Web Intelligence (WI)
Some Research Challenges
[IJCAI '03 Invited Talk]

Jiming Liu
Web Intelligence Consortium (WIC) &
Department of Computer Science
Hong Kong Baptist University
jim@comp.hkbu.edu.hk
IJCAI '03 8/15/2003

Acknowledgements and Credits

❖ Profs. Ning Zhong, Yiyu Yao,
  Edward A. Feigenbaum, Setsuo Ohsuga,
  Benjamin Wah, Philip Yu, Lotfi A. Zadeh, and
  Xindong Wu, etc.
❖ WIC Technical Committee
❖ WIC Research Centers in Australia, Canada, India,
  Japan, and Spain, among others
❖ Students and Post-doc/Visitors at
  ❖ A.M.D. Lab (HKBU)
  ❖ AAMAS/AOC Group (HKBU)
The preparation has benefited from a peaceful break in Grand Canyon, on the way to IJCAI'03 …

Outline

- Background
- Challenges
  - Semantic Web + Planning
  - Distributed Agents + Coordination
  - Social Networks + Self-Organization
- Issues and Directions
What is Web Intelligence (WI)?

WI explores the fundamental roles as well as practical impacts of

- Artificial Intelligence (AI) (e.g., knowledge representation, planning, knowledge discovery, agents, and social intelligence) and

- Advanced Information Technology (IT) (e.g., wireless networks, ubiquitous devices, social networks, and data/knowledge grids)

on the next generation of Web-empowered systems, environments, and activities

Four Levels of WI Support

Level-4
Application-level ubiquitous computing and social intelligence utilities

Level-3
Knowledge-level information processing and management tools

Level-2
Interface-level multi-media presentation standards

Level-1
Internet-level communication, infrastructure, and security protocols

2003-8-29
**Semantic Web**

- **Ontology:** Define meanings and relationships of vocabularies (in terms of classes and properties)
- **Semantic Web:** Add semantic meanings to Web information based on pre-defined ontology
- **Benefits:** Enable better human-computer communications as well as software agents access
- **Example:** DARPA Agent Markup Language (DAML)
Planning

- Planning example: STRIPS
- States: conjunctions of ground literals
  
  \[
  \text{At(Home)} \land \text{Sell(Supermarket, Banana)} \land \text{Sell(Supermarket, Milk)} \land \text{Sell(Hardware Store, Drill)}
  \]

- Goals: conjunctions of literals (possibly containing variables)
  
  \[
  \text{Have(Drill)} \land \text{Have(Milk)} \land \text{Have(Banana)} \land \text{At(Home)}
  \]

  \[
  \text{At(?x)} \land \text{Sells(?x, Milk)}
  \]
Operators in STRIPS

Action: Buy(?x)
Preconditions: At(?store) \land Sell(?store, ?x)
Effects:
- additions: Have(?x)
- deletions:

Action: Go(?there)
Preconditions: At(?here)
Effects:
- additions: At(?there)
- deletions: At(?here)

POP Algorithm

- Ordering constraint
  \[ S_i < S_j \]
  - Step \( S_i \) occurs before step \( S_j \)
- Casual link
  \[ S_i \rightarrow S_j \]
  - \( S_i \) achieves the precondition \( c \) of \( S_j \)
- Open condition
  - Precondition that is not causally linked

Source: Russell & Norvig
Semantic Web + Planning
(with Kelvin Tsang)

