
Welcome
Welcome

Committee
- Dr. Arvin Agah, Professor in Charge
- Dr. Victor Frost
- Dr. Costas Tsatsoulis
Problem and Solution Overview
Overview of Presentation

- Problem and Solution Overview
- Playing Go
Overview of Presentation

- Problem and Solution Overview
- Playing Go
- Relevant Computational Methods
Overview of Presentation

- Problem and Solution Overview
- Playing Go
- Relevant Computational Methods
  - Genetic algorithms
  - Traditional programs
- State of the Art
Overview of Presentation

- Problem and Solution Overview
- Playing Go
- Relevant Computational Methods
  - Genetic algorithms
  - Traditional programs
- Implementation
Problem and Solution Overview
Playing Go
Relevant Computational Methods
State of the Art
– Genetic algorithms
– Traditional programs
Implementation
Experiments and Results
Overview of Presentation

● Problem and Solution Overview
● Playing Go
● Relevant Computational Methods
   – Genetic algorithms
   – Traditional programs
● State of the Art
● Implementation
● Experiments and Results
● Conclusion
   – Contributions
   – Limitations
   – Future
Introduction to Go

- Perfect Information
Introduction to Go

- Perfect Information
- Board is 19 by 19
Perfect Information

Board is 19 by 19

Two players
Introduction to Go

- Perfect Information
- Board is 19 by 19
- Two players
- Territory
Introduction to Go

- Perfect Information
- Board is 19 by 19
- Two players
- Territory
- Capturing
Go has simple rules, but tactics and strategies are complex
- Go has emergent complexity
- Multiagent systems have emergent complexity
Go has simple rules, but tactics and strategies are complex
– Go has emergent complexity
– Multiagent systems have emergent complexity
Current go programs play on a beginner level, why?
Go has simple rules, but tactics and strategies are complex
- Go has emergent complexity
- Multiagent systems have emergent complexity

Current go programs play on a beginner level, why?

<table>
<thead>
<tr>
<th>Search Ply</th>
<th>Go</th>
<th>Chess</th>
<th>Checkers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>361</td>
<td>20</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>129,960</td>
<td>400</td>
<td>49</td>
</tr>
<tr>
<td>3</td>
<td>445,145,640</td>
<td>approx. 10,000</td>
<td>approx. 343</td>
</tr>
</tbody>
</table>
Goals of This Project

- Use multiple agents to suggest solutions based on a narrow world perspective
Goals of This Project

- Use multiple agents to suggest solutions based on a narrow world perspective
- Bring these solutions together to obtain a better overall solution
Goals of This Project

- Use multiple agents to suggest solutions based on a narrow world perspective
- Bring these solutions together to obtain a better overall solution
- Agents are not fully decentralized
Goals of This Project

- Use multiple agents to suggest solutions based on a narrow world perspective
- Bring these solutions together to obtain a better overall solution
- Agents are not fully decentralized
- Algorithmic composition of individual agent solutions
Goals of This Project

- Use multiple agents to suggest solutions based on a narrow world perspective
- Bring these solutions together to obtain a better overall solution
- Agents are not fully decentralized
- Algorithmic composition of individual agent solutions
- Illustrate this method in a non-trivial environment: go
Multiagent Architecture
- Specialized agents
- Each has its own perspective of the game
- Outputs an array representing move qualities
Multiagent Architecture
- Specialized agents
- Each has its own perspective of the game
- Outputs an array representing move qualities

Agents connected via a summation network to generate output
- No communication
- Allows a passive combination of agent output into a solution
**Research Approach**

- **Multiagent Architecture**
  - Specialized agents
  - Each has its own perspective of the game
  - Outputs an array representing move qualities

- Agents connected via a summation network to generate output
  - No communication
  - Allows a passive combination of agent output into a solution

- Weights for this network were evolved using genetic algorithms
Network weights are four-bit integers
Network weights are four-bit integers
These four-bit integers make up chromosome
Network weights are four-bit integers
These four-bit integers make up chromosome
Extra bits at the end of chromosome are available
– Extra bits for internal use by agents
– Extender agent uses these extra bits
Diagram of Summation Network
Why Use Go?

