III

COMPARE AMONG DIFFERENT ALGORITHMS AND MODELS AND ITS IMPROVEMENT

	SPM	**DPM**	**MDPM**
average	443.73m	434.00m	429.86m
Std.	461.28m	458.02m	454.71m
67%	434.61m	428.57m	420.71m
90%	1074.88m	1049.03m	1037.03m
95%	1543.48m	1509.56m	1496.39m
Minimum	1.20m	1.38m	1.00m
Maximum	2994.80m	2997.11m	2989.49m
Sample size	116284	116284	116284

TABLE IV

MLE RESULT OF DIFFERENT DIRECTIONAL SIGNAL PROPAGATION MODELS

	SPM	**DPM**	**MDPM**
average	338.50m	334.80m	324.22m
Std	375.67m	369.61m	362.38m
67%	331.44m	333.11m	316.62m
90%	736.03m	710.98m	687.15m
95%	1113.43m	1086.60m	1079.41m
Minimum	0.72m	0.86m	0.07m
Maximum	2995.55m	2999.87m	2998.11m
Sample size	116284	116284	116284

TABLE V

BE RESULT OF DIFFERENT DIRECTIONAL SIGNAL PROPAGATION MODELS

Model	MLE	BE	Imp.
SPM	443.73m	338.50m	23.71 %
DPM	434.00m	334.80m	22.86 %
MDPM	429.86m	324.22m	24.58 %

TABLE VI

COMPARE WITH THE RESULTS OF MLE AND BE

as the estimation of the MS location based on this directional signal propagation model. In this paper, we present a new directional signal propagation model—the Modified Directional Propagation Model(MDPM), which derives from the Directional Propagation Model (DPM) and combines the merits of EPM and SPM, and we also provide two Bayes Estimates to provide the estimation of the MS location for two different loss functions. And we denote these two Bayes Estimates as **BE** and **MLE**.

We compare these different algorithms and directional signal propagation models in the following table. We give the value of average of error and its standard deviation and the succuss ratio as the criteria to see the effects of these different algorithms and models. In order to look into the effect of different algorithms, we also provide its improvement between the result of MDPM with **BE** and its compared algorithms. And the results have shown in Table (III).

The succuss ratio includes two cases: one is the assumption of the algorithm, the other is the criteria of our computing. Since some algorithms presented in Table (III) have its model assumptions, if the snapshot information does not fit this model assumptions, it will incur no estimation. For example, EPM with the Geometric Algorithm and EPM with the Iterative Algorithm all have a model assumption, which is that the number of signals we received is greater than 2, since these algorithms all use three BSs information to provide the MS location. On the other hand, in the GSM network, the MS receives the RSS and TA from a serving cell, if an estimation of the MS location we get is not within this TA zone, we think it is a worse case, then we will discard this solution, it also incurs no solution. These two cases which incur no solution of the MS location will reduce our success ratio.

Based on the Table (III), we can draw a conclusion that MDPM with **BE** has the best performance in terms of average of errors among these algorithms. Furthermore, these directional signal propagation models (SPM, DPM and MDPM) have higher succuss ratio than those of these geometric models (CG,CT, EPM). Since these directional signal propagation models have no restriction of the model assumption, even we only received the signal from one antenna, we also can provide an estimation of the MS location by these directional signal propagation models. And these directional signal propagation models with **BE** have done a better performance than those with **MLE** in terms of average error. Since the result of **BE** is the expectation of the MS location within a TA zone, while the result of **MLE** is just the local maximum of its likelihood function within a TA zone. Namely, the result of **MLE** is the one that maximizes the poster pdf within a TA zone, and the result of **BE** is a weighted mean of the MS location and its corresponding probability within a TA zone.

In order to look into the effects of different directional signal propagation models, we also propose three tables to describe its results: Table (IV), Table (V) and Table (VI). Based on Table (IV), Table (V) and Table (VI), we can draw two conclusions. One is that MDPM has done the best performance among these statistical models; the other is that the results of **BE** are better than those of **MLE**.

VII. Conclusions and Future works

In this paper, we have presented a directional signal propagation model—the Modified Directional Propagation Model (MDPM), which derives from DPM and combines the merits of SPM and EPM. And we summarize a uniform formula for the Uniform Directional Signal Propagation Model, which includes all signal propagation models in this paper. We use a view of Bayes statistics to provide a Bayes Estimate of the parameters of this directional signal propagation model, then provide an estimation of the MS location based on our model. Since the Bayes Statistics uses the information of survey data and the prior information about these parameters to provide an estimation, the effect of Bayes Estimate is not worst than the one only uses the information of survey data. But the Bayes Estimate is not unique, it depends on its loss function, namely, the Bayes Estimate depends on a criteria we choose. And we just present a 2-D estimation of the MS location in our paper, but our method is easy to extend into a 3-D space to provide a 3-D solution.

Based on our model, we have proposed two Bayes Estimates to calculate the real data. One is a Bayes Estimate based on a quadratic loss function, we denote it as **BE**; the other is a Bayes Estimate which maximizes the poster p.d.f. of parameters, which is same with the local maximum of the likelihood function of the survey data we have based on we assume that the prior information of the parameter obeys a uniform distribution, we denote it as **MLE**. And **BE** is the expectation value of the MS location and its probability within a TA zone, while **MLE** is the local maximum likelihood function estimate within a TA zone.

We compare our model with some existing models and algorithms. And the results are shown that our model is better than other existing models and algorithms in terms of average error. That is, our model has the best performance in terms of average error among these compared models. Although the number of our model parameters is larger than that of other model, the computational cost does not increase too much since we also present an iterative method to provide a Bayes Estimate of the parameters of our model. It is worth increasing the number of model parameters in order to provide a more accurate model to describe the relationship of a directional signal propagation between the RSS and the surrounding environment.

MDPM is a directional signal propagation model, which has high computational success ratio. However, that the defect of a directional signal propagation model requires more computational cost. While we have presented a geometric model—the Ellipse Propagation Model (EPM), to provide an estimation of the MS location, and this geometric model is more efficient than that of a directional signal propagation model. We need to choose an equilibrium point between efficiency and accuracy in applications.

During our research, we found that signals do fluctuate at the same place. Signal attenuation can be affected by some surroundings conditions, such as weather and car movement. The fluctuating signals will induce more errors in our estimation. In order to provide a more accurate estimation of the MS location, we need to reduce the effect of signal fluctuation. As for our future work, we will try to provide a filtering method to reduce the effect of signal fluctuation.

References

[1] Peter H. Dana, *Global Positioning System Overview*, The University of Texas, http://www.colorado.Edu/geography/gcraft/notes/gps/gps.html.

[2] Richard Walter Klukas, Gerard Lachapelle, and Michel Fattouche, *Cellular Telephone Positioning Using GPS Time Synchronization*, The University of Calgary, http://www.geomatics.ucalgary.ca/Papers/Thesis/GL/97.20114.RKlukas.pdf.

[3] Federal Communications Commission, "Revision of the commission's rules to ensure compatibility with enhanced 911 emergency calling systems," Report and Order and Further Notice of Proposed Rulemaking, Tech. Rep. CC Docket No. 94-102, July 1996.

[4] Kaveh Pahlavan, Prashant Krishnamurthy, *Principles of Wireless Networks a Unified Approach*. Pearson Education, Inc., 2002.

[5] Svein Yngvar Willassen, Steinar Andresen, *A Method of implementing Mobile Station Location in GSM*, Norwegian University of Science and Technology, http://www.willassen.no/msl/bakgrunn.html.

[6] P.Bahl, V.N. Padmanabhan, A. Balachandran, "Enhancements to the RADAR User Location and Tracking System," Microsoft Research, Tech. Rep., February 2000, technical Report MSR-TR-00-12.

[7] N.Bulusu, J.Heidemann, and D.Estrin, "GPS-Less Low Cost Outdoor Localization for Very Small Devices," *IEEE Personal Comm.*, vol. 7, no. 5, pp. 28–34, 2000.

[8] M. Hata, "Emprirical Formula for Propagation Loss in Land Mobile Radio Services," *IEEE Transactions on Vehicular Technology*, vol. 29, pp. 317–325, Aug. 1980.

[9] T.S.Rappaport, J.H.Reed, B.D.Woerner, "Position Location Using Wireless Communications on Highways of the Future," *IEEE Comm. Magazine*, vol. 34, pp. 33–41, 1996.

[10] Teemu Roos, Petri Myllymäki, Herry Tirri, "A Statistical Modeling Approach to Location Estimation," *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 59–69, January-March 2002.

[11] M. McGuire, K. Plataniotis, A. Venetsanopoulos, "Estimating Position of Mobile Terminals with Survey Data," *EURASIP Journal on Applied Signal Processing*, vol. 2002, no. 1, pp. 58–66, January 2002.

[12] Teemu Roos, Petri Myllymäki, Herry Tirri, Pauli Misikangas, Juha Sievänen, "A Probabilistic Approach to WLAN User Location Estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, July 2002.

[13] M. McGuire, K. Plataniotis, A. Venetsanopoulos, "Estimating Position of Mobile Terminal from Path Loss Measurements with Survey Data," *Wireless Communications and Mobile Computing*, vol. 3, no. 1, pp. 51–62, February 2003.

[14] Junyang Zhou, Kenneth Man-Kin Chu, Joseph Kee-Yin Ng, "Providing Location Services within a Radio Cellular Network using Ellipse Propagation Model," in *Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA 2005)*, Taipei, Taiwan, March 28-30 2005, pp. 559–564.

[15] Joseph K. Ng, Stephan K. Chan, And Kenny K. Kan, "Location Estimation Algorithms for Providing Location Services within a Metropolitan area based on a Mobile Phone Network," in *Proceedings of The 5th International Workshop on Mobility Databases and Distributed Systems(MDDS 2002)*, Aix-en-Provence, France, September 2002, pp. 710–715.

[16] Joseph Kee-Yin Ng, Stephen Ka Chun Chan, and Shibin Song, "A Study on the Sensitivity of the Center of Gravity Algorithm for Location Estimation," Hong Kong Baptist University, Tech. Rep., May 2003, http://www.comp.hkbu.edu.hk/tech-report/tr03014f.pdf.

[17] Kenny K.H. Kan, Stephen K,C. Chan, and Joseph K. Ng, "A Dual-Channel Location Estimation System for providing Location Services based on the GPS and GSM Networks," in *Proceedings of The 17th International Conference on Advanced Information Networking and Applications(AINA 2003)*, Xi'an, China, March 2003, pp. 7–12.

[18] Kenneth M. Chu, Karl R.P.H. Leung, Joseph K. Ng, and Chun H. Li, "Locating Mobile Stations with Statistical Directional Propagation Model," in *Proceedings of the 18th International Conference on Advanced Information Networking and Applications (AINA 2004)*, Fukuoka, Japan, March 2004, pp. 230–235.

[19] Junyang Zhou, Kenneth Man-Kin Chu, Joseph Kee-Yin Ng, "An Improved Ellipse Propagation Model for Location Estimation in facilitating Ubiquitous Computing," in *Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, Hong Kong, Aug. 17-19 2005.

[20] Kenneth M. Chu, and Joseph K. Ng, "Estimating Propagation Parameters using a Modified EM Algorithm for Mobile Location Estimation," Hong Kong Baptist University, Tech. Rep., Nov. 2004, http://www.comp.hkbu.edu.hk/tech-report/tr04007f.pdf.

[21] M. McGuire, K. Plataniotis, A. Venetsanopoulos, "Data Fusion of Power and Time Measurements for Mobile Terminal Location," *IEEE Transactions on Mobile Computing*, vol. 4, no. 2, pp. 58–66, March-April 2005.

[22] "Inverse square law," http://hyperphysics.phy-astr.gsu.edu/hbase/forces/isq.html.

[23] T. Rappaport, *Wireless Communications: Principles & Practice*. New Jersey: Prentice Hall PTR, 2002.

[24] Samuel Kotz, Wu Xizhi, *Mordern Bayesian Statistics*. China Statistics Press, 2000, ch. 2, p. 9.

[25] James O. Berger, Jia Naiguang, Wu Xizhi, *Statistical Decision Theory and Bayesian Analysis*. China Statistics Press, 1998, ch. 4, pp. 176–178.

[26] A.P. Dempster, N.M.Laird, D.B.Rubin, "Maximun Likelihood from Imcomplete Data via the EM Algorithm," *J.Royal Statistical Socity*, pp. 1–38, 1977.

[27] Chen Xiru, *Introduction of Mathematic Statistics* . Science Press, China, 1997.

[28] Chen Chuanzhang, Jin fulin, Zhu Xueyan, OYang Guangzhong, Ed., *Mathematics Analysis Lecture (The first Part)*, 2nd ed. High Education Publishing Company, 1993, ch. 5, pp. 174–175.

Processing Precision-Constrained Approximate Queries in Wireless Sensor Networks

Minji Wu Jianliang Xu
Hong Kong Baptist University
Kowloon Tong, Hong Kong
{alexwu,xujl}@comp.hkbu.edu.hk

Xueyan Tang
Nanyang Technological University
Singapore
asxytang@ntu.edu.sg

Abstract

A lot of efforts have been devoted to improving energy efficiency for wireless sensor networks by exploring distributed data storage and in-network query processing techniques. In this paper, we propose a generic two-tier data storage strategy for answering precision-constrained approximate queries in a sensor network. The basic idea is to keep two versions of data in the network. A high-precision version is kept at the sensor node that captures the data while a low-precision version is maintained at the base station. We develop query processing and node refreshment strategies for various types of approximate queries under the two-tier storage. Our extensive experiments show that the two-tier storage strategy outperforms the basic centralized storage scheme by an order of magnitude in terms of network lifetime under various system configurations.

1 Introduction

The rapid development in sensing and wireless communication technologies has made the availability of wireless sensor networks. Wireless sensor networks can be used in a wide range of practical applications such as habitat monitoring and environment monitoring. For example, the conservation of endangered species in Hong Kong (such as Chinese White Dolphins and Romer's Tree Frog) is hampered by insufficient knowledge on their status [19]. The current practice is to send human beings to the habitat sites of these species to collect their status information. However, this approach introduces potential disturbance. The use of networked sensors not only eliminates the potential impacts of human presence, but also enables data collection at scales and resolutions that are difficult to achieve through traditional instrumentation [17].

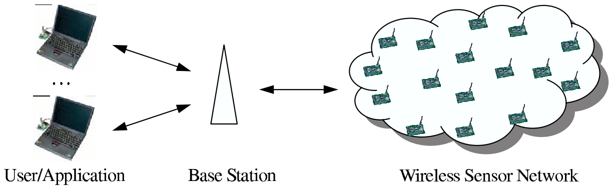


Figure 1. Sensor Network Architecture

A wireless sensor network is typically constructed of a base station and a large number of sensor nodes scattered in an area of interest (see Figure 1). These sensor nodes are equipped with sensing, data processing, and communication components to collect local measurements, process and exchange information about the environment. They are usually battery powered. Replacing the batteries is not only costly but also impossible in many situations. As such, energy efficiency is a critical consideration in the design of sensor networks. There have been significant research efforts towards energy-conserving sensor networks. However, most of the existing studies have focused on the design of sensing architectures and protocols in support of *exact* answers to user queries.

Here we take a different approach to improve energy efficiency. We exploit the trade-off between data quality and energy consumption to extend network lifetime by investigating approximate queries with precision guarantees. Many sensor applications are willing to tolerate a certain degree of error in data due to either the application nature or the high resource constraints in sensor networks. For example, to save energy, a user retrieving the temperature reading of a sensor may allow an error of one degree. In this case, the retrieved value is acceptable as long as it is within ±1 degree of the actual reading.

In this paper, we propose a generic two-tier data storage strategy for answering precision-constrained approximate queries. The basic idea is to keep two versions of data in the network. A high-precision version is kept at the sensor node that captures the data. Meanwhile, the same data with a lower precision is replicated at the base station. The imprecision of low-precision data at the base station is bounded by an *approximation range*. An update in reading will not be sent to the base station if the new value remains within the approximation range. Thus, a query can be answered by the base station if the user's required precision is weaker than that of the result computed based on low-precision data. Otherwise, some of the nodes need to be refreshed to improve the data precision, in which two fundamental issues arise: 1) How to determine the to-refresh node set? As the costs for sensor nodes to report their readings to the base station differ from one another, it is important to pick up the set of sensor nodes that incurs the minimal energy consumption. 2) Upon deciding the set of to-refresh nodes, how to refresh them in an energy-efficient way? In some types of queries, we do not need to refresh all the nodes in the to-refresh set to resolve the answer.

We develop detailed query processing and node refreshment strategies for various types of approximate queries (including ID-based, range, top-k, and aggregate queries) under the proposed two-tier data storage. Extensive experiments are conducted to evaluate the performance of the two-tier storage strategy using real trace data. The results show that the two-tier storage strategy substantially outperforms the basic centralized storage and local storage schemes under various system configurations.

The rest of this paper is organized as follows. Section 2 reviews the related work. Section 3 gives some preliminaries of the work. Section 4 presents the proposed two-tier data storage strategy and develops the query processing and node refreshment techniques for various types of queries. We evaluate the performance of our proposed techniques in Section 5. Finally, Section 6 concludes this paper.

2 Related Work

Distributed data storage for wireless sensor networks has been investigated in the literature [3, 10, 15, 21]. In the TinyDB project, Madden *et al.* [10] proposed a pull-based acquisitional query processing (ACQP) model, where the sensors control where, when, and how often the data is acquired and delivered to query processing operators. The Cougar project [3] employed a hybrid pull-push model: sensed data is pushed to some selected view nodes, from which the data is pulled by queries. Ratnasamy *et al.* [15] proposed a data-centric storage model: the sensor reading is pushed to the sensor node nearest to some geographical location hashed from a predefined key. Only equality queries are supported by data-centric storage.

In-network query processing techniques have been studied for various data storage models [5, 7, 22]. These studies examined exact queries only. Query evaluation techniques over imprecise data have been investigated by Lazaridis and Mehrotra [8], which quantified data quality with *set-based uncertainty* and *value-based uncertainty*. They then proposed a cost efficient processing technique for quality-aware relational queries. However, how to collect imprecise data was not discussed in [8]. Deshpande *et al.* [2] have most recently developed a model-driven data acquisition architecture that employs statistical modelling techniques to efficiently answer one-shot queries with high confidence.

In-network data aggregation, where data values are aggregated as forwarded by the network, has been receiving increasing attention recently [1, 9]. Continuous precision-constrained aggregate queries were studied in [4, 16]. The key issue is how to allocate the error budget to the sensor nodes involved in the aggregation tree. Sharaf *et al.* [16] implemented a uniform error allocation scheme. Deligiannakis *et al.* [4] improved it by developing an adaptive algorithm to allocate more error tolerances to the nodes that can reduce more network traffic. Neither of these studies considered balancing the energy consumption of sensor nodes to extend network lifetime. In a recent work [18], we proposed an error allocation algorithm to optimize network lifetime. However, these techniques developed for continuous queries are not applicable to one-shot queries. To the best of our knowledge, this is the first effort to investigate one-shot approximate queries for wireless sensor networks.

Other related work includes querying approximate data over distributed caches and streams. Olston *et al.* [13] studied error-bounded aggregate queries over distributed data streams [13]. An adaptive scheme for precision adjustment at each individual source was proposed to reduce the communication cost. Inspired from [14], Han *et al.* [6] developed an adaptive precision setting algorithm for precision-constrained data collection in a single-hop sensor network. However, their work was confined to collection of individual sensor readings. In contrast, this paper proposes a generic two-tier data storage strategy in support of various types of queries in multi-hop sensor networks.

3 Preliminaries

We assume the wireless sensor network is composed of a base station and many sensor nodes. Each sensor node measures the local physical phenomena (e.g., temperature, humidity, and light) at a *fixed* sampling rate and reports to the base station if necessary. The base station and sensor nodes are equipped with wireless interfaces to communicate with each other. Since the wireless transmission range is limited, a routing infrastructure (such as TAG tree [9]) is established to relay data between the base station and the sensor nodes in the network.

The base station serves as an interface for external users to pose queries to the sensor network. Users are interested in various types of precision-constrained approximate queries, e.g. (their definitions will be detailed in Section 4):

Q1 Find the temperature reading (within ±1°C) of Sensor Node 1. (ID-based Query)

Q2 Find the sensors whose temperature readings are above 100°C (within an error of 5°C). (Range Query)

Q3 Find the k sensors (within an error of 1°C) with the highest temperature readings. (Top-k Query)

Q4 Find the average temperature reading (within an error of 1°C) of the sensors. (Aggregate Query)

To answer these queries, two basic data storage strategies exist:

- **Centralized Storage (CS):** Each sensor node reports to the base station whenever a new reading is sampled. Note that the report of sensor readings cannot make use of the in-network aggregation technique. When the base station receives a query, the result can be immediately computed based on the stored up-to-date readings.
- **Local Storage (LS):** The sensed data is stored on the local node only. When the base station receives a query, the query is sent to the node involved (for ID-based queries) or flooded throughout the whole network (for range, top-k, and aggregate queries). The query result is then collected by the base station via the routing infrastructure. For top-k and aggregate queries, the result collection can take advantage of in-network aggregation to save energy costs.

Both of these two strategies have some performance disadvantages. The CS strategy suffers from a high update cost while the LS strategy incurs a high querying traffic. Furthermore, they do not take advantage of the error allowed in the query answer to improve system performance. In the next section, we propose a more efficient data storage strategy.

4 Two-Tier Data Storage

4.1 Overview

Taking advantage of users' error tolerances, we propose a generic *two-tier* data storage strategy to support processing of various types of approximate queries (including ID-based, range, top-k, and aggregate queries). The base station serves as the first tier (referred to as *centric storage*) that stores imprecise sensed data, while each sensor node serves as the second tier (referred to as *local storage*) that stores exact up-to-date data. Consider a sensor node i. The imprecision of the data stored at the base station is bounded by a certain error represented by an *approximation range*, i.e., a stored value v_i with an approximation range of e_i means that the actual value must lie in the *approximate interval* $[l_i, h_i]$, where $l_i = v_i - \frac{e_i}{2}$ and $h_i = v_i + \frac{e_i}{2}$. At each sampling instance, if the newly sensed value v'_i is within a difference of $\frac{e_i}{2}$ from the previously reported value v_i, the new value v_i is kept at the sensor node locally, otherwise an update message is sent to the base station to replace v_i by v'_i in the centric storage (this is called *source-initiated update*). In this way, a lot of updating traffic can be saved. However, if the precision of stored data is insufficient to answer a query issued to the base station, we will have to refresh the data from the local storage (this is called *query-initiated refreshment*), which incurs communication overhead.

The general query processing under the two-tier storage takes three steps. First, the base station computes a tentative result based on stored imprecise data. Second, if the tentative result is not sufficiently precise, the base station refreshes the readings from a (selected) subset of the sensor nodes. After refreshment, the approximation ranges of those refreshed nodes are shrunk to zero. Note that the refreshed values remain up-to-date till the next sampling instance, after which the approximation range returns to e_i. Finally, the base station re-evaluates the query based on refreshed data. In the following, we detail the query processing techniques for different types of queries.

4.2 ID-based Query

An approximate ID-based query is interested in the reading of a particular sensor node (e.g., Node 1), with a precision constraint of R. It is acceptable as long as the returned value is within a deviation of R of the true reading.

Recall that the sensor reading kept at the base station is in the form of $[l_i, h_i]$. If $R \geq h_i - l_i$, meaning the stored data has a higher precision than the expected, it is immediately returned to the user. Otherwise, the stored data does not meet the precision requirement, we have to send a refresh message to the desired sensor node to probe its latest reading. By doing so, we shrink the approximation range to zero (till the next sampling instance), thereby satisfying the precision requirement of the query.

4.3 Range Query

In this type of queries, we are interested in the sensor nodes whose readings are within a specified range $[L, H]$. With a precision constraint of R, we are required to find out all sensor nodes whose readings are in $[L + R, H - R]$ and to exclude those whose readings are not in $[L - R, H + R]$. The nodes whose readings are within $[L - R, L + R]$ or $[H - R, H + R]$ may or may not be returned.

By examining the approximate value $[l_i, h_i]$ of each node i, we divide the sensor nodes into three groups T^+, T^-, $T^?$, which respectively represent the nodes who can be returned, the nodes who must not be returned, and the rest. A node i is categorized in T^+ if $l_i > L-R$ and $h_i < H+R$. It is categorized in T^- if $h_i < L+R$ or $l_i > H-R$. If none of these conditions is satisfied, the node is categorized in $T^?$. The nodes in $T^?$ must be refreshed because we are not sure whether they should be included in the query result.

Take Q2 as an example, in which the query asks for the nodes whose readings are greater than 100°C with a precision constraint of 5°C. Thus, as illustrated in Figure 2, for the nodes that hold $l_i >$ 95, we throw them into T^+, and for the nodes who hold $h_i <$ 105, we throw them into T^-. We will refresh all the other nodes and combine the qualified nodes with T^+ as the final result.

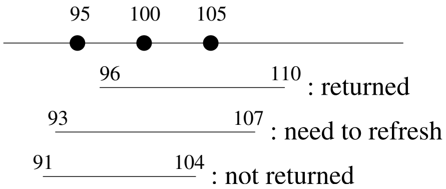


Figure 2. Processing Range Query

4.4 Top-k Query

In a top-k query, the user wants to get the k nodes with the highest (or lowest) readings. Recall that the reading of node i is approximated with an interval of $[l_i, h_i]$. Given a precision constraint of R, an approximate top-k query retrieves the (ordered) set of sensor nodes $\mathcal{T}$ with the highest readings:

$$\mathcal{T} =< n_1, n_2, \cdots, n_k >,$$

where $\forall i > j, h_{n_i} \leq l_{n_j} + R$ and $\forall l \neq n_i(i = 1, 2, \cdots, k), h_l \leq min\{l_{n_1}, l_{n_2}, \cdots, l_{n_k}\} + R$.

The evaluation of an approximate top-k query is much different from that of the previous two types of queries. Given an ID-based query or range query, the set of to-refresh nodes is uniquely determined. However, for a top-k query, whether a node needs to refresh depends on the relative order of its reading against the other sensor nodes. We divide the refreshing process into two steps: selecting to-refresh nodes and processing refreshment.

When the base station receives a top-k query with a precision constraint of R, it sorts the sensor nodes based on their current approximate readings. Without loss of generality, the nodes are sorted by the upper bounds of their approximate intervals. Suppose $n_1, n_2, \cdots, n_k$ is the tentative top-k list. We will return this list immediately if no node in the list has an overlap with any other node by greater than R in the approximate interval. Otherwise, we need to refresh some nodes to resolve the top-k order. To do so, we define *refreshing candidates* (RC_i) with respect to each node i in the tentative top-k list as follows:

$$RC_i = \begin{cases} \emptyset & \text{if } \forall j, h_j - l_i \leq R, \\ \{i\} \cup \{j \mid h_j - l_i > R\} & \text{otherwise.} \end{cases}$$

Note that the refreshing candidate sets with respect to different nodes may overlap. Figure 3 shows an example top-2 query among 4 nodes (with approximate intervals of [5, 8], [3, 7], [2, 6], and [1, 5], respectively). Assume the precision constraint $R = 1$. The RC_i sets for nodes 1 and 2 are $\{1, 2\}$ and $\{2, 3, 4\}$, respectively.

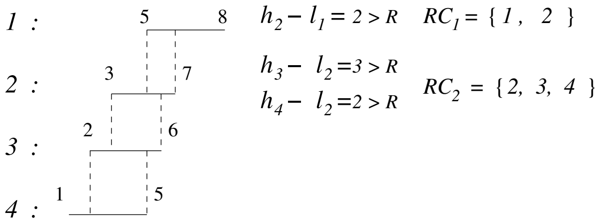


Figure 3. Finding out RC_i in Top-k Query

A straightforward refreshment strategy is to refresh all nodes in $RC = \bigcup_{i=1}^{k} RC_i$. We call it *full refreshment*. However, this might not be necessary because the refreshments of some nodes may eliminate the need to refresh other nodes. Consider the early example. $RC = \{1, 2, 3, 4\}$. Suppose we choose to refresh node 2 first, and assume that the current reading of node 2 is 5.5. After refreshment, the approximation interval $[3, 7]$ of node 2 is replaced by its exact reading of 5.5. Hence, RC_1 and RC_2 are updated with empty sets. Thus, we can assert that the top-2 list is $\langle 1, 2 \rangle$ without refreshing nodes 1, 3, and 4 anymore.

