# Towards Social Complexity Reduction in Multiagent Learning: The AdHoc method

Michael Rovatsos and Marco Wolf

{rovatsos,wolf}@informatik.tu-muenchen.de

Computer Science Department

Technical University of Munich

# Motivation

- Open multiagent systems (MAS)
  - ➡ necessity of modelling peer agents to achiev coordination
- Large-scale MAS imply only occasional encounters with acquainted peers
  - ➡ acquiring and maintaining information about *individual* agents hard and/or ineffici
- Is employing models of whole *classes/types* of other agents a solution?
- Objective: application of the principle of *social complexity reduction* to artificial agent societies

# Overview

1. **Introduction**
2. **The** ADHOC **Heuristic**
3. **Application to Multiagent IPD Games**
4. **Experimental Results**
5. **Conclusions & Outlook**

# Introduction

# Introduction

- In human societies reducing the number of models of others is a common phenomenon, e.g.
  - roles in organisations,
  - stereotypes,
  - legal regulations,
  - ...
- Learning opponent models is a prominent issue in multiagent learning
- Applying classification techniques to such opponent models has not received much attention
- Our aim: to combine opponent modelling techniques with classification

# Introduction (contd.)

- Advantages: Using a limited number of models
  - reduces computational cost,
  - is adequate for modelling *bounded rationality*,
  - speeds up learning of the models by increasing learning data.
- Assumptions:
  - no prior knowledge about others' goals or strategies,
  - other's strategies need not be fixed over time,
  - no benevolence assumptions, no common goals,
  - active, on-line learning during interaction.

# The ADHOC Heuristic

# The ADHOC heuristic

- ADaptive Heuristic for Opponent Classification
- Evolves up to $k$ opponent classes $\mathcal{C}$ with a (crisp) membership function $m : A \rightarrow \mathcal{C}$ for an arbitrary set of opponents $A$.
- Assumes an underlying Opponent Modelling Method (OMM) that returns an opponent model $OM(c)$ for each class and which
  - is capable of adequately describing the opponent's behaviour,
  - allows for computing the similarity $S(a, c)$ between peer $a$ and opponent class $c$,
  - makes it possible to determine an optimal behaviour *towards* class $c$.

# The AdHoc heuristic (II)

The heuristic

- processes data observed during encounter $e = \langle (s_0, t_0), \ldots (s_l, t_l) \rangle$ with utility $u_i(s_l, t_l)$ for the modelling agent $a_i$,
- determines the optimal class for $a_j$ while constantly updating similarity values $S(a_j, c)$ for *all* classes,
- adapts the model of a class through use of the OMM for agent $a_j$ in case of weak similarity,
- handles similarity values in case of model adaptation.

Central sub-procedure: OptAltClass

# The ADHOC heuristic (III)

**procedure** OPTALTCLASS$(a_j, e, \mathcal{C}, k, b)$

**if** $\mathcal{C} \neq \emptyset$ **then**

$\mathcal{C}_{max} \leftarrow \{c | S(a_j, c) \text{ maximal } \wedge S(a_j, c) > b\}$

**if** $\mathcal{C}_{max} \neq \emptyset$ **then**

**return** $\arg\max_{c \in \mathcal{C}_{max}}$ QUALITY$(c)$

**else**

**if** $|\mathcal{C}| < k$ **then**

**return** NEWCLASS$(e)$

**else**

**return** OPTALTCLASS$(a_j, e, \mathcal{C}, k, -\infty)$

**end if**

**end if**

**else**

**return** NEWCLASS$(e)$

**end if**

# The ADHOC Heuristic (IV)

- Heuristic "quality" function in our implementation:

$$\begin{aligned} \textsc{Quality}(c) &= \alpha \cdot \frac{\#\mathit{CORRECT}(c)}{\#\mathit{ALL}(c)} + \beta \cdot \frac{\#\mathit{corr}}{\#\mathit{all}(} \\ &+ \gamma \cdot \frac{\#\mathit{agents}(c)}{\#\mathit{known\_agents}} \\ &+ (1 - \alpha - \beta - \gamma) \cdot \frac{1}{\textsc{Cost}(\textsc{c})} \end{aligned}$$

