Play-Based Team Coordination

Manuela Veloso, Brett Browning
CORAL Research Group
Carnegie Mellon

Boeing Treasure Hunt
Individual Visit

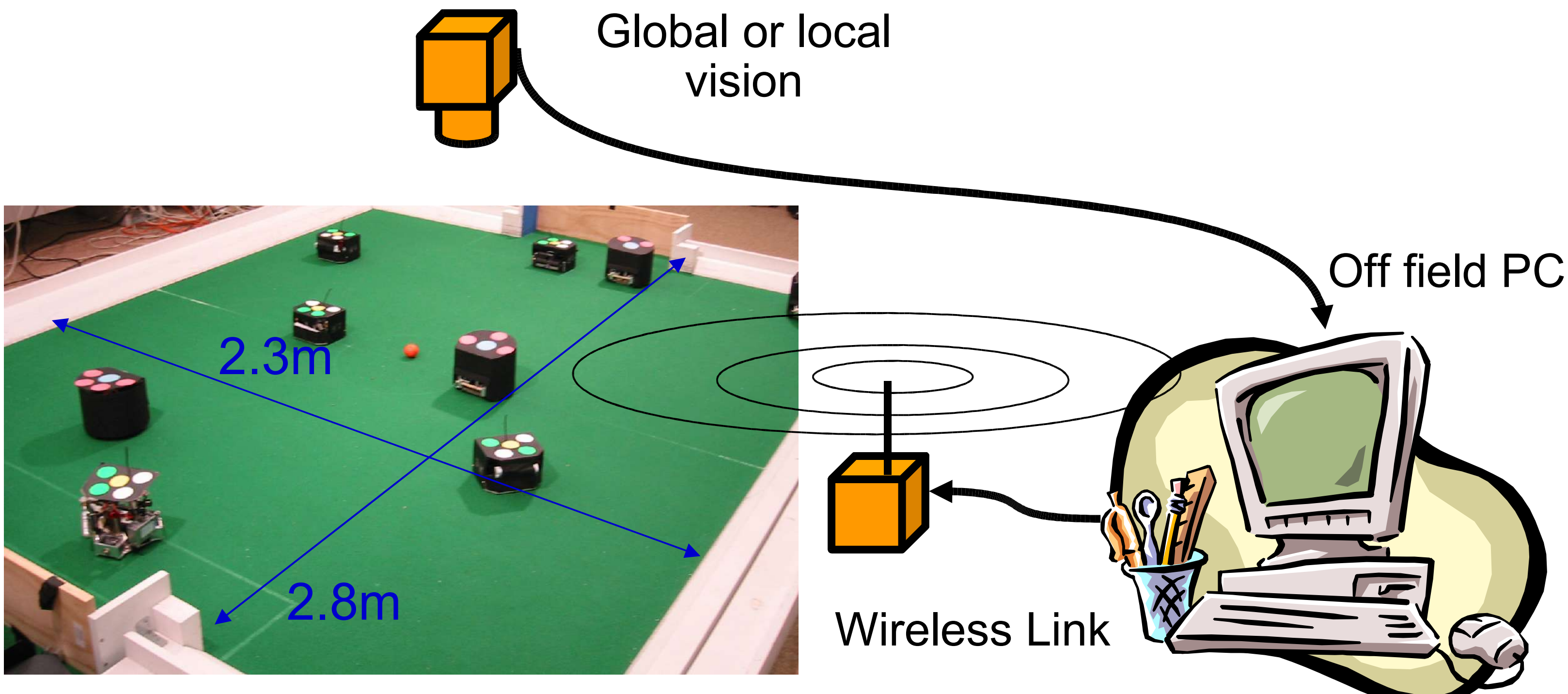
Introduction

- How do we get autonomous teams to work in highly dynamic, adversarial task
- Main challenges
  - Single robot and team autonomy
  - Synchronizing activities
  - Opportunistic events (and other dynamic changes)
  - Robust monitoring of actions and failure recovery
  - Adapting strategy in response to opponents and/or team performance

Must all work in *real*-time

CMDragons Small-Size Robots

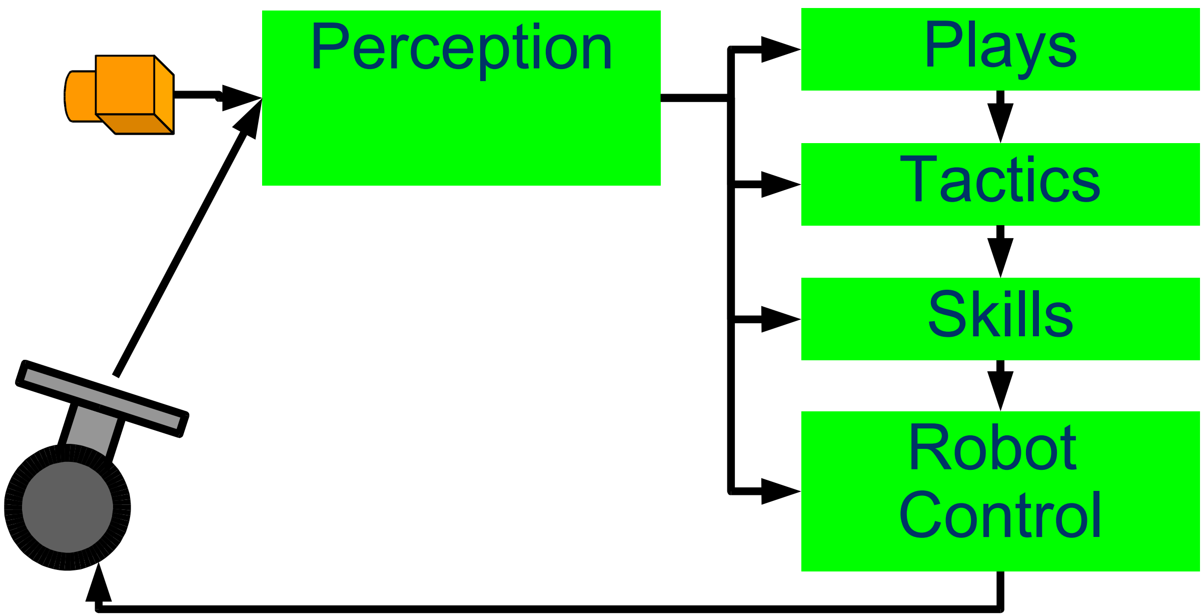
- 5 robots on each team and goal ball
- Global vision and/or local vision allowed
- System must be autonomous as a whole



