TRANSFER LEARNING OF SEARCH HEURISTICS FOR CONSTRAINT SATISFACTION

(Introducing Constrained Heuristic Search to the Soar Cognitive Architecture)

(30th Soar Workshop, University of Michigan)

Sean A. Bittle
Mark S. Fox

May 27th, 2010
Agenda

- Introduction: The Problem
- Implementation
- Experiments and Results
- Summary and Future Work
Introduction

“What is the problem?”

Constraint Satisfaction Problems (CSP)

What variable and value should we select next?

D = \{R,G,B\}
Introduction

“What is the problem?”

A central challenge in efficiently solving CSP’s is variable and value ordering.

Current approaches use heuristics to guide ordering.

Tradeoff between generality and effectiveness

Can we learn variable and value ordering heuristics that are effective and general?
Introduction

- Our goal is to demonstrate that computers can apply knowledge learned from one set task(s) to achieve superior performance on a different set of task(s).

Constraint Satisfaction Problems (CSP)

- Our thesis is that we can abstract and learn variable and value ordering search heuristics from a set of task(s) that can be used to secure improved problem solving performance on a different set of task(s).
Introduction

“What is transfer learning?”

Transfer learning (TL) is the ability to extract knowledge learned for an original (source) set of tasks in one domain to solve problems on a different (target) set of tasks in order to improve performance or enhance future learning [DARPA, 2005]
Implementation

“What is CHS-Soar-RL?”

- Combines constraint and rule based reasoning
- Soar: Symbolic cognitive architecture developed by Newell, Laird and Rosenbloom, 1983
  - Two forms of learning (Chunking and RL)
- Constrained Heuristic Search (CHS) developed by Fox, Sadeh and Baykan, 1989
  - Problem Topology
  - Problem Textures
  - Problem Objective
Implementation

“What are texture measures?”

- A texture measurement is a technique for distilling information embedded in the constraint graph into a form that heuristics can use.

<table>
<thead>
<tr>
<th>Ordering</th>
<th>Name</th>
<th>Texture</th>
<th>Heuristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Degree (DEG)</td>
<td>$C_i$, number of constraints to other unassigned variables linked to variable.</td>
<td>Select max texture.</td>
</tr>
</tbody>
</table>
Implementation

“What are texture measures?”

- A texture measurement is a technique for distilling information embedded in the constraint graph into a form that heuristics can use

<table>
<thead>
<tr>
<th>Ordering</th>
<th>Name</th>
<th>Texture</th>
<th>Heuristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Degree (DEG)</td>
<td>$C_v$, number of constraints to other unassigned variables linked to variable.</td>
<td>Select Max texture.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>DEG</th>
<th>DEG Actual</th>
<th>DEG Normalized</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1</td>
<td>2</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>V2</td>
<td>3</td>
<td>0.6</td>
<td></td>
</tr>
<tr>
<td>V3</td>
<td>3</td>
<td>0.6</td>
<td></td>
</tr>
<tr>
<td>V4</td>
<td>3</td>
<td>0.6</td>
<td></td>
</tr>
<tr>
<td>V5</td>
<td>2</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>V6</td>
<td>5</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>V7</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>
Implementation

“How do we solve problems?”

CHS-SOAR PROBLEM SPACE

Initial State → Propagation

VARIABLE ORDERING

VALUE ORDERING

DEG Textures

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEG</td>
<td>0.0</td>
</tr>
<tr>
<td>DEG</td>
<td>0.4</td>
</tr>
<tr>
<td>DEG</td>
<td>0.6</td>
</tr>
<tr>
<td>DEG</td>
<td>1.0</td>
</tr>
</tbody>
</table>

LCV Textures

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCV</td>
<td>?</td>
</tr>
<tr>
<td>LCV</td>
<td>?</td>
</tr>
</tbody>
</table>

Operators

Goal State

Propagation

State
Implementation

“How do we learn?”