- Traditional search provides little help
Why Use Go?

- Traditional search provides little help
- Complex
Why Use Go?

- Traditional search provides little help
- Complex
- Heavily pattern-oriented
Why Use Go?

- Traditional search provides little help
- Complex
- Heavily pattern-oriented
- Unsolved now and in the near future
Why Use Go?

- Traditional search provides little help
- Complex
- Heavily pattern-oriented
- Unsolved now and in the near future
- Analogues to more complex environments
  - Local versus global concerns
  - Many choices at any point
  - Adversarial
The Go Board
Groups
Playing Go (continued)

- Groups
- Eyes
Groups

Eyes

Live and Dead Stones
Capturing
Ko and Seki Rules
Territory
Scoring
Scoring
Other board sizes
Playing Go (continued)

- Scoring
- Other board sizes
- Handicaps
Random search
Genetic Algorithms

- Random search
- Populations
Genetic Algorithms

- Random search
- Populations
- Chromosomes representing parameters or solutions
• Genetic Algorithms

• Random search
• Populations
• Chromosomes representing parameters or solutions
• Fitness functions
Genetic Algorithms

- Random search
- Populations
- Chromosomes representing parameters or solutions
- Fitness functions
- Crossover, mating, and mutations
Multiagent Systems

- Autonomous agents
- Autonomous agents
- Sense environment
Multiagent Systems

- Autonomous agents
- Sense environment
- Interacts with environment
- Autonomous agents
- Sense environment
- Interacts with environment
- Cooperative or adversarial
• No Soft Methods
Good Traditional Programs

- No Soft Methods
- Müller
  - Patricia trees variant
  - 3000 pattern database
Good Traditional Programs

- No Soft Methods
- Müller
  - Patricia trees variant
  - 3000 pattern database
- Many Faces of Go
  - Opening database of 45,000 moves
  - Pattern database of 1000 patterns
  - 200 rules hardcoded
• No Soft Methods
• Müller
  – Patricia trees variant
  – 3000 pattern database
• Many Faces of Go
  – Opening database of 45,000 moves
  – Pattern database of 1000 patterns
  – 200 rules hardcoded
• Others
  – Patterns
  – Try to create small set of possible moves to look into
Other Programs Using Genetic Algorithms

- Da Silva
  - Evaluation function evolved
Other Programs Using Genetic Algorithms

- Da Silva
  - Evaluation function evolved
- Jeffrey Greenberg
  - Evolved Prolog-like statements
Other Programs Using Genetic Algorithms

- Da Silva
  - Evaluation function evolved
- Jeffrey Greenberg
  - Evolved Prolog-like statements
- Ecological Models
  - Sets of rules were evolved
Other Programs Using Genetic Algorithms

- Da Silva
  - Evaluation function evolved
- Jeffrey Greenberg
  - Evolved Prolog-like statements
- Ecological Models
  - Sets of rules were evolved
- Neural Network Hybrids: SANE
  - Neural network configuration and weights evolved
  - Entire board fed into neural network
Other Programs Using Genetic Algorithms

- Da Silva
  - Evaluation function evolved
- Jeffrey Greenberg
  - Evolved Prolog-like statements
- Ecological Models
  - Sets of rules were evolved
- Neural Network Hybrids: SANE
  - Neural network configuration and weights evolved
  - Entire board fed into neural network
- Common Themes
  - Small Boards
  - No Meta-processing
  - Not multiagent
Support Classes

- Bit-level operations for Stone class for speed
- Bit-level operations for Stone class for speed
- Board class is a 1D array of stone classes
Support Classes

- Bit-level operations for Stone class for speed
- Board class is a 1D array of stone classes
- Game class is a linked list of Boards
Support Classes