With Michael Bowling, James Bruce

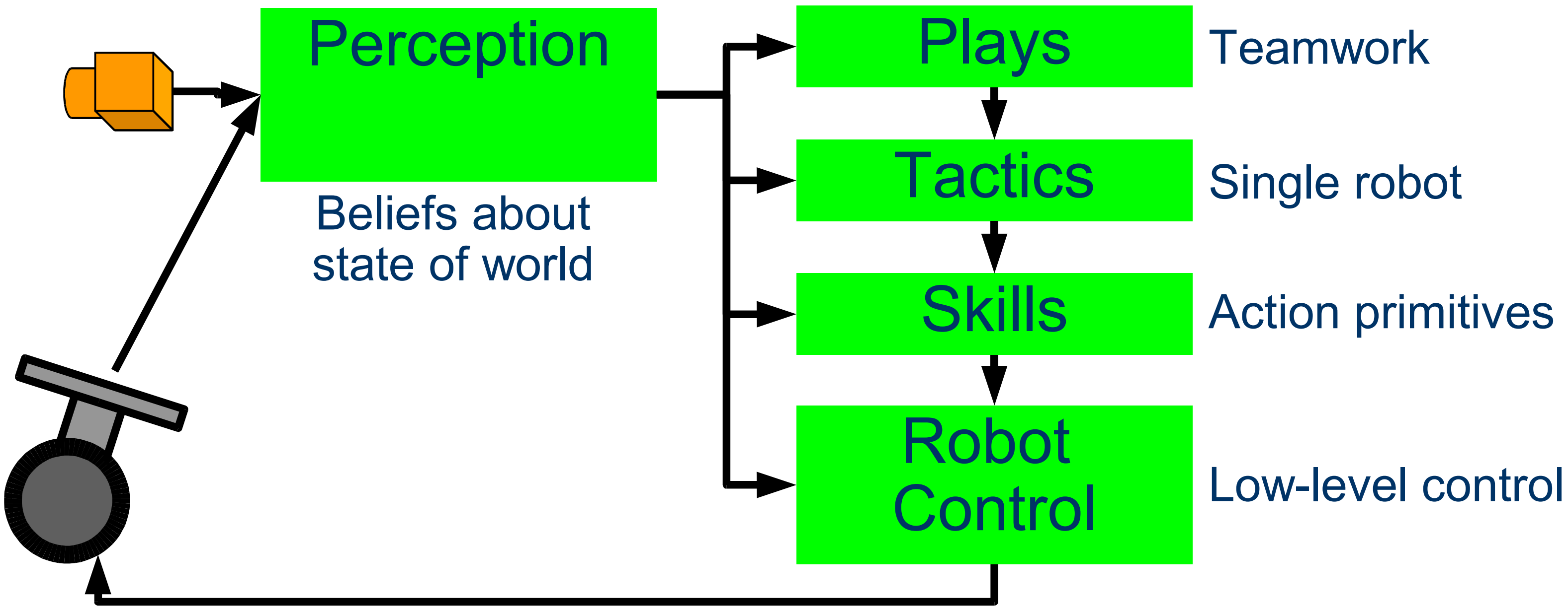
Skills, Tactics, Plays

- Skills -- encode complex low-level actions
- Tactics -- encapsulate single robot behavior
- Plays -- For adaptive team strategy

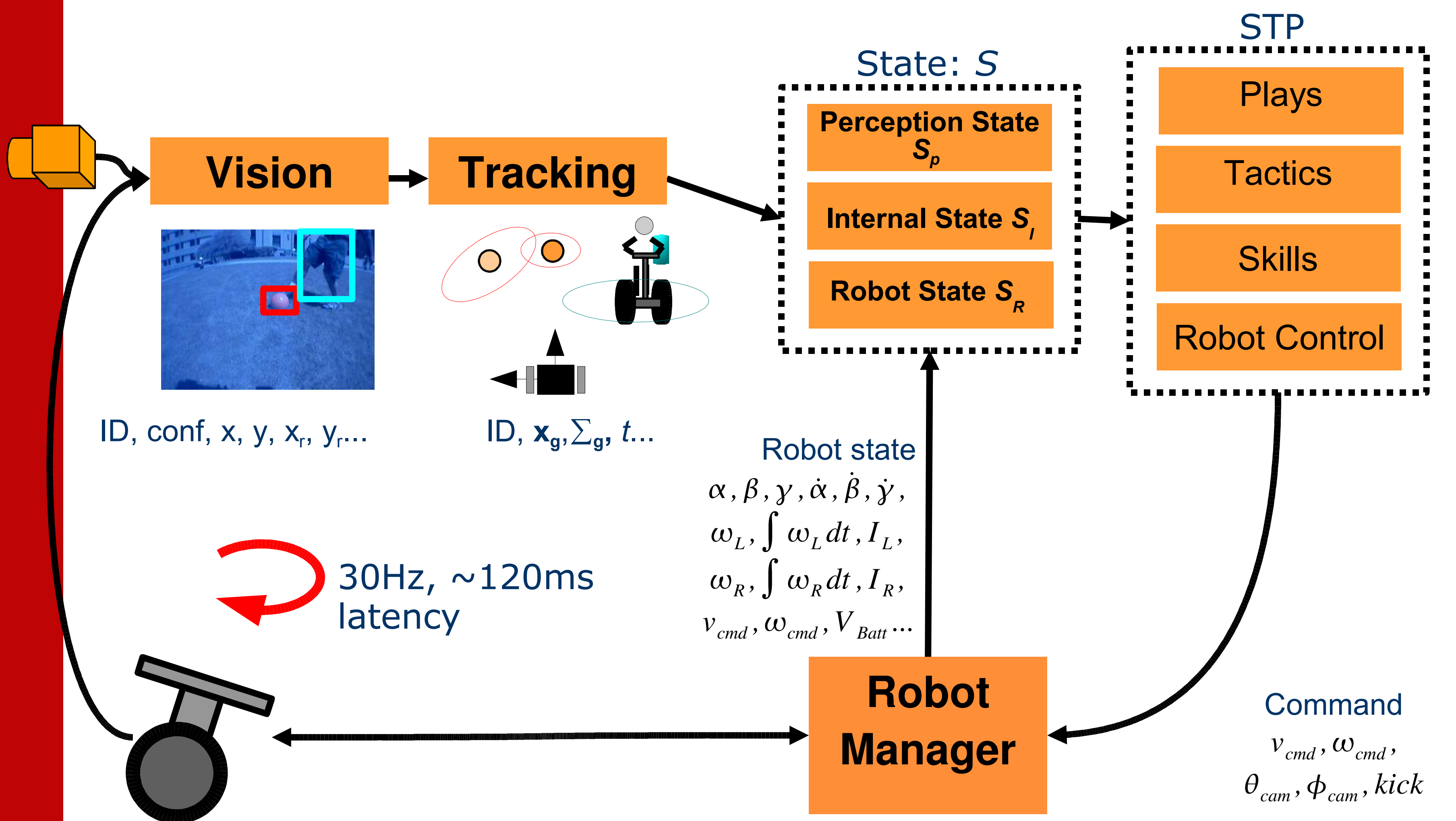


Skills, Tactics, Plays

- Skills -- encode complex low-level actions
- Tactics -- encapsulate single robot behavior
- Plays -- For adaptive team strategy



