

CADIAPLAYER COMMANDS

HILMAR FINNSSON

These first are the standard Communication Protocol for a GGP agent.

(START [matchid] [role] [kif] [startclock] [playclock][GDL])	Start a new game, player replies after thinking on startclock with "READY".
(PLAY [matchid] NIL)	The first play message (when no moves have been made), player replies after thinking on playclock with its first move in kif format.
(PLAY [matchid] ([last move of all roles]))	All other play messages, the player performs them moves included and replies with its next move after thinking on the playclock.
(STOP [matchid] ([last move of all roles]))	Same as above PLAY message but game is in terminal position when the moves included have been played. Player replies with "DONE".

These are custom made for manipulating CADIAPLAYER.

(QUIT)	Terminate the program.
(CONF)	Expected goal value if the player is allowed to finish the current game.
(LOAD [matchid] [role] [kif] [startclock] [playclock])	Same as START but no thinking on startclock is done before replying with "READY".
(STATE)	Player replies with the current state.
(MOVE [matchid] ([legal moves for all roles]))	The player executes the moves included and replies with its new state.
(UNDO)	Undo last state transition of the player and the player replies with the new current state.
(FILE [matchid] [role] ([filename])[startclock] [playclock])	Same as START but the game description is read from a file instead of being included in the message.
(THINK)	Make the player think on the startclock and reply with the move it would choose in the current state.
(AVAIL)	The player replies with all moves available for all roles.
(TERM)	If the current game is at a terminal state the player replies with "1", "0" otherwise.
(GOAL)	The player replies with the goals of all roles calculated from the current state. If no goal for a role can be proven it is returned as -1.
(HALT)	Discards the current game and game description and replies with "DONE".
(INFO [matchid] ([filename]))	The player writes the current in-memory game-tree to the file given and replies with "done" when done.