This fact suggests that we can refresh in rounds. In each round, we choose a subset of RC to refresh. When we get the refreshed reading(s), we update the RC_i set for each node i in the tentative top-k list. This process is repeated until all the RC_i sets become empty. We propose two round-based refreshment strategies:

- **Batch:** Starting from the top-1 node, we refresh the RC_i set of one top-k node in each round.
- **Sequential:** We refresh one node per round. In each round, we select to refresh the node that appears in most RC_i sets. We expect that refreshing such a node is most helpful in resolving the order confusion.

4.5 Aggregate Query

There are five types of standard aggregate queries: SUM, MAX, MIN, COUNT, AVG. The MAX and MIN queries can be viewed as top-1 queries. The COUNT query returns the cardinality of the sensor nodes and can always be computed exactly. The AVG and SUM queries differ by only a constant which is the number of sensor nodes. Therefore, we shall focus our discussion on the SUM query here.

If the reading of each sensor node i maintained at the base station has an approximation range e_i, the SUM aggregation can be computed with an approximation range $E = \sum_{i=1}^{n} e_i$, where n is the number of sensor nodes in the network.

If the query precision constraint R is greater than E, the result is returned by the base station without refreshing the reading of any sensor node. Otherwise, if R is smaller than E, some sensor readings have to be refreshed to refine the query result. Let T be the to-refresh node set. Since refreshing the reading of a sensor node reduces its approximation range to zero, to meet the precision requirement of the query, T must satisfy

$$\sum_{i \in T} e_i \geq E - R. \quad (1)$$

The refreshment can make use of in-network aggregation to improve energy efficiency. Specifically, on receiving up-to-date readings from more than one children, an intermediate node aggregates the readings before forwarding them upstream. For SUM aggregation, the partial aggregate result is simply the sum of the readings received. In-network aggregation cuts down the volume of data sent over the upper-level links in the routing tree.

We define the subtree rooted at each child of the base station as a *region*. Since these children relay packets between the base station and the other nodes in their respective regions, they consume much more energy than the others. We therefore call these nodes the *hot-spot nodes*. In order to prolong the network lifetime, we should conserve the energy at these hot-spot nodes. This implies the following design philosophy of refreshing:

- We should distribute the to-refresh nodes in as few regions as possible. This is because due to in-network aggregation, the volume of data sent by a hot-spot node to the base station is independent of the number of sensor nodes refreshed in the corresponding region. To save the energy consumption at hot-spot nodes, it is desirable to reduce the number of regions involved in the refreshment.
- When selecting regions for refreshment, we favor those with more residual energy.
- When the number of to-refresh sensor nodes is smaller than that in one region, we should choose the nodes that lie closer to the base station. This helps reduce the number of sensor nodes involved in relaying the up-to-date readings and thus the network-wide total energy consumption.

We propose to construct the to-refresh node set as follows. Starting from an empty to-refresh node set, we continue to insert nodes into the set until the total approximation range of the nodes in the set adds up to $E - R$. In this process, the regions are sequentially examined. For each region, all nodes in the region are inserted to the to-refresh node set if the insertion does not make the total approximation range greater than $E - R$. Otherwise, only a subset of the nodes in the region are inserted to increase the total approximation range of the to-refresh node set to $E - R$. The subset of nodes are selected in increasing order of their distances to the base station. We propose two examination orders of the regions:

- **Max-Size:** Our first strategy favors large regions to minimize the number of regions involved in the refreshment, i.e., the regions are examined in decreasing order of their sizes.
- **Max-Energy:** The second strategy favors the regions with more residual energy to balance the energy consumption among regions. That is, we examine the regions in descending order of the residual energy of their hot-spot nodes. Note that since the hot-spot nodes are located near the base station, it is practically easy to maintain the residual energy of these nodes (e.g., by piggybacking the energy information on the refresh messages).

5 Performance Evaluation

5.1 Simulation Setup

We have developed a simulator based on ns-2 (version 2.26) [11] and NRL's sensor network extension [12] to evaluate the proposed two-tier storage strategy. The simulator includes the detailed models of the MAC and physical layers for wireless sensor networks. The sensor nodes can operate in one of three modes: sending message, receiving message, and sleeping. These modes differ in energy consumption. The energy consumption for sending a message is determined by a cost function: $s \cdot (\alpha + \beta \cdot d^q)$, where s is the message size, α is a distance-independent term, β is the coefficient for a distance-dependent term, q is the component for the distance-dependent term, and d is the distance of message transmission. We set α=50 nJ/b, β=100 pJ/b/m^2, and q=2 in the simulation. The energy consumption for receiving a message is given by $s \cdot \gamma$, where γ is set at 50 nJ/b. The power consumption in sleeping mode is set at 0.016 mW. For simplicity, the energy overhead of mode switching is ignored. We set the size of a data update message at 8 bytes, and the size of a refresh message at 4 bytes. The initial energy budget at each sensor node was set at 0.1 Joule.

We simulated a multi-hop network of 120 sensor nodes (see Figure 4 for the layout). The sensor readings were simulated using real traces provided by the Live from Earth and Mars (LEM) project [24] of University of Washington. We used the temperature and humidity traces logged

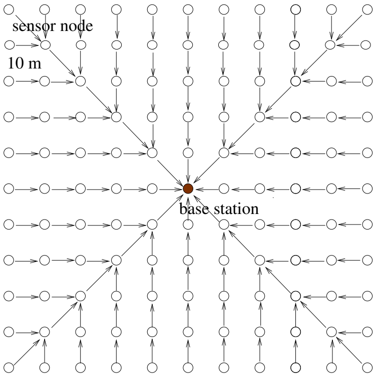


Figure 4. Network Layout

by the station at the University of Washington from Aug. 2004 to Aug. 2005 in our experiments. Each trance consists of more than 500,000 readings captured at a sampling interval of one minute. We extracted many different subtraces starting at randomly selected timepoints. Each subtrace contained 20,000 readings. The subtraces were used to simulate the physical phenomena in the immediate surroundings of different sensor nodes. In the simulation, the interval between two successive readings was assumed to be one time unit. The following two metrics are used in the performance comparison:

- **Network Lifetime**: As in the previous work [20, 23], the network lifetime is defined as the time duration before the first sensor node runs out of power. It serves as the primary metric in the performance evaluation.
- **Average Energy Consumption**: It is defined as the total amount of energy consumed in the network averaged for all sensor nodes.

In what follows, we first evaluate the query processing heuristics developed for top-k queries and aggregate queries. We then compare the performance of the proposed two-tier storage strategy against the basic centralized storage and local storage schemes with mixed types of queries.

5.2 Refreshment Strategies for Top-k Queries

In this section, we evaluate the performance of the three node refreshment strategies (proposed in Section 4.4) for top-k queries. We set k at 5 and the query precision constraint of each query at a value randomly selected from an interval of [0,1]. The temperature trace was used in this set of experiments. Figure 5 plots the network lifetime under different approximation range settings. When the approximation range is smaller than 0.5 (i.e., the stored data at the base station is relatively precise), the refreshment strategies have a similar performance since refreshments are rarely needed. With increasing the approximation range, the three strategies achieve different network lifetimes. The batch and sequential strategies are much better than (sometimes double the lifetime of) the full refreshment. By maximizing the utility of each refreshment, the sequential refreshment shows the best performance in all cases tested. It is also interesting to observe that the performance curve of each strategy forms a '$\cap$' shape. The network lifetime shortens when the approximation range is set too small (due to a large amount of source-initiated updates) or too large (due to a large amount of query-initiated refreshments). This suggests there exists an optimal setting of approximation range. We leave the study of optimization of the approximation range as an important future work.

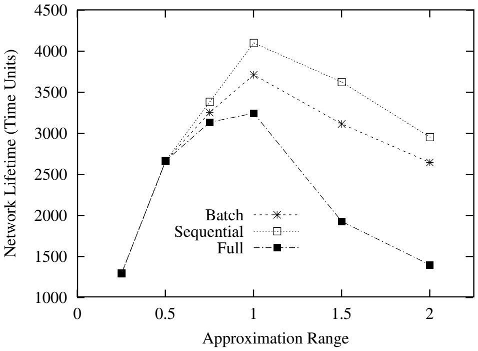


Figure 5. Performance for Top-k Queries

5.3 Node Selection for Aggregate Queries

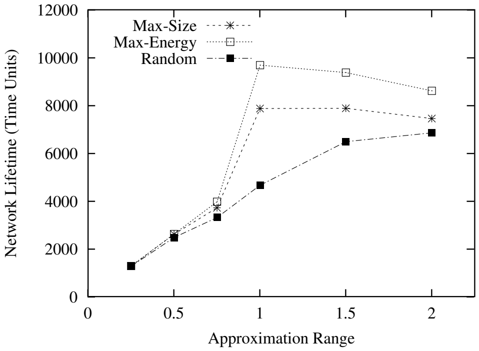
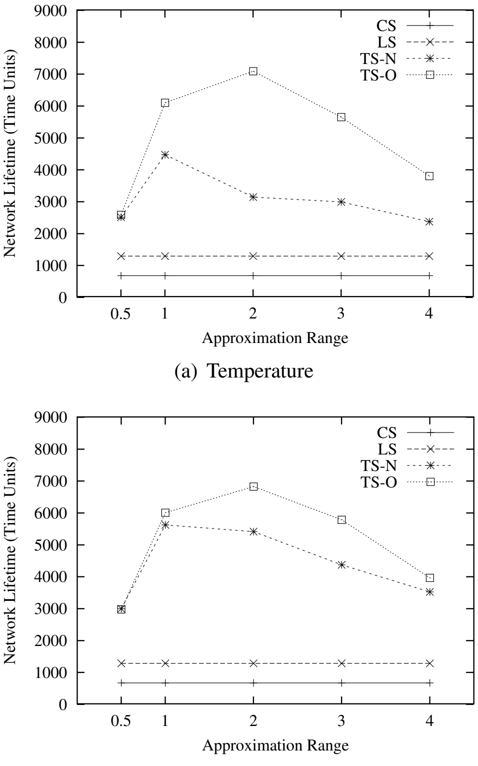


Figure 6. Performance for Aggregate Queries

In this section, we evaluate the schemes for selection of to-refresh nodes in processing aggregate queries. In addition to the Max-Size and Max-Energy schemes proposed in Section 4.5, we also include a *Random* selection scheme for comparison. The Random scheme randomly selects the to-refresh nodes, which serves as a baseline scheme. We tested AVG aggregate queries using the temperature trace with precision constraints uniformly distributed in the range of [0, 1]. Figure 6 shows the result under different approximation range settings. As observed in the last subsection, when the approximation range is small, all the three schemes show a similar performance. Their performance differences are obvious with an approximation range larger than 0.5. Clearly, the Max-Energy scheme beats the other two schemes by more than 25%. Max-Energy performs better than Max-Size, which implies that it is more important to balance the energy consumption of each region rather than the overall network traffic.

5.4 Performance Evaluation of Two-Tier Storage



(a) Temperature

(b) Humidity

Figure 7. Lifetime vs Approximation Range

In this section, we evaluate our proposed two-tier storage (TS) against the basic centralized storage (CS) and local storage (LS) strategies under a mixed-type query environment. We simulated four types of queries, i.e., ID-based, range, top-10, and AVG queries. We set the query rate at one per time unit by default. At each querying instance, a query type is randomly selected among the four types and a precision constraint is set to a random value in the range of [0, 1]. We evaluate two versions of TS strategy: *TS-O* in which the best node selection and refreshment schemes (i.e., Sequential and Max-Energy) for top-k and AVG queries are used, and *TS-N* in which the basic node selection and refreshment schemes (i.e., Full and Random) are used. As shown in Figures 7a and 7b, both TS-O and TS-N substantially outperform CS and LS. In particular, TS-O improves the lifetime against CS by an order of magnitude and against LS by several times. For a similar reason explained in Section 5.2, the network life increases first and drops next as the approximation range increases.

Figure 8 shows the average energy consumption for each of the storage strategies under comparison. It is interesting to observe that the improvement of TS-O over CS, LS, TS-N in terms of energy consumption is not as high as that in terms of network lifetime. This suggests that the node selection and refreshment schemes in TS-O are particularly optimized for the metric of network lifetime.

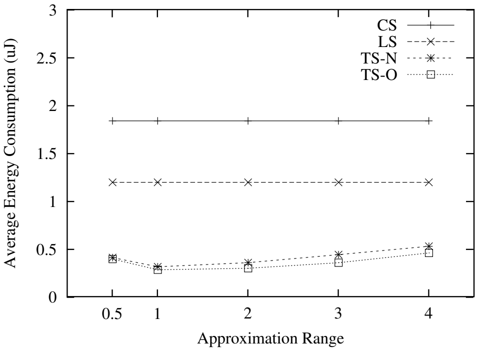


Figure 8. Energy vs Approximation Range (Temperature)

We also evaluate the proposed two-tier storage with different query patterns. Figure 9 shows the result by varying the query rate. The approximation range was set at 1. Again, TS-O and TS-N perform much better than CS and LS. As expected, all the storage strategies except CS deteriorate with increasing query rate. When the query rate is increased from 0.25 to 2, compared to LS, the performance downgrade of TS-O and TS-N is a bit smaller (i.e., 56% and 58% vs 66%). Figure 10 plots the result with different settings of query precision constraints. While CS and LS remain constant in performance as the precision constraint is relaxed, TS-O and TS-N can take advantage of the relaxed precision requirement to further improve network lifetime significantly.

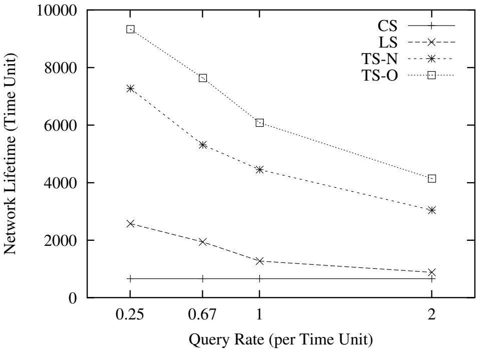


Figure 9. Lifetime vs Query Rate (Temperature)

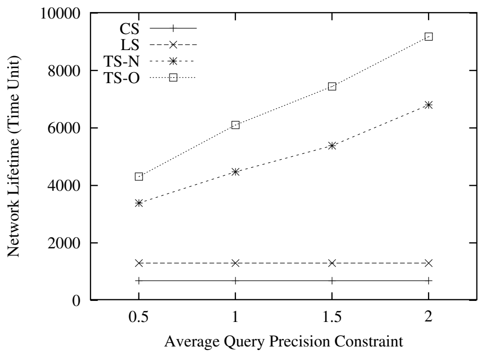


Figure 10. Lifetime vs Precision Constraint (Temperature)

6 Conclusion

This paper has proposed a two-tier data storage strategy in support of precision-constrained approximate queries in wireless sensor networks. By storing high-precision data at the sensor nodes while maintaining low-precision duplicates at the base station, the proposed strategy attempts to balances the energy consumption between data updating and querying. We have developed the query processing and node refreshment strategies for various types of approximate queries under the two-tier storage. Extensive experiments have been conducted to evaluate the performance of the proposed two-tier storage strategy using real trace data. The results show that the two-tier storage strategy soundly outperforms the basic centralized storage and local storage schemes.

As for future work, we are going to investigate the optimal setting of approximation range for the two-tier storage. This paper did not consider the query predicates; we plan to extend the work to the queries with predicates. We also plan to build a small-scale testbed using Motes to measure the performance of different storage strategies.

References

[1] J. Considine, F. Li, G. Kollios, and J. Byers. Approximate aggregation techniques for sensor databases. In *IEEE ICDE*, March 2004.

[2] A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *VLDB*, August 2004.

[3] A. Demers, J. Gehrke, R. Rajaraman, J. Trigoni, and Y. Yao. The Cougar project: A work-in-progress report. In *SIGMOD Record*, 32(4): 53-59, Dec. 2003.

[4] A. Deligiannakis, Y. Kotidis, and N. Roussopoulos. Hierarchical in-network data aggregation with quality guarantees. In *EDBT*, March 2004.

[5] J. Gehrke and S. R. Madden. Query processing in sensor networks. *IEEE Pervasive Computing*, 2004.

[6] Q. Han, S. Mehrotra, and N. Venkatasubramanian. Energy efficient data collection in distributed sensor environments. In *IEEE ICDCS*, March 2004.

[7] X. Li, Y. J. Kim, R. Govindan, and W. Hong. Multi-dimensional range queries in sensor networks. In *ACM SenSys*, Nov. 2003.

[8] I. Lazaridis and S. Mehrotra. Approximate selection queries over imprecise data. In *IEEE ICDE*, March 2004.

[9] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. TAG: A tiny aggregation service for ad-hoc sensor networks. In *USENIX OSDI*, Dec. 2002.

[10] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. The design of an acquisitional query processor for sensor networks. In *ACM SIGMOD*, June 2003.

[11] The network simulator - ns-2. http://www.isi.edu/nsnam/ns/.

[12] NRL's sensor network extension to ns-2. http://nrlsensorsim.pf.itd.nrl.navy.mil/.

[13] C. Olston, J. Jiang, and J. Widom. Adaptive filters for continuous queries over distributed data streams. In *ACM SIGMOD*, June 2003.

[14] C. Olston, B. T. Loo, and J. Widom. Adaptive precision setting for cached approximate values. In *ACM SIGMOD*, May 2001.

[15] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu. Data-centric storage in sensornets with GHT, a geographic hash table. *ACM/Kluwer MONET*, 8(4), 2003.

[16] M.A. Sharaf, J. Beaver, A. Labrinidis, and P.K. Chrysanthis. TiNA: A scheme for temporal coherency-aware in-network aggregation. In *ACM MobiDE*, Sept. 2003.

[17] R. Szewczyk, *et al.* Habitat monitoring with sensor networks. *Communications of ACM*, June 2004.

[18] X. Tang and J. Xu. Extending network lifetime for precision-constrained data aggregation in wireless sensor networks. In *IEEE INFOCOM*, April 2006.

[19] World Wildlife Fund - Hong Kong. http://www.wwf.org.hk/eng/conservation/spe_cons/.

[20] M. Wu, J. Xu, X. Tang, and W.-C. Lee. Monitoring top-k query in wireless sensor networks. In *IEEE ICDE*, April 2006. (Poster)

[21] J. Xu, and X. Tang, and W.-C. Lee. EASE: Energy-conserving Approximate StoragE for querying object tracking sensor networks. In *IEEE SECON*, Sept. 2005.

[22] Y. Xu, W.-C. Lee, J. Xu, and G. Mitchel. Processing window queries in wireless sensor networks. Proc. In *IEEE ICDE*, April 2006.

[23] O. Younis and S. Fahmy. Distributed clustering for ad-hoc sensor networks: A hybrid, energy-efficient approach. In *Proc. IEEE INFOCOM*, March 2004.

[24] Live from Earth and Mars (LEM) Project. http://www-k12.atmos.washington.edu/k12/grayskies/.

Algorithm to support Temporal Consistency of Sensor Data for Transaction Processing Applications in Broadcast Environments [*]

Chui Ying Hui, Joseph Kee-Yin Ng

Department of Computer Science, Hong Kong Baptist University,
Kowloon Tong, Hong Kong
Tel:(852)-3411-7864 Fax:(852)-3411-7892
Email:{cyhui, jng}@comp.hkbu.edu.hk

Abstract

This paper presents a performance study on various broadcast algorithms in a Real-Time Information Dispatch System. The objective of the study is to design efficient broadcast program scheduling algorithm for providing fresh data in timely manner for applications connecting to sensor networks. We propose and perform a series of simulation experiments. Simulation results show that our proposed broadcast algorithm can further reduce the probability of mobile read only transactions missing deadlines in broadcast environments.

1 Introduction

The demand for real-time information in many emerging applications fueled by the Internet, mobile networks and sensor networks has been increasing. In real-time computing, transaction correctness includes using data that is timing-consistent in addition to the timing constraints imposed to a transaction. Examples are monitoring and tracking systems with data feed from sensor networks. In these applications, data may change continuously to reflect the real world state such as current location or stock prices and queries for such real-time data have to be responded in a short and predictable time limit. Also these applications are connected to sensor networks via a wireless framework, the timing-consistency between the device readings and the current values manipulated by transactions is of equivalent importance to the serializability. In these applications, the requirement for the timely interaction between transactions submitted by mobile clients and the data collected by sensors embedded in the physical world is crucial to the high performance of the systems. Current transaction processing systems, which are not time-cognizant, are poor in supporting timeliness requirement and temporal consistency of real-time data. This paper develops a transaction algorithm for providing fresh data in timely manner for applications connecting to sensor networks.

The rest of the paper is organized as follows. In Section 2, we present the related work and in Section 3, we discuss about the system model. We describe a number of scheduling algorithms and propose our algorithm in Section 4. In Section 5, we discuss the simulation setup, the experiments and present the results of the performance evaluation. Finally, in Section 6, we summarize our research findings and discuss some possible future work.

2 Related Work

In recent years, many efficient data dissemination methods have proposed particularly for read-only transactions [2, 5, 6, 13, 14, 15, 17]. Many of them are based on data broadcast or on-demand transmission. However, most of the previous studies are not for time-constrained mobile computing systems and they aim to improve the response time instead of meeting the deadline requirements.

In [4], a system called PSoup is developed for applications with streaming data in environments such as sensor networks. The system supports queries that require access to both data that arrive prior to an dafter the query by continuously materializing and maintaining the results of the query in a Results Structure. A user interacts with PSoup by initially registering a query specification with the system such that the user may repeatedly invoke the results of the query at later times. In this current implementation, PSoup

[*] The work is supported in part by the RGC Earmarked Research Grant from the HKSAR Government under the grant number: RGC HKBU 2174/03E

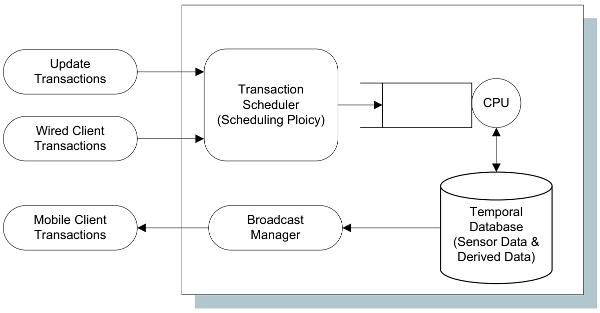

Update Transactions
Wired Client Transactions
Mobile Client Transactions
Transaction Scheduler (Scheduling Ploicy)
CPU
Broadcast Manager
Temporal Database (Sensor Data & Derived Data)

Figure 1. Overall System Architecture

allows the user only to retrieve data corresponding to the current window and there is no transaction semantics.

A broadcast scheduling facility, namely broadcast disk [1], for non-uniform access to data was proposed. Data are interleaved into a stream for dissemination on the broadcast channel based on their data access frequencies. Various ways of combining broadcast and on-demand channels for data dissemination were proposed [3, 8]. Air indexes for improving the tune-in time for mobile users have been studied in [9]. Lee et al. addressed general queries over wireless channels with a semantic-based broadcast approach [10]. The described here, however, focuses on data access read-only requests form mobile clients.

3 System Model

Figure 1 shows the overall architecture for a typical broadcast system for real-time data dissemination. As indicated in the figure, we have identified three types of transactions operation on the system, which by itself consists of three major components, namely the Transaction Scheduler, the Broadcast Manager and the Temporal Database. While the Transaction Scheduler will enforce our proposed scheduling policy for the update transactions. The Temporal Database is partitioned in tow two sections to hold sensor data and derived data. The communication between the mobile client and the system is by means of a relatively slow communication link through the Broadcast Manager. Since we we dealing with Read-only transactions for the mobile client, there is no upload link from the mobile client to the system.

3.1 Update Transactions

By Update Transactions, we meant continuous streams of data arriving at the system from the sensor network. These are write-only transactions that keep on updating the sensor data in the temporal database. A sensor data item composes of a value, d_{value}, to reflect its current state, the data item is also associated with a timestamp, $d_{timestamp}$, recording its time of creation, and a timing interval, d_{avi}, for its data validity. The system has to guarantee the freshness of the data when being read and the timeliness of the data so that the temporal database within the system is reflecting the situation of the real world. Since update transactions can be a periodic task or an aperiodic task updating a set of sensor data items.

3.2 Wired Client Transactions

Wired Transactions are actually background transactions with mixed real-time tasks. These transactions can be a real-time tasks, can either be periodic or a periodic, reading sensor data and producing derived data in the temporal databases, or it can be non-real-time tasks operating on both sets of data in the system and provide a background workload for the system.

3.3 Mobile Client Transactions

For the Mobile Client Transactions, these are Read-Only transactions that read either one or both sets of sensor data and derived data from the temporal database.

3.4 Transaction Scheduler

With three kind of transactions in our system, the transaction scheduler is used to pick the right task to execute. Update transactions are assigned with higher priority level than that of wired client transaction. Among update transactions, they are scheduled in First-Come-First-Server (FCFS) manner. Among wired client transaction, they are scheduled in Earliest-Deadline-First (EDF) manner.

3.5 Temporal Database

The data objects in the Temporal Database are divided into non-temporal data objects and temporal data objects. Each of them is associated with a write timestamp (WTS), which is the last update time of the data object. Temporal data objects are only updated by periodic update transactions. That means that other wired client transactions will not update the values of these temporal data objects. All temporal data objects are updated periodically at the same rate asynchronously. A version number or an expiration time is associated with each temporal data objet for the versioning and the time validity interval protocols respectively. Version number denotes the ith version of the temporal data object.

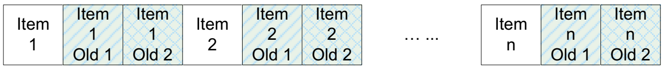


Figure 2. One By One

3.6 Broadcast Manager

Because of the relatively slow link between the mobile client and the system, mobile client has to rely on the broadcast manager to complete the transaction in time. In each broadcast cycle, the broadcast manager broadcasts the values of all data objects and the control information in the form of a single flat broadcast disk structure [1, 7].

4 Scheduling Algorithms

In the following sections, we will present each of the algorithms we have explored.

4.1 Pure BCC-TI

No old version will be broadcasted by Pure BCC-TI [11]. In each broadcast cycle only the current values of each items will be broadcasted.

4.2 Clustering

In the clustering approach [16], all current values of each items are broadcast first and older versions of each items are broadcast at the end of each broadcast cycle. Consequently, long-running read-only transactions that read old versions are penalized since they have to wait for the end of the broadcast to read such versions.

4.3 Old Versions on New Disk

With the new disk approach [16], a new disk is created to hold any old versions. The relative frequency of the disks with the current versions is maintained by simply multiplying their frequency by a positive number m so that the slow disk that carries the old versions in m times slower than the disks with the current versions. The new disk approach is easily adaptive. Old versions can be placed on faster disks when there are many long-running transactions and on slower disks when most transactions need current values.

4.4 One By One

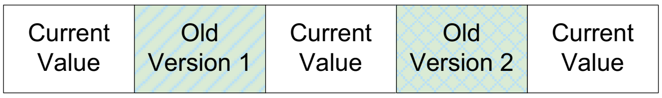


Figure 3. Candyfloss I

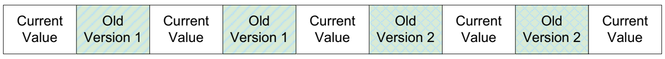


Figure 4. Candyfloss II

Following the one by one approach, one way to structure the broadcast is to broadcast all versions of each item successively. older versions of an item are placed right after with the current values of that item. One By One works well when each transaction may access any version of an item with equal probability.