Robot Autonomy



Hierarchical Control

- Perception encapsulated in world state s
- Cognition broken into
  - A set of plays $\{P_{1,} P_{2,} \ldots, P_{N_t}\}$
  - A set of tactics $\{T_{1,} T_{2,} \ldots, T_{N_t}\}$
  - A set of skills $\{Sk_{1,} Sk_{2,} \ldots, Sk_{N_{sk}}\}$
  - Low level robot control: Navigation, motion control

Hierarchical robot control:
Plays use tactics,
Tactics use skills,
skills use robot control layer

STP World State

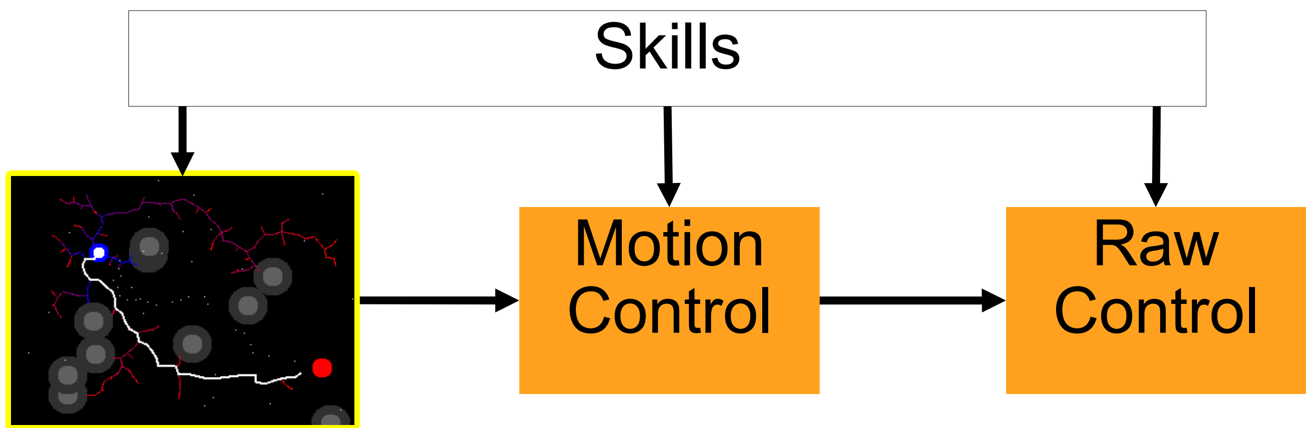
- State space *S* formed from
  - S_P Perception state – Vision and tracking output
  - S_R Robot state – Internal sensors within Segway RMP
  - S_I Internal state – Execution time,

$$S = S_P \times S_R \times S_I$$

Example vision processing

Robot Control Layer

- Provides low level action primitives for STP
- *Choice* of actions to use
  - Path planning – Using ERRT [Bruce & Veloso, 03]
  - Motion control – Trapezoidal control in velocity space
  - Raw control – Direct velocity commands to robot



Commands $(x_t, y_t, \theta_t, v_{max}, \omega_{max})^T$ $(x_t, y_t, \theta_t, v_{max}, \omega_{max})^T$ $(v_{cmd}, \omega_{cmd}, kick)^T$

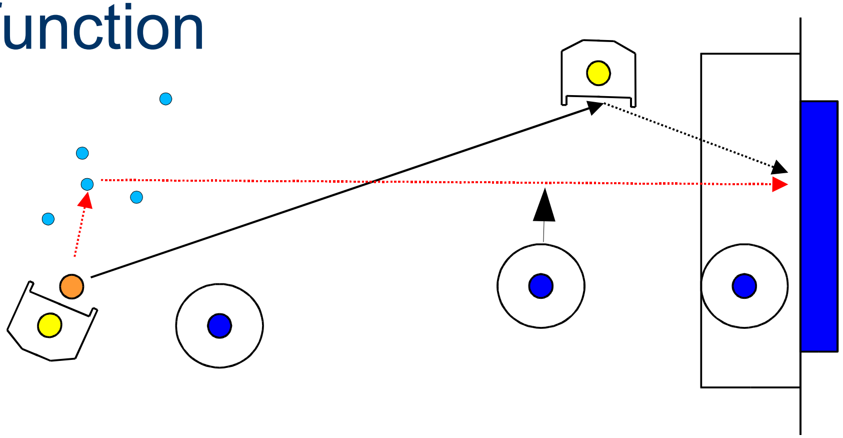
Skills

- Each skill Sk_i is a *focused* control policy
- Control policy $\pi_i : S \times A \times P \rightarrow [0,1]$
  - *A* is set of actions to robot control
  - *P* is set of parameters (set by tactics)
  - Defined as a stochastic policy, but usually just a deterministic policy
- Focused policy
  - Policy meaningful only in applicable states $S_{A_i} \subseteq S$
  - Policy only defined over meaningful states:

$$\pi_i(s) = 0 \quad \forall\, s \notin S_{A_i}$$

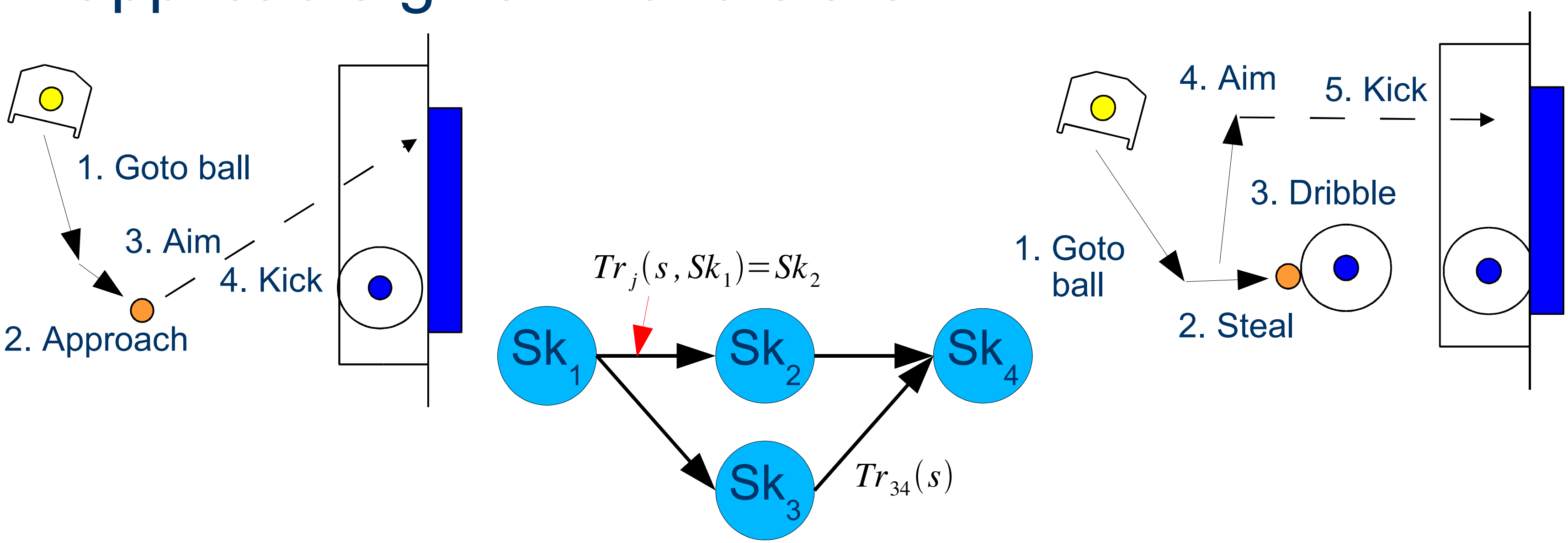
Tactics

- Each tactic T_j defines how skills will execute
- Consists of
  - A function to set parameters for executing skill
  - An augmented finite state machine (AFSM) of skills
- Tactic evaluation function:
  - Evaluated each decision cycle $Ev_j : S \times P \rightarrow [0,1]$
  - Common approach: Stochastic sampling of an objective function



Augmented Finite State Machine

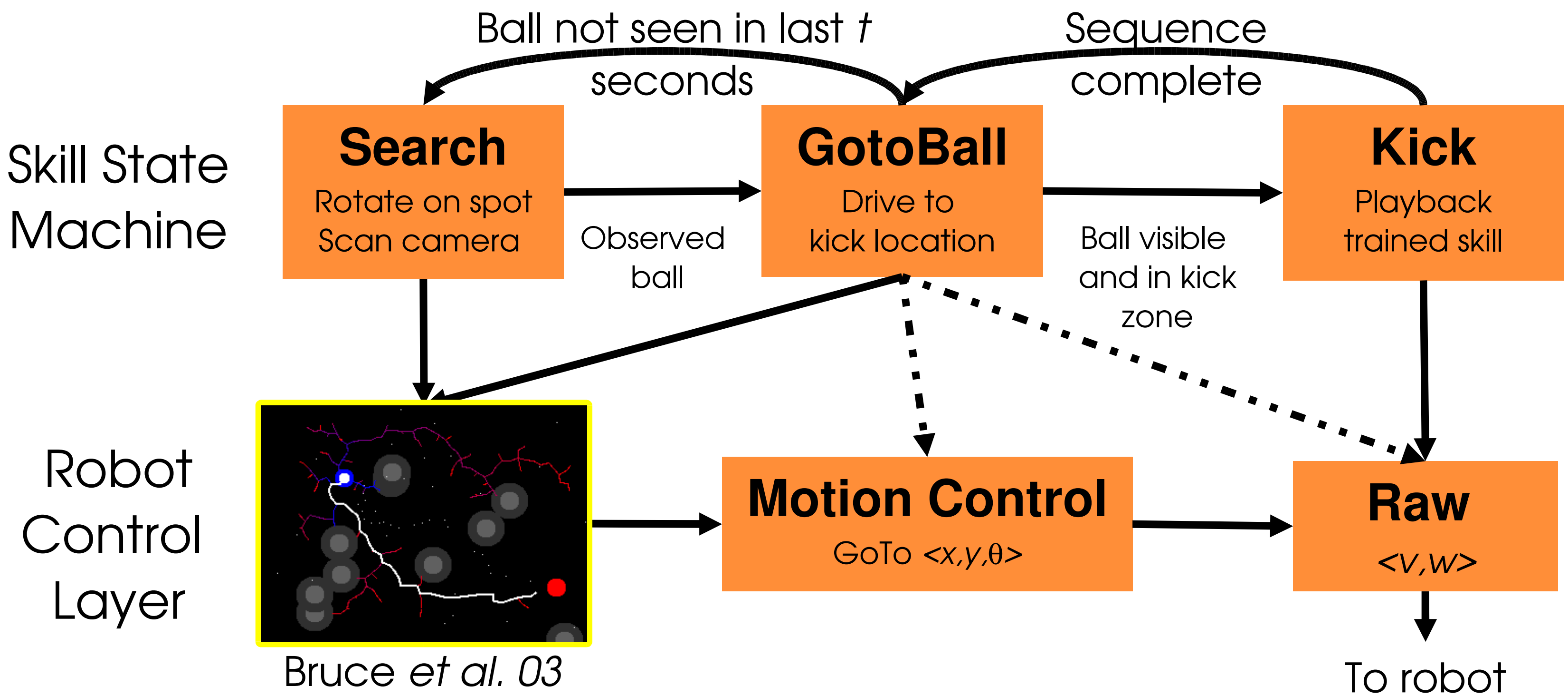
- Consists of currently executing skill
- Transition function $Tr_j : S \times Sk \rightarrow Sk$
  - Determines skill to execute in next decision cycle
- Transitions must ensure current skill is applicable given world state



Self transitions not shown

A Tactic/Skill Example

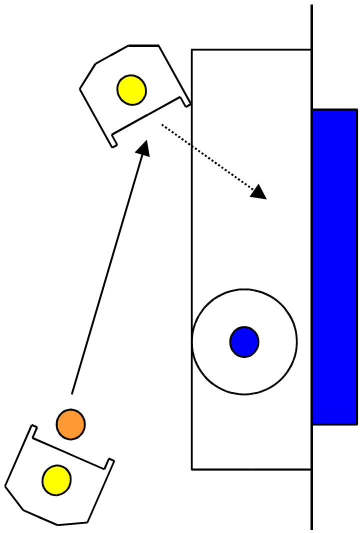
- *ChaseBall* for chasing and kicking balls



Hysteresis is essential to negate decision boundary problems caused by noise and uncertainty

Plays as Team Plans

- A play is a team *plan* with tactics as operators on the joint state
- Must allow for:
  - Flexible execution
  - Execution monitoring
  - Selective plays
  - Flexible parameterization
  - Easy play creation
  - Adaptation