Learning Phase

Source Task(s)

SOURCE TASKS

Source task represented as a binary constraint graph

CHS-Soar-RL

Learning
Long Term Memory
CHS-Soar-RL Agent
Decision Cycle
Working Memory

External Agent

Soar (Rule Based Reasoning)

CHS (Constraint Based Reasoning)

Learned Heuristics

Chunks

RL (Epsilon-Greedy)

RL (Subgoaling)

[Condition] -- [Action]

Learning search heuristics for variable and value ordering.

Testing Phase

CHS-Soar-RL

Learning
Long Term Memory
CHS-Soar-RL Agent
Decision Cycle
Working Memory

External Agent

Target Task(s)

Target task represented as a binary constraint graph

Job 1
Op → Op → Op

Job 2
Op → Op → Op

Job n
Op → Op → Op

Target Task

Target Task
Experiments

Investigate target task initial performance improvement via transfer learning for 3 different learning approaches and 2 different sets of source tasks:

Experiment 1: Source Tasks: RGP (Random Generated CSP’s)
Experiment 2: Source Tasks: Mix (set of CSP’s comprised of: MCP, FAP & JSP)

<table>
<thead>
<tr>
<th>Target Tasks</th>
<th>Base</th>
<th>Instance Size (# of variables)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Base</td>
<td>1x</td>
</tr>
<tr>
<td>Size multiple of base</td>
<td></td>
<td>1x</td>
</tr>
<tr>
<td>Map Coloring (MCP)</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>Job Shop Scheduling (JSP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frequency Assignment (FAP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Traveling Salesman (TSP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Randomly Generated (RGP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N-Queens (NQP)</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>
Experiments

Evaluation of target task performance based on:

- No Transfer (random selection)
- Transfer (with training from source tasks)
- Benchmark uses standard CSP heuristics (i.e. MRV, DEG, LCV)

Allows us to compare two metrics:

- Transfer Ratio: Performance change with transfer over no transfer
- Benchmark Ratio: Performance change with transfer over benchmark
Experiment 1: (RGP Source Tasks)

Transfer Ratio: How did we perform with transfer over the no transfer case?

For FAP, MCP and JSP we observe approximately 50% improvement for Chunks and RL (Subgoaling).
Experiment 1: (RGP Source Tasks)

Benchmark Ratio: How did we perform with transfer over the benchmark case?

Chunks

RL (Epsilon-Greedy)

RL (Subgoaling)

Overall negative benchmark performance for all three learning approaches.
Experiment 2: (Mix Source Tasks)

Transfer Ratio: How did we perform with transfer over the no transfer case?

For FAP, MCP and JSP we observe approximately 75% improvement for Chunks and RL (Subgoaling).
Experiment 2: (Mix Source Tasks)

**Benchmark Ratio: How did we perform with transfer over the benchmark case?**

For FAP, MCP and JSP we observe 9% improvement for Chunks and RL (Subgoaling) over benchmark case.
Experiment 2: (Mix Source Tasks)

Results suggest “Mixed” source tasks provide more opportunities to encode search heuristic knowledge, particularly for value ordering over RGP.
Experiment 2: (Mix Source Tasks)

Frequency Assignment Problem (n=100)

Non max or min texture values deliver superior performance

DEG and LCV Heuristics

MRV Heuristic
Summary

- Progress to date has demonstrated transfer learning for variable and value ordering in binary CSPs by combining constraint and rule based reasoning.

- Initial performance improvement (type 1) of:
  - 50% above the no transfer case
  - 9% above the benchmark case
  - Sensitive to the type of source training tasks.

- Expanding the expressiveness of the search heuristics by encoding intermediate (i.e. non min/max) texture details allows us to secure improved performance over benchmark heuristics for selected target tasks.

- Did not confirm the inherent benefit of reinforcement learning which allows us to dynamically revise the numerical preferences.
Current/Future Work

- Duration and diversity of training
- Improved texture evaluation functions (e.g. tree search)
- Additional (more insightful) texture measurements
- New structural features (i.e. density, tightness)
- Additional problem types (i.e. tasks)
Questions

TRANSFER LEARNING OF SEARCH HEURISTICS FOR CONSTRAINT SATISFACTION
(Introducing Constrained Heuristic Search to the Soar Cognitive Architecture)
Sean A. Bittle
Mark S. Fox