



TECHNISCHE
UNIVERSITÄT
DRESDEN

Faculty of Computer Science, Institute for Artificial Intelligence, Computational Logic Group

TAKT

A General Game Player
(Peter Großmann)

Dresden, 31.03.2010

Overview Of Model / Algorithms

- most game tree search is done by iterative deepening:
 - terminates, if time is up or memory exceeds certain limit
 - stores found nodes in transposition table
- nonterminal states are evaluated by goal distance
- singleplayer: iterative deepening + greedy search
- turntaking multiplayer games: MaxN
- simultaneous multiplayer games: nash equilibria

Goal Distance Heuristic

- all nonterminal states are evaluated by the goal distance heuristic
- tries to reach terminal states, if goal rule is satisfied, else try to avoid terminal states
- identified structures:
 - order relations
 - amounts
 - boards
 - singletons

Singleplayer

- splits start / play clock phase into two parts:
 - firstly iterative deepening to get an "idea" of the game and find out all boundary nodes
 - secondly greedy search with the initial set of the previous found boundary nodes
- stores the move sequence to the current best found node

Turntaking Multiplayer Games

- treats every node as a maximation node for the current player in turn
- every player tries to reach his maximum
- stores all nodes in a transposition table until a better evaluation of this node has been found

Simultaneous Multiplayer Games

- plays according to found nash equilibria (game theory)
- eliminates all dominated strategies / rows (hyper planes for > 2 players)
- calculates in each node the nash equilibria according to the values of the successor nodes:
 - exactly one equilibrium found: play this strategy
 - > 1 equilibrium found: calculate the mean value for all involved strategies and choose best
 - 0 equilibria found: calculate mean value for all strategies for all players and restart procedure

Improvements / TODOs

- replace 2 player turntaking zerosum games again by alpha beta pruning / negamax (currently problems with transposition table)
- improve speed of goal distance heuristic evaluation (consumes a lot of cpu time)
- improve reasoning speed by automatic rule / clause reordering
- analyze order relations semantically and not syntactically (currently too slow for domains with cardinality $> 20k$)