Role 0
- Dribble to P_1
- Pass to R_2
- Wait for loose ball

Role 1
- Wait for Pass at P_2
- Receive Pass
- Shoot

[Bowling, Browing, Veloso, ICAPS 04]

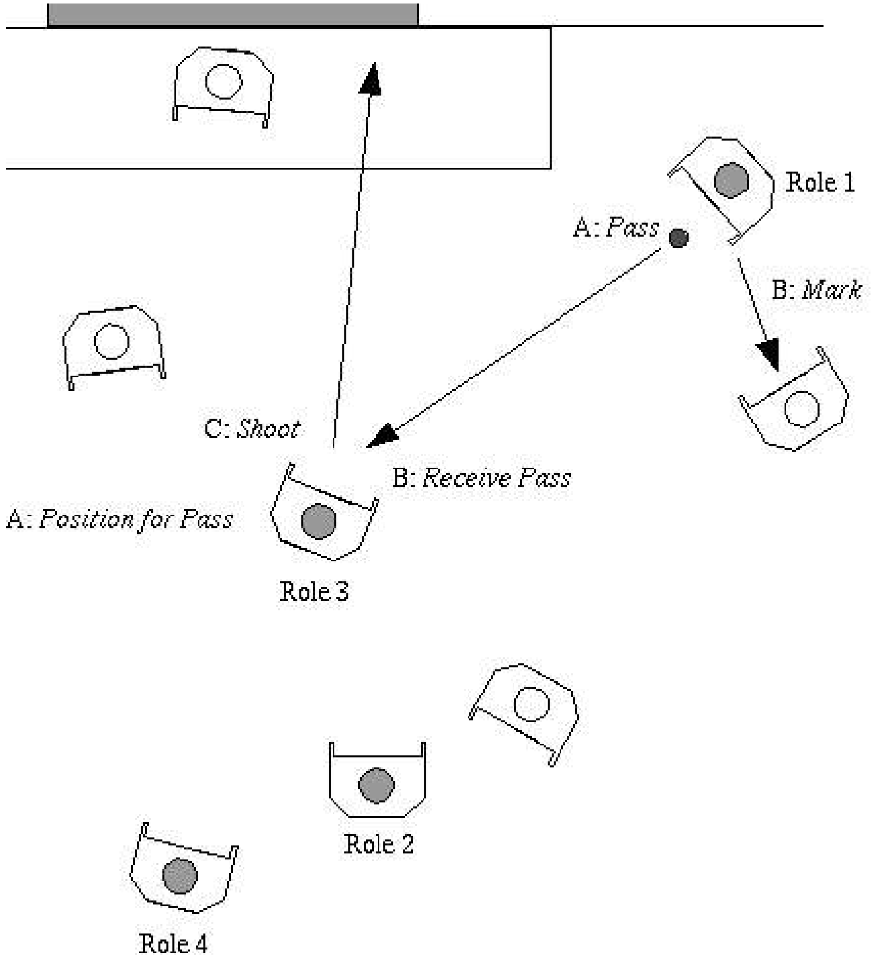
Elements of a Play

- Play language
  - Human understandable and new plays easily added
- Team plan
  - Sequence of single robot actions for each role
- Selective execution and execution monitoring
  - Pre-conditions for selection using boolean predicates
  - Termination conditions using boolean predicates
- Tactic and parameter definitions
  - *Actions* for plays
  - Flexible coordinate frames
- Role assignment
- Play adaptation

An Example Play

APPLICABLE offense
DONE aborted !offense

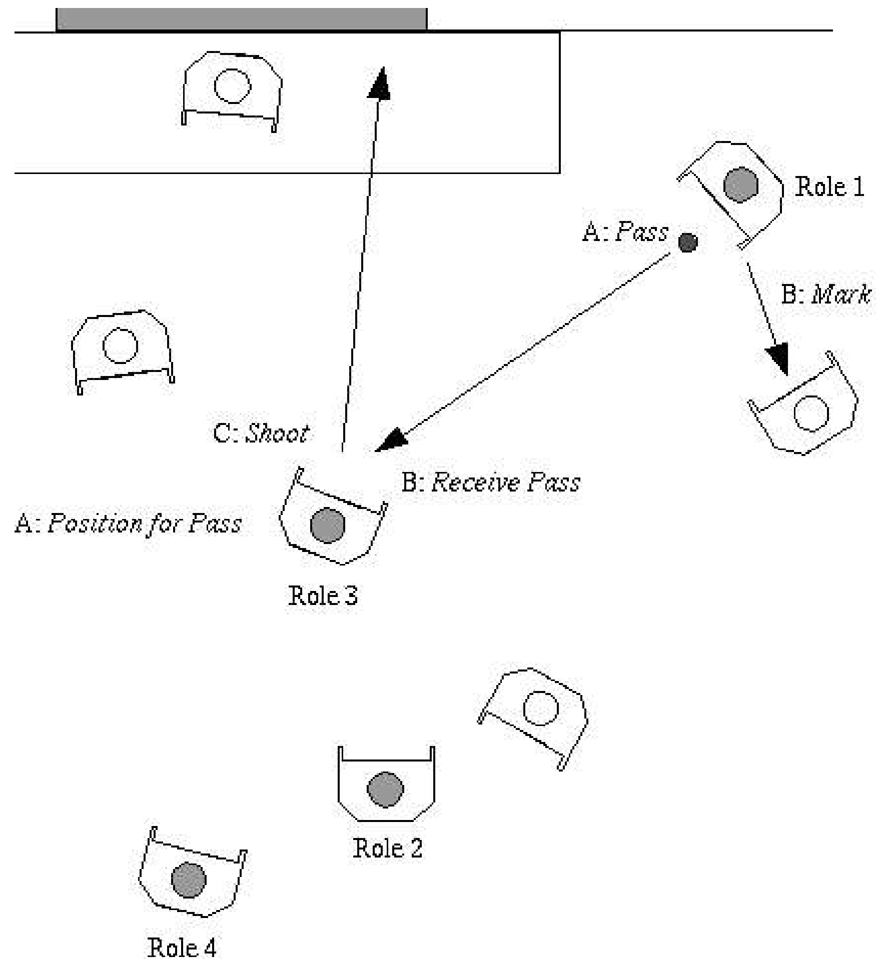
ROLE 1
pass 3
mark best_opponent
ROLE 2
block
ROLE 3
pos_for_pass R B 1000 0
receive_pass
shoot A
ROLE 4
defend_lane