- The planning agent is **goal-directed**
- A **plan** is a sequence of **actions** to achieve the goals, given an initial state
- A **logic-based language** is used to describe the problem
- General planner is based on **Partial Order Planning (POP)**, coupled with **heuristic search**
- Meanings and relationships of the words in the documents are specified in **ontologies**
- **Planning information** is interpreted from semantic Web documents

```xml
<?xml version="1.0"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:daml="http://www.daml.org/2001/03/daml+ont#">
<daml:Ontology daml:versionInfo="1.0">
<daml:comment>An example for logic ontology</daml:comment>
</daml:Ontology>
<-- Term -->
<rdfs:Class rdf:ID="Term"/>
<-- Variable -->
<rdfs:Class rdf:ID="Variable">
<daml:subClassOf rdf:resource="# Term"/>
</rdfs:Class>
<-- Constant -->
<rdfs:Class rdf:ID="Constant">
<daml:subClassOf rdf:resource="# Term"/>
</rdfs:Class>
<daml:DatatypeProperty rdf:ID="value">
<daml:domain rdf:resource="#Constant"/>
<daml:range rdf:resource="http://www.w3.org/2000/10/XMLSchema#string"/>
</daml:DatatypeProperty>
<-- Predicate -->
<rdfs:Class rdf:ID="Predicate">
<daml:subClassOf rdf:resource="#Term"/>
</rdfs:Class>
<daml:ObjectProperty rdf:ID="arguments">
<daml:domain rdf:resource="#Predicate"/>
<daml:range rdf:resource="#Term"/>
</daml:ObjectProperty>
<-- Common Property -->
<daml:DatatypeProperty rdf:ID="name">
<daml:domain rdf:resource="#Variable"/>
<daml:range rdf:resource="http://www.w3.org/2000/10/XMLSchema#string"/>
</daml:DatatypeProperty>
</rdf:RDF>
```
### Ontology Modeling

```xml
<!-- Planning domain -->
<daml:Class rdf:ID="Domain"/>
<daml:ObjectProperty rdf:ID="operators">
  <daml:domain rdf:resource="#Domain"/>
  <daml:range rdf:resource="#Operator"/>
</daml:ObjectProperty>
<daml:ObjectProperty rdf:ID="initial">
  <daml:domain rdf:resource="#Domain"/>
  <daml:range rdf:resource="logic:Predicate"/>
</daml:ObjectProperty>
<daml:ObjectProperty rdf:ID="goal">
  <daml:domain rdf:resource="#Domain"/>
  <daml:range rdf:resource="logic:Predicate"/>
</daml:ObjectProperty>
```

### Instance Files

- DAML instance files: To encode planning information in Semantic Web documents that use the vocabularies in
  - logic ontology (logic-onto.daml)
  - planning ontology (plan-onto.daml)

- shopping-problem.daml

**Action:** Buy(?x)
**Preconditions:** At(?store) ^ Sell(?store, ?x)
**Effects:**
  - additions: Have(?x)
  - deletions:
Examples of Operators

```xml
<!-- Buy(?x) -->
<plan:Operator>
<!-- rdf:comment -->Buy the ?there</rdf:comment>
<plan:action>
  <logic:Predicate>
    <logic:name>
      <xsd:string rdf:value="Buy"/>
    </logic:name>
    <logic:arguments rdf::parseType="daml:collection">
      <logic:Variable>
        <logic:name>
          <xsd:string rdf:value="?x"/>
        </logic:name>
      </logic:Variable>
    </logic:arguments>
  </logic:Predicate>
</plan:action>

<!-- Preconditions -->
<plan:preconditions rdf::parseType="daml:collection">
  <!-- At(?store) -->
  <logic:Predicate>
    <logic:name>
      <xsd:string rdf:value="At"/>
    </logic:name>
    <logic:arguments rdf::parseType="daml:collection">
      <logic:Variable>
        <logic:name>
          <xsd:string rdf:value="?store"/>
        </logic:name>
      </logic:Variable>
    </logic:arguments>
  </logic:Predicate>
</plan:preconditions>

<!-- Sell(?store, ?x) -->
<logic:Predicate>
  <logic:name>
    <xsd:string rdf:value="Sell"/>
  </logic:name>
  <logic:arguments rdf::parseType="daml:collection">
    <logic:Variable>
      <logic:name>
        <xsd:string rdf:value="?store"/>
      </logic:name>
    </logic:Variable>
    <logic:Variable>
      <logic:name>
        <xsd:string rdf:value="?x"/>
      </logic:name>
    </logic:Variable>
  </logic:arguments>
</logic:Predicate>
</plan:preconditions>
```
OntoPlan

Goals & constraints

Planning Agent  
Domain Viewer  
Plan Viewer

DAML file

Ontology files

Semantic Web

OntoPlan: Domain & Plan Viewers

Domain & Plan Viewer

Semantic Web Planning Domain Viewer

Planning Domain (PDL):