- OPTALTCLASS is used throughout the top-level heuristic to find the most appropriate class for a peer $a_j$

# The AdHoc Heuristic (V)

**procedure** AdHoc($a_j, e, k$)

$\forall c \in \mathcal{C}.S(a_j, c) \leftarrow correct(a_j, c) \div all(a_j)$

**if** $m(a_j) = \perp$ **then** $m(a_j) \leftarrow$ OptAltClass$(a_j, e, \mathcal{C}, k, 1)$ **else**

**if** $m(a_j)$ doesn't predict $e$ correctly **then**

**if** $S(a_j, m(a_j)) \leq \delta \vee m(a_j)$ is very stable **then**

$m(a_j) \leftarrow$ OptAltClass$(a_j, e, \mathcal{C}, k, \rho_1)$

**end if**

$c' \leftarrow$ OptAltClass$(a_j, e, \mathcal{C}, k, \rho_2)$

**if** $c' \in \mathcal{C} \wedge c'$ is very stable **then** $m(a_j) \leftarrow c'$

Om-Learn$(m(a_j), e)$

**if** $m(a_j)$ has been modified **then**

$\forall m(a') \neq m(a_j).S(a', m(a_j)) \leftarrow 0$

**end if**

**end if**

**end if**

# The AdHoc Heuristic (VI)

Control flow during encounters:

1. Action selection:
   (a) If $a_j$ is encountered for the first time, OptAltClass is called *after each turn* to determine the most suitable class.
   (b) Else, $OM(m(a_j))$ is used throughout the encounter.
2. After the encounter is over, the classification procedure is called.
3. Empty classes are erased from $\mathcal{C}$.

# Application to Multiagent IPD Games

# Application scenario (I)

- Well-understood application example: Iterated Prisoner's Dilemma games.
- Payoff matrix:

| $a_i$ \ $a_j$ | C | D |
|---|---|---|
| C | (3,3) | (0,5) |
| D | (5,0) | (1,1) |

- Simulations consist of fixed-length IPD games between randomly matched agents on a toroidal grid (with random agent movement).

# Application scenario (II)

Apply a combination of the $US - L^*$ algorithm [Carmel & Markovitch, 1996] and Q-Learning [Watkins & Dayan, 1992] as OMM:

- Model opponents as deterministic finite automata (DFA): transitions depend on own actions, states are labeled with other's actions
- Lookup table for Q-values uses DFA states as MDP states
- Boltzmann exploration:

$$P(z) = \frac{e^{Q(s,z)/T}}{\sum_{z'} e^{Q(s,z')/T}}$$

# Application scenario (III)

Properties of the opponent modelling method:

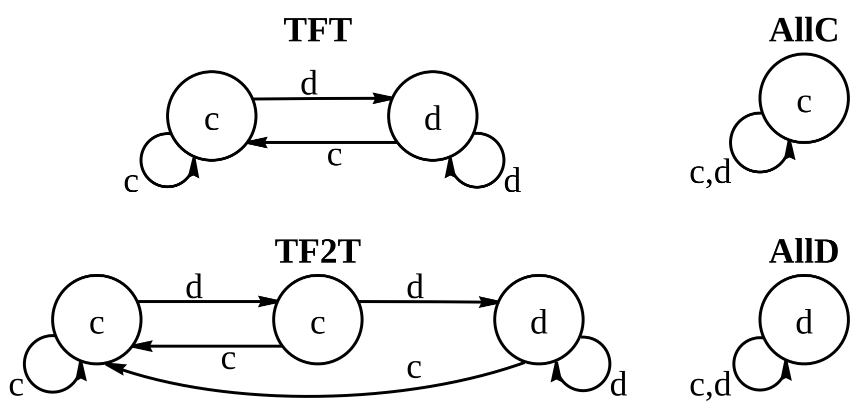
- Guaranteed to converge to a DFA consistent with the peer automaton
- Can be easily combined with RL methods
- Similarity function is easy to define by checking correct prediction of an interaction sequence
- Models can*not* be improved incrementally
- Limited expressiveness, esp. it does *not* cater for non-deterministic behaviour

# Experimental Results

# Experimental results (I)