An Example Play

APPLICABLE offense
DONE aborted !offense

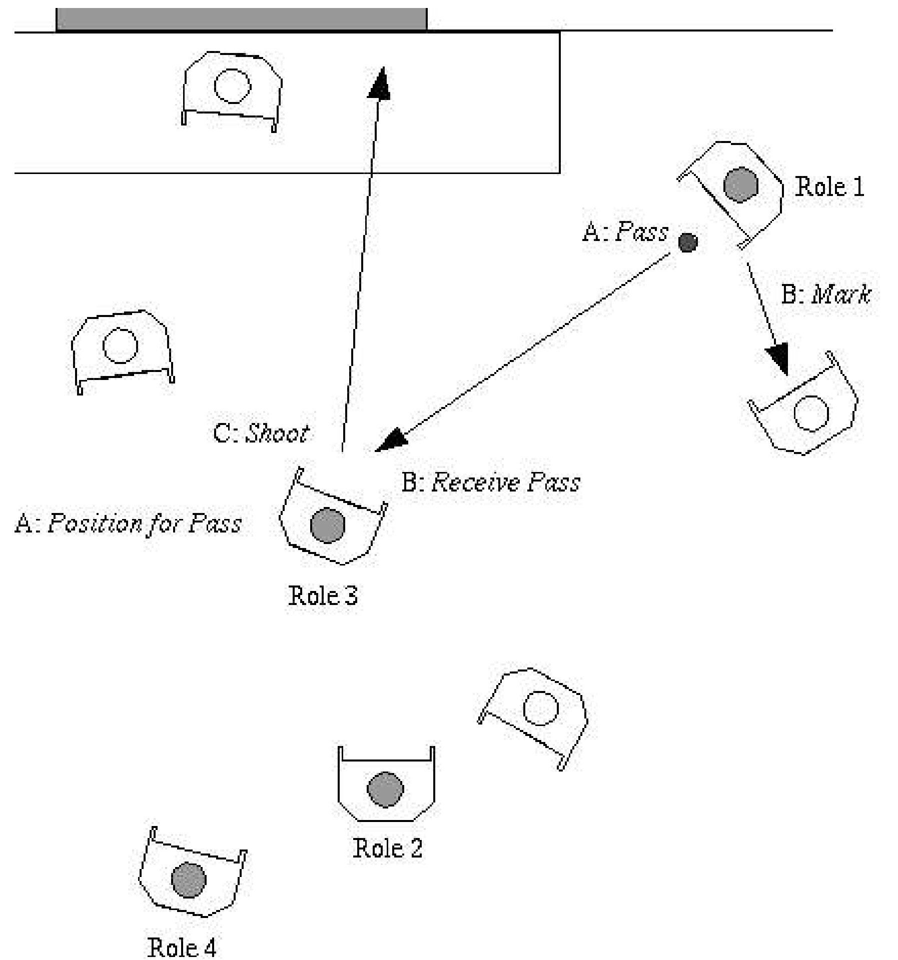
ROLE 1
pass 3 
mark best_opponent
ROLE 2
block
ROLE 3
pos_for_pass R B 1000 0
receive_pass
shoot A
ROLE 4
defend_lane



An Example Play

APPLICABLE offense
DONE aborted !offense

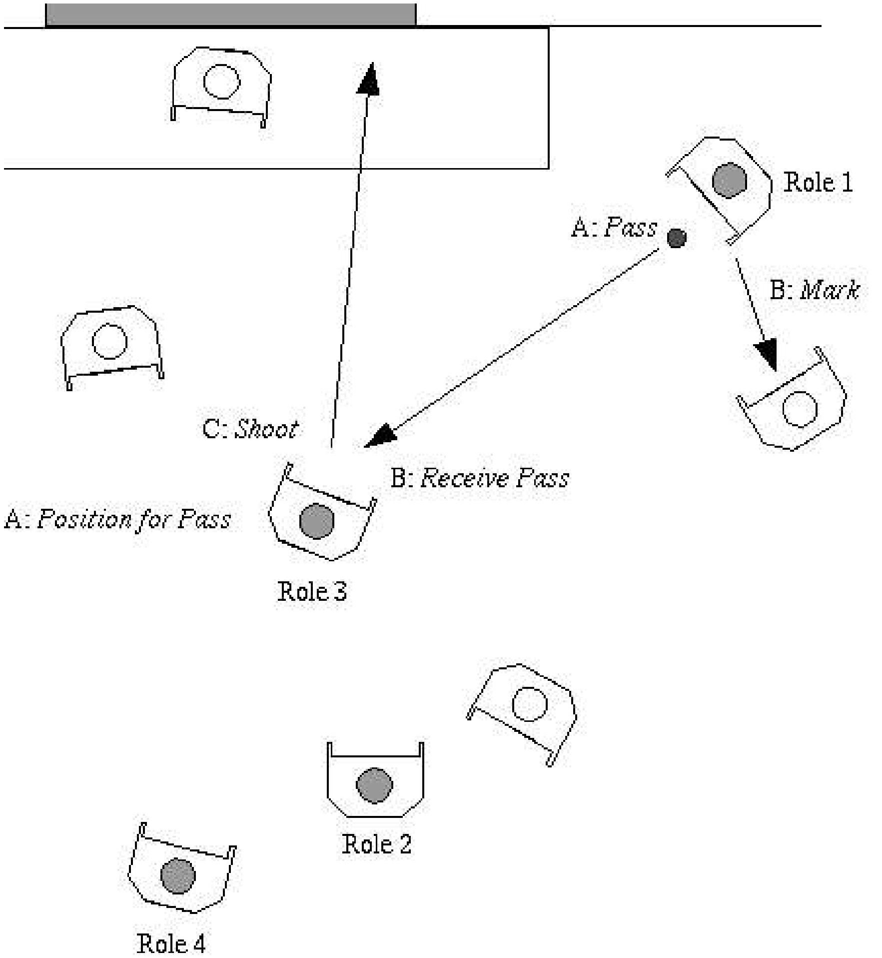
ROLE 1
pass 3
mark best_opponent
ROLE 2
block
ROLE 3
pos_for_pass R B 1000 0
receive_pass 
shoot A
ROLE 4
defend_lane