- Bit-level operations for Stone class for speed
- Board class is a 1D array of stone classes
- Game class is a linked list of Boards
- Probability Board class
  - Parallel to board array
  - Each offset is a move quality
  - Summation, normalization, and scaling provided
  - Spin
Moderator class, a template
- Moderator class, a template
- Multiagent genetic algorithm player
• Moderator class, a template
• Multiagent genetic algorithm player
• Genetic algorithm trainer player
  – Fitness function
Text User Interface

P=pass, A3=play at (A,3), R=redisplay, Q=quit
Move> []
Graphical User Interface
Agents

- Random
Agents

- Random
- Follower
Agents

- Random
- Follower
- Opener
Agents

- Random
- Follower
- Opener
- Capture
Agents

- Random
- Follower
- Opener
- Capture
- Tiger’s Mouth
Agents

- Random
- Follower
- Opener
- Capture
- Tiger’s Mouth
- Extender
  - Uses GA values internally
Experiments Overview

- Each Agent Individually
Experiments Overview

- Each Agent Individually
- Random Agent
Experiments Overview

- Each Agent Individually
- Random Agent
- Multiagent
Experiments Overview

- Each Agent Individually
- Random Agent
- Multiagent
- GA parameters
  - Crossover 0.4
  - Mutation 0.0333
  - Population size: 10 and 100
## Results of Single Agents, Capturer

<table>
<thead>
<tr>
<th>Generation</th>
<th>Max</th>
<th>Min</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Sumfitness</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>7.85e-09</td>
<td>0.777</td>
</tr>
<tr>
<td>1</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>2.95e-05</td>
<td>0.777</td>
</tr>
<tr>
<td>2</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>2.95e-05</td>
<td>0.777</td>
</tr>
<tr>
<td>3</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.00181</td>
<td>0.777</td>
</tr>
<tr>
<td>4</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.00181</td>
<td>0.777</td>
</tr>
<tr>
<td>5</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0142</td>
<td>0.777</td>
</tr>
<tr>
<td>6</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0142</td>
<td>0.777</td>
</tr>
<tr>
<td>7</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0397</td>
<td>0.777</td>
</tr>
<tr>
<td>8</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0777</td>
<td>0.0397</td>
<td>0.777</td>
</tr>
</tbody>
</table>
Results of Single Agent Experiment: Extender

<table>
<thead>
<tr>
<th>Generation</th>
<th>Max</th>
<th>Min</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Sumfitness</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0415</td>
<td>0.0301</td>
<td>0.0365</td>
<td>0.00428</td>
<td>0.365</td>
</tr>
<tr>
<td>1</td>
<td>0.0089</td>
<td>0</td>
<td>0.0354</td>
<td>0.03219</td>
<td>0.354</td>
</tr>
<tr>
<td>2</td>
<td>0.0777</td>
<td>0.0037</td>
<td>0.0389</td>
<td>0.0347</td>
<td>0.389</td>
</tr>
<tr>
<td>3</td>
<td>0.0788</td>
<td>0.00656</td>
<td>0.0394</td>
<td>0.0641</td>
<td>0.394</td>
</tr>
<tr>
<td>4</td>
<td>0.0753</td>
<td>2.7e-10</td>
<td>0.0409</td>
<td>0.0667</td>
<td>0.409</td>
</tr>
<tr>
<td>5</td>
<td>0.0821</td>
<td>8.1e-9</td>
<td>0.0436</td>
<td>0.0899</td>
<td>0.436</td>
</tr>
<tr>
<td>6</td>
<td>0.0797</td>
<td>5.57e-09</td>
<td>0.0458</td>
<td>0.0893</td>
<td>0.458</td>
</tr>
<tr>
<td>7</td>
<td>0.0669</td>
<td>0</td>
<td>0.0468</td>
<td>0.102</td>
<td>0.468</td>
</tr>
<tr>
<td>8</td>
<td>0.0861</td>
<td>0</td>
<td>0.0449</td>
<td>0.105</td>
<td>0.449</td>
</tr>
</tbody>
</table>