Candyfloss

With this approach, old versions of each item are broadcasted separately interleave with all current values of each items. For instance, say in Figure 3 we have a broadcast organization for keeping 2 old version. The first chunk to be broadcast is the all current values of each item then followed by the first old versions of each item. After that is all current values of each item again. Then the second old versions of each item. At the end will be the current values of each item. Old versions of each item and all current values of each items are interleave with each other. This approach is adaptive. The broadcast frequency of old versions can be various. Old versions can be broadcast more frequency when there are many long-running transactions and less when transactions need current values. Figure 4 shows yet another example. Old versions of each item are broadcast 2 times more than that in Figure 3.

5 Performance Evaluation

We have constructed and conducted a series of simulation experiments to look into the performance of each scheduling algorithm on our broadcasting system. In the following sections, we will discuss about the simulation setup, the workload being used, performance metrics, and finally present our research findings in these experiments.

5.1 Simulation Experiments

5.1.1 Setup

We have written a simulator using CSIM to simulate the mobile transactions and the data dissemination and all the simulation experiments were executed on the Windows XP platform with an Intel 2.4GHz CPU. As indicated by Figure 1, the simulation model consists of a server, a number of mobile clients and a broadcast manager. In the whole system, there are three types of transactions. They are mobile transactions submitted by the mobile clients, wired client transactions processed at the server, and update transactions for installing the latest values of the temporal objects in the database at the server. Mobile transactions are read only and wired client transactions consist of both read and write operations. For data objects accessed by mobile and server transactions, 40% are temporal data objects and 60% are non-temporal data objects. Each data object in a class has an equal chance of being accessed by an operation. Each temporal data object is updated asynchronously at a fixed interval by an update transaction.

Transactions are generated at the transaction generator and are lined up in the CPU queue according to the scheduling discipline. When the CPU is available, the transaction at the front of the CPU queue will be submitted to the CPU for processing. To read a data object, transactions need to line up in the disk queue for data access. The transactions will repeat these steps until all operations are processed. During each operation, transactions will check if they have missed the transaction deadlines. For versioning and time validity interval, temporal data consistency will be checked as well. Of a transaction can commit, it will line up in the disk queue again for installing the pre-written values into the database. All active transactions will be checked to see if they have accessed the data objects written by the validating transaction. Those active transactions that have read the data objects need to be restarted.

The ROTs generated by mobile clients read data objects from the broadcast disks. In each operation, they will listen and wait for the requested data objects to be broadcasted. When the data object are read, they will adjust their lower bound of timestamp intervals to reflect their position in the serialization order. Version numbers and the end of validity intervals will be recorded in the case of versioning and time validity interval respectively. Of control information is listened, they will adjust their upper bound of timestamp intervals as well. Ages of temporal data objects read by ROTs will be checked in the case of versioning.In each operation, transaction deadline is checked. Data deadline is checked too in the case of time validity interval. If the ROTs can complete the last operation without missing the transaction deadline, they can commit autonomously.

Transactions are processed until either they are committed or the transaction deadline is missed. The deadline of a transaction $d(T)$ arrived at $a(T)$ with predicted execution time $p(T)$ is assigned by following formula.

$$\mathrm{d(T)} = a(T) + slackfactorxp(T)$$

where $p(T)$ for mobile transactions = broadcast cycle time x transaction length + client inter-operation delay x (transaction length -1), and $p(T)$ for wired client transactions = (disk service time + CPU service time) x transaction length. The transaction length is the number of operations in a transaction.

Items	Values
Mobile clients	
Transaction length (number of read operations)	8
Mean inter-operation delay	65,536 bit-times (exponentially distributed)
Mean inter-operation delay	131,072 bit-times (exponentially distributed)
Slack factor	2.0-6.0 (uniformly distributed)
Concurrency control protocol	BCC-TI
Probability of accessing a temporal object	0.4
Number of Clients	10
Server	
Transaction length (number of operations)	8
Transaction arrival rate	1 per 100K bit-times
CPU service time	0.983 K bit-times (15 ms)
Disk service tiem	1.638 K bit-times (25ms)
Probability of accessing temporal data objects	0.4
Probability of writing non-temporal data objects	0.8
Total number of data objects in database	300
Number of temporal objects in database	30
Size of data object (including object ID)	8 K bits
Timestamp size	8 bits
Concurrency control protocol	OCC-FV
Priority scheduling	Higher Priority First
Period of temporal objects	5,000 K bit-times

Table 1. A Summary on the Workload used in our study

There are two priority levels for transaction scheduling at the server. Update transaction are assigned with a higher priority level than that of wired client transactions. Among update transactions, they are scheduled in First-Come-First-Serve (FCFS) manner. Among wired client transactions, they are scheduled in Earliest-Deadline-First (EDF) manner.

The time unit is in K bit-times, the time to transmit 1 K bits in broadcast environments. For a broadcast bandwidth of 64Kbps. 1 M bit-times is equivalent to approximately 15s and the mean inter-operation delay and the mean inter-transaction delay are 1s and 2 s respectively.

At the stat of each broadcast cycle, the server fills the broadcast disk with the data. Each broadcast disk contains the control information followed by the broadcast of all the data objects in the database. The control information consists of the timestamps and write sets of the committed wired client transactions and update transactions during the last broadcast cycle. Version numbers of temporal data objects are contained as well. The broadcast of all the data objects contains the values and the write timestamps of the data objects. The current version numbers and the ends of time validity interval are broadcasted along with the temporal data objects.

5.1.2 Workload being used

In order to test our algorithm rigorously, Table 1 shows a summary on the data we used to test our algorithms throughout the simulation experiments. Except explicit statement, we examined the execution of the various scheduling algorithms using the default parameters.

5.2 Performance Metrics

In this study, we concern about the real-time performance on broadcast strategies. Since each mobile read only transaction has its own deadline to meet, the number of mobile read only transaction that can finish on time naturally becomes one of the performance indicator. Hence, in our simulation experiments, performance are measured by the followings:

Commit Rate Commit Rate indicates how many transaction can be committed before their deadlines expired. Hence,

$$\text{Commit Rate} = \frac{c}{N}$$

$$\text{where} \begin{cases} \text{c = Number of Transaction Finished in Time} \\ \text{N = Total Number of Transaction in the System} \end{cases}$$

Restart Rate Restart Rate indicates the average number of restarts before a transaction leaves the system. Hence,

$$\text{Restart Rate} = \frac{r}{N}$$

$$\text{where} \begin{cases} \text{r = Number of Transaction Restarted} \\ \text{N = Total Number of Transaction in the System} \end{cases}$$

5.3 Simulation Results

5.3.1 Effect of Different Transaction Arrival Mean

We will first look at the effect of different transaction arrival mean. In Figure 5, we varied the transaction arrival mean form 30 K bit-times to 250 K bit-times, the commit rate of for different algorithms is shown. We fixed the transaction length for each read-only transaction at 8. All algorithms show the trend that when transaction arrival mean is low, the commit rate is low, and when we increase the transaction arrival mean, all algorithms show an increase in commit rate.

Our proposed Candyfloss perform the best in terms of commit rate. The Clustering algorithm does not perform as well especially when the transaction arrival mean is really low (i.e., at 30 K bit-times).

Commit rate is one of the performance measures of our concern. When we look at the restart rate, a clear picture picture can be drawn. Figure 6 shows the restart rate at

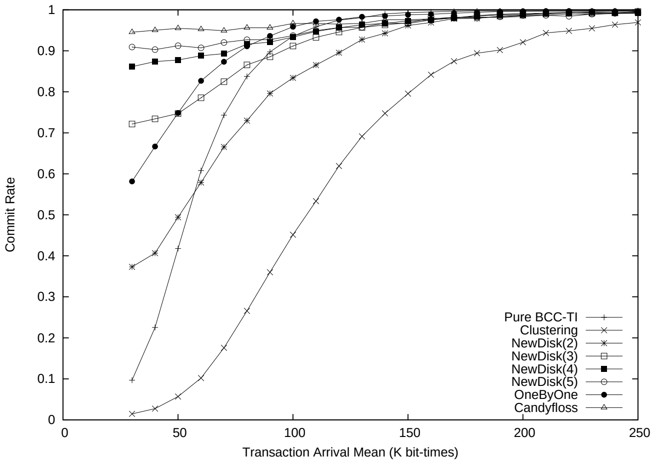


Figure 5. Transaction arrival mean vs. Commit Rate

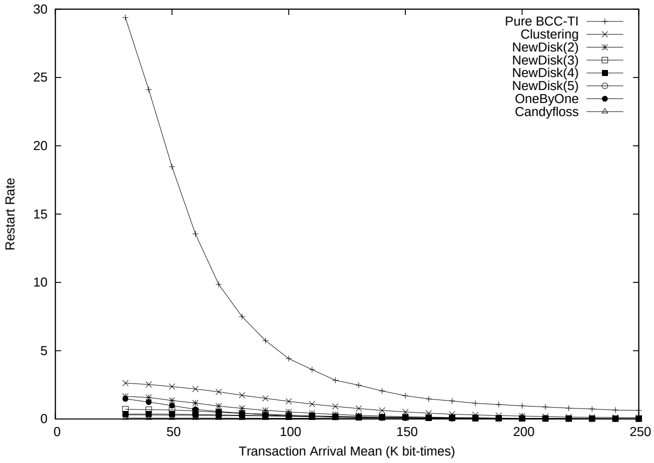


Figure 6. Transaction arrival mean vs. Restart Rate

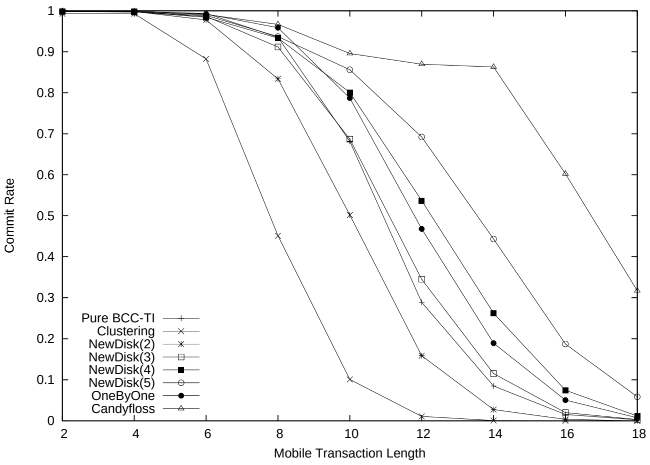


Figure 7. Transaction Length vs. Commit Rate

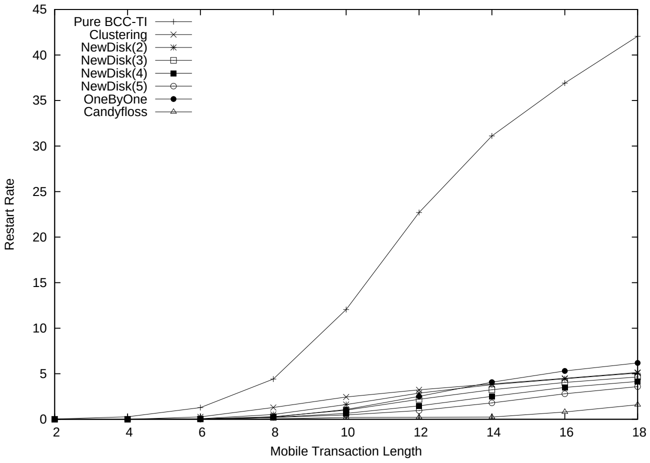


Figure 8. Transaction Length vs. Restart Rate

various transaction arrival mean with different algorithms. From Figure 6, we can observe that as we increase the transaction arrival mean from 30 K bit-times to 250 K bit-times, the restart rate decrease accordingly.

Our proposed Candyfloss performs the best in terms of restart rate and at the same time allow most of mobile read-only transaction commit before their transaction deadline.

5.3.2 Effect of Different Transaction Length

In Figure 7, we varied the transaction length from 2 operations to 18 operations . The commit rate of all algorithm decrease when the length of the mobile transactions increases. The length of the mobile transactions is defined as the number of read operations in a mobile transaction. To read more number of data objects, a mobile transaction may span a larger number of broadcast cycles. As a result, it is more likely for it to miss the deadline and more difficult for it to meet the deadline after restart. In addition, the prolonged execution time also increases the chances of having data conflicts with server transactions because more server transactions will be executed concurrently with the mobile transaction. The performance of Clustering is the worst because mobile read-only transaction have to wait for the end of the broadcast to locate old versions, thus their span increases as does their probability of missing transaction deadline. Our proposed Candyfloss out performs all other algorithms. In particular, with transaction length of 18.

Figure 8 shows the restart rate of all algorithm. It show that our proposed algorithm - Candyfloss perform the best. Pure BCC-TI is still performing badly.

6 Summary & Future Work

In this paper, we presented a performance study on various broadcast algorithms to support Temporal Consistency of Sensor Data for Transaction Processing Applications in Broadcast Environments. The objective of the study is to develop a transaction algorithm for providing fresh data in timely manner for applications connecting to sensor networks. We have performed a series of simulation experiments. Simulation results show that our proposed broadcast algorithm not only succeeds in providing on-time commit of mobile read-only transaction but also reduce the restart rate of mobile read-only transaction.

As for future work, we will take a closer look on data freshness, transaction deadline and postponement of getting item from broadcast cycle. All these could be affected by different broadcast algorithms.

References

[1] S. Acharya, R. Alonso, M. Franklin and S. Zdonik. Broadcast disks: data management for asymmetric communication environments. In *Proceedings of the 1995 ACM SIGMOD international conference on management of data*, pages 199–210, San Jose, California, USA, 1995.

[2] S. Acharya, M. Franklin, S. Zdonik. Balancing Push and Pull for Data Broadcast. In *Proceedings of the ACM SIGMOD*, Tucson, Arizona, May, 1995.

[3] D. Aksoy and M. Franklin. R x W: A scheduling approach for large-scale on-demand data broadcast. *IEEE/ACM Transactions on Networking*, 7(6):846–860, December 1999.

[4] S. Chandrasekaran and M.J. Franklin. Streaming Queries over Streaming Data. In *Proceedings of the 28th VLDB Conference*, pages 203–214, Hong Kong, Auguest, 2002.

[5] A. Datta, A. Celik, J. Kim and D.E. VanderMeer. Adaptive Broadcast Protocol to Support Power Conservant Retrieval by Mobile Users. In *Proceedings of Internatinoal Conference on Data Engineering*, 1997.

[6] J. Fernandez, K. Ramamritham. Adaptive Dissemination of Data in Real-Time Asymmetric Communication Environments. In *Proceedings of Euromicro Conference on Real-Time Systems*, June, 1999.

[7] Q. Hu, W.-C. Lee and D.L. Lee. Indexing Techiques for Wireless Data Broadcast under Data Clustering and Scheduling. In *Proceedings of The Eighth international Conference on Information and Knowledge Management (CIKM '99)*, Kansas City, Missouri, November 2-6, 1995.

[8] T. Imielinski and S. Viswanathan. Adaptive wireless information systems. In *Proceedings of SIGDBS Conference*, pages 19–41, Oct, 1994.

[9] T. Imielinski, S. Viswanathan and B.R. Badrinath. Energy efficiency indexing on air. In *Proceedings of ACM SIGMOD'94*, pages 25–36, May, 1994.

[10] K.C.K. Lee, H.V. Leong and A. Si. A semantic broadcast scheme for a mobile environment based on dynamic chunking. In *Proceedings of IEEE ICDCS'2000*, pages 522–529, April, 2000.

[11] V.C.S. Lee, J.K. Ng, J.Y.P. Chong and K. Lam. Maintaining Temporal Consistency in Broadcast Environments. In *Proceedings of MDM 2004*, pages 284–292, 2004.

[12] C.S. Peng and K.J. Lin. A Semantic-Based Concurrency Control Protocol for Scheduling Mixed Read-time Tasks. In *Proceeding of 1996 IDDD Real-Time Technology an dApplications Symposium*, papes 59–67, Brookline, MA, USA, June, 1999.

[13] E. Pitoura and P.K. Chrysanthis. Scalable Processing of Read-Only Transactions in Broadcast Push. In *Proceedings of International Conference on Distributed Systems*, May, 1999.

[14] E. Pitoura and P.K. Chrysanthis. Exploiting Versions for Handling Updates in Broadcast Disks. In *Proceedings of Very Large Data Base Conference*, Sept, 1999.

[15] E. Pitoura. Supporting Read-Only Transactions in Wireless Broadcasting. In *Proceedings of the DEXA'98 Workshop on Mobility in Databases and Distributed Systems*, August, 1998.

[16] E. Pitoura and P.K. Chrysanthis. Multiversion Data Broadcast. In *IEEE Transactions on Computers*, 51(10):1224-1230, October, 2002.

[17] J. Shanmugasundaram, A. Nithrakashyap, R. Sivasankaran and K. Ramamritham. Efficient Concurrency Control for Broadcast Environments. In *Proceedings of ACM Intrnational Conference on Management of Data*, Philadelphia, 1999.

Regularized Selective Ensembles of Base Learners

Zhili Wu, Chunhung Li, Jiming Liu
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong, Hong Kong
Email:{vincent,chli,jiming}@comp.hkbu.edu.hk

Abstract

The ensemble of base learners such as classifiers or regressors has shown performance benefits in many reported work. However, methods for selecting and combining multiple base learners are often heuristic in nature and do not have a well-defined objective function. We propose a set of linear prediction approaches and optimize this combination task with regularized objective functions. Specifically, for regression, we suggest a novel regularized form considering both margin maximization and square loss minimization, with a close-form solution obtainable; for classification, we borrow the setup in typical one-norm and two-norm SVMs, trying to regulate both the hinge loss of outputting labels and the classifying margin. These regularized setups for ensemble learning can further guide the selection of base learners. Improved performances over bagging and other selective ensemble methods can be achieved.

1 Introduction

Ensemble is a useful technique where the outputs of a set of base learners are combined to form a unified prediction [26], e.g. neural network ensemble [13], and the random forest of decision trees [34]. Taken classification tasks as examples, a typical ensemble learning is to construct a collection of individual classifiers, then obtain the class prediction by voting the outputs of the individual classifiers in the ensemble. Many researchers have demonstrated that ensembles generally outperform the best single classifier in the ensemble [25]. Typical applications of ensembles includes face recognition [11], hand written word recognition [10], medical diagnosis [36],etc.

Bagging (bootstrap aggregation) [2] and boosting (e.g. Adaboost [7, 8]) are two major techniques for constructing ensembles [24, 5]. Both techniques are thought to generate different base learners by training on different subsets of training samples [5]. In bagging, each training set is constructed by drawing a certain number of examples uniformly (with replacement) from the original training set. In boosting, the typical Adaboost algorithm starts from a set of weights over the original training set, and constructs new training sets through *boosting by sampling* or *boosting by weighting*. In *boosting by sampling*, it constructs a new training set by drawing examples (with replacement) from the original training set with probability proportional to their weights. In *boosting by weighting*, the entire original training set and the weights are input into the base algorithms which can accept a weighted training set directly. The weights in Adaboost are then adjusted after the training of a base learner is completed, in the manner of increasing the weights of misclassified examples, and decreasing those of the correctly classified examples.

To construct a good ensemble, much effort has been focused on the diversity and accuracy of the underlying base learners [19]. For bagging, it is revealed that unstable base learners sensitive to the sampled training sets is usually needed to achieve diversity. For boosting, it doesn't require the unstable condition for base classifiers because it can realize diversified base learners through re-weighting training sets. Some other approaches many aiming at further diversifying base learners are available, by inducing randomness into the base learning algorithm [5], or manipulating the input attributes [16], and the model outputs [4].

Not like the diversity and accuracy issues, the issue of how to combine base learners in an ensemble has not aroused enough attention. Ensemble learning typically adopts weighted or unweighted voting for prediction combination [1, 28]. In the unweighted majority (plurality) voting for classification, the class with the most votes from all base classifiers is regarded as the prediction by the ensemble. In weighted voting which is more typical in regressor ensemble, base learners have different weights associated with, their outputs are then weighted and linearly combined. In this fashion, Adaboost comes up with a set of weights for combination at the end of training all base learners, but these weights have been shown to overfit the final model [14].

For the combination of base learners in an ensemble, the stacked generalization [32] is a very general framework to follow. It is to learn an upper-level meta-classifier based on the predictions of all base classifiers. A recent study [25] shows stacking enabled by multi-response model trees performs better than selecting the best single classifier in the ensemble. The stacked generalization is even drawn a relationship with meta-learning, which is about ”learning to learn” [30]. It is an ensemble schema allowing easy combination of heterogenous base learners, which enables base learners to be trained by several different learning algorithms, rather than a single learning algorithm typically used in bagging and boosting.

Other than the issue of learning how to combine base learners, choosing which set of base learners for the combination is also important. Each base learner in an ensemble usually is not tuned to be optimal. On the other hand, some base learners in the ensemble might be redundant, due to the large overlapping of training sets used by base learners, or the difficulty in achieving diversity of some stable base learning algorithms. It is thus helpful to consider the selective combination of base learners. Along this direction, some approaches can be found, which deal with adding base learners dynamically [18], learning the optimal combination of neural networks [29] or doing selective ensemble [35]. The selective ensemble [35] has some common points with stacked generalization, by learning a set of weights through genetic algorithms upon the outputs of base learners. It has been reported with better performance than typical bagging and boosting approaches by properly selecting the controlling parameters.

This paper focuses on the issue of how to select and combine base learners, through learning regularized optimal combination of base learners and conducting selection based on the learned weights. In section two, the regularized objective formulation for base learner combination of regression and classification will be presented, followed by the the discussion of selecting base learners and the implementation. In section three, comparisons are provided among our approaches and the genetic algorithm-based ensemble (GASEN) [35] and bagging . At the end, there will be the discussion of limitation and open problems.

2 Formulation and Solution

2.1 Base Learner Combination

Given a set of n examples $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ to be classified into the positive/negative class, or regressed into real values, that is, in both cases, into a target output vector $\mathbf{Y} = \{y_i\}_{i=1}^n$, an ensemble of N base learners can give N predictions to each example. This actually results in a prediction matrix $\mathbf{G}$ with the size of $n \times N$, while $\mathbf{G}_{ij} = g_j(\mathbf{x}_i)$, the output for i-th example by the j-th base learner.

From the perspective of voting for or linearly combining the outputs of base classifiers, liking bagging or boosting, the following form is implied or approximated,

$$f : \mathbf{G}\boldsymbol{\beta} \rightarrow \mathbf{Y}, \tag{1}$$

where $\boldsymbol{\beta} \in \mathbf{R}^N$ is a weighting vector. In bagging, majority vote in binary classification or averaging in regression are to let all $\beta_i = \frac{1}{N}$ (the *sign()* function is further used for classification). Note the bagging approach does not have a clear objective function to quantify the goodness of the $\boldsymbol{\beta}$ but a set of homogenous constants. In Adaboost the weights $\boldsymbol{\beta}$ to base learners are obtained through the updating of the weights to training examples, that is, the weights $\boldsymbol{\beta}$ are indirectly learned from the process of updating the weights to training examples. Adaboost has been understood to greedily minimize the exponential loss in terms of classification or regression error, which however may result in over-fitting problems [14]. Hereby we think a remedial way for both is to formulate the learning of the $\boldsymbol{\beta}$ into optimizing an objective function with a good control mechanism on over-fitting, and solve the objective function in a systematic manner.

2.2 The Regularized Combination of Base Regressors

Following the form in Eq. 1, as well as those in GASEN, Adaboost and Least Square SVM [27], we adopt the linear combination of base regressors, with the weight sum normalized to one, that is,

$$y(\mathbf{x}_i) = \sum_{j=1}^{N} \beta_j g_j(\mathbf{x}_i)$$

with respect to

$$\sum_{j=1}^{N} \beta_j = 1.$$

We further adopt the following square loss same as the one in GASEN. As a remark, other loss functions, like the epsilon-insensitive loss in typical support vector regression, are also feasible and under our current investigation.

$$\xi_i = (y(\mathbf{x}_i) - \sum_{j=1}^{N} \beta_j g_j(\mathbf{x}_i))^2$$

Instead of solely minimizing the total loss as in GASEN, or the exponential transformation as in Adaboost, we suggest an additional measure of margin to maximize, which is proportional to the reciprocal of $\frac{1}{2}||\boldsymbol{\beta}||^2$. Balanced by a user-searched constant C, these two terms can be written into an integrated form,

$$\min\{\frac{C}{2}\sum_{i=1}^{n}\xi_i + \frac{1}{2}||\boldsymbol{\beta}||^2\}. \tag{2}$$

Since the total loss of ξ over all n examples is

$$\sum_{i=1}^{n}\xi_i = \sum_{j,k=1}^{N}\beta_j K_{jk}\beta_k,$$

where K_{ik} is defined to be the overall correlation between the errors of the $j-$th and $k-$th base regressors [35] for predicting all examples, which is described by the following formula,

$$K_{jk} = \sum_{i=1}^{n}(g_j(\mathbf{x}_i) - y(\mathbf{x}_i))(g_k(\mathbf{x}_i) - y(\mathbf{x}_i)).$$

A close form solution to Eq. 2 can be obtained, as shown in the appendix A.

2.3 The Regularized Combination of Base Classifiers

In the same way of regressor combination, for classification it is not a good choice to solely minimize the empirical prediction error when combining base classifiers. If adopting the linear combination of base classifiers, the criteria of maximizing the margin associated with the linear classifying hyperplane can be steadily used. Among many variants of regularized forms, the l_2 SVM-alike formulation, as presented in the following, is one of the most familiar to the machine learning community.

Assuming $f_i = \sum_{j=1}^{N}\beta_j g_j(\mathbf{x}_i) + \beta_0$ is the class output to the $i-$th example by the $j-$th base classifier (before taken the *sign*). The β_0 is an offset constant, which can also be forced to zero in some SVM formulations [17]. The objective function for base classifier combination is then

$$\min C\sum_{i=1}^{n}\max\{1 - y_i f_i, 0\} + \frac{1}{2}||\boldsymbol{\beta}||^2, \tag{3}$$

where the term $\max\{1 - y_i f_i, 0\} = (1 - y_i f_i)_+$ is also called hinge loss, which has shown to be useful for measuring the loss of binary class output. After transformed into the dual form, this objective form can be solved by typical quadratic programming routines, or simply by calling more efficient SVM packages.

Under the general framework of regularization [6], many variants of the above objective function are possible for regularized base classifier combination.

To change the penalty term $||\boldsymbol{\beta}||^2$ to $||\boldsymbol{\beta}||^0$ or $||\boldsymbol{\beta}||^1$, zero or one norm SVMs [31, 37] are formulated, respectively. They have the property of forcing some entries of $\boldsymbol{\beta}$ to be zero, which implies that by solving them, the way of combining learners and the selection of learners can be simultaneously fulfilled with. In our experiment, we also tested the 1-norm SVM formulation, which can be solved through linear programming, as remarked in Appendix B.

In data analysis, the multiclass tasks by SVM-alike formulations haven't fully solved. The use of such formulations for the combination and selection of base learners in ensemble learning thus also leaves the multiclass issues open. However, the approach of decomposing a mutliclass task into multiple binary tasks can always be used. In our experiment, the one-vs-one method is used. In appendix B, some directions of integrated formulations for multiclass ensemble are preliminarily researched, with the exemplification under our current investigation.

2.4 Base Learner Selection Strategies

The selection and deletion of some base learners are to reduce the complexity of the ensemble model, without sacrificing the accuracy too much (it might even lead to performance improvement).