An Example Play

APPLICABLE offense
DONE aborted !offense

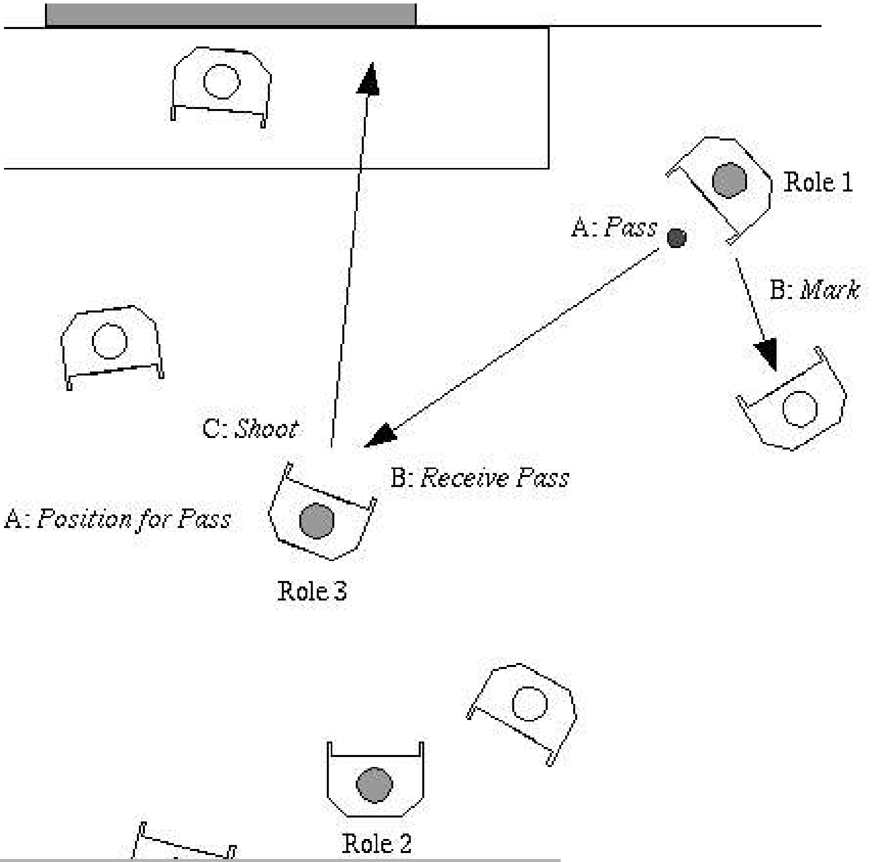
ROLE 1
pass 3
mark best_opponent
ROLE 2
block
ROLE 3
pos_for_pass R B 1000 0
receive_pass
shoot A 
ROLE 4
defend_lane



An Example Play

APPLICABLE offense
DONE aborted !offense

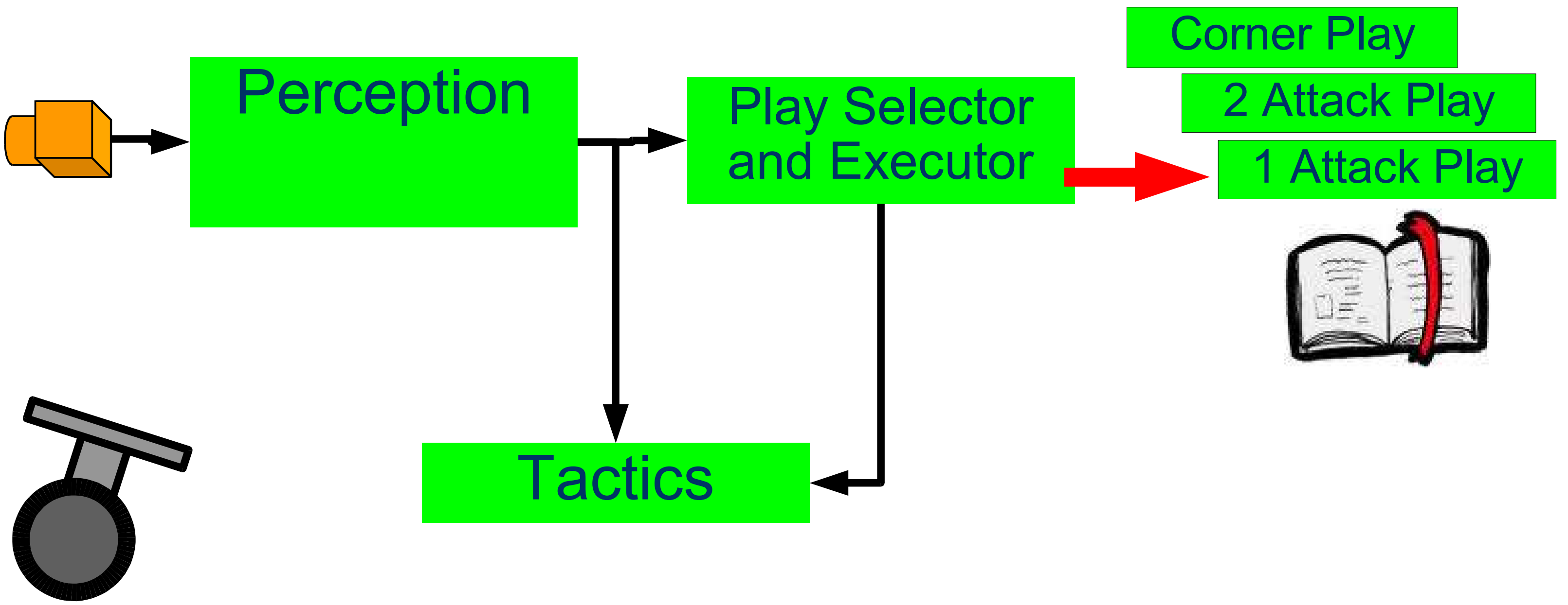
ROLE 1
  pass 3
  mark best_opponent
ROLE 2
  block
ROLE 3
  pos_for_pass R B 1000 0
  receive_pass
  shoot A
ROLE 4
 defend_lane



Assign reward on play termination

Playbook for Team Strategy

- Each play describes a course of action
- Multiple plays describe team strategy
- Stochastic selection mechanism
  - Adapt play selection likelihood based on performance



Adaptive Play Selection

- Plays selected stochastically from applicable set P_A
  - Each play p_j has an associated weight w_j
  - Probability of selection:

$$P(p^t = p_j) = \frac{e^{w_j^t}}{\sum_{p_i \in P_A} e^{w_j^t}}, \quad \forall p_j \in P_A$$