Fitness By Generation (Extender Agent Only)
Results of Multiagent Experiment

<table>
<thead>
<tr>
<th>Generation</th>
<th>Max</th>
<th>Min</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Sumfitness</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0881</td>
<td>0.0435</td>
<td>0.0537</td>
<td>0.0138</td>
<td>0.537</td>
</tr>
<tr>
<td>1</td>
<td>0.108</td>
<td>0.0245</td>
<td>0.0539</td>
<td>0.0454</td>
<td>0.539</td>
</tr>
<tr>
<td>2</td>
<td>0.121</td>
<td>0.0106</td>
<td>0.0604</td>
<td>0.0575</td>
<td>0.604</td>
</tr>
<tr>
<td>3</td>
<td>0.119</td>
<td>3.49e-10</td>
<td>0.0694</td>
<td>0.0865</td>
<td>0.694</td>
</tr>
<tr>
<td>4</td>
<td>0.105</td>
<td>3.15e-09</td>
<td>0.0812</td>
<td>0.0867</td>
<td>0.812</td>
</tr>
<tr>
<td>5</td>
<td>0.109</td>
<td>2.82e-09</td>
<td>0.0848</td>
<td>0.103</td>
<td>0.848</td>
</tr>
<tr>
<td>6</td>
<td>0.108</td>
<td>2.15e-09</td>
<td>0.0877</td>
<td>0.103</td>
<td>0.877</td>
</tr>
<tr>
<td>7</td>
<td>0.134</td>
<td>0</td>
<td>0.087</td>
<td>0.113</td>
<td>0.87</td>
</tr>
<tr>
<td>8</td>
<td>0.14</td>
<td>0</td>
<td>0.0798</td>
<td>0.118</td>
<td>0.798</td>
</tr>
</tbody>
</table>

Fitness by Generation (All Agents)
Results of Multiagent Experiment: Large Population
Comparison Plot

Comparison of Best Fitness Achieved

Fitness

Agent

Opener  Tiger  Follower  GroupStats  Extension  All Agents  Large Pop.
Results Summary (Multiagent, Population 100)

- 0.143 Highest fitness
Results Summary (Multiagent, Population 100)

- 0.143 Highest fitness
- 0.076 Highest mean fitness
0.143 Highest fitness
0.076 Highest mean fitness
Student’s T-test
– T-test, 100 population confidence: 3.89E-21
– T-test, 10 population confidence: 5.04E-4
Contributions

- Unique approach to go
Contributions

- Unique approach to go
- Probabilistic methods for go
Contributions

• Unique approach to go
• Probabilistic methods for go
• Multiagent paradigm
Contributions

- Unique approach to go
- Probabilistic methods for go
- Multiagent paradigm
- Scalability
Limitations

- Board Size
Limitations

- Board Size
- Number of Agents
Limitations

- Board Size
- Number of Agents
- Time to run genetic algorithms
  - Training sets
  - Populations
  - Larger summation networks
  - Generations
Limitations

- Board Size
- Number of Agents
- Time to run genetic algorithms
  - Training sets
  - Populations
  - Larger summation networks
  - Generations
- Programmer’s knowledge of go
Future

- Larger population size
Future

- Larger population size
- Larger board size
Future

- Larger population size
- Larger board size
- More agents
Future

- Larger population size
- Larger board size
- More agents
- Agents of higher complexity
Future

- Larger population size
- Larger board size
- More agents
- Agents of higher complexity
- Larger summation network
Questions

- Thank You
Questions

- Thank You
- Thread Pools
Questions

- Thank You
- Thread Pools
- Search
Questions

- Thank You
- Thread Pools
- Search
- Minimax
Questions

- Thank You
- Thread Pools
- Search
- Minimax
- Dr. Arvin Agah
• Thank You
• Thread Pools
• Search
• Minimax
• Dr. Arvin Agah
• Committee
Questions

- Thank You
- Thread Pools
- Search
- Minimax
- Dr. Arvin Agah
- Committee
- Texts
  - Genetic Algorithms in Search Optimization, and Machine Learning
  - Numerical Recipes in C: The Art of Scientific Computing