As a simple analysis which is similar to the one in Gasen, if constant weights are used in regression (Eq. 2), and then if the $k-$th base regressor is removed, the change of objective function becomes

$$\triangle L \propto \text{const} - 2(N-1)^2\sum_{j\neq k}\mathbf{K}_{jk} - (N-1)^2\mathbf{K}_{kk}.$$

It is thus intuitive to know the base learner with large training error can be deleted to get large reduction of objective value (Note the $\mathbf{K}_{kk}$ is the overall error of the $k-$th base regressor over all training examples).

For the general cases of various weights of $\boldsymbol{\beta}$, the magnitude of $\boldsymbol{\beta}$ can be used as the selection criterion. The entries of $\boldsymbol{\beta}$ with small magnitudes can be regarded to be less salient for the ensemble, the corresponding base learners can be safely removed without changing the model's optimality and performance too much. Pruning by the magnitude of weights has been used in many feature selection methods for data analysis (e.g. SVM-RFE [12], wrapper). For linear models used here, it can be further shown the selection based on the magnitudes of $\boldsymbol{\beta}$ is also coincident with the strategy of pruning based on objective function change [3, 12].

2.5 The Implementation

Our current implementation of base leaner combination and selection can be summarized as follows:

- Train multiple base learners (e.g. Neural networks, Decision Trees, SVMs, even their mixture) based on different data split, feature subsets and randomness injection.

- For a separate validation set with labels/responses known, take all the predictions from the base learners, and train a linear combination based on Eq. 2 or Eq. 3 or the 1-norm variant (Possibly tuning the model parameters like C to ensure a good generalization ability).
- For small β_j less than a preset threshold, their associated base classifiers are deleted from the ensemble, the remaining weighting set $\boldsymbol{\beta}$ is accordingly normalized or retrained.

3 Experiments

3.1 Datasets

We tested several synthetical regression datasets from [2]. The total dataset size is 5000 for each task. In our testing, noise is further generated to make the tasks more difficult.

Friedman #1 $y = 10\sin(\pi x_1 x_2)+20(x_3-0.5)^2+10x_4+5x_5$ (4)

$$\text{Friedman \#2 } y = \sqrt{x_1^2 + (x_2x_3 - (\frac{1}{x_2x_4}))} \quad (5)$$

$$\text{Friedman \#3 } y = \tan^{-1}\frac{x_2x_3 - \frac{1}{x_2x_4}}{x_1} \quad (6)$$

The first classification dataset is the numerical version of the Credit (German) data from the UCI machine learning repository [23]. It is an unbalanced binary classification task. It has 700 positive data points and 300 negative ones, each has 24 numerical features.

The second dataset is the chess data, also from the UCI machine learning repository. It has 36 nominal features, 1669 positive examples and 1527 negative ones. It is converted into a numerical data matrix with the dimension of 38.

The third dataset is the waveform data. It has 5000 examples and three evenly distributed classes. In our testing, only 2000 examples are sampled as the total set.

3.2 Experimental Setup and Results

We mainly follow the experimental settings for the GASEN in the selective ensemble paper [35], and compare our approaches with bagging and GASEN.

For each classification dataset, we use half of the randomly drawn examples to form the original training data set, on which an ensemble of 20 neural networks are trained by bootstrap sampling. The remaining half of examples are then evenly divided into a separate validation set and a testing set. Taking the Credit (German) data of 1000 points as an example, 500 points are randomly drawn out to form the original training set. For the remaining 500 points, 250 are taken as a validation set and the other 250 as a testing set.

Based on the training set, twenty neural networks are trained upon 20 different bootstrapped samples. After training the base neural networks, their prediction over the validation set are then used to learn the selection and combination strategy. The RESEN-l_1 and RESEN-l_2 denoted our l_1 and l_2-norm regularization based selective ensemble of neural networks. In this setting, learners are selected by the magnitudes of weights and after selection the final outputs to the testing set are the majority vote of the remaining base learners. Correspondingly the genetic algorithm for learning the weighted combination of base neural networks (GASEN) is compared. The whole process is repeated ten times to get an average error reporting.

Several variants of GASEN and RESEN are implemented. They are GASEN-w, GASEN-wa, RESEN-w-l_1, RESEN-wa-l_1, RESEN-w-l_2, and RESEN-wa-l_2. GASEN-w, RESEN-w-l_1 and RESEN-w-l_2 use the evolved weights to select base neural networks, and combines the predictions of the selected neural networks with the normalized/relearned version of their evolved weights. GASEN-wa, RESEN-wa-l_1 and RESEN-wa-l_2 do not select the base neural networks, just do a weighted combination of the outputs of all twenty base networks. All results are divided by the results of bagging, which just combines all base networks through majority vote or averaging.

Table 1. Selecting and Combining Base Classifiers by Regularized Ensemble (RESEN) and Genetic Algorithms (GASEN), Error Relative to Bagging

	Credit(German)	Chess	Waveform
GASEN	1.036	1.022	0.986
RESEN-l_2	0.994	0.911	0.961
RESEN-l_1	0.996	0.933	0.992
GASEN-w	1.107	1.011	0.975
RESEN-w-l_2	1.000	0.811	0.941
RESEN-w-l_1	1.039	0.688	0.970
GASEN-wa	1.068	1.022	0.994
RESEN-wa-l_2	0.997	0.833	0.981
RESEN-wa-l_1	1.029	0.711	0.970
# of selected	7	8	10

From Table 1, it can be noted RESEN based approaches outperform GASEN and bagging in general. The GASEN series, though verified to outperform bagging in many other datasets and settings, are found possible to perform worse than bagging, as for the German and Chess data. It might be due to the unbalanced property of the German data, and the easy separability of the Chess task. It can be noted GASEN-w and GASEN-wa, which utilize weighted combination of base learners, do not bring performance improvement. This is because the weights obtained from GASEN do not ensure good generalization, thus may cause overfitting issues. To the contrary, the weighted combination based on RESEN usually has some improvements. Howver, the reduced combinations by RESEN, like the RESEN-w-l_2 and RESEN-w-l_1, have some advantages over the combination of all base learners, as denoted by RESEN-wa-l_2 and RESEN-wa-l_1. The typical number of selected base learners is 8, the final model is thus much less complicated than the one based on all twenty base learners. The 1-norm based RESEN approaches perform best in the chess task, and comparable to the 2-norm based RESEN in other tasks. An interpretation to its performance is its sparse solution, which makes the selection of top base learners more accurate for the less nonlinear Chess task.

The experiment of regressor ensemble follows a simpler setup. The averaged combination of the selected base regressors by GASEN and RESEN is tested and compared to bagging. The result is shown in Table 2. Both GASEN and RESEN clearly outperform the bagging approach.

Table 2. Selecting and Combining Base Classifiers by Regularized Ensemble (RESEN) and Genetic Algorithms (GASEN). Error Relative ($4/20$ base learners) to the Bagging of all 20 Base Learners

	Friedman #1	Friedman #2	Friedman #3
GASEN	0.610	0.574	0.637
RESEN	0.521	0.477	0.635

4. Discussion and Conclusion

This paper presents a regularization framework for base learner selection and combination. This formulation can result in better weights for selecting and combining base learners. Experimental results show the approach performs better than the GASEN and bagging approach.

For base regressors in an ensemble, we adapt an regularized form with the objective of margin maximization added. For classification, the standard l_1 and l_2 SVMs are used for selection and combination. Other variants of regression ensemble, and the multiclass ensemble are under further investigation.

The current setup of ensemble learning deals with inductive learning only. However, it is interesting to study the possibility of semisupervise ensemble. Moreover, the strong correlation among base learners also makes the manifold learning possible. Along this direction, we will further the study of semisupervised regularized manifold learning [33] to the ensemble scenarios.

Under an even broader background, our regularized selective ensemble can be regarded as a very limited case of agent learning. Here the base learners, in a very basic form of agents, are required to target at an objective function, with the competition and cooperation underlying implemented. As a possible future aim, we would like to see how the autonomous computing which defines richer agent action can be applied to the analysis and improvement of selective ensemble learning.

Acknowlegement

I would like to express great thanks to Dr. Ji Zhu for his assistance to me in the study of multiclass and 1-norm SVMs.

Appendix A. The close form solution of the regressor ensemble

By First Eq. 2 can be written into a vectorial form,

$$\min\{\frac{1}{2}\boldsymbol{\beta}'(\mathbf{K}+C\mathbf{I})\boldsymbol{\beta}\}$$
$$\boldsymbol{\beta}'\mathbf{e}=1,$$

where $\mathbf{e}$ is a vector of all ones. And then an unrestricted lagrange multiplier η can be introduced,

$$\min L=\frac{1}{2}\boldsymbol{\beta}'(\mathbf{K}+C\mathbf{I})\boldsymbol{\beta}-\eta(\boldsymbol{\beta}'\mathbf{e}-1).$$

Take the derivative over $\boldsymbol{\beta}$ and set all to zero, we have

$$\partial_{\boldsymbol{\beta}}L=(\mathbf{K}+C\mathbf{I})\boldsymbol{\beta}-\eta\mathbf{e}=0.$$

With the proper selection of C, $(\mathbf{K}+C\mathbf{I})$ will be invertible, hereby we have,

$$\boldsymbol{\beta}=\eta(\mathbf{K}+C\mathbf{I})^{-1}\mathbf{e}.$$

Since the sum of all $\boldsymbol{\beta}$ is one, then

$$\eta=1/(\mathbf{e}'(\mathbf{K}+C\mathbf{I})^{-1}\mathbf{e})$$

and finally

$$\boldsymbol{\beta}=(\mathbf{K}+C\mathbf{I})^{-1}\mathbf{e}/(\mathbf{e}'(\mathbf{K}+C\mathbf{I})^{-1}\mathbf{e}).$$

To reduce the complexity, iterative methods are also possible to derive to approximate the solution since the above form is equivalent to solving a linear system.

Appendix B. Multiclass Ensemble

In [21, 15], an integrated multicategory 2-norm svm is proposed by encoding the multiclass labels in a special way. It can be used to model the multiclass ensemble with two-norm margins.

$$\min \frac{C}{n}\sum_{i=1}^{n}\mathbf{L}(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+ + \frac{1}{2}\sum_{l=1}^{k}||\boldsymbol{\beta}_l||^2,$$

$$\text{subject to}: \sum_{l=1}^{k}\mathbf{f}_l(\mathbf{x}_i)=\sum_{l=1}^{k}\sum_{j=1}^{N}(\boldsymbol{\beta}_{lj}g_{lj}(\mathbf{x}_i)+\beta_{l0})=0$$

For a k-class task, the class label of each $\mathbf{x}_i$ is encoded into a column vector $\mathbf{y}_i$ with the length of k, with the l-entry be 1 if the class of $\mathbf{x}_i$ is $l \in (1,\ldots,k)$, while other entries be $-\frac{1}{k-1}$. And $\mathbf{L}(\mathbf{y}_i)$ generates a $0/1$ row vector for $\mathbf{y}_i$, with the corresponding entry to be 0 if $\mathbf{y}_{il}=1$, while the entries corresponding to $\mathbf{y}_{il}=-\frac{1}{k-1}$ to be 1. In terms of ensemble learning, assuming each of the N base learners has been adapted to giving outputs complied with the encoding schema of $\mathbf{y}$, that is, the j-th base learner subsumes a set of k sub-component learners $g_{.j}()$. The j-th base learner is thus associated with a set of k weights $\boldsymbol{\beta}_{.j}$, and will give k outputs to each $\mathbf{x}_i$, with only one of the outputs approaching 1 while the other outputs approaching $-\frac{1}{k-1}$, which in fact has been shown to implement the Bayes decision rule. And this objective function can be solved through quadratic programming.

Based on the above formulation for 2-norm multicategory regularized ensemble, a formation can be proposed for multi-category 1-norm regularized ensemble.

$$\min \frac{C}{n}\sum_{i=1}^{n}\mathbf{L}(\mathbf{y}_i)\cdot(\mathbf{f}(\mathbf{x}_i)-\mathbf{y}_i)_+ + \frac{1}{2}\sum_{l=1}^{k}||\boldsymbol{\beta}_l||^0,$$

$$\text{subject to}: \sum_{l=1}^{k}\mathbf{f}_l(\mathbf{x}_i)=\sum_{l=1}^{k}\sum_{j=1}^{N}(\boldsymbol{\beta}_{lj}g_{lj}(\mathbf{x}_i)+\beta_{l0})=0,$$

where $|\mathbf{w}_j|$ is the 1-norm of the feature weight vector $\mathbf{w}_j$.

This new formulation for 1-norm svm can be solved by linear programming (by introducing $\boldsymbol{\beta}_l=\mathbf{p}_l-\mathbf{q}_l$ or by other approximation methods [9, 22]). It might be possible to derive the whole solution path of this multicategory 1-norm ensemble [38] and [20], which might help parameter adjustment.

References

[1] E. Bauer and R. Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine Learning*, 36(1-2):105–139, 1999.

[2] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[3] Y. L. Cun, J. S. Denker, and S. A. Solla. Optimum brain damage. In *Advances in Neural Information Processing Systems 2*, pages 598–605, 1990.

[4] T. G. Dietterich. Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 1857:1–15, 2000.

[5] T. G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–157, 2000.

[6] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13(1):1–50, 2000.

[7] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995.

[8] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156, 1996.

[9] G. Fung and O. L. Mangasarian. A feature selection newton method for support vector machine classification. *Computational Optimization and Applications*, 28(2):185–202, 2004.

[10] S. Gunter and H. Bunke. An evaluation of ensemble methods in handwritten word recognition based on feature selection. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on ICPR'04*, pages 388–392, Washington, DC, USA, 2004. IEEE Computer Society.

[11] S. Gutta, J. Huang, B. Takacs, and H. Wechsler. Face recognition using ensembles of networks. In *Proceedings of the 13th International Conference on Pattern Recognition*, volume 4, pages 50–54, 1996.

[12] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46(1-3):389–422, 2002.

[13] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(10):993–1001, 1990.

[14] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning, Data Mining, Inference and Prediction*. Springer Verlag, 2000.

[15] S. I. Hill and A. Doucet. Adapting two-class support vector classification methods to many class problems. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 313–320, New York, NY, USA, 2005. ACM Press.

[16] Y. Jiang and Z.-H. Zhou. Editing training data for knn classifiers with neural network ensemble. In *Lecture Notes in Computer Science*, Berlin: Springer, 2004.

[17] T. Joachims. Making large-scale support vector machine learning practical. In A. S. B. Schölkopf, C. Burges, editor, *Advances in Kernel Methods: Support Vector Machines*. MIT Press, Cambridge, MA, 1998.

[18] J. Kolter and M. Maloof. Dynamic weighted majority: A new ensemble method for tracking concept drift. Technical Report CSTR-20030610-3, Department of Computer Science, Georgetown University, Washington, DC, June 2003.

[19] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 231–238. The MIT Press, 1995.

[20] Y. Lee and Z. Cui. Characterizing the solution path of multicategory support vector machines. *To appear, Statistica Sinica*.

[21] Y. Lee, Y. Lin, and G. Wahba. Multicategory support vector machines, theory, and application to the classification of microarray data and satellite radiance data. *Journal of the American Statistical Association, Theory and Methods section*, 99:67–81, 2004.

[22] O. L. Mangasarian. Exact 1-norm support vector machines via unconstrained convex differentiable minimization. *JMLR*, 6, 2006.

[23] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz. UCI repository of machine learning, 1998.

[24] D. Opitz and R. Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, 1999.

[25] B. Z. Saso Dzeroski. Is combining classifiers with stacking better than selecting the best one? *Machine Learning*, 54(3):255–273, 2004.

[26] P. Sollich and A. Krogh. Learning with ensembles: How overfitting can be useful. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 190–196. The MIT Press, 1996.

[27] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[28] G. Tsoumakas, L. Katakis, and I. Vlahavas. Effective Voting of Heterogeneous Classifiers. In *The 15th European Conference on Machine Learning (ECML) and the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)* , Pisa, Italy, September 2004.

[29] N. Ueda. Optimal linear combination of neural networks for improving classification performance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(2):207–215, 2000.

[30] R. Vilalta and Y. Drissi. A perspective view and survey of metalearning. *Artificial Intelligence Review*, 18(2):77–95, 2002.

[31] J. Weston, A. Elisseeff, B. Scholkopf, and M. Tipping. The use of zero-norm with linear models and kernel methods. *JMLR*, 2003.

[32] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[33] Z. L. Wu, C. H. Li, J. Zhu, and J. Huang. Semi-supervised manifold svm. In *Submitted to ICPR2006*.

[34] Z. L. Wu, J. Zhu, and C. H. Li. Hierarchical random forest (in preparation).

[35] Z. H. Zhou and et.al. Selectively ensembling neural classifiers. In *Proceedings of the International Joint Conference on Neural Networks*, volume 2, 2002.

[36] Z.-H. Zhou and Y. Jiang. Medical diagnosis with c4.5 rule preceded by artificial neural network ensemble. *IEEE Transactions on Information Technology in Biomedicine*, 7(1):37–42, 2003.

[37] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-norm support vector machines. In *NIPS*, 2003.

[38] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. L1 norm support vector machines. In *NIPS*, 2003.

Dissimilarity Learning for Nominal Data

Victor Cheng, C.H.Li

Department of Computer Science
Hong Kong Baptist University
Hong Kong

ABSTRACT

Nominal data is often found in data mining and classification problems. While Boolean metric is often used for comparing nominal data, it would be useful for various data mining and classification task if we could actually infer the degree of similarity between different nominal values. We propose constructing minimum error based intra-feature metric matrices to provide distance measure for nominal data so that metric or kernel algorithms can be used. For each nominal attribute, a matrix is first initialized to constant values with zero diagonal and then the off diagonal values are tuned based on minimizing the training error. The optimized matrices give the distance information between nominal values and thus can be referenced in various metric algorithms. Experimental results with classical Boolean nominal metric and C4.5 on various datasets show that the proposed approach gives superior performance.

1. INTRODUCTION

In many machine learning classification problems, we are requested to label a given set of data. Nearest neighbor, neural network and support vector machine [1],[2] are very popular algorithms for these problems. The common ground between them is that they all work on measuring either the similarity or the distance between data points. For instance, nearest neighbor algorithms label data points according to the posterior probability formulated by neighbor training labels which are selected according to their "distance" from the data point. While for neural networks, similar outputs will be obtained if the input vectors are "close". Thus, it is clear that the concept of metric and similarity is very important.

Metric measurement with Euclidean distance is commonly used. However, they require all attributes of patterns to have ordinal structure. Real-valued or ordinal discrete valued feature vectors have ordinal properties and hence their similarity can be measured easily. But metric information for nominal attributes cannot be easily found. For example, we cannot measure the distance between two patterns both containing an attribute "Taste", *T*, which may take one of the following properties:

$T \in$ {salty, sweet, sour, bitter, tasteless}.

For such datasets with nominal attributes, the evaluation or the definition of metric is often difficult. Although rule-based algorithms and decision tree algorithms such as ID3[3], CART[4] and C4.5[5] can be employed, many elegant methods based on metric cannot be applied. Another serious drawback is that we are limited from using many kernel methods.

A simple way of defining a metric for the nominal attributes is the overlap metric or the *Boolean* nominal metric where the distance between two nominal attributes is zero when the attribute value is identical and the distance is one otherwise. However, this Boolean metric does not allow the representation of the similarity between similar nominal features. For example, in cases where 'sour' are more closely associated with 'sweet' than the taste 'bitter', a distance between one and zero should be assigned to describe this relationship between the nominal values of 'sour' and 'sweet'. The value distance metric is designed to measure distance between nominal values using conditional probabilities[6]. An improved distance function is introduced to handle mixed data type[7]. However, both of these approaches do not make use of the classification error feedback in designing the metric.

In follows, we propose using intra-feature metric matrices to provide distance measure for nominal data. In tackling a nominal attribute having *n* possible properties, we do not prefer finding a set of numerical values and assign to each possible property for that attribute. As nominal data may not be ordinal, a nominal attribute may require *n-1* dimension vector to represent all possible properties and thus increase the dimension of feature space significantly. Instead, we propose to construct an intra-feature metric matrix (IFMM) having dimension *nxn* for each nominal attribute. Although it seems *nxn* matrix is more troublesome than *n-1* dimension vector, the matrix actually for metric information storage purpose and does not involved in pattern classification process. Hence, the complexity of classification is not affected. In this IFMM, rows and columns represent all the possible properties of the attribute and the matrix element S_{ij} represent the distance between the property *i* and property *j* of that attribute. For example, the intra-feature metric matrix for the attribute *T* described previously has the following format:

$$\begin{pmatrix} 0 & S_{12} & S_{13} & \cdots\cdots & S_{1d} \\ S_{21} & 0 & S_{23} & \cdots\cdots & S_{2d} \\ S_{31} & \cdots & 0 & \cdots & S_{3d} \\ \vdots & \cdots & \cdots & 0 & \vdots \\ S_{d1} & \cdots & \cdots & S_{d,d-1} & 0 \end{pmatrix}$$

where S_{ij} is the similarity value between the *i*-th and *j*-th properties (say salty vs. sweet) of the attribute *T*. The elements in the matrix are determined by minimizing the output error during the training phase. First, each element of the matrix is initialized to a value, say 1.0 plus a small random value, except the diagonal elements are set to zero because metric of same property should be zero. The IFMM is also a symmetric matrix as Euclidean metric is symmetric. With this initialization, an optimization method such as gradient descent is employed to tune the matrices. Cross-validation are then used to further tune the matrices. After tuning, the values can be referred in Euclidean distance or Gaussian kernel computation. Experimental results show that not only the metric information of nominal data can be obtained, but also the classification accuracy can be improved compared with Boolean matrix method. Nevertheless, it should be noted that the tuned values are problem specific and cannot be transferred to other problems.

In Section 2, we describe the theory and techniques to construct intra-feature metric matrices for nominal attributes. Section 3 describes the techniques to weight the pattern attributes and Section 4 gives our experiment results and Section 6 is the conclusion.

2. INTRA-FEATURE METRIC MATRICES

In this section we first assume that all the attributes of patterns are nominal. Patterns with mixed nominal and numerical attributes will be handled latter. Also, for simplifying the representation, the word "vector" will be still used for cases even though nominal attributes are present. Considering a classification learning, we are given a training set of labeled instances. Each training instance is described by a feature vector ***x*** containing nominal attributes f_i (*i*=1,…,m) and a categorical class label *y*. If f_i having possible properties $\{p_r : r = 0,1,...,d\}$, the corresponding intra-feature metric matrix is denoted by $S_{ij,f}$ where the *f* is an index to the different attributes.

The numerical values of $S_{ij,f}$ are set to (1.0 + a small random value) except diagonal elements are set to zero, then they are tuned by minimizing the training error. In the training phase, training exemplars are first divided into training group and testing group. Exemplars in training group play the role as labeled neighbor and tested by exemplars in testing group. We use a nearest neighbor classifier with Gaussian kernel for training because its performance is comparable to other classifiers and it is relatively simple and intuitive. The classifier has the following structure.

No. of neighbours : 8

Gaussian kernel : $W_{nk} = \exp(-d_{nk}^2 / 2\sigma^2)$

$$d_{nk}^2 = \sum_i \sum_f S_{uv,f}^2$$

$$\sigma = \frac{1}{M} \sum_n d_{nk}$$

where the *u* corresponds to the *i*-th attribute of feature vector indexed by *n* and *v* corresponds to the *i*-th attribute of feature vector indexed by *k*, *f* is the feature attribute index.

The posterior probability for pattern *k* is given by the weighted regression of the *n* different nearest neighbours of *k*,

$$P_k = \frac{\sum_n W_{nk} O_{nk}}{\sum_n W_{nk}}$$

where O_{nk} is the neighbor's label and it can take the value 1 or 0. Let the mean square training error be

$$E = \frac{1}{2} \sum_k (1 - P_k)^2 .$$

With using the gradient descent method, the matrix element $S_{ij,f}$ for the attribute *f* are adjusted, in the *n+1* iterations, according to:

$$S_{ij,f}^{n+1} = S_{ij,f}^{n} - r \frac{\partial E}{\partial S_{ij,f}^{n}} \qquad (1)$$

where $\frac{\partial E}{\partial S_{ij,f}} = -\sum_k (1 - P_k) \frac{\partial P_k}{\partial S_{ij,f}}$ *(2)*

We also have

$$\frac{\partial P_k}{\partial S_{ij,f}} = \frac{\partial}{S_{ij,f}} \left(\frac{\sum_n W_{nk} O_{nk}}{\sum_n W_{nk}} \right)$$

$$= \frac{\sum_n \frac{\partial W_{nk}}{\partial S_{ij,f}} (O_{nk} - P_k)}{\sum_n W_{nk}} \qquad (3)$$

$$\frac{\partial W_{nk}}{\partial S_{ij,f}} = W_{nk} \frac{\partial}{\partial S_{ij,f}} \left(\frac{-d_{nk}^2}{2\sigma} \right)$$

$$= \frac{-d_{nk}}{\sigma^2} \left(\frac{\partial d_{nk}}{\partial S_{ij,f}} \right) + \frac{d_{nk}^2}{\sigma^3} \left(\frac{\partial \sigma}{\partial S_{ij,f}} \right) \qquad (4)$$

As $d_{nk} = \left(\sum_l S_{v_l w_l}{}^2 \right)^{1/2}$

$$\frac{\partial d_{nk}}{\partial S_{ij,f}} = \frac{1}{2} \left(\sum_l S_{v_l W_l}{}^2 \right)^{-1/2} \left(2 S_{v_f W_f} \frac{\partial S_{v_f w_f}}{\partial S_{ij,f}} \right) \qquad (5)$$

$$= \begin{cases} \frac{S_{v_f w_f}}{d_{nk}} & \text{if } (v_f{=}i, w_f{=}j) or (w_f{=}i, v_f{=}j) \\ 0 & \text{otherwise} \end{cases} \qquad (6)$$

Similarly

$$\frac{\partial \sigma_k}{\partial S_{ij,f}} = \begin{cases} \sum_n \frac{S_{v_{nf} w_{nf}}}{d_{nk}} & \text{if } (v_{nf}{=}i, w_{nf}{=}j) \text{ or } (w_{nf}{=}i, v_{nf}{=}j) \\ 0 & \text{otherwise} \end{cases} \qquad (7)$$

As a result, the values in the intra-feature metric matrix can be adjusted by first using (6) and (7) and then substitute back and eventually calculated using (1). Iterations can be stopped if the changes of the norm of all matrices are smaller than a given threshold. In simulations, we find that only a small number of iterations (10-30 iterations) are enough. Furthermore, in derivation of the equations, it is assumed that there is no change of neighbors during tuning. This may not be true because changes in metric matrix may cause the change of neighbors and thus discontinuities may also be raised. Nevertheless, discontinuities due to changing neighbors do not bring us problems unless we have a very large tuning constant *r*. First, changing of one neighbor, even introduce different label value, does not cause a great change in posterior probability calculation because there are 8 neighbors, and, more important, the effect is further reduced significantly by the Gaussian kernel.

Since there are always limited number of training exemplars and dividing them into training group and testing group further reduce exemplars in calculating training error *E* in (1), cross validation should be used to improve the situation and prevent over training by re-sampling. In fact, our simulation results show that this technique is very useful in boosting classifier performance. Although the equations derived by considering all attributes of pattern vectors are nominal. It is still useful for cases that pattern feature vectors contain both numeric and nominal attributes because our criterion is the same: Find a metric measure for nominal attributes that minimize the training error.

Furthermore, the regularization approach can also be introduced for the construction of the intra-feature matrices [9]. A regularization term can be introduced to stabilize the training process. The regularization energy can be written as

$$E = \frac{1}{2} \sum_k (1 - P_k)^2 - \alpha \sum_i \sum_f (1 - S_{uv,f})^2 \qquad (8)$$

where α is the regularization constant with real positive value. With a high value of α, the trained intra-feature matrices will be closer to the Boolean matrices. The advantages of the regularization approach is that the regularization term helps to bound the coefficient of the intra-feature matrices and is especially suitable for cases where limited training samples limits the use of part of the training sample for the cross-validation.

It should be noted also that the proposed approach is very general and could be further applied to different metric-based machine learning algorithms besides the nearest neighbor classifier. The main purpose is to provide a framework for general metric information for nominal data so that classification algorithms using metric measures can be applied.

3. RESULTS AND COMPARISION

To see whether the metric information obtained with the above approach is useful, various data sets containing nominal data from UCI Repository for machine learning are used for testing[8]. For each dataset, all patterns with missing attributes are removed because we hope differences in testing error are due to metric manipulation only and do not relate to any algorithms or probability manipulations for missing attributes. The used data sets are mushroom, monks-1, monks-3, credit-application, and tic-tac-toe.

For each dataset, three algorithms are used. C4.5, using Boolean metric for calculating nearest neighbor (Boolean metric KNN), IFMM version nearest neighbor (IFMM KNN). The Boolean metric is the classical method of considering match or mismatch in nominal values by setting 1 as non-match and 0 as match. In the matrix representation of IFMM, the Boolean metric equivalent to setting all off-diagonal entries as 1 and diagonal entries to 0. The C4.5 algorithm without pruning is also included in our tests for comparing the proposed algorithm's performance. All training data are selected randomly and the process is repeated 10 times so that the results are more reliable. Since monks-1 and monks-3 data sets have their own specified training sets and testing sets, we only repeat the process by randomizing initial conditions in gradient descent iterations. In each test, only a small portion of data are selected as training data for reflecting the practical situation that labeled data in many classification problems are expensive and they are usually obtained in small amount. The summary for the testing data is given in the following table.

	Training Data	Testing Data	Total Attributes	Nominal Attributes
Mushroom	282	5362	22	22
Credit Applicatio n	194	459	15	9
Monks-1	124	432	6	6
Monks-3	122	432	6	6
Tic-tac-toe	190	768	9	9

Table 1: Summary of the datasets

In all tests, the training data is divided into two groups, the training group and the testing group. In the training phase, we take 2/3 of the training data as the training group and the resting data used for training phase testing. Furthermore, cross validation is also used so that all the training data can perform both the roles of training group and testing group, and, of course, reduce the effect of over training also. After tuning the matrix, the classifier is tested with all testing data and the classification error is given in the Table 2.

	Boolean Metric KNN	**IFMM KNN**	**C4.5**
Mushroom	1.0%	0.9%	**0.4%**
Credit Application	16.7%	**14.5%**	18.4%
Monks-1	17.1%	**0%**	23.4%
Monks-3	9.3%	**3.1%**	7.4%
Tic-tac-toe	17.8%	**9.1%**	27.1%

Table2: Classification error in testing

From the table it is observed that in all five sets of results, the IFMM give lower classification error than Boolean metric based Nearest Neighbor algorithm especially for the cases monk-1 and monk-3. Furthermore, in four sets of experimental results, the IFMM has much lower classification error than the C4.5. One interesting observation is that the accuracy of both KNN algorithms have poorer classification accuracy than C4.5 for mushroom dataset. It is because this dataset is highly dominated by a few attributes and this property makes tree algorithms superior. Comparing the results of IFMM with the results in other nominal metrics[7], the IFMM has much lower errors in Monks-1 and Credit Application, while the IFMM has slightly higher error in Monks-3. The results in mushroom is not comparable as the number of training samples are different. Thus it is observed that the IFMM performs superior on different datasets.

Finally, it is worth mentioning that the problem of metric measure for nominal data is very interesting. Without prior information, it may be not practical to find the metric measure for nominal attribute. However, when a criterion is defined, say minimizing training error in our case, it becomes a well-defined problem where solutions can be obtained. The created metric matrix reflects the metric infrastructure of a nominal attribute for minimizing training error. If some other criteria is defined, the created matrix may be different.

4. REFERENCES

[1] Richard O. Duda, Peter E. Hart & David G. Stork, "*Pattern Classification*", Wiley-Interscience, 2001.

[2] Vladimir Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

[3] J.R. Quinlan, Induction of decision trees. *Machine Learning*, 1(1), 81-106, 1986.

[4] Leo Breiman, Jerome Friedman, Charles J Stone, R.A. Olshen, *Classification and Regression Trees*, Chapman and Hall/CRC, 1984.

[5] J.R. Quinlan, *C4.5: Programs for Machine Learning*, San Mateo, CA:Morgan Kaufmann, 1993.

[6] C. Stanfill and D. Waltz, Toward Memory-Based Reasoning, Comm. of the ACM, 29(12), 1213-1228, 1986.

[7] D.R. Wilson and T.R. Martinez, Improved Heterogeneous Distance Functions, Journal of Artificial Intelligence Review, 11(1-5), 273-314, 1997.

[8] Blake, C.L. & Merz, C.J. (1998). UCI Repository of machine learning databases, Irvine, CA: University of California, Department of Information and Computer Science.

[9] Girosi, F., Jones, M. & Poggio, T. (1995), Regularization theory and neural networks architectures ', *Neural Computation* 7, 219--269.

Intelligent Agents in Resources Allocation

Ng Ka Fung
Hong Kong Baptist University
kfng@comp.hkbu.edu.hk

Abstract

My research topic is applying intelligent agents in resources allocation, to optimize the speed, the utility and success rate of computing resources acquisition.
The research is focused on the strategies agents can be used in acquiring resources in an effective manner.

1. Introduction

Grid computing environment features vast availability of computing resources of various kinds, where users can acquire for accomplishing their tasks, and usually a large pool of requests are pending to be processed, be it a computational task to solve sequential analysis problem, or a request for leasing a large amount of storage capacity for a certain time frame. It is universally acknowledged that users prefer to acquire required resources in grid environment to satisfy their needs while spending at little as possible, while service providers would prefer to maximize their return on investment for leasing their resources to users.

Our research studies the behavior of both kinds of entities by utilizing agent technologies to negotiate for resources, and investigates into the strategies that can help optimizing utilities on each side while boosting overall grid performance.

2. Negotiation

Agents in the simulated grid computing environment are divided into two regiments, resources consumer and resources provider agents. Each agent represents a grid entity, as described in section 3, and carry out negotiation with another regiment to acquire/lease required/possessed resources. The negotiation process is described as follows:

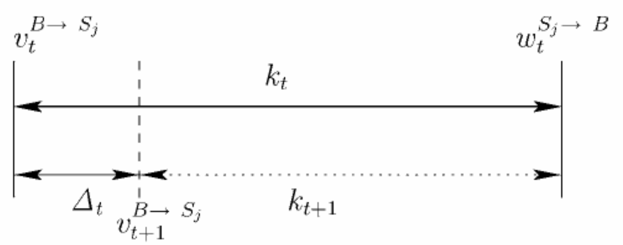


Figure 1. Negotiation Mechanism

Alternating offers protocol: Consumer agents begins the negotiation by proposing an initial offer price to a task towards resource provider agents, and then each resource provider agents evaluates the proposal, it either accepts or proposing a higher price, and return the proposal to the consumer agent [1]. The negotiation terminates either on 1) either agent has reached their negotiation deadline, or 2) either agent has accepted the offer/counteroffer.

Let $v_t^{B\to Sj}$ be the utility of the proposal obtained by the consumer agent, if its proposal is being accepted, and $w_t^{Sj\to B}$ be the utility of the proposal obtained by the provider agent, if its proposal is being accepted, and k_t be the spread between the two proposals. Using the alternating offers protocol, agents make concession Δt in the next negotiation round t+1, such that k_{t+1} is the spread of the next round [1]. Utility is evaluated from price proposal, which is described in section 4.

Each task is assigned a negotiation price set [a, b] where a, b > 0 and b > a. Negotiation begins with initial price a and reserve price b, which consensus will not be reach for any price above b. On the contrary each service provider is also assigned a negotiation price set [c, d], where c, d > 0 and c > d. c is thus the initial price that provider agents begins its negotiation, and d is the reserve price of leasing its resources.

The mechanism of making concession is based on opportunity, time and competition [1].

$$k_{t+1} = f\left[\boldsymbol{O}\left(n_t^B, v_t^{B\to S_j}, \left\langle w_t^{S_j\to B}\right\rangle\right) \boldsymbol{C}\left(m_t^B, n_t^B\right), \boldsymbol{T}(t,\tau,\lambda)\right] k_t.$$

where O is the opportunity function, C is the competition function and T is the time function. O, C and T functions are defined as follows:

$$\boldsymbol{O}\left(n_t^B, v_t^{B\to S_j}, \left\langle w_t^{S_j\to B}\right\rangle\right) = 1 - \prod_{j=1}^{n_t^B} \frac{v_t^{B\to S_j} - w_t^{S_j\to B}}{\left(v_t^{B\to S_j} - c^B\right)}$$

$$\boldsymbol{C}\left(m_t^B, n_t^B\right) = 1 - \left[\left(m_t^B - 1\right)/m_t^B\right]^{n_t^B}$$

$$T(t,\tau,\lambda) = 1 - (t/\tau)^{\lambda}$$

O, C and T functions together determine the amount of concession to be made in the next round. O evaluates all proposals and counter proposals. O evaluates all proposals and counter proposals utility of all its trading alternatives from all trading partners n_t^B. By considering all available trading alternatives, it is actually evaluating the opportunity of reaching a deal in the next round. c^B in the O function is the utility obtained with conflict occurs, in other words it is the utility when no consensus could be reached. It is defined as zero in value. $v_t^{B\to Sj}$ is the utility of agents obtained if its trading partner accepts its proposal, and $w_t^{Sj\to B}$ is the utility an agent can obtain if its accept the counterproposal from its trading partners.

C function considers the competition in the grid, which is essentially the user agent to service provider agent ratio. The more providers the more intense pressure on provider agents to negotiate with consumer agents, and vice versa for consumer agents. m_t^B is the number of agents on the same side and n_t^B is the number of trading partners.

While O and C take all trading partners into consideration, T function determines a particular task's negotiation deadline. As approaching of deadline pushes agents negotiation on intense pressure to reach a deal, the quicker the deadline approaches, the more concession it should gives. t is the consumed negotiation rounds and Γ is the total number of negotiation rounds permitted. λ is a real number from 0 to 1, which involves strategies with time function. For details of the relevant strategies with these functions please refer to [1].

Negotiation of the testbed grid agents (section 3) reuses all these three concession functions.

This research project focused on concession optimization in negotiation, meaning to minimize concession, while obtaining high success rate in a speedy manner.

3. Testbed and Simulation

Currently a testbed has been implemented to simulate the grid environment and the resources trading market for agents to trade required resources. Figure 2 shows a simplified architecture of the testbed.

The grid negotiation testbed consists of (i) a transaction service agent (TSA) which records all successfully negotiated deals information such as agents' utility for reaching the agreement and the current grid utilization level for statistic purpose, (ii) a set of grid resources represented by a set of provider agents, (iii) a set of grid resources consumer agents with a list of tasks to be executed, (iv) an agent registry which provide directory services and list of all agents and (v) a grid controller.

Resource Provider and Consumer Agents: Once agents are generated to represent their respective consumers and providers, they register themselves with the agent registry so trading partners are able to locate them. Consumer agents start negotiation with provider agents once a task arrives, until either an agreement is reached, or the negotiation deadline is reached. Provider agents response to request negotiation on the resources they can sell until the end of their lifespan.

*Grid Controlle*r: The grid controller is responsible for simulating the entrance of resource consumer agents and provider agents to the market, to represent resource consumer and provider respectively. Additionally it generates the resources that provider possesses and tasks consumers need to dispatch for execution. It coordinates the exchange of negotiation messages among resource provider agents and consumer agents. To facilitate comparison of performance between agents with and without relaxation, the controller is responsible to divide consumer agents into two groups by enabling relaxation mechanism to only half of the consumer agents.

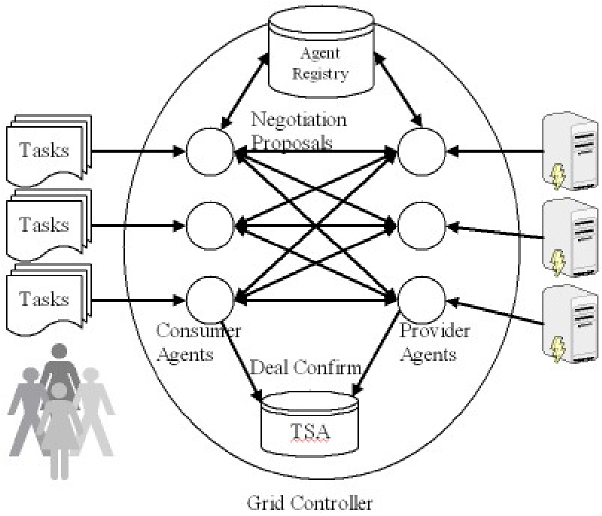


Figure 2. Testbed Architecture

Tasks: Each computational task is associated with two parameters: million of instructions to be executed and the speed of number of million instructions to be completed in one second. Each is randomly assigned a price range for agents to negotiate with the provider agents.

Service Providers: Each service provider must possess certain capabilities that can be leased to any grid agents, such as computational capacity or storage capacity.

Information: To resemble the complexity and difficulty for agents to negotiate in grid environment, agents only have information on it's own states – such as utilization, negotiation price offers, and its history such as number of failure. They are also assumed the information of agents contact point from the agent's registry. However they don't have information on complete view such as global utilization, or other agents' negotiation price set, etc.

4. Performance Measure

Major performance measures of the grid simulation system are (i) success rate, the proportion of tasks successfully acquired required resources, (ii) the speed of acquiring resources and (iii) the utility of the agents, that is the budget spent for successfully acquiring resources.

Success Rate	$R_{success} = N_{success} / N_{total}$
Expected Utiltiy	$U_{expected} = (\sum U_{success} + \sum U_{fail}) / N_{total}$ $= \sum U_{success} / N_{total}$
$N_{success}$	Number of tasks reached consensus
N_{total}	Total number of tasks negotiated
$U_{success}$	Average utility of a task that reached consensus
$U_{fail} = 0$	Average utility of a task that no consensus reached

The speed of acquiring resources corresponds to the number of negotiation rounds spent until a consensus is reached on both agents. The less rounds spent, the faster the acquisition.

Utility function: Let l_{min} and l_{max} be the initial price and reserve price respectively for tasks in consumer agent, (the reverse for provider agent), and l be the price that consensus is reached by both side, then

Consumer: $U_{success} = v_{min} + (1 - v_{min}) (l_{max} - l) / (l_{max} - l_{min})$
Provider: $U_{success} = v_{min} + (1 - v_{min}) (l - l_{min}) / (l_{max} - l_{min})$

where v_{min} is the minimum utility that agents will get for reaching a deal at reserve price. In this experiment the value of v_{min} is defined to be 0.1. A value that is too close to zero would imply there is little differences with not reaching a deal, where a value too high would make agents making concession easily to aim for success rate.

5. Experiment Settings

To investigate and search for adequate agent's strategies that could adapt and react to wide variety of different grid environments, simulations under various situations are going to be performed.
(I) Market Density
A probability that the grid controller may generate a new agent to the grid market in each negotiation round, which may either be a provider side or a consumer.

(II) User to Service Provider Ratio
User to service provider ratio effectively affects the success rate of acquiring resources, as a matter of fact this is a key element of competition. Tuning the ratio would simulate stressing either side of agents to reach consensus on making a deal, or a balanced market can be obtained by using a 1:1 ratio.

(III) Mean Event Rate
Each user will be assigned a mean event rate, the probability that a task will arrive in any negotiation round during the agent's lifespan. The rate is used to simulate

(IV) Agent's Lifespan
Each agent survives, or remains active in the grid for a certain time only. This simulates the environment that the

grid is highly dynamic – users and service providers are continuously going in and out of the grid.

(V) Negotiation Deadline
Task under negotiation must reach a deal before the deadline or it will be classified as failed. Shorter deadline implies agents are under more stress to reach consensus.

These parameters create a large space of combinations of grid environment and thus complexity that agents need to react autonomously.

6. Current Progress

We did some experiments with aforementioned simulation and different experiment parameters. Preliminary results are obtained and analysis is on the way. Further development and implementation is also on progress, and we are drafting a technical report for the analysis on the preliminary results.

7. References

1. Equilibria, Prudent Compromises, and the “Waiting” Game, Kwang Mong Sim, IEEE Tranactions on Systems, Man and Cybernatics Part B. Vol 35. No 4, August 2005
2. Agents that react to changing market situations, Kwang Mong Sim and Chung Yu Choi, IEEE Transactions on Systems, Man and Cybernetics Part B. Vol 33, No.2, April 2003

Service-Oriented Implementation of Distributed Data Mining

Ho-Fai Wong, Xiaofeng Zhang, William K. Cheung
Department of Computer Science
Hong Kong Baptist University, Hong Kong
{hfwong, xfzhang, william}@comp.hkbu.edu.hk

Abstract

Current implementations of distributed model-based data mining (DMDM) are mainly focused on parallel computing architecture, in which the computation environment is aggregated in homogeneous computing resources, performance of the system is largely controlled by the executing hardwares. In this paper, the service-oriented architecture (SOA) based DMDM is purposed and demonstrated. In the service-oriented design, a set of portable, self-contained data mining modules are implemented as web services and distributed throughout the web. Services orchestration is performed therefore the performances of the mining process can be optimized. By using this framework, 3 goals can be achieved: (1) the processes can be run in heterogenic computation resources, (2) greater flexibility can be gained during implementations, and (3) the mining processes can be easily scalable.

1 Introduction

Data mining is one of the hottest research issue in information science recent years. Data mining, also known as knowledge-discovery in databases (KDD), is the practice of automatically searching large stores of data for patterns. To do this, data mining uses computational techniques from statistics and pattern recognition [11, 6]. To perform data mining in computation system, most data mining algorithms assume that the data collected / extracted from the production system have been properly stored at a server, waiting for some subsequent computationally intensive data crunching process. 2 main challenges raised: Privacy concerns and computation resources limitations. To solve these challenges, distributed data mining (DDM) approach is purposed to minimize the harms caused by the challenges [9].

A DDM problem is concerned with how to discover patterns from a complete set of data which are however partitioned and physically distributed at different remote data sources that makes direct data sharing infeasible. Many DDM techniques are purposed such as [8, 4]. In this paper, service-oriented architecture (SOA) approach is adopted for designing the distributed model-based data mining. 3 advantages are gained from the service-oriented distributed model-based data mining (DMDM) approach: (1) data privacy is protected, (2) network bandwidth requirements can be minimized, and (3) the computational complexity of the global data analysis is reduced.

Service-oriented architecture (SOA) is a loosely coupled, conventional XML-based, distributed computing environments. SOA aims to provide seamless integration of self-contained computational services so that they can communicate and coordinate with each other to perform some activities. This concept blossoms in the past few years due to the recent advent of Web Services. XML-based Web Services standards such as WSDL, UDDI, SOAP and BPEL are widely adopted for building intra / inter-organizational systems, with the objective of making their systems more interoperable. Recently some existing data-mining middleware projects are built on the distributed Services-oriented platform [3, 2]. One of the example on SOA enabled application is Weka4WS [10]. Weka4WS is recently implemented with distributed data mining properties and can be used on the Globus toolkit (GT) - one of de facto standard of API in grid environment. The grid, starting from the GT3, has become service-oriented [5].

There are 2 parts for the distributed model-based data mining (DMDM) systems in SOA: local data generalizations and global analysis. For local data generalization, data abstractions are extracted from the distributed data sources, and the privacy of the data is preserved since the original data will be abstractized before further processing. For global analysis, all local data abstractions are collected and global models are learnt in the global analysis services. To implement the proposed DMDM, the conventional Web Services standards are adopted for ease the services interactions. 2 main standards are emphasized : Web Services Description Language (WSDL) is adopted to standardize the web services interface and the Business Process Execution

Language for Web Services (BPEL) is adopted for web services execution.

The organization of the paper is shown as follow: Brief specifications and formulations for distributed data mining modeling discussed in [13, 12] are given in the first part of the paper, Agglomerative Hierarchical Clustering (AGH) is applied in local data generalization and the modified Expectation-Maximization (EM) algorithm is adopted for the global model learning. The second part of the system is the architecture design of the SOA-based DMDM system, the design considerations will be discussed in this section. And a SOA-based data mining application is implemented and experiments are tested on this platform, results shows that the granularity and number of process in local abstractions are crucial for speed up the DMDM.

2 Model-based Data Mining in Service-Oriented Platform

From the system development point of view, the advantages of adopting the service oriented architecture for DMDM are at least three-fold. First, we only need to focus on the implementation of the data-mining Web services without taking care of the interoperation details. Second, we can easily modify DMDM applications via reconfigurating the flow done by editing the BPEL specification. Third, the local data granularity concern specified as data privacy policy can be readily be supported by the design of SOA.

2.1 Service-Oriented Design with WSDL and BPEL

In services-oriented design, the construction of the DMDM can be viewed as 3 key components:

Services discovery To seek the necessary web services for the data mining. Users could search the suitable mining services in UDDI registries.

Services composition and orchestration To organize the web services to serve the desired mining purposes. For composing an application based on web services, Web Service Description Language (WSDL) is a standard which is describing the services' interface.

Services execution To execute the selected web services to achieve the mining goals. Business Process Execution Language (BPEL) is a web services execution language for executing the selected mining services.

In this paper, the main focuses are on the syntactic service composition and the service execution with the uses of the WSDL and BPEL. After discovering a pool of useful web services from UDDI registries, each services is evaluated in order to choose the relevant services for execution. Web Service Description Language (WSDL) is an XML-based services description language, which specifies the interface of the web services. It defines the functions provided by the services, input and output semantics of the services. During the syntactic services composition, the functions of the services determine the role the services should be performed in the process, for example, any web service has data provisioning function is acted as the data provider in DMDM system.

Very often, the output of a service will be fed to the subsequent services as the input parameter. In WSDL documents, input and output parameters of each function as well as the formats of the parameters are defined in detail. To orchestrate different services, one of the key criteria is to convert the output parameters into the right formats with respect to the input parameters of its subsequent services. Otherwise, unexpected exceptions might be caused if different formats of data are used in parameter assignments.

To execute the composed web services, Business Process Execution Language (BPEL), the emerging standard of an XML-based process description language, is the most suitable candidate which specifies the DMDM process. BPEL enables the top-down realization of Service Oriented Architecture (SOA) through composition, orchestration, and coordination of Web services. BPEL provides a relatively easy and straightforward way to compose several Web services into new composite services called business processes [7].

A BPEL document can glue those loosely coupled web services into a structured, well-organized business process, and it can be executed on any web services execution environment that support BPEL. 4 advantages are induced when using the BPEL to execute the processes: (1) the implementation of the services can just focus on the functions without considering the interoperations between processes, preserving the loosely coupled nature of the service-oriented design. (2) Flexibility of implementation can be gained due to the services are broken down into modules. (3) The process can be easily scaled up or down in BPEL without affecting the low level implementation of the services by updating the number of parallel flows in the processes, speed up can be gained also if higher degree of parallelism are implemented. (4) BPEL can simplify the messy web services calls, as a BPEL document can be viewed as a process, or a single web service, therefore user can call the BPEL process once, and it performs all the necessary services calls.

Some syntax are frequently used in BPEL documents in order to execute the web services [1]:

`<partnerLink>` Each `<partnerLink>` belongs to one web services provider. The role of the service is also stated in attribute `partnerRole`.

Figure 1. Segment of a BPEL document

```
<variables>
    <variable name="dataRes"
        messageType="nsxml3:loadDataResponse"/>
    <variable name="distReq"
        messageType="nsxml1:distanceRequest"/>
    <variable name="distRes"
        messageType="nsxml1:distanceResponse"/>
    <variable name="clusterReq"
        messageType="nsxml2:AggloClusteringRequest"/>
    <variable name="dataReq"messageType="
        nsxml3:loadDataRequest" />
</variables>
<invoke name="cs8205-data" partnerLink="CS8205-Data"
    portType="nsxml3:DataStore" operation="loadData"
    outputVariable="dataRes" inputVariable="dataReq">
</invoke>
```

`<receive>` Receives the user request. Without receiving any input parameter by the user, the BPEL process will not perform anything. The receive tag contains an attribute `operation`, which is the function name for the user to call the BPEL process.

`<variable>` Variable declaration for temporarily storage of the processing data, the variable format should be either message type variable (e.g. input message of a web services) or simple XML type (e.g. integer specified in XML Schema).

`<sequence>` All procedures stated inside the tag `<sequence>` will be executed sequentially.

`<flow>` To execute the services in parallel, all pipelined services should be inside `<sequence>`, tagged by `<flow>`. The number of pipeline flows are determined by the number of `<sequence>`.

`<invoke>` Invoke the necessary web services. When the service is invoked, the service provider, `<partnerLink>`, is needed to be specified. Furthermore, the port type of the service and the functions of this service are specified in attributes `portType` and `operation` respectively. The variable containing the input parameter and the variable going to receive the returning results from the service are also needed to be specified.

`<assign>` Variable assignment, `<copy>` is required inside `<assign>` for variable assignment. Each `<copy>` performs one variable assignment operations and the values of variable stated inside `<from>` tag will be assigned into the variable stated inside `<to>` tag. Both `<from>` and `<to>` should be appeared in pairs.

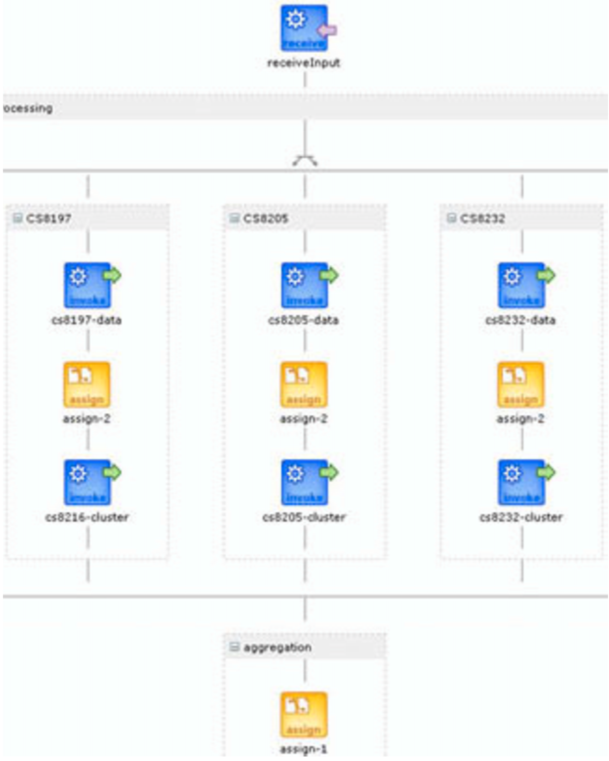


Figure 2. A workflow composed by Business Process Execution Language (BPEL).

`<reply>` Reply the user together with the execution results of the BPEL process. It contains an attribute `operation`, which is the function name of the BPEL process. The function name is the same as that stated in `<receive>`.

2.2 Architecture

In DMDM system purposed in this paper, it can be partitioned into 3 main steps: local data provisioning service, local data generalization (with AGH implemented) and global data analysis (with GMM learning implemented). Local data provisioning is a data queries services, and it provides data for mining. Local data generalization performs the model-based clustering algorithms to extract the hierarchical local data abstractions. And the global data analysis service gathers all local data abstraction and performs machine learning algorithms on these abstractions. Reference to the introduction, the first 2 steps can be grouped into local data generalization part of DMDM system, and the third step can be invoked after the first part of operations are completed.

To gain the speed up by using BPEL, the first 2 steps are split into pipeline flows and each flow aggregated with both steps together to minimize the communication overhead. One design issue is how to break down the DMDM process into steps, and thus services to be coupled via message exchange. If the breakdown is too fine, the communication overhead is too coarse, the reusability of the services will be limited, the effects of these overhead are shown in the experiment section, it is largely related to the granularity of the data.