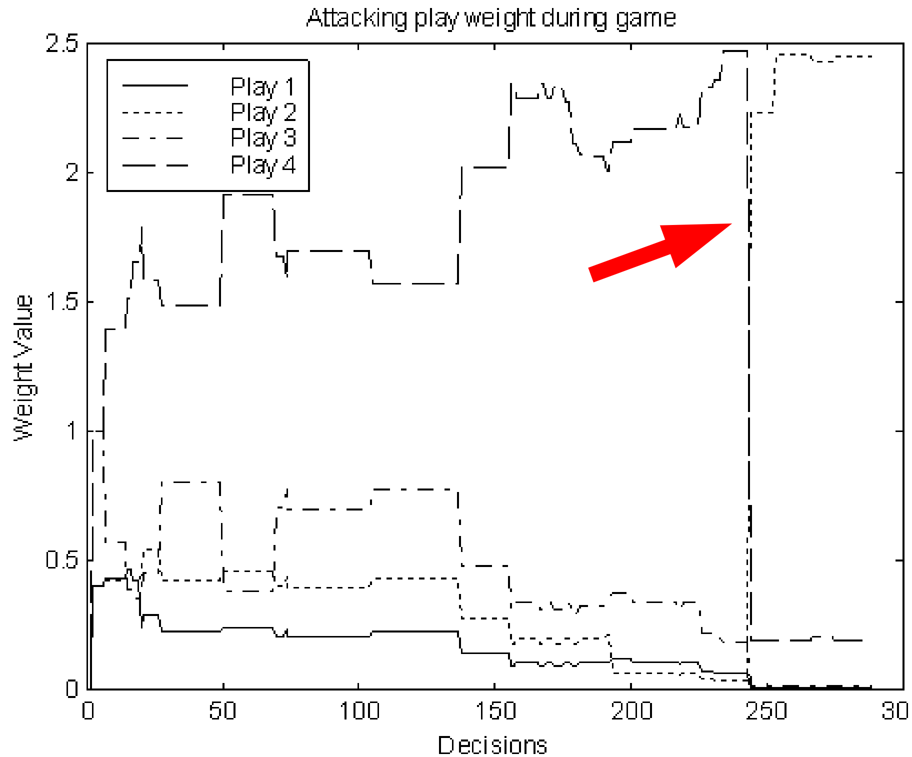
- Play weights adapted based on performance

$$\tilde{w}_j^t = w_j^{t-1} (K_j^t)^{1/P(p^t = p_j)}$$
$$w_j^t = \frac{\sum_{p_i \in P_A} w_i^{t-1}}{\sum_{p_i \in P_A} \tilde{w}_i^t} \tilde{w}_j^t \qquad \forall p_j \in P_A$$

Play Performance

- Robust adaptation to opponents, without requiring opponent model
- Easy to develop new plays
- Allows experimentation in strategy without hurting team performance

- Centralized implementation
- Team performance only as good as best play
- Performance limited by underlying tactics and skills



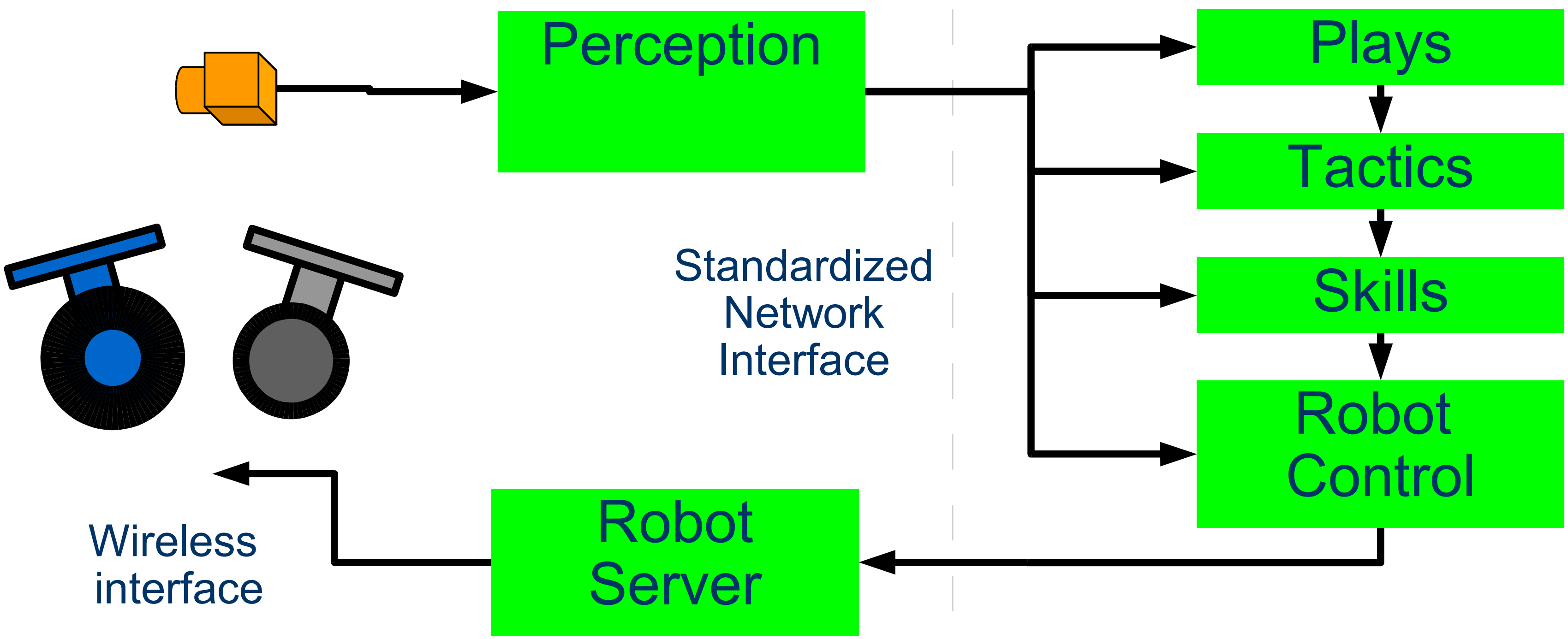
Offense Play Weights
[CMDragons vs RoboDragons, RoboCup 2003]

RoboCup Pickup Team

- Joint small-size pickup team with RoboDragons
  - Prof. Naruse, Aichi Prefectural University, Japan
- Features
  - RoboDragons have fast robots
  - CMDragons have good software
- Distance and language barrier make tightly coordinated development infeasible

Pickup Team Case Study

- Joint team
  - CMDragons software driving RoboDragons robots
  - Standardized vision, action interfaces



Team Performance

- Enabled joint team
  - Competitive game after 1 setup day
  - Major stumbling block was misunderstanding of rotation frame of reference
- Effectiveness
  - Won round robin group to reach semi-finals
  - Reached 4th place out of 20 teams overall
- Lessons
  - Standardized interfaces can enable pickup teams
  - Shared frames of reference are problematic
  - Flexible implementations are essential

Treasure Hunt Challenges

- Distributed challenges
  - Play selection
  - Role assignment
  - Monitoring of execution
  - Synchronized stepping requires communication
- Pickup team challenges
  - Agreement on tactic meaning
  - Agreement on plays
  - Dynamic sub-teaming

End of Talk

CORAL

cooperate - observe - reason - act - learn

http://www.cs.cmu.edu/~coral

http://www.cs.cmu.edu/~robosoccer/segway

Thank you!