OntoPlan OntoPlan

2003-8-29 21

2003-8-29 22
Application

A consultation system for computer configurations
1. To suggest compatible hardware components
2. To meet user preference

Hardware Ontology: hardware-onto.daml
Distributed product information (from manufacturers) are located in distributed instance files, as well as in local sources

Example of Operators

Socket of the motherboard fits the CPU
FSB of the motherboard is compatible with the bus speed of the CPU

Action: BuyCPU(?x)
Preconditions: CPU(?x), HaveMotherboard(?y), platform(?x, ?socket), cpuPlatform(?y, ?socket), systemBusSpeed(?x, ?fsb), maxFSB(?y, ?fsb)
Effects: add: HaveCPU(?x)
del: 
WI Challenge #2

Distributed Agents + Coordination
Example: Distributed Scheduling

Given:
- A group of people, each of whom has specific available time slots
- A set of constraints among people (e.g., persons A and B will not be present at the same time)

Find:
- An available time slot when all constraints are satisfied

Satisfiability Problems (SAT)

<table>
<thead>
<tr>
<th>Person and time slots</th>
<th>Variable and its domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraint</td>
<td>Clause</td>
</tr>
<tr>
<td>Persons involved in a constraint</td>
<td>Literals in a clause</td>
</tr>
<tr>
<td>Meeting scheduling</td>
<td>SAT</td>
</tr>
</tbody>
</table>

CNF: 
\[ F = \bigwedge_{i=1}^{m} C_i \text{ where } C_i = \bigvee_{j=1}^{K} L_j \]

? = 'and', V = 'or', C_i = clause, L_j = literal (variable or its negation)
Local Search
(Selman, Levesque, and Mitchell, 1992; Gu, 1992)

- Take the solution space (i.e., the Cartesian product of all variable domains) as a search space, and search it based on a certain rule
  - Randomly select one position as the start point to search
  - At each step, move to a neighboring position according to the rule (i.e., heuristic)

<table>
<thead>
<tr>
<th>Method</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSAT</td>
<td>1992</td>
</tr>
<tr>
<td>WalkSAT</td>
<td>1994</td>
</tr>
<tr>
<td>GWSAT</td>
<td>1994</td>
</tr>
<tr>
<td>WalkSAT/Tabu</td>
<td>1997</td>
</tr>
<tr>
<td>GWSAT/Tabu</td>
<td>1997</td>
</tr>
<tr>
<td>Novelty</td>
<td>1997</td>
</tr>
<tr>
<td>R-Novelty</td>
<td>1997</td>
</tr>
<tr>
<td>Novelty+</td>
<td>1999</td>
</tr>
<tr>
<td>R-Novelty+</td>
<td>1999</td>
</tr>
<tr>
<td>SDF</td>
<td>2001</td>
</tr>
<tr>
<td>UnitWalk</td>
<td>2002</td>
</tr>
</tbody>
</table>

If

- people are distributed in different places
- the time slot information is NOT centralized

Then

- centralized local search methods become ineffective

How to solve it?
Multi-Agent SAT (MASSAT) 
(with X. Jin)

- Use multiple computational agents
- Decompose the search space into several sub-spaces
- Each agent decides how to locally search a sub-space (i.e., its environment)
- Through local interactions between agents and their environments, the agents coordinately find a global solution to the given problem

Formulation

- Divide variables into u groups
- Agent $a_i$: variable group $V_i = \{v_{i1}, \ldots, v_{ik}\}$
- Environment $e_i$ of $a_i$: Cartesian Product of the variable group, $D_i = D_{i1} \times \ldots \times D_{ik}$
- Agent’s position $e_{ij}$: $i^{th}$ value combination in $D_i = D_{i1} \times \ldots \times D_{ik}$
- The position combination of all agents $\{e_{1j}, \ldots, e_{ij}, \ldots\}$: a possible solution
- Basic move strategies of agent $a_i$:
  - $\Psi_i: D_{i1} \times \ldots \times D_{ij} \times \ldots \times D_i$ → $D_i$
  - Best-move: $\Psi_i^{\text{best}} = e_{ij}^{i'}, \text{ st, } \text{eval}(e_{ij}^{i'}) = \text{Max } \text{eval}(e_{ij}^{k}) \text{ (for all } e_{ij}^{k} \in D_i)$
  - Better-move: $\Psi_i^{\text{better}} = e_{ij}^{i'}, \text{ st, } \text{eval}(e_{ij}^{i'}) > \text{eval}(e_{ij}^{l})$
  - Random-move: $\Psi_i^{\text{random}} = e_{ij}^{i'}$, $e_{ij}^{i'}$ is a random position in $D_i$
MASSAT Procedure

