A.I. projects you didn't know you are interested in …

Michael Buro
http://www.cs.ualberta.ca/~mburo
mburo@cs.ualberta.ca

Outline

- My Background & Interests
- 4 Projects
  - Evaluation Function Construction
  - Selective Search
  - Generic Game Server
  - Real-Time Strategy Game AI

My Background and Interests

- Heuristic Search in AI
  - **Evaluation Function Construction**
    - GLEM
  - **Selective Game-Tree Search**
    - Multi-ProbCut
- Opening Book Learning
  - Using heuristic evaluations
  - Waste of time

More Background and Interests

- Incomplete Information Games
  - Virtually perfect heuristic backgammon race player (including doubling)
  - Recently solved the sumo game
- **Generic Game Server**
  - One server binds them all …
- **Real-time Decision Support**
  - Real-time strategy (RTS) games

Applications

- **Logistello**
  - crushed the human Othello World-champion 6-0 in 1997, still among the best programs
- **Logistello-server**
  - connected to NEC's Internet Portal
- **AmsBot**
  - Was the strongest Amazons program in 2000, in hibernation now
- **GGS – Generic Game Server**
- **ORTS – Open Real-Time Strategy game programming environment**

1. Evaluation Function Construction

- Challenge:
  - Given 100 million labeled training positions of your favorite game or puzzle
  - Construct a fast evaluation function that predicts the final game result or solution length with high confidence
- Approach:

  Thousands of indicator features + Regression
- Works great in **Othello**, what about **chess** or **go** or **single agent problems**?
- Topics: Automatic pattern search, over-fitting, distributed regression

2. Forward Pruning

- Idea: exploit correlation of look-ahead search results
- "If a Queen behind and no compensation in sight, don't bother looking farther ahead"
- Multi-ProbCut uses a simple linear model to predict opinion changes
- Works well in **Othello**, what about **chess** or **go** or **single agent problems**?
- Topics: More sophisticated model. When does MPC work? When does it not?

3. Generic Game Server

- Players and game services connect to central server
- Uniform and **open message protocol**
- Uniform game **archive format**
- **Fair game modes** that overcome color bias in random starting positions
- **One applet** for all games
- Currently just board games
- To do: Class hierarchy that allows to add games of other types, port the applet to C++, support more games, **connect Chinook!**

4. Real-Time Strategy Games

- Very popular PC games. Million-sellers!
  - –WarCraft, StarCraft (Blizzard)
  - –Age of Empires (Ensemble Studios)
- Players set up **economy**, build **armies** and struggle over **resources** in a 2.5D world
- **real-time – 5+ simulation cycles/sec**

What makes RTS games hard?

- Imperfect information
- Hundreds of objects
- Micro-actions, tricks of the trade don't work
- Real-time action!

Computer opponents are stupid because they don't

adapt, look-ahead, grasp spatial and temporal relations, collaborate, ...

Easy for humans!

RTS game projects

- RTS programming environment, server
- State space abstraction & planning
- Opponent modeling, learning
- TD learning of low-level behavior
- Dealing with incomplete information: When and where to scout? What are the opponents' intentions?
- Finding safe routes fast ("path-finding")
- …

Real-Time Decision Support Meetings

Fridays 14:00-15:00

CSC 333

Starting today!

http://www.cs.ualberta.ca/~mburo/rtds.html

Mailing list: rtds@cs.ualberta.ca

Interested?

Here is how you can reach me:

Ath 3-37
492-1763
mburo@cs.ualberta.ca