A sample design of the DMDM system by using BPEL is shown as fig 2. The first 2 steps in the DMDM system can be duplicated and executed in pipeline manners. All the results will be aggregated into a global data analysis service. One of the key design issue is the syntactic services orchestration, which concerns the input-output semantics matching. As data formats of input or output parameters are different with different services, it cannot directly be fed from ones output into the others input directly. In our design, BPEL documents need to handle the integrity of the parameters' formats. For example, the global data analysis requires an array of matrices as the one of the input parameter however the local data generalization only export a set of abstraction matrices. BPEL needs to aggregate all the abstractions altogether to form an array in order to coping with the input parameter of the global data analysis services.

2.3 Formulation

In this paper, Agglomerative Hierarchical Clustering (AGH) is implemented for local data generalization. The basic idea of AGH is as follow: When initialization, all data are assigned as a cluster randomly. Abstractions of each cluster, including the mean μ, covariance matrix Σ, square of the expectation $E(X^2)$, are computed. For data with dimension D, a D-by-D covariance matrix for each cluster is initialized with random values which prevents singular value. Furthermore, the Euclidean distances of each cluster are computed which are used as the clustering criteria during the clusters merging purposes.

During the merging process, 2 clusters with minimum distance are merged and a larger cluster formed, the data points under both clusters are aggregated to form a new set of the new cluster. The means, covariance, expectation and the distance of the newly formed clusters are computed.

For any i^{th} and j^{th} clusters which are going to merge, the abstractions of the newly merged clusters, l^{th}, are computed as follows:

$$\mu_l = \frac{N_i\mu_i + N_j\mu_j}{N_i + N_j} \quad (1)$$

$$E(X_l^2) = \frac{N_i E(X_i^2) + N_j E(X_j^2)}{N_i + N_j} \quad (2)$$

$$\Sigma_l = E(X_l^2) - \mu_l^T \mu_l \quad (3)$$

where the square of the expectation $E(X_l^2) = E(X_l^T X_l)$, N_i and N_j are the number of data points in clusters i^{th} and j^{th} respectively . The merging processes are performed iteratively until a single cluster left finally, i.e. there are at most N levels of hierarchical clusters formed. The number of clusters (number of data abstraction set) depends on the level of hierarchy retrieved. Higher the level, less clusters abstractions can be retrieved, and less information are found as well.

For each level of hierarchy, Gaussian mixture model (GMM) is applied on the local / global data clustering. Assume that there are totally L local sources. Let $t_i \epsilon D_t$ denotes a data item of dimension d at the l^{th} local source, Θ_l denotes the set of parameters of the local GMM with K_l components for abstracting the l^{th} source. The corresponding probability density function can be written as,

$$p_{local}(t_i|\Theta_l) = \sum_{j=1}^{K_l} \alpha_{jl} p_j(t_i|\theta_l j) \quad (4)$$

where $\sum_{j=1}^{K_l} \alpha_{jl} = 1$, and

$$p_j(t_i|\theta_{lj}) = \frac{1}{(2\pi|\Sigma_j|)^{d/2}} \exp(-\frac{(t_i - \mu_j)^T \Sigma_j^{-1}(t_i - \mu_j)}{2}) \quad (5)$$

After all Θ_l is computed, they will be aggregated to form an aggregated GMM at the global server by simply adding the pdf of the local GMMs together and recomputing their mixing portions of the enlarged set of local components by referring the number of data abstracted by each of them. Modified Expectation-Maximization (EM) algorithm is implemented on the global data modeling services.

The global model based learning, given that a global GMM with M components, the probability density function can be written as,

$$p_{global}(t_i|\Theta_g) = \sum_{k=1}^{M} \alpha_k p_k(t_i|\theta_k).$$

The global GMM's parameters can be learnt by using a modified EM algorithm [12]. The posterior probability for a local GMM component which is generated by a global GMM component can be computed as

$$R_{lk} = \frac{\alpha_k \exp\{-\zeta D(p_{global}(t|\Theta_k^{ML})||p_{local}(t|\Theta_l^{ML}))\}}{\sum_{i=1}^{M} \alpha_k \exp\{-\zeta D(p_{global}(t|\Theta_k^{ML})||p_{local}(t|\Theta_l^{ML}))\}} \quad (6)$$

where $D(P||Q)$ denotes the distance between two probabilistic models P and Q. If P and Q are identical, then $D(P||Q) = 0$. $D(p_{global}(t|\Theta_k^{ML})||p_{local}(t|\Theta_l^{ML})$ can be derived as

$$\ln \frac{|\Sigma_l|^{\frac{1}{2}}}{|\Sigma_k|^{\frac{1}{2}}} + \frac{1}{2}[trace(\Sigma_l^{-1}\Sigma_k) + (\mu_k - \mu_l)^T \Sigma_l^{-1}(\mu_l - \mu_k) - d]$$

The new M-step can be given as

$$\mu_k = \frac{\sum_{l=1}^{L} R_{lk}\mu_l}{\sum_{l=1}^{L} R_{lk}} \quad (7)$$

$$\alpha_k = \frac{1}{L}\sum_{l=1}^{L} R_{lk} \quad (8)$$

$$\Sigma_k = \frac{\Sigma_{l=1}^{L} R_{lk}(\Sigma_l + \mu_l \mu_l^T)}{\Sigma_{l=1}^{L} R_{lk}} - \mu_k \mu_k^T. \quad (9)$$

where μ_l and Σ_l are the local component's mean and covariance matrix, and μ_k an Σ_k are those for the global one.

The E-Step and M-Step iterate alternatively until the parameter estimates converge.

3 Experiments

To evaluate the performance of the proposed DMDM being specified in BPEL and executed on the service-oriented BPEL platform, a set of 2000 two-dimensional data points was first generated from a pre-defined GMM with five components. Then, the data set was randomly partitioned and assigned to the storage of the different local data provision services as local data sources. To mimic the distributed environment, web services were built and deployed at different machines for the testing. For each local data abstraction service, a hierarchy of GMMs as well as the privacy measures for all the levels were computed and stored when they were first invoked. Then, the privacy measure for each local abstraction was set and the corresponding set of local model parameters were sent to the global services for aggregation and GMM learning. By applying previously proposed method, the performance of global model under different local privacy requirements were compared. Regarding the details of the execution environment, all the web services were deployed in Apache tomcat version 5.5 servers, which are equipped with Axis version 1.2 web service library. The Java SDK version was 1.4.2. For the service execution, the Oracle BPEL execution engine version 10.1.2 were used.

3.1 Evaluation of Degree of Parallelism and Speed up

To evaluate the systems efficiency, we first performed the DMDM process based on different numbers of local sources. The average speed up gained in 5 independent runs was reported in Table 3.1). Note that Column 2 to Column 4 show the time spent for the corresponding steps and Column 5 shows the overall execution time. All the timings were reported by the BPEL engine. The communication cost was not explicitly specified but can be easily deduced. As expected, the speed up increased as the number of local sources increased. Also, it was found that the communication overhead between the services throughout the process was quite significant (from 8 to 50 % of the total execution time). For the mining results, figure 3 plots the clustering results from 2 to 5 separate local data generation processes. The location of global clusters have a bit differences by using different number of local data generation sources. One of the possible causes is due to the data set of each local data generation service has been changed once the number of distribution changed, changing of size of the data set might change the distribution of the data as well. Therefore, the abstractions generated from the local data generation services are differed.

no. of process	data transfer	local clustering	global clustering	execution time
2	1	20 - 30	2	73.2
3	1	15 - 21	2	35.8
4	1	5 - 15	2	28.3
5	1	5 - 12	2	16.4

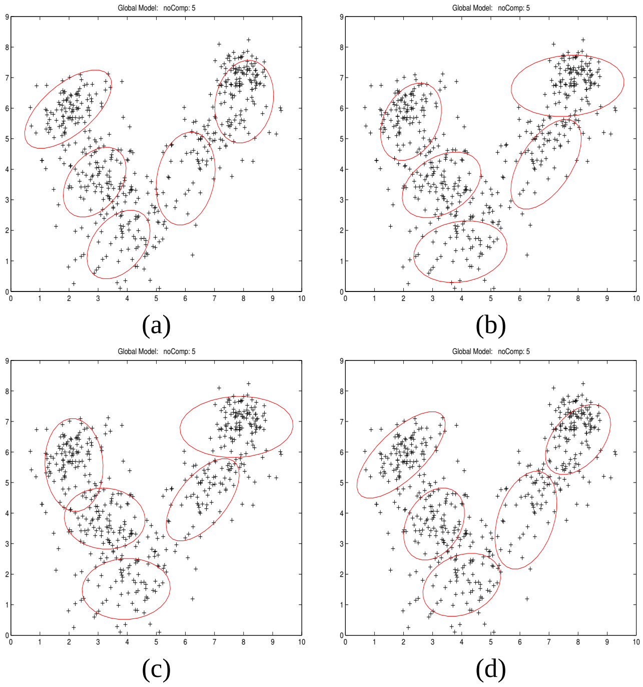
Table 1. Average execution time (in second) with different number of parallel process

3.2 Evaluation of Data Granularity and Speed up

To further investigate the effect of the communication overhead, we conducted another experiment by deliberately reducing the data size and executed again the DMDM process with different numbers of local sources (see Table 2). It was observed that when the overall number of data items decreased from 2000 to 500, the overall execution time for running the DMDM process with five local sources was found to be in fact even higher than that with four local sources. It can be explained by the fact that the cost for the computational parts will decrease up to a limit as the data are partitioned into smaller parts. However, the cost for the BPEL engine to communicate with the services increases constantly and up to a point that the communication cost dominates and the benefit brought by the parallelism diminishes.

3.3 Conclusion

In this paper, we demonstrate the design and architecture of the distributed model-based data mining (DMDM) appli-



(a) (b) (c) (d)

Figure 3. 5 global clusters with (a) 2, (b) 3, (c) 4 and (d) 5 parallel local clustering processes.

no. of	execution time with data size			
process	500	1000	1500	2000
2	15.8	21.0	39.0	73.2
3	9.7	10.9	20.9	35.8
4	**9.2**	**10.4**	18.8	28.3
5	10.7	14.1	**16.3**	**16.4**

Table 2. Average execution time (in second) with different number of parallel process in different size of data set

cation based on service-oriented architecture (SOA). By using the emerging XML-based web service standards such as UDDI, WSDL, BPEL etc, a flexible, scalable DMDM application are built with significant improvement of speed up. BPEL, one of the widely adopted web service orchestration and execution language, can effectively build a DMDM application with enormous benefits. Performance evaluation shows that even communication overhead cannot be negligible during the execution, however we can optimize the performance by adjusting the number of pipeline flows and the granularity of the local data. For future work, it will be focused on the performance improvements on distributed data mining algorithms plus implementing the intelligences on the SOA-based DMDM, especially on service discovery and service composition.

3.4 Acknowledgement

This work is partially supported by RGC Central Allocation HKBU 2/03C.

References

[1] Business process execution language for web services version 1.1. Technical report, BEA, IBM, Microsoft, SAP AG, Siebel, May 2003.

[2] M. Cannataro and C. Comito. A data mining ontology for grid programming. In *Proceedings of the First International Workshop on Semantics in Peer-to-Peer and Grid Computing*, Budapest, 20-24 May 2003.

[3] M. Cannataro, D. Talia, and P. Trunfio. Design of distributed data mining applications on the knowledge grid. In *Proceedings National Science Foundation Workshop on Next Generation Data Mining*, Baltimore, November 2002.

[4] R. Chen and S. Krishnamoorthy. A New Algorithm for Learning Parameters of a Bayesian Network from Distributed Data. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002)*, pages 585–588, Maebashi City, Japan, December 2002. IEEE Computer Society.

[5] I. Foster. Globus toolkit version 4: Software for service-oriented systems. In *IFIP International Conference on Network and Parallel Computing*, LNCS 3779, pages 2–13. Springer-Verlag, 2005.

[6] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT Press, Cambridge, MA, 2001.

[7] M. Juric. A hands-on introduction to bpel. http://www.oracle.com/technology/pub/articles/matjaz_bpel1.html.

[8] H. Kargupta, B. Park, D. Hershberger, and E. Johnson. Collective Data Mining: A New Perspective Towards Distributed Data Mining. In H. Kargupta and P. Chan, editors, *Advances in Distributed and Parallel Knowledge Discovery*, pages 133–184. MIT/AAAI Press, 2000.

[9] A. Prodromidis and P. Chan. Meta-learning in distributed data mining systems: Issues and approaches, 2000.

[10] D. Talia, P. Trunfio, and O. Verta. Weka4WS: a wsrf-enabled weka toolkit for distributed data mining on grids. In *Proc. of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2005)*, volume 3721 of *LNAI*, pages 309–320. Springer-Verlag, Porto, Portugal, October 2005.

[11] Wikipedia. Data mining. http://en.wikipedia.org/wiki/Data_mining.

[12] X. Zhang and W. K. Cheung. Learning Global Models Based on Distributed Data Abstractions. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 2005)*, pages 1645–1646, Edinburgh, August 2005.

[13] X. Zhang, C. Lam, and W. K. Cheung. Mining local data sources for learning global cluster models via local model exchange. *The IEEE Intelligent Informatics Bulletin*, 4(2), 2004.

Lightweight Piggybacking for Packet Loss Recovery in Internet Telephony

Wing Yan Chow, Yiu Wing Leung
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong, Hong Kong
Email: {wychow, ywleung}@comp.hkbu.edu.hk

Abstract

Internet Telephony has shown a substantial growth in recent years because of its huge potential. A challenge to Internet Telephony is packet loss, which affects voice quality. In this study, we propose a packet loss recovery scheme called lightweight piggybacking for Internet telephony. Using this scheme, the source telephone gateway applies two stages of erasure coding on the low-bit-rate versions of the original voice streams, such that the resulting redundant packets (called lightweight redundant packets) are very small and can be shared by all the original voice streams. Then the source gateway piggybacks these lightweight redundant packets to the original voice streams for transmission to the destination telephone gateway. Compared with the existing packet loss recovery schemes, the proposed scheme needs a smaller redundancy but achieves a smaller packet loss probability because the small redundancy can be fully utilized via sharing among all the original voice streams. We will conduct simulation experiments for performance evaluation.

1. Introduction

Internet Telephony has shown a substantial growth in recent years because of its huge potential [1]. It can be classified into three types: computers to computers, computers to telephones, and telephones to telephones (see Figure 1). In particular, the third type is useful to the general public which may have difficulties in operating computers or accessing to the Internet. It makes use of telephone gateways to bridge local telephone network and the Internet for voice transmission [1-2]. Compared with traditional telephone services, Internet Telephony can significantly reduce maintenance cost and service charge, especially for long-distance calls [2]. Qovia's study [3] shows that delay and packet loss significantly affect voice quality. To tackle these problems, packet loss recovery [4] and concealment [4] methods are used.

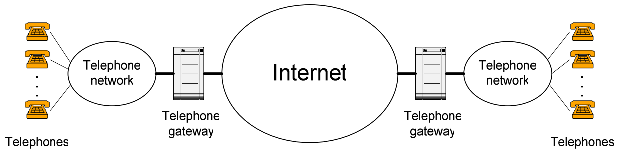


Figure 1. An Internet telephony system based on telephones to telephones

Packet loss recovery methods include XOR Packet Recovery [4] and Forward Error Correction (FEC) [4]. In the XOR Packet Recovery scheme, the source adds a redundant packet by performing bit-by-bit exclusive-OR on n packets. The destination recovers a lost packet by an XOR operation if $n - 1$ packets are received. In FEC, erasure coding [5] is used to produce $n - k$ redundant packets from the original k packets. If the destination can receive any k out of n packets, the lost packets can be recovered. Erasure coding is adopted in shared packet loss recovery [2]. As shown in Figure 2, there are multiple active voice streams in a telephone network. The telephone gateway uses erasure coding to produce redundant packets from these streams. Then the streams can share the redundant packets for loss recovery.

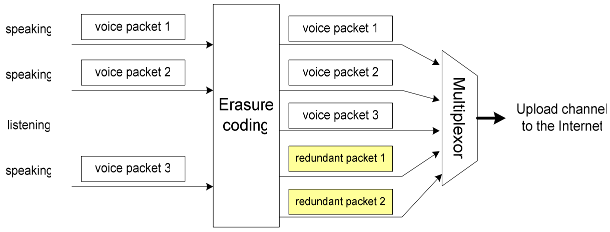


Figure 2. Shared packet loss recovery for multiple active voice streams in Internet Telephony [2]

Error concealment methods comprise interpolation [4], interleaving [4] and piggybacking [6]. Interpolation means that the destination can use neighboring packets to estimate the lost packet. If interleaving is used, audio data units are re-sequenced before transmission. This minimizes the chance of consecutive packet loss. For piggybacking, the source produces low-bit rate audio packets and attaches them to neighboring packets. When a packet is lost during transmission, it can be replaced by the piggybacked packet.

In this paper, we propose an enhanced version of piggybacking called *Lightweight Piggybacking*. The key feature of this scheme is to integrate piggybacking and shared packet loss recovery. Adopting multilayer coding and erasure coding, the source can compress and produce *extra lightweight redundant packets* from multiple voice streams. These packets provide a low-bit-rate voice stream for packet loss recovery. They are *piggybacked* to the voice streams during transmission. In every two contiguous packetization periods, the lightweight packets are *shared* among multiple voice streams. The destination can use the lower quality stream to substitute multiple lost streams. With packet sharing, the total size of redundant packets and packet loss probability are significantly reduced. Lightweight Piggybacking not only saves bandwidth, but also enables more efficient packet loss recovery.

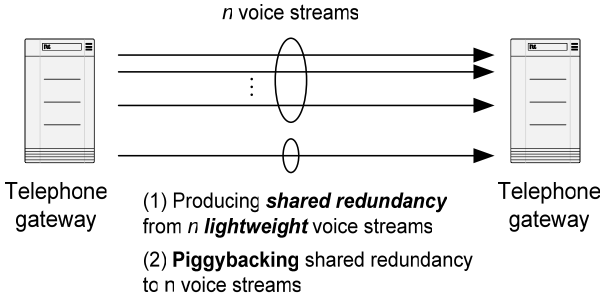

n voice streams
Telephone gateway
(1) Producing ***shared redundancy*** from *n* ***lightweight*** voice streams
(2) **Piggybacking** shared redundancy to n voice streams
Telephone gateway

Figure 3. Piggybacking shared redundancy to voice streams

2. Lightweight Piggybacking

In this section, we discuss the details of Lightweight Piggybacking. Four main steps are involved in the proposed scheme, i.e. forming compressed voice packets, computing redundant voice packets, computing lightweight redundant voice packets and piggybacking.

Step 1: Forming Compressed Voice Packets

In this step, we produce compressed voice packets by dropping less significant bits of voice streams.

A telephone session is active [7] if there is voice stream transmission. We denote the number of active sessions as *n.* Thus the total number of voice packets in each packetization period is *n* [2]. The destination continuously buffers packets in two consecutive packetization periods. In case there is packet loss in current period, redundant packets in next period can provide packet loss recovery. Using multilayer coding [8-9], compressed voice packets can be produced. Given that the user receives more layers, the voice quality is better. But the required bandwidth is therefore larger.

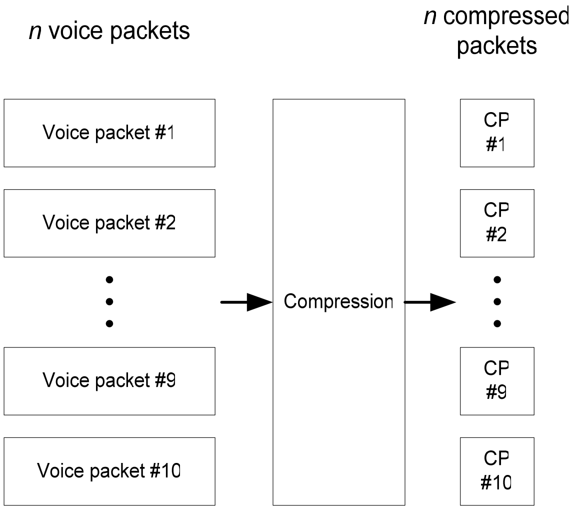


Figure 4. Compression of voice packets. In current period, the less important bits of *n* voice packets are dropped to form *n* compressed packets (number of original voice packets *n* = 10).

Based on the idea of multilayer coding, the source produces compressed voice packets by dropping the less important bits of original voice packets. As shown in Figure 4, *n* compressed voice packets are produced in current period. We define the compressed packets as *lightweight packets* because the packet size is much smaller than original packets. Since the compression method does not require many complicated calculations, it will not significantly delay packet transmission. The source can adjust the compression ratio of voice packets depending on bandwidth requirement. It can choose to drop different number of bits provided that the voice quality is acceptable after compression.

Step 2: Computing Redundant Voice Packets

Having produced lightweight packets, we use erasure coding to compute redundant voice packets.

Instead of producing redundant packets from original voice packets, the source uses the *n* lightweight packets. The redundant information from these packets will be used for packet loss recovery. Erasure coding [5] is adopted in producing redundant packets. The key idea of this coding is to encode *k* blocks of source data into *n* blocks of encoded data where *k* is smaller than *n*. As a result, there is redundant information in the *n* – *k* blocks of encoded data. In each packetization period, there are originally *k* packets. The source produces *n* – *k* redundant packets and transmits all *n* packets to the destination. If the destination can receive any *k* out of *n* packets, the lost packets can be recovered. In short, the received *k* packets contain redundant information of the original *k* packets. The destination can use erasure coding to recover the lost packets.

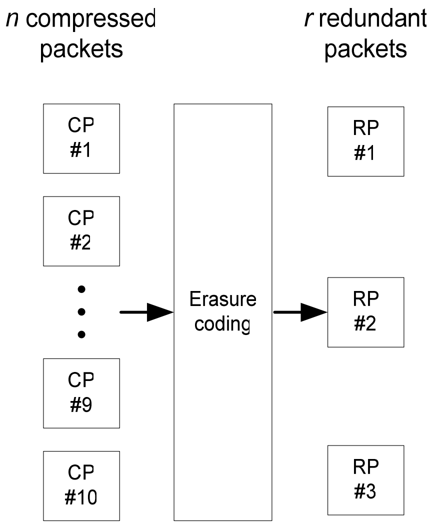


Figure 5. First erasure coding of voice packets. There are *r* redundant packets produced from *n* compressed packets (number of compressed packets *n* = 10, number of redundant packets *r* = 3).

At this point, there are *n* lightweight packets at the telephone gateway (see Figure 5). The source uses erasure coding to produce *r redundant lightweight packets* from *n* lightweight packets. These redundant packets contain redundant information from original voice packets. They can act as substitutes for original voice packets when there is packet loss. The number of redundant packets *r* can be adjusted for better bandwidth utilization. If the source intends to increase the chance of packet loss recovery at the receive side, it needs to produce more redundant packets. Then there is more redundant information to recover lost packets. However, increasing the number of redundant packets will expand the required bandwidth for transmission. There is always a tradeoff between packet loss recovery and bandwidth. Too many redundant packets produced at this step will also affect packet loss recovery. Details will be discussed in next step.

Step 3: Computing Lightweight Redundant Voice Packets

In this step, we further reduce the size of redundant lightweight packets by fragmentation.

The source needs to fragment the *r* redundant lightweight packets into smaller pieces before transmission. To illustrate more clearly the detailed steps, consider Figure 6. Each redundant lightweight packet is fragmented into *p* pieces, forming *extra lightweight packets*. If a packet is fragmented into more pieces, the size of each piece is smaller. Consequently, the total size of redundant packets will be greatly reduced. However, this may reduce the chance of packet loss recovery. The receiver needs to receive more pieces correctly in order to recover all lightweight packets.

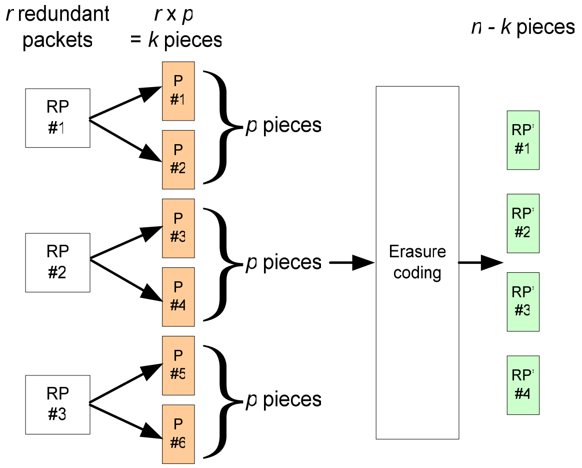


Figure 6. Fragmentation and second erasure coding of voice packets. Each redundant packet is fragmented into *p* pieces. After fragmentation, there are totally *rp* pieces (number of redundant packets *r* = 3, number of pieces *p* = 2). The source then produces *n* - *k* redundant pieces from second erasure coding (number of original voice packets *n* = 10, total number of pieces *k* = *rp* = 6).

After fragmentation, there are totally *rp* pieces. To achieve better results in packet loss recovery, the total number of pieces *rp* should not exceed the number of outgoing voice packets *n*, i.e. $rp < n$. Under this scheme, the destination gets at least *k* out of *n* packets for packet loss recovery. It should receive all *rp* pieces so as to reconstruct *r* redundant packets. Therefore, the *k* described in Step 2 should be equal to *rp*. Now the source has *k* pieces of redundant packets. To attach the *k* pieces on *n* voice packets, it still needs to produce *n* – *k* redundant pieces by second erasure coding. Eventually, there are totally *n* redundant pieces (extra lightweight packets) for *n* voice packets.

Step 4: Piggybacking

To achieve the goal of packet sharing, we need to attach the extra lightweight packets to original voice packets.

Together with the original *k* pieces, the *n* – *k* redundant pieces from second erasure coding are attached to the voice packets in next period. This scenario is depicted in Figure 7. This approach is different from traditional Piggybacking, which requires redundant packets to be attached to packets of the same period. Each voice packet now contains an extra lightweight redundant packet for packet loss recovery. In case there is packet loss in current period, the lost packets can still be recovered by the extra lightweight packets in next period. When the destination receives any *k* out of *n* packets in next period, it can use erasure coding to produce compressed packets from received packets and extra lightweight packets. After substitution, the lost packets are replaced with compressed packets. Although the voice quality produced by compressed packets is relatively lower, it is better than inserting silence or noise to replace the lost packets. The total size of redundant packets is significantly reduced compared with the existing method of piggybacking as well.

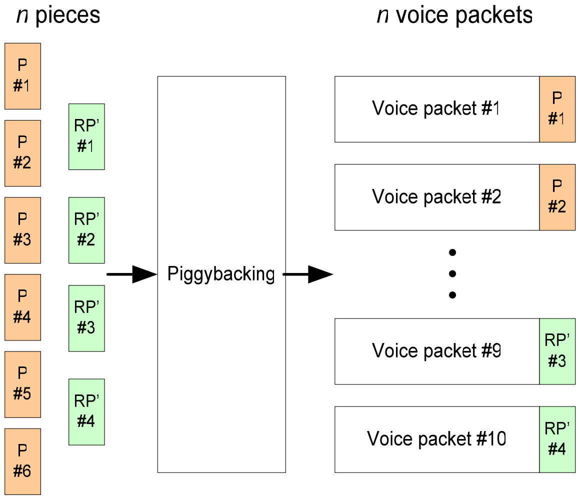


Figure 7. Piggybacking of voice packets. The *n* - *k* redundant pieces are combined with the original *k* pieces to become *n* pieces (total number of pieces *n* = 10, number of original pieces *k* = 6). They are piggybacked on the *n* voice packets of next period.

3. Further Work

We plan to conduct simulation experiments for performance evaluation.

4. References

[1] IEEE Communications Magazine, "Special Issue on Internet Telephony", vol. 38, no. 4, April 2000.

[2] Y. W. Leung, "Shared Packet Loss Recovery for Internet Telephony", *IEEE Communications Letters*, vol. 9, no. 1, January 2005, pp. 84-86.

[3] C. Shim, L. Xie, B. Zhang, and C.J. Sloane, "How Delay and Packet Loss Impact Voice Quality in VoIP", *Qovia, Inc. White Papers*, December 2003, pp. 1-10.

[4] C. Perkins, O. Hodson, and V. Hardman, "A Survey of Packet Loss Recovery Techniques for Streaming Audio", *IEEE Network*, September/October 1998.

[5] L. Rizzo, "Effective Erasure Codes for Reliable Computer Communication Protocols", *ACM Computer Communication Review*, vol. 27, April 1997, pp. 24-36.

[6] J. F. Kurose and K.W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Chapter 6, 2nd ed., Addison Wesley, 2003.

[7] G. Scheets, M. Parperis, and R. Singh, "Voice over the Internet: A Tutorial Discussing Problems and Solutions Associated with Alternative Transport", *IEEE Communications Surveys*, vol. 6, no. 2, April 2004, pp. 22-31.

[8] S. McCanne, M. Vetterli, and V. Jacobson, "Low-Complexity Video Coding for Receiver-Driven Layered Multicast," *IEEE JSAC*, vol. 16, no. 6, August 1997, pp. 983-1001.

[9] B. Girod, K. W. Stuhlmüller, M. Link and U. Horn, "Packet Loss Resilient Internet Video Streaming", *VCIP'99, Proc. SPIE*, vol. 3653, January 1999, pp. 833-844.

The Client-based Framework For Privacy-Preserving Location-based Data Access

DU Jing

Abstract

In many applications adopting spatial databases, clients desire not only the server to respond as soon and correctly as possible, but also the client's precise location can be cloaked from the server. Thus we propose such a novel client-based framework to meet the requirement of privacy preserving . At the time of writing this paper, some research jobs concerning such a framework have been finished and some are still in progress. In this paper we present the main idea of our framework first. Then we discuss details of finished research tasks, which concern R-tree, range nearest neighbor search, dynamic location cloaking and etc. Finally we briefly present the research tasks still in progress.

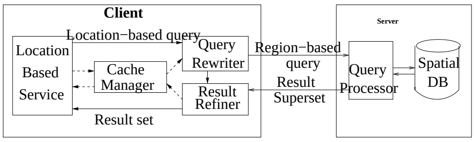


Figure 1. Proposed client-based framework for privacy-preserving location-based data access

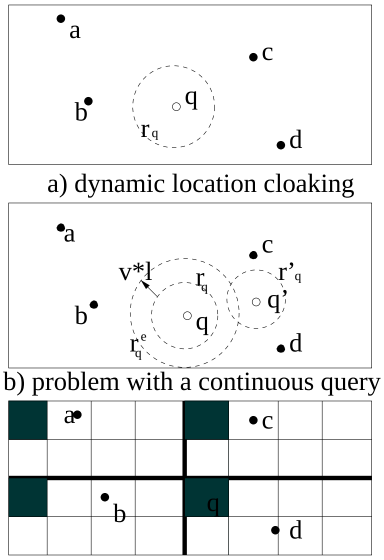
1 Introduction

We consider a client-server architecture, where clients are mobile and equipped with wireless interface to communicate with the server. We assume clients are location-aware - they can position their own locations (e.g., using GPS or WLAN based positioning) or obtain their locations from a trusted location server. Clients are interested in querying information related to their current locations with location privacy retained. Existing middleware-based solutions heavily rely on the trust in middleware service provider. Furthermore, the middleware must seek for the client's consent before use of location data in accordance with laws, which complicates the system administration. We propose a novel client-based framework in which the needs of trust in service provider and consent from client are relieved.

As shown in **Figure 1**, the proposed framework employs a query rewriter in the client to transform a location-based query to a region-based query such that the resolution of current location is reduced before it leaves the client, thereby protecting location privacy. Based on received region-based query, the server evaluates a result superset, which is returned to the client. Finally, the actual result set is computed by a result refiner in the client. To illustrate, in **Figure 2a**, instead of providing current location q and the query of finding the nearest object, the client submits an uncertain region r_q and the query to the server. The server then returns the set of objects that are potentially a nearest object of some point in r_q , i.e.,{b,c,d}. At last, the client uses the exact location q to find out the actual nearest object, i.e., c. Many challenging issues arise in this framework, including how to efficiently transform queries, how to evaluate result supersets, and how to organize the data within a superset. Moreover, the proposed framework includes an optimal caching module to improve data access performance. We propose a privacy-preserving caching technique and address various cache based querying and cache management issues.

The location-based queries we shall examine include range query (e.g., finding the shopping malls within 5 miles), k-nearest-neighbor search (kNN, e.g., finding the nearest gas station), and k-nearest-surrounders search (kNS, e.g., finding the surrounding solider in a certain angle). While range and kNN queries have been extensively researched in the past decade, kNS is an emerging type of spatial queries.

At the time of writing this paper, the research jobs concerning queries transforming, rectangle-shaped region nearest neighbor search and circle- shaped region nearest neighbor search has been done. The details are discussed in the following sections.



a) dynamic location cloaking

b) problem with a continuous query

c) static location cloaking

Figure 2. An Illustrative Example

2 Cloaking Techniques

How should we transform a location-based query before it is sent to the server? In essence, the problem is how to cloak the current location by an uncertain region. In this paper we assume the area of an uncertain region is used to specify the privacy requirement. For example, a user can specify it's acceptable to be located within an area of 1 square mile when she is in a shopping mall or within an area of 10 square miles when she is in the Disneyland. We propose two cloaking techniques:

- Dynamic Cloaking. A random uncertain region is dynamically generated based on current location, e.g., r_q in **Figure 2a**.
- Static Cloaking. The service area is pre-partitioned into a set of grid cells, each of which is further divided into a number of sub-cells. As shown in **Figure 2c**, the service area is partitioned into 4 cells with each consisting of 8 sub-cells. Given current location q, the sub-cell covering q and the corresponding sub-cells in other cells constitute the (discrete) uncertain region (i.e., shaded sub-cells in **Figure 2c**).

Currently we have implemented dynamic circle-shaped cloaking and dynamic rectangle-shaped cloaking.

3 R-Tree and Range Nearest Neighbor Search

One issue concerning the framework is how to evaluate location-based queries with respect to a cloak region (or a set of cloak regions) in the server side.

Some current technologies have been adopted in our framework.

R-Tree[1] index is used in spatial database to organize the spatial point data and facilitate executing various kinds of queries. The evaluation of a range query would be straightforward since it is still a (larger) range query. It is more complex to evaluate kNN queries, in which all potential k nearest objects of some point within the region should be returned. While point-based kNN search has been extensively studied, little work has been done on region-based kNN queries. Regarding that the only work we are aware of is a recent paper by Hu and Lee[2]. According to it there are two important observations: 1) all objects within the region should be returned since they are nearest objects to themselves; 2) all potential nearest objects outside the region must be nearest objects to some point on the region boundary. Thus, a region-based query can be decomposed into a range query and a kNN query with respect to the region boundary. When the boundary is composed of several line segments, finding out kNN of the boundary is equivalent to separately finding out kNN of those line segments. And this problem has been gently settled by Tao et al[3]. In our framework, these achievements have been absorbed to construct the component of handling rectangle-shaped kNN queries.

4 Circle-shaped kNN

4.1 Motivation

As mentioned earlier, the clients are mobile and are equipped with wireless interfaces to communicate with the server. Therefore the energy consumption of clients is a considerable issue. Based on the current research results we want to exploit the feasibility of reducing the clients' load within the framework, say, energy consumed to execute queries.

Our consideration is we could try to choose an optimal cloaking shape to reduce the size of result superset returned by the server side when the cloaking area is fixed, and therefore lower energy consumed to download and refine result supersets.

Immediately the idea of adopting circles as cloaking shapes occurs to us. As mentioned earlier, a region-based query can be decomposed into a range query and a kNN query with respect to the region boundary. As the area of cloaking shape is fixed, if we can shorten the boundary of the cloaking shape, theoretically it's possible to get smaller result superset. When the area of shape is fixed, of what type of shape the boundary is the shortest? Of course, a circle.

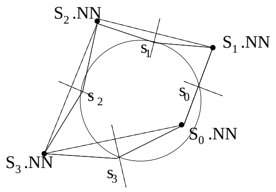


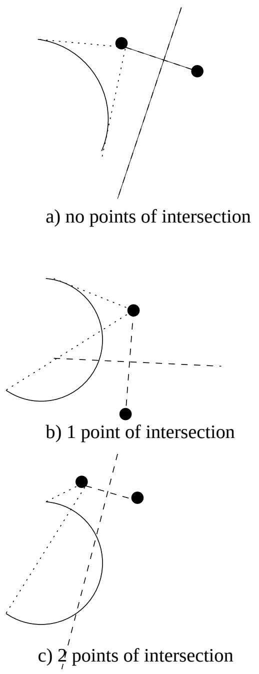
Figure 3. An Illustrative Example For Circle-shaped RNN

4.2 In-Memory Processing Techniques

We first focus on the case when k =1. Very similar to CNN[3], the objective of a circle-boundary NN query is to retrieve the set of nearest neighbors of a circle boundary, c, together with the resulting list SL of split points. For each split point s_i ∈ SL (0 ≤ i<|SL|-1): s_i ∈ c and all points in curve interval [s_i, $s_{i+1(mod|SL|)}$] have the same NN, denoted as si.NN. Refer to **Figure 3** for an example.

In order to avoid multiple database scans, we aim at reporting all split and the corresponding covering points with a single traversal. We start with the SL empty and corresponding covering point sets empty, and incrementally update the SL during query processing. At each step, SL contains the current result with respect to all the data points processed so far. The final result contains each split point s_i that remains in SL after termination together with its nearest neighbor s_i.NN.

Processing a data point p involves updating SL, if p is closer to some point u ∈c than its current nearest neighbor u.NN (i.e., if p covers u). An exhaustive scan of c is intractable because the number of points is infinite. On the other hand, the lemma 3.1and lemma 3.2 in [3] does not hold for the circle boundary. So, for each data point p, a simple exhaustive scan is performed for each split point and corresponding curve interval. We use the following algorithm to check whether the new coming data point p covers part of currently checked split curve interval and identify new split points, if any. Refer to **Figure 4**.We compute the intersection of the curve interval with the perpendicular bisector of p and the covering point. If there are no points of intersection, we must identify whether the new coming point p covers the whole curve interval taking the place of the old covering point or the old covering point still holds. If there are one or two points of intersection, we must identify which parts of the curve interval are covered by the old covering point and which parts are covered by the new point. After all curve intervals are checked, we scan the updated curve intervals and merge the intervals which share the same covering point and are connected.


a) no points of intersection

b) 1 point of intersection

c) 2 points of intersection

Figure 4. In-Memory Processing Techniques

4.3 Secondary Memory Pruning Techniques

Still like CNN, our algorithms employ branch-and-bound techniques to prune the search space. When we traverse the R-tree index, if a leaf entry (i.e., a data point) p is encountered, we just process it using aforementioned in-memory processing techniques; if a intermediate entry is encountered, we will identify whether it's possible that the sub-tree of it contains qualifying data points using several heuristics. Thus we can avoid unnecessary secondary memory access.

Heuristic 1: Given an intermediate entry E and a circle boundary c, the subtree of E may contain qualifying points only if mindist(E, c) $< SL_{MAXD}$, where mindist(E, c) denotes the minimum distance between the MBR of E and c, and SL_{MAXD} is the maximum distance between a covering point and the corresponding curve interval.

Heuristic 2: Given an intermediate entry E and a circle boundary c, the subtree of E must be searched if and only if there exists a curve interval such that $CI_{MAXD} >$ mindist(CI, E), where CI_{MAXD} denotes the maximum distance between this curve interval and its covering point and mindist(CI, E) is the minimum distance between the MBR of E and the curve interval.

Heuristic 3: Entries (satisfying heuristics 1 and 2) are accessed in increasing order of their minimum distances to the circle boundary c.

These heuristics can be deduced easily from the three heuristics given in [3].

4.4 The kNN Query(k>1)

In the situation when k>1, the algorithms doing pruning are almost the same as when k=1 except that CI_{MAXD} should be replaced by the maximum distance between this curve interval and its k^{th} (i.e., the farthest) NN.

On the other hand, the in-memory techniques used here are a little different. For each curve interval, a sweeping algorithm is adopted to identify new split points, if any. Here we have a new coming data point p. At the beginning, the sweep point is the starting point of this interval and the candidate point is p. Then we find out the intersections between the curve interval and ⊥(p, NN_1), between the curve interval and ⊥(p, NN_2), ..., between the curve interval and ⊥(p, NN_k) . Intersections that fall out of the interval are discarded. Amongst the remaining ones, the intersection that has the shortest distance to the starting point becomes new split point. And new generated sub-interval must be given updated NNs. We apply the sweeping algorithm on the remaining part of the original interval with the candidate point changed to the point which fails to become one of NNs of new generated sub-interval. We do sweeping until no new split point is generated. Finally a scan-and-merge processing is applied.

4.5 Experiments

We have done a series of experiments to compare the performance of rectangle cloaking NN query handling algorithms and circle cloaking NN query handling algorithms.

The scenario of experiment is as follows. We set up a R-tree based spatial database server containing about 20000 2D data, which are the coordinates of populated places of USA plus Mexico[4]. Then we use an iPaq Pocket PC, which the client software sits on, to communicate with the server through wireless LAN. And we have done eight groups of experiments using different parameters. In every group we randomly send 100 queries to the server, separately using circle-shaped cloaking and rectangle-shaped cloaking.

As expected, the average size of result superset of circle-shaped RNN query is smaller than that of rectangle-shaped RNN query. Please refer to **Table 1**.

Unexpectedly, the measure figures of power consumption don't support our prediction. Please refer to **Table 2** first. As we can see, obviously the rectangle cloaking NN query handling consumes much less power of the iPaq Pocket PC than the circle cloaking NN query handling.

How does this result come? Following is the analysis.

The energy consumption in the client side consists of three parts: 1) the energy used to send queries, 2) the energy used to wait for responses, and 3) the energy used to download and refine result superset. As to part 1 both rectangle and circle consume almost the same energy. As to part 3 circle consumes less due to a little smaller result superset. But as to part 2, in the process of experiment, we find when circle cloaking is adopted the waiting time is about twice of that used by rectangle cloaking; therefore circle consumes more due to longer waiting time. Finally the consumed energy from part 3 becomes crucial. And we temporarily think that the reason why circle cloaking gives responses slower is the circle-shaped RNN query handling algorithm is much more complicated than the rectangle-shaped RNN query handling algorithm.

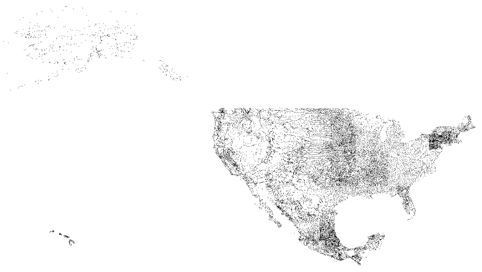


Figure 5. The spatial dataset used in experiments

In one word, when R-tree is adopted to organize spatial data, the rectangle cloaking is more efficient.

5 Conclusion and Future Work

There are many research tasks to refine the framework and what we have done is just a small part. In the following days, we will focus on addressing the following issues.

It's observed that dynamic cloaking doesn't work well for continuous queries. Consider the example shown in **Figure 2b**. Suppose the client issues another query at location q' with an uncertain region r'_q . If the system knows the maximum moving speed of the object , v, it can draw an ever-expanding region $(r^e{}_q)$ of the previous region r_q based on v and the elapsed time l since the last query. Thus, one can figure out that the client must reside in the intersection region of $r^e{}_q$ and r'_q , which is smaller than the expected area for privacy requirement if $r'_q \not\subset r^e{}_q$. We shall develop analytical models and conduct simulation experiments to study the optimal cloaking techniques for different scenario.

In the paper by Hu and Lee[2], when handling RNN query, they evaluate the range query and the kNN query with respect to the region boundary separately. A more efficient solution would be to integrate the evaluation of these two queries: the returned results of range query can be used to help pruning the search space of kNN query.

Another issue is how to organize data within the result superset that is returned to the mobile client. When the

Table 1. The comparison of result superset size between rectangle cloaking NN query and circle cloaking NN query

Experiment Parameters	Rectangle-shaped kNN Query	Circle-shaped kNN Query
cloaking area: 4π k-value: 1	209	200
cloaking area: 4π k-value: 2	211	218
cloaking area: 4π k-value: 4	237	233
cloaking area: 4π k-value: 8	261	254
k-value: 4 cloaking area: 1π	81	72
k-value: 4 cloaking area: 2π	131	125
k-value: 4 cloaking area: 4π	237	233
k-value: 4 cloaking area: 8π	428	416

Table 2. The comparison of client side energy consumption between rectangle cloaking NN query and circle cloaking NN query (mJ per query)

Experiment Parameters	Rectangle cloaking kNN Query	Circle cloaking kNN Query
cloaking area: 4π k-value: 1	212.3290452	488.261863
cloaking area: 4π k-value: 2	733.9649435	859.8681606
cloaking area: 4π k-value: 4	1616.630162	2356.447412
cloaking area: 4π k-value: 8	4816.670414	9798.434888
k-value: 4 cloaking area: 1π	550.8215683	664.2931839
k-value: 4 cloaking area: 2π	789.806747	1153.908125
k-value: 4 cloaking area: 4π	1616.630162	2356.447412
k-value: 4 cloaking area: 8π	4816.670414	5097.955781

size of the result superset is large, efficient refinement of actual results is important given limited CPU and battery capacity of mobile devices. Towards this end, the first approach builds some auxiliary index structure (e.g., quad-tree-like structure) over the data, which is returned along with the superset. A disadvantage of this approach is that more data is transferred on the wireless link, and , hence potentially consuming more energy of the client. The second approach arranges the data within the superset according to some well-defined ordering function (e.g., Hilbert Curve or Z-ordering). We shall examine the performance of these different heuristics. Moreover, we can analyze the probability of each object being actual result. For the example shown in **Figure 2a**, object c has a higher probability to be the actual NN than b and d because c is the NN for most points inside r_q . We shall investigate how to compute the probabilities and make use of such probability information to optimize the index structure (e.g., developing an unbalanced tree).

Moreover, the aforementioned optional caching module to improve data access performance will also be examined.

References

[1] Antonin Guttman. R-trees: A dynamic index structure for spatial searching. In SIGMOD Conference, Boston, Massachusetts, pages 47-57, 1984

[2] Haibo Hu and Dik Lun Lee. Range Nearest Neighbor Search. 2005.

[3] Y. Tao, D. Papadias, and Q. Shen. Continuous nearest neighbor search. In VLDB Conference, Hong Kong, China, pages 287-298, 2002.

[4] Yannis Theodoridis. Spatial datasets: an unofficial collection. http://dke.cti.gr/People/ytheod/research/datasets/spatial.html

An Adaptive Controller to Guarantee Proportional Delay Differentiation on Clustered Web Servers

Chan Ka Ho
Department of Computer Science
Hong Kong Baptist University
Kowloon Tong
Hong Kong
khchan@comp.hkbu.edu.hk

Abstract

Nowadays, it is often desirable to isolate performance of different services and different classes of requests from each other. A lot of works have been done on the controller of a standalone server to allocate resources for different classes of requests.

However, there are still problems of building a controller for achieving performance isolation in distributed, clustered servers for distributed resources and requests of different classes. This paper introduces basic networking ideas, controller of a single server, design of the dispatcher of clustered web servers.

Keywords: QoS, Web Servers, Dispatcher, Resources

1 Introduction

Because of the wide spread usage of web services, the number of accesses to popular web site is always increasing. Moreover, the number of service types is increasing. Users can get static page, dynamic page, database information, audio, video, etc.

No matter clients request what kind of services, it is a must for the server to serve it. It is not possible to have a machine that satisfies all the requirements. Therefore, the concept of quality of service is addressed.

During overload, not all requests can be served in a timely manner. Hence, performance-enhancing mechanisms that provide better service to premium customers even during server overload are of major importance. In [6], classical Proportional-Integral (PI) controller is proposed to guarantee the delay ratios between different classes. However, it cannot satisfy results on some performance measures. A more robust controller design will be discussed in later section.

The clustered web servers serve the requests in a different manner compared with the single server. In [16], locality aware distribution is proposed. It is more desirable for different web servers among clustered web servers to serve requests of different services. It is often expected that a fraction of server resources is reserved for a service, independent from the present demand for other services. A mechanism of request distribution by implementing a centralized Cluster Reserves has been proposed in [11]. The distribution of requests is based on objects requested and the current status information of the back-end servers. The details of request distribution among clustered web servers will be discussed later.

We propose to implement a Fuzzy Controller for controlling request distribution of the clustered web servers. The controller gathers information from the back-end nodes (servers) and the types of object requested to choose how the requests are forwarded and to decide which server to serve the client. The brief idea of the controller, dispatcher and the distribution algorithm will be discussed.

The rest of the paper is organized as follows. Section 2 introduces some background information of the networking protocol (HTTP/1.0 and HTTP/1.1), and web server architecture. As our project mainly focused on web servers. In Section 3, we introduce the QoS of single web server. Section 4 discusses the basic idea of the clustered web server QoS including the introduction of packet forwarding, TCP handoff and TCP splicing. In Section 5, we are going to introduce the dispatcher we designed and the algorithm will be employed. Conclusion and future work will be presented in Section 6.

2 Background

Hypertext Transfer Protocol, HTTP [4], is the most important in web server. HTTP is one of the application of TCP/IP protocol suite. Moreover, there are two versions of the protocol, HTTP/1.0 [14] and HTTP/1.1 [13].

In short, HTTP is used between a browser and a web server. A browser sends a GET request to a server, then the server send the requested item as a response. The following sections introduce web server architecture, idea of QoS and both versions of HTTP thoroughly, in sense of application differences, characteristics of the protocol suite and how we can establish the client-server connection. Examples will be shown by some diagrams drawn by log of tcpdump [17].

2.1 HTTP server architectures

Apache [1] is one of the most well-known web servers. In order to discuss HTTP server architectures, Apache server is a good example. The following paragraphs in this section will focus on the Apache architecture.

Apache is a Process-Based Web Server. At the starting phase, lots of processes are forked for serving requests. One of the available, forked processes in the process pool is responsible to serve one request.

2.2 Quality of Service

Quality of Service is the main idea of this paper. According to the point of view of clients, the quality of the service depends on how long it can receive the response after the request is sent (end-to-end delay). The end-to-end delay of a web service includes communication delay on the network, connection delay on the server and the processing delay on the request. It is notices that the server-side delays contribute a significant portion.

In [6], Service Delay Guarantees are introduced. Relative Delay Guarantee and Absolute Delay Guarantee are two types of service delay guarantees. For Absolute Delay Guarantee, a desired absolute delay is assigned to each priority class. The absolute delay guarantee requires that all classes receive satisfactory delays if the server is not overloaded. For Relative Delay Guarantee, a desired relative delay is assigned to each priority class. It is required that the connection delay ratio of classes should obey the ratio of desired relative delays. It is useful only if there are requests from different priority classes. Otherwise, the corresponding relative delays constraint is considered to be satisfied by default.

2.3 HTTP/1.0

As discussed in [7], HTTP operates at the application level, which is stateless, bi-directional transfer. Moreover, it supports caching, intermediaries, and is capable for negotiation. There are two types of HTTP/1.0 message types: *requests and responses* [18].

DNS lookup, TCP connection establishment will take place for the HTTP request. The format of the request is as follow:

request-line
headers
blank line

The request-line is in form of
GET http://www.comp.hkbu.edu.hk/en HTTP/1.0

On the other hand, the format of the response is as follow:

status-line headers blank line body

The status-line is in form of

HTTP-version response-code response-phrase

The first line of the response is the status-line, [14, 18]. 3-digit numeric response code is applied in the categories of *success*, *redirection*, *client error* and *server error*. In the header, there is a numbers of fields, including: Content-Length, Content-Type, Content-Language and so on. The details about the HTTP/1.0 implementation will not be discussed in this section, since it is beyond our purpose of introduction.

HTTP/1.0 uses a TCP connection for each object transfer. The three-way handshaking is performed to connect the client and the server. Fig. 1 is the graphical representation of the connection establishment, serving and the termination of the TCP. After the setup of TCP connection, the GET request will be sent. When the server receives the request, a copy of the requested item will be sent, and then the TCP connection will be closed. At the same time, the client reads data from the TCP connection continuously until it encounters an *end of file* condition. When the client side receives *end of file*, the connection will be terminated.

HTTP/1.0 is simple and easy to implement. However, each request consumes one TCP connection which leads a number of drawbacks affecting both network and the server performance.

Firstly, the interactive action between server and client is a big problem for HTTP/1.0. The data size of interactive activities is usually small. Therefore, the packet size of each TCP segment is usually smaller and very likely to have a size less than Maximum Segment Size (MSS) [17]. Due to interaction, large amount of small packets are transmitted between server and client. It leads high overhead. Furthermore, large amount of small size packets leads the network congestion which lowered the performance. In [8], Nagal algorithm is introduced to get rid of the problem, but it turns out to the result of slow reaction and unacceptable response time for the client.

Besides, the TCP characteristics, slow start [19], also introduces performance problem. As slow start adds additional Round Trip Time (RTT) [17] to the connection, the response time of the servers will be affected.

Lastly, it is possible for the control blocks to make server busy. Control block is a special feature of TCP for Transactions(T/TCP). This is caused by the HTTP server closed the connection. Then, there will be a TCP TIME_WAIT delay on server [18]. Server receives large amount of control blocks which degrade the network performance and the service provided by the server.

2.4 HTTP/1.1

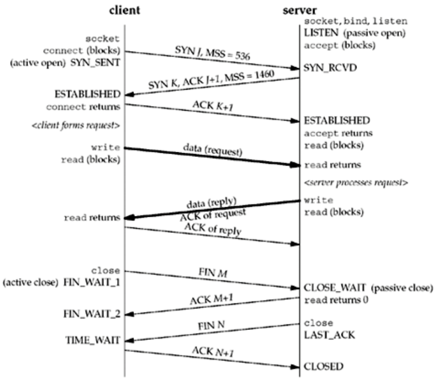


Fig 1. Server and client interaction using HTTP/1.1

HTTP/1.1 is a further developed version of HTTP based on HTTP/1.0. They share similar properties, including request, response and header format. However, HTTP/1.1 introduced persistent connections [13] to solve the problem of HTTP/1.0. Persistent connection allows multiple requests from the same client to reuse an opened TCP connection, which makes it unnecessary to establish and terminate new TCP connection for every request. It is much more efficient with less overhead.

The chief advantage of persistent connections is reducing overhead [7], because of less TCP connection establishment and termination. After the TCP connection is opened, the connection will be kept for certain amount of time, which can lower response latency, overhead, buffers usage and CPU time consumption. The performance can be further optimized by pipelining request. It is especially attractive in situations of getting giant files, for example, multiple images, videos for the requested page.

On the other hand, persistent connections bring a peculiar server bottleneck. When all processes are actively processing requests or waiting for another request on a persistent connection, all new incoming requests must wait until a process become available. Therefore, determine when to terminate the connection is extremely essential. In current practice, HTTP/1.1 normally sends a length followed by the item [7].

3 QoS of Single Server

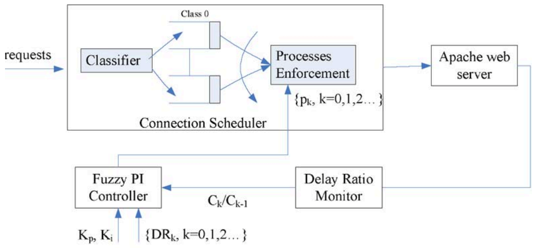


Fig 2. The system architecture of fuzzy PI controller

In [6], Lu proposed a classical PI controller to guarantee the delay ratios, as discussed in Relative Delay Guarantee, between different classes. Since no accurate model for the nonlinear web server, the server is modeled as a second order system and the parameter is decided by system identification. However, PI controller is based on accurate linear model. Therefore, it cannot guarantee satisfactory results on some performance measurement, such as oscillating effect and settling time.