For \( i = 1 \) to MAX-Cycles
   If all clauses are satisfied Then return \( T \); 
   For all distributed agents 
      Select one (or more) of three behaviors 
      \{Best-move, Better-move, or Random-move\}; 
   Perform selected behavior(s); 
   EndFor 
   If no agent moves Then 
      Modify the weights of unsatisfied clauses; 
   End 
   Update \( T \) according to new positions of agents; 
EndFor

Comparison Based on Time Step

- Benchmark SAT problem packages from SATLIB
- Time step is the minimum unit
  - At each time step of SDF, only one variable is flipped
  - In MASSAT, agents move asynchronously

\[
W_{t+1}^i = \begin{cases} 
W_t^i & \text{if } T(C_t^i) = T\text{-true} \\
W_t^i + \delta & \text{if } T(C_t^i) = \text{False} 
\end{cases}
\]

where \( W_t^i \) is the weight of clause \( C_t^i \) at time \( t \).
\( \delta \) is a learning rate.
**MASSAT: Variable-Based Agent Representation**

Nodes: Each agent represents a variable in an SAT problem.

Edges: If two variables appear in a common clause, there will be an edge between the agents corresponding to these two variables, indicating these two agents need to coordinate their respective values.

**Clause-Based Agent Representation**

Nodes: Each agent represents a clause in an SAT problem.

Edges: If clauses represented by two agents have a common literal, there will be an edge between these agents, which indicates these two agents need to coordinate.
Agent Networks

- How does the topology of an agent network reflect the complexity of solving distributed SAT?
  - Jin and Liu (AAMAS’03) have experimentally proven that constraint satisfaction in a small-world is inefficient

- Do agent networks formed in MASSAT have small-world topologies?

Small-World

1. Highly clustered
2. The shortest path length between any two nodes is small

WI Challenge #3

Social Networks + Self-Organization
Social Networks

- A Social Network comprises a group of people with a pattern of interactions among them.

- A Social Network is a self-organizing structure of users, information, and communities of expertise or practice.

Self-Organization: Game of Life

*Conway, 1970*

- ‘Life’ rules are applied to an initial population of live cells (i.e., black circles)
Social Networks + Self-Organization: Game of RoboNBA (with C. H. Ng)

- Live cells ➔ Distributed player agents
- ‘Life’ rule ➔ Decision/strategy
- Environment ➔ Virtual court
- Patterns ➔ NBA-like games

Basketball competition is a complex behavior:
- Team players interact **locally**
- It is difficult to predict a **complete** match

Example: **Moving Action** (1 of 8)

- blue circle – defender’s old position
- red circle – offender’s old position
- blue square – defender’s new position
- red square – offender’s new position
The teams used for testing the accuracy of the system:

<table>
<thead>
<tr>
<th>Team Name</th>
<th>Rank in NBA</th>
<th>Wins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dallas Maverick</td>
<td>1st</td>
<td>0.75%</td>
</tr>
<tr>
<td>Philadelphia Sixers</td>
<td>10th</td>
<td>0.592%</td>
</tr>
<tr>
<td>Washington Wizards</td>
<td>20th</td>
<td>0.461%</td>
</tr>
<tr>
<td>Cleveland Cavs</td>
<td>29th</td>
<td>0.197%</td>
</tr>
</tbody>
</table>

Dallas Mavericks VS Philadelphia Sixers (5 matches)
Mavericks win Sixers 4 - 1 (averages 90.6 - 79.2)
Dallas Mavericks VS Philadelphia Sixers
Mavericks win Sixers 107-94

<table>
<thead>
<tr>
<th>TEAM</th>
<th>DJ</th>
<th>MN</th>
<th>FGMA</th>
<th>3OMA</th>
<th>FTMA</th>
<th>OFF</th>
<th>DEF</th>
<th>TOT</th>
<th>AST</th>
<th>PF</th>
<th>ST</th>
<th>TO</th>
<th>BS</th>
<th>PTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mavericks</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Michael Finley</td>
<td>G</td>
<td>40</td>
<td>8-14</td>
<td>1-4</td>
<td>2-2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>Shawn Bradley</td>
<td>C</td>
<td>24</td>
<td>4-7</td>
<td>0-0</td>
<td>0-0</td>
<td>4</td>
<td>5</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>Walt Williams</td>
<td></td>
<td>13</td>
<td>4-4</td>
<td>3-3</td>
<td>0-0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>Rauf LaFrentz</td>
<td></td>
<td>7</td>
<td>0-2</td>
<td>0-0</td>
<td>0-0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Avery Johnson</td>
<td></td>
<td>2</td>
<td>0-1</td>
<td>0-0</td>
<td>0-0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Dallas Jazz</td>
<td></td>
<td>DNP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| TOTAL        |    | 240| 42-85| 12-22| 11-11| 9   | 30  | 39  | 24  | 22  | 10 | 7  | 107| 49.4% 54.5% 100.0% Team Rebs: 3 Total TO: 10

2003-8-29 Source: www.nba.com

RoboNBA Games
(30 matches)

<table>
<thead>
<tr>
<th>TEAMS</th>
<th>Sixers (right)</th>
<th>Wizards (right)</th>
<th>Cavs (right)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mavericks (left)</td>
<td>22 – 8 (87.0 – 83.7)</td>
<td>24 – 6 (86.0 – 73.5)</td>
<td>25 – 5 (82.7 – 72.9)</td>
</tr>
<tr>
<td>(real: 107.91)</td>
<td>(real: 106.101)</td>
<td>(real: 114.93)</td>
<td></td>
</tr>
<tr>
<td>Sixers (left)</td>
<td>18 – 12 (78.7 – 76.5)</td>
<td>24 – 6 (75.4 – 69.0)</td>
<td>14 – 16 (69.3 – 74.3)</td>
</tr>
<tr>
<td>(real: 89.90)</td>
<td>(real: 116.102)</td>
<td>(real: 93.91)</td>
<td></td>
</tr>
<tr>
<td>Wizards (left)</td>
<td>14 – 16 (69.3 – 74.3)</td>
<td>24 – 6 (75.4 – 69.0)</td>
<td>22 – 8 (87.0 – 83.7)</td>
</tr>
<tr>
<td>(real: 93.91)</td>
<td>(real: 116.102)</td>
<td>(real: 107.91)</td>
<td></td>
</tr>
</tbody>
</table>

2003-8-29
Issues and Directions

Top 10 Issues

1. Goal-directed Services (best means/ends)
2. Personalization (identity)
3. Social & psychological context (sensitivity)
4. PSML (representation)
5. Coordination (global behavior)
6. Meta-knowledge (planning control)
7. Semantics (relationships)
8. Association (roles)
9. Reproduction (population)
10. Self-aggregation (feedback)
The Wisdom Web

◆ To enable humans to gain practical wisdom of living, working, and playing
  ❖ Wisdom: (Webster Dictionary, page 1658)
    The quality of being wise; knowledge, and the capacity to make due use of it: knowledge of the best ends and the best means

◆ To provide
  ❖ a medium for knowledge and experience (e.g., of the Grand Canyon) sharing
  ❖ a supply of self-organized resources for driving sustainable knowledge creation and scientific or social development/evolution

Summary

◆ Background
◆ Challenges
  ❖ Semantic Web + Planning: OntoPlan
  ❖ Distributed Agents + Coordination: MASSAT
  ❖ Social Networks + Self-Organization: RoboNBA

◆ Issues and Directions
  ❖ The Wisdom Web