A fuzzy controller [20] is proposed for proportional delay differentiation. It is independent of any accurate models, so it is a good choice of feedback controller. But the main drawback of the fuzzy controller is large amount of parameters to be tuned which is very difficult to make initial approximate adjustment [12]. Furthermore, it also depends on the quality of the expert knowledge.

In previous work, we have proposed a fuzzy PI controller. The controller shares the advantage of classic PI controller and the fuzzy controller. Comparing with the classic PI controller, fuzzy PI controller is more robust. In heavy load situation, the settling time is shortened, less and lower range of oscillation are shown. Besides, in the steady state, it performs as well as the classic PI controller. Comparing with the fuzzy controller, the fuzzy set defined by the fuzzy PI controller is much more simpler. Therefore, in the phase of initial state of parameter, it is more convenient. Except the variable setting, the problem of oscillation is less when compare with the fuzzy controller as well.

The system architecture includes Apache web server, connection scheduler, Fuzzy PI controller and the Delay Ratio Monitor as shown in Fig 2.

The connection scheduler listens to the port and accepts every incoming connection requests; then it classifies requests into different classes. Moreover, it maintains a FIFO queue and a process counter for each class. The processes are the server processes of the Apache web server which serve requests from clients. Besides, there is a delay ratio monitor, which is used to monitor the delay ratio periodically. And the delay ratio will become one of the input of the fuzzy PI controller for the next period of time. The last module is the fuzzy PI controller which control the connection scheduler to distribute request for different time slots.

A series of experiments have been done for the fuzzy PI controller. First of all, the classic PI controller is implemented to find out the value of parameters by using the White Noise Input [6]. Then, the values are used to setup the fuzzy PI controller. At the same time, the controller results (same as classic PI controller) are gathered, which is used as a control experiment. The results of fuzzy PI controller are compared with the control results. A better performance can be clearly shown by plotting the graphs for comparison.

4 QoS of Clustered Web Server

The clustered web servers are different from the single web server. The classical PI controller, fuzzy controller and fuzzy PI controller cannot be implemented directly, as we cannot monitor the performance accurately for the whole cluster. And the concept of performance measurement of the whole clustered web server is not clearly defined.

For cluster controlling system, load balancing and the locality [16] are essential features. The main advantages of the system are enhancing performance, increasing secondary storage scalability and the specialized back-end node. In order to balance the load of web servers within the cluster and to provide client-blinded service [15], a front-end machine is required. When request arrived at the front-end, some decisions are made to determine which back-end node should serve it.

4.1 Packet Forwarding

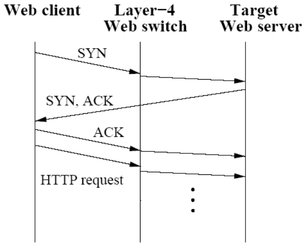


Fig 3. Layer-4 routing operations

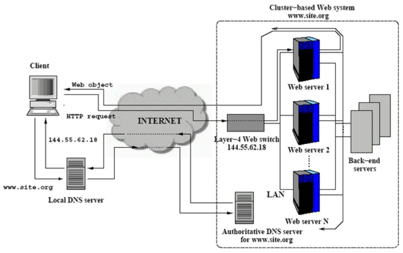


Fig 4. Packet forwarding architecture

The packet forwarding mechanism implements a layer-4 switch that the packet is forward on top of the transport layer. The operation of layer-4 routing is shown in Fig 3.

Packet forwarding is an approach assuming all web switch and servers nodes are on the same subnet and all the machines share the same IP address. The IP address is shared among all machines. Therefore, Address Resolution Protocol (ARP) must be disabled to avoid collision.

As discussed in [15], all the machines have the same IP address, the inbound packets reach the web switch because the ARP is disabled. Hence, the web switch forward the packets to the target server by rewriting the physical (MAC) address of the destination server and retransmitting the frame on the network. Rewriting the MAC address is to rewrite the Data Link layer without modifying the TCP/IP header. The target server receives the packets with the source address of the client and the destination address of itself, so the server can response to the client directly. The packet forwarding architecture is shown in Fig 4.

4.2 TCP Splicing

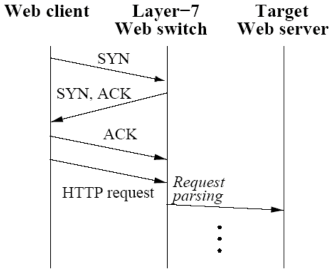


Fig 5. Layer-7 routing operations

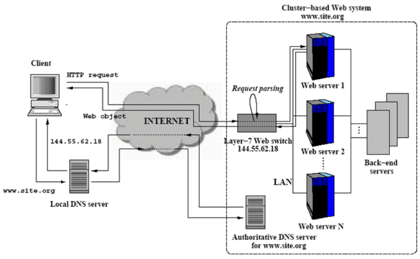


Fig 6. TCP splicing architecture

Compare with packet forwarding, TCP splicing [15, 5, 9] is a layer-7 switch. The operation of layer-7 routing is shown as Fig 5. The packet forwarding is occurred in the network level between network interface driver and the TCP/IP stack. Furthermore, it is directly controlled by the operating system.

Once TCP connection is established between web switch and the client, one of the persistent TCP connection between web switch and the back-end server node will be chosen. And the two connections will be spliced together. The packets are forwarded from web switch to the target server node. The web switch changes TCP and IP header of the packets for the packets going to be forwarded. Since the IP and TCP header are modified, the source of the request for the back-end server node is the web switch. Therefore, it is a must for the response to pass through the web switch for the changing of IP and TCP header again. As we can see that all packets from and to the clusters pass through the web switch, the web switch will become the performance bottleneck. The TCP splicing architecture is shown in Fig 6.

4.3 TCP Handoff

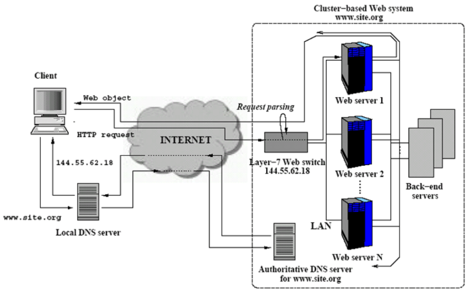


Fig 7. TCP Handoff architecture

From the previous section, the performance bottleneck of TCP splicing is the web switch as all packets from and to the clusters need to passes through it. TCP handoff mechanism solves the problem of the bottleneck by using a one-way architecture [15].

After the TCP connection is established between web switch and the client, it is handed off to a target server. TCP handoff serializes the state of the existing client TCP connection and the web switch, and then instantiates a TCP connection with the chosen back-end servers. The mechanism remains transparent to client, since the response from the back-end node appeared from the web switch and the acknowledgement from the client is transmitted to the web switch and forwarded to the back-end node. TCP handoff works on top of TCP and runs on the web switch and the servers, so changing of the operating systems among web switch and back-end nodes are required. Using a loadable kernel modules to the UNIX systems is the usual practice for the implementation of the TCP Handoff. The TCP handoff architecture is shown in Fig 7.

Using the TCP handoff, the problem of performance bottleneck problem is eased. Moreover, the scalability of the clustered web servers can be enhanced [16].

5 Dispatcher

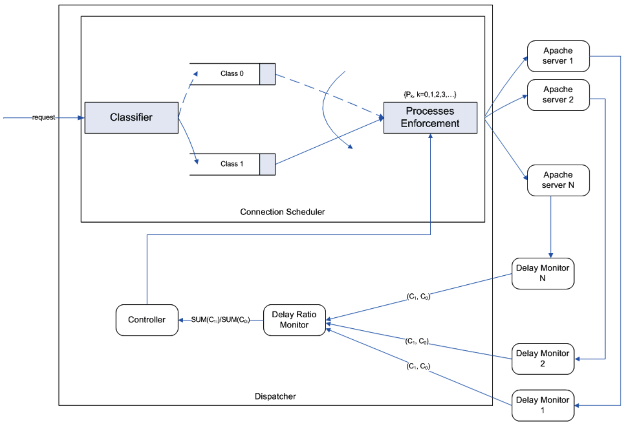


Fig 8. Dispatching mechanism of the web clusters

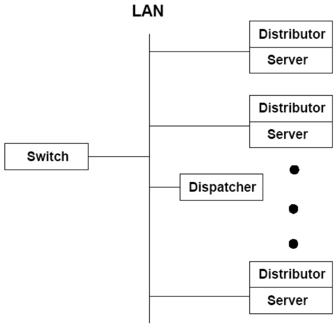


Fig 9. network nature of the clusters

In [16, 10, 11], Aron raised the idea of Locality-Aware Request Distribution (LARD) for the dispatching of the requests to the clustered web servers. The idea of this dispatching algorithm is considering the request of client and the status among back-end servers. However, the clustered web servers consider all the requests at the same priority. As more and more web sites would like to provide better service for premium class users, the LARD dispatching algorithm is not very suitable. Therefore, adaptive controller for the clustered web servers is designed.

However, large amount of difficulties, (e.g., where to place the monitor, how to dispatch the request to the back-end servers), present in the design of controller for clustered web servers.

The decision of how to monitor the server performance is very important for an adaptive controller. The adaptive controller is a negative feedback system, so the output of the previous time period is an important input. We design to adopt TCP handoff for distribution of the requests. However, the total output of the clustered web servers is very hard to determine since TCP handoff is a one-way architecture of layer-7 application. Therefore, we design to implement a monitor for each back-end server. Then, the output of each server will be collected by the dispatcher and the total output will be calculated. According to the output, the client requests and the output will become the input of the controller for the next time slot. According to fuzzy logic and the inputs, the dispatcher (the adaptive controller) decides which back-end node (web server) should serve for specific requests. Design of the dispatcher (the adaptive controller) is shown in Fig 8.

As discussed in TCP handoff, a loadable module for the operating systems of all machines is required. All the machines in the clustered web servers are running FreeBSD [3] version 5.4. The back-end nodes are two same configuration server machines. Moreover, the dispatcher is different from them. On the other hand, the client machines will be operated on top of the Fedora Core 4 [2]. There are three lower-end machines acting as the clients machine to do the trace based experiment for our web clusters. All the server machines and the dispatcher are connected by a high-speed(10/100MB) switch, the connection is as shown in Fig 9. The clients generate requests to the cluster via the Internet to the switch, and the switch re-direct the packets to the back-end nodes act as an distributors. Finally, the dispatcher decides which server is responsible to the request. The architecture is similar to the clusters in [10, 11].

The framework is still in the construction phase. No experiments have been done on top of our clustered web servers.

6 Conclusions and Future Works

With the wide spread usage of the web services and the number of access to popular web sites is ever increasing, single server for the serving purpose become less feasible. The use of clustered web servers are more suitable for the growth trend. According to the investigation of the network usage, we can see that content aware and load balancing are extremely essential for the clustered web servers. Furthermore, the proposed controller works well in single web server, it is foreseeable that the performance of clustered web servers will be enhanced after applying the controller.

There are lots of improvement area for the dispatcher. We proposed to implement the fuzzy controller in the dispatcher of the clustered web servers. However, large amount of parameters must be tuned which is closely related to the expert knowledge. It may be not very good for the implementation of the clustered systems, which may lead to design new dispatching controller. Besides, the oscillating effect between the steady point may occur. In order to get rid of the problem, decoupling techniques are crucial for the dispatcher.

Locality aware and the Weighted Round Robin techniques are used to distributes the requests. For the back-end node, the hardware configuration may be different. Therefore, Weighted Round Robin may not be suitable for this circumstance. The average delay of each server can be accounted by the decision making procedure.

All the above listing may be the future works for fine tuning the dispatcher of clustered web servers.

References

[1] Apache Project.
http://www.apache.org/.

[2] Fedora Core 4 – Red Hat Linux.
http://www.redhat.com/en_us/USA/fedora/.

[3] FreeBSD 5.4.
http://www.freebsd.org/.

[4] HTTP - Hypertext Transfer Protocol.
http://www.w3.org/Protocols/.

[5] A. Cohen, S. Rangarajan, and H. Slye. On the Performance of TCP Splicing for URL-Aware Redirection. *Procedings of the 2nd USENIX Symposium on Internet Technologies and Systems*, 1999.

[6] C. Lu, Y. Lu, T. F. Abdelzaher, J. A. Stankovic, and S. H. Son. Feedback Control Architecture and Design Methodology for Service Delay Guarantees in Web Servers. *IEEE Transactions on Parallel and Distributed Systems*, 2005.

[7] Douglas E. Comer. *Internetworking with TCP/IP*. Englewood Cliffs, N.J. : Prentice-Hall, 1993.

[8] J. Nagle. Congestion Control in IP/TCP Internetworks. *IETF RFC 896*, 1984.

[9] K. Fall and J. Pasquale. Exploiting In-Kernel Data Paths to Improve I/O Throughput and CPU Availability. *Proceedings of the Winter 1993 USENIX Conference*, 1993.

[10] M. Aron, D. Sanders, P. Druschel, and W. Zwaenepoel. Scalable Content-aware Request Distribution in Cluster-based Network Servers. *Proceeding of the 2000 Annual Usenix Technical Conference*, 2000.

[11] M. Aron, P. Druschel, and W. Zwaenepoel. Cluster Reserves: A Mechanism for Resource Management in Cluster-based Network Servers. *Proceeding of the ACM Sigmetrics 2000 International Conference on Measurement and Modeling of Computer Systems*, 2000.

[12] P. Pivonka. Comparative Analysis of Fuzzy PI/PD/PID Controller Based on Classical PID Controller Approach. *Proceeding of 2002 IEEEWorld Congress on Computational Intelligence*, 2002.

[13] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol–HTTP/1.1. *IETF RFC 2616*, 1999.

[14] T. Berners-Lee, R.Fielding, H. Frystyk. Hypertext Transfer Protocol–HTTP/1.0. *IETF RFC 1945*, 1996.

[15] M. C. V. Cardellini, E. Casalicchio and P. S. Yu. The state of the art in locally distributed web-server systems. 2001.

[16] V. S. Pai, M. Aron, G. Banga, M. Svendsen, P. Druschel, W. Zwaenepoel, and E. Nahum. Locality-Aware Request Distribution in Cluster-based Network Servers. *Proceeding of the ACM 8th International Conference on Architectural Support for Programming Languages and Operating Systems*, 1998.

[17] W. Richard Stevens. *TCP/IP Illustrated, Volume1: The protocols*. Addison-Wesley, 1994.

[18] W. Richard Stevens. *TCP/IP Illustrated, Volume3: TCP for Transactions, HTTP, NNTP, and the UNIX Domain Protocols*. Addison-Wesley, 1994.

[19] W. Stevens. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. *IETF RFC 2001*, 1997.

[20] Yaya Wei, Chuang Lin, Xiaowen Chu, T. Voigt, F. Ren. Fuzzy Control for Guaranteeing Absolute Delays in Web Servers. *IJHPCN*, 2005.

Studies on Location Estimation in Library Environments

Wilson M. Yeung

Abstract

In this paper, we review the current positioning algorithms. Through the revision, we discuss the limitations of the current algorithms in library environment. A preparatory experiment was conducted and the result shows that the signal attenuation circumstance due to the metal bookshelves in library is similar to the situation in office environment, in where the signal is distorted by wall.

1 Introduction

Wireless communication technology has gratefully enhanced in recent years. People can use mobile devices to communicate, retrieve information from Internet, and perform location estimation. Based on wireless location estimation technology, some location-based services such as library book-finding system and electronic tour guide become available. In fact, there are some limitations, e.g. Body effect [1] [6], Trailing Effect [6], on location estimation in indoor environment. Positioning in library environment is difficult, as the radio signal is blocked and reflected by bookshelves and books [6]. Providing location-aware service in library require high accuracy of estimation, therefore enhancement on positioning technique is required.

In this paper, we are going to review the mainstream location estimation algorithms and investigate the problem of positioning in library environment. We also present the preliminary research works and the experimental result.

2 Positioning Algorithms

2.1 Triangulation

Triangulation algorithm use signal readings to set up a propagation model, and then gathers the intersection points among these models. The smallest triangle among these intersecting points is then formed. The centroid of the smallest triangle is the result. If the number of intersection points is not enough to form a triangle, it takes the average of intersection points as the result. The propagation model can be empirical either theoretical. Some research works [1][6] shows that the performance of triangulation using empirical propagation model is, about 11.8%[6], better than using radio (theoretical) propagation model.

2.2 Weighted Center of Gravity (CG)

Weighted Center of Gravity is based on competition of signal strength between neighboring Access Points (Antenna). It's defined as:

$$\begin{cases} \hat{x} = \dfrac{x_0/rss_0^\alpha + x_1/rss_1^\alpha + \ldots + x_n/rss_n^\alpha}{1/rss_0^\alpha + 1/rss_1^\alpha + \ldots + 1/rss_n^\alpha} & \text{(1a)} \\ \hat{y} = \dfrac{y_0/rss_0^\alpha + y_1/rss_1^\alpha + \ldots + y_n/rss_n^\alpha}{1/rss_0^\alpha + 1/rss_1^\alpha + \ldots + 1/rss_n^\alpha} & \text{(1b)} \end{cases}$$

Where ($\hat{x}$, $\hat{y}$) are the coordinate of the estimated location; the $(x_1, y_1)\ldots(x_n, y_n)$ and $rss_1\ldots rss_n$ represent the actual location and signal strength of corresponding Access Point; and α is a environment dependant variable [3] [6], which should be calculated based on the signal sample data in the environment.

2.3 Smallest M-vertex Polygon (SMP)

Similar to Triangulation, SMP looks up the database to find out the locations with the closest signal reading, i.e. the shortest Euclidean distance in signal space [1], for each antenna. Suppose M access points exist in the environment, and each access point promotes at least one possible location. These locations are treated as candidates, and used to from an M-vertex polygon among all combinations. The centroid of the smallest polygon is the estimation result.

2.4 Fingerprinting

The basic idea of fingerprint is finding out the nearest neighbor(s) in signal space (NNSS) [1]. A signal strength tuple (snapshot), is a group of signal readings taken from multiple access points nearby at a particular moment. The principle of fingerprinting is looking up a snapshot in the local database which is the most similar to the current snapshot. Euclidean displacement in signal space will be used to determine similarity [6]. The marker point which provides the most similar snapshot is the result location. However, the result would be discrete, which means the result location must be one of the sampling locations. Therefore an interpolated [6] design has been proposed to generate a non-discrete result with top K locations according to their probability.

3 Problems Definition

3.1 Alogorithm

Both Triangulation and Weighted-CG algorithm are based on antenna-centric approach [6] which focuses on the antenna and propagation model. However, the signal propagation model is distorted by the bookshelves in the library due to blockage and reflection. Our experiment result shows the circumstances of signal propagation, we discuss it in Section 4. Even we can use empirical model, which is set up by the location data, as a replacement, the enhanced performance is still not good enough, comparing [6] to Fingerprinting. However, antenna-centric approach algorithm is worth to pay effort on it, because the cost of re-calibration due to the environmental change is smaller than location-centric approach algorithms. When environment is changed, we can change the model parameters to adapt.

Fingerprinting and SMP are based on location-centric approach [6] which is based on sampling location and signal strength samples. Although fingerprinting obtains the best result in the library environment [6], the location-centric approach based algorithms are sensitive to the environment changes. In other words, if the library environment is changed, say by redecoration or reallocating the bookshelves, re-calibration [4] is required, and the cost of re-calibration can be considerably high.

3.2 Sliding Window

Although the library environment is a relatively static, sometimes the fluctuation of radio signal can be considerably large because of body effect or movement. The general solution is to use fixed-size sliding window to absorb, by averaging the value usually, the large fluctuation, hence to attain a smooth estimation result. However, if the size of sliding windows is too large, although we can obtain a smooth estimation result, but the response may be seriously lagged. On the other hand, if the size of the sliding window is too small, it cannot absorb the large fluctuation. Currently, the window size is fixed and environment dependant.

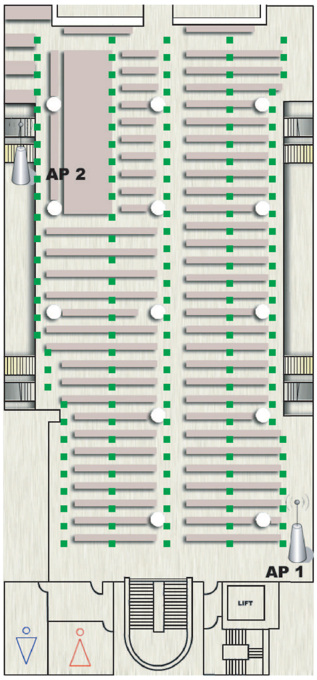


Figure 1. Floor plan

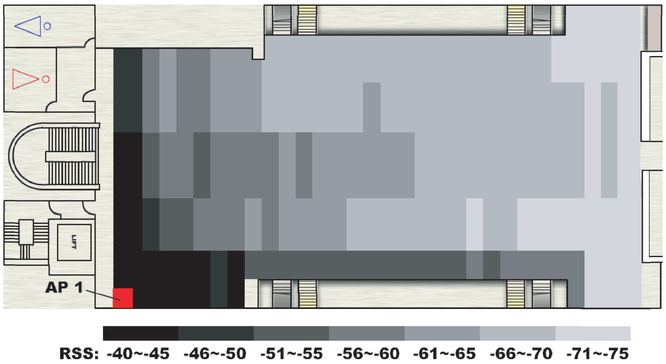

LIFT
AP 1
RSS: -40~-45 -46~-50 -51~-55 -56~-60 -61~-65 -66~-70 -71~-75

Figure 2. Signal Propagation of Access Point 1

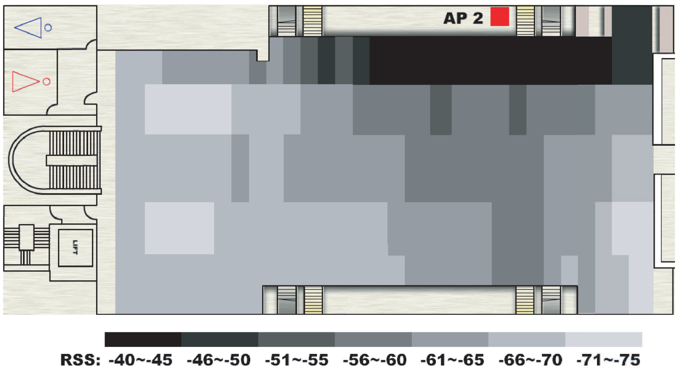

AP 2
LIFT
RSS: -40~-45 -46~-50 -51~-55 -56~-60 -61~-65 -66~-70 -71~-75

Figure 3. Signal Propagation of Access Point 2

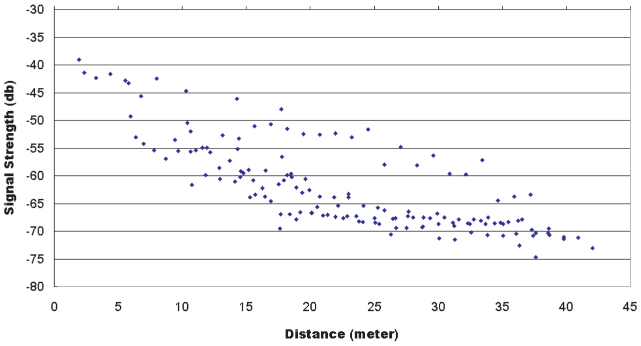


Figure 4. Signal Strength Distribution of Access Point 1

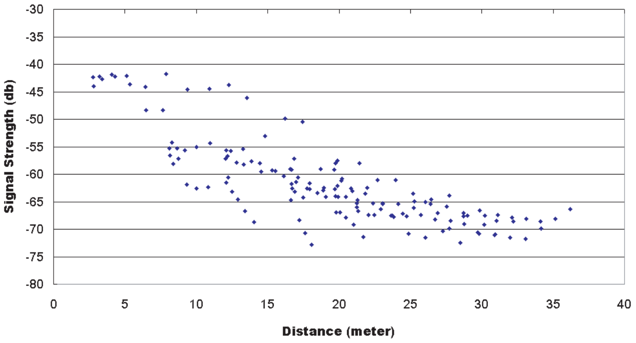


Figure 5. Signal Strength Distribution of Access Point 2

4 Preparatory Experimental Studies

In order to investigate the real signal propagation circumstances in the library environment, we obtained a collection of sample data from a floor of library. Our test site is the fifth floor of a seven-storey library building. The total area of the floor is about $773.78m^2$. In the test site, signals from 6 access points can be detected. Two of them are installed on the fifth floor while two of them are installed on forth floor. The remaining two access point are installed on the half floor between fifth and forth floor. Our experiment focused on the access points which installed on the selected floor, i.e. fifth floor, only. We obtained the relative signal strength (RSS) information in each of the 4 directions at 147 distinct locations on the floor (see figure 1). For each direction, 50 samples signal strength samples are received, so we obtained 29400 samples in total. The average signal reading from both access points at 147 sampling locations is individually plotted. (See figure 2 & 3.) Besides that, the relationship between transmitter-receiver separation distance and signal strength of both access points are also plotted. (See figure 4 & 5.)

Figure 2 & 3 shows the average signal strength at different location of fifth floor. Generally, the locations near the access point obtain stronger signal while weak signal is received at the location far from the access point. However, some locations obtain weaker signal strength than the locations nearby. All of these locations are between the bookshelves. Furthermore, the locations where had line of sight to access point obtained stronger signal than the locations between the bookshelves.

Figure 4 and Figure 5 show the signal strength against the transmitter-receiver separation distance. We observed a common phenomenon in both figures. The locations where had approximately same distance from the access point obtained different signal strength, while the difference can be considerably large. For example, the signal strength received at two locations, which had approximately 24.5 m from access point 1, are approximately 15 db apart. This phenomenon also occurred on access point 2. (See figure 5). The location with stronger signal is, again, the locations had line of sight to access point.

In fact, our experimental result is similar to a previous research [2] which had proposed a location tracking system in office environment. The main obstacle in their test site is wall and their result shows that the signal at a location where between the walls is attenuated. They had proposed Wall Attenuation Factor (WAF) model which enhanced the performance of triangulation in indoor environment.

5 Future work

For the future work, we are going to investigate Weighted-CG with an independent α values [6] for each of the access point. It is because the signal and empirical propagation model of each access point may not be similar to the other one. Based on the experimental result shows that the signal attenuation due to bookshelves is similar as wall, we are going to investigate Wall Attenuation Factor model [2] in library environment. Besides that, as the signals from the access point located at other floor is received, we will work on Floor Attenuation Factor propagation model [5] too. We will use these models to set up a better empirical model for both Triangulation and Weighted-CG. Furthermore, the adaptive sliding window which can dynamically choose the size should enhances the performance of positioning, and we will pay effort on it too.

References

[1] P. Bahl, A. Balachandran, and V. Padmanabhan. Enhancements to the RADAR user location and tracking system. Technical report, Microsoft Corporation, February 2000.

[2] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM (2)*, pages 775–784, 2000.

[3] K. K.-H. Kan, S. K.-C. Chan, and J. K.-Y. Ng. A dual-channel location estimation system for providing location services based on the gps and gsm networks. In *Proceedings of the 17th International Conference on Advanced Information Networking and Applications*, pages 7–12. IEEE Computer Society, March 2003.

[4] H. Laitinen, S. Ahonen, S. Kyriazakos, J. Laheenmaki, R. Menolascino, and S. Parkkila. Cellular location technology. Technical report, November 2001.

[5] S. Y. Seidel and T. S. Rapport. 914 mhz path loss prediction model for indoor wireless communications in multi-floored buildings. *IEEE Trans. on Antennas & Propagation*, February 1992.

[6] W. H. Wong, J. K. Ng, and W. M. Yeung. Wireless lan positioning with mobile devices in a library environment. In *Proceedings of ICDCS-MDC 2005 Workshop, Columbus, Ohio, USA.*, pages 633–636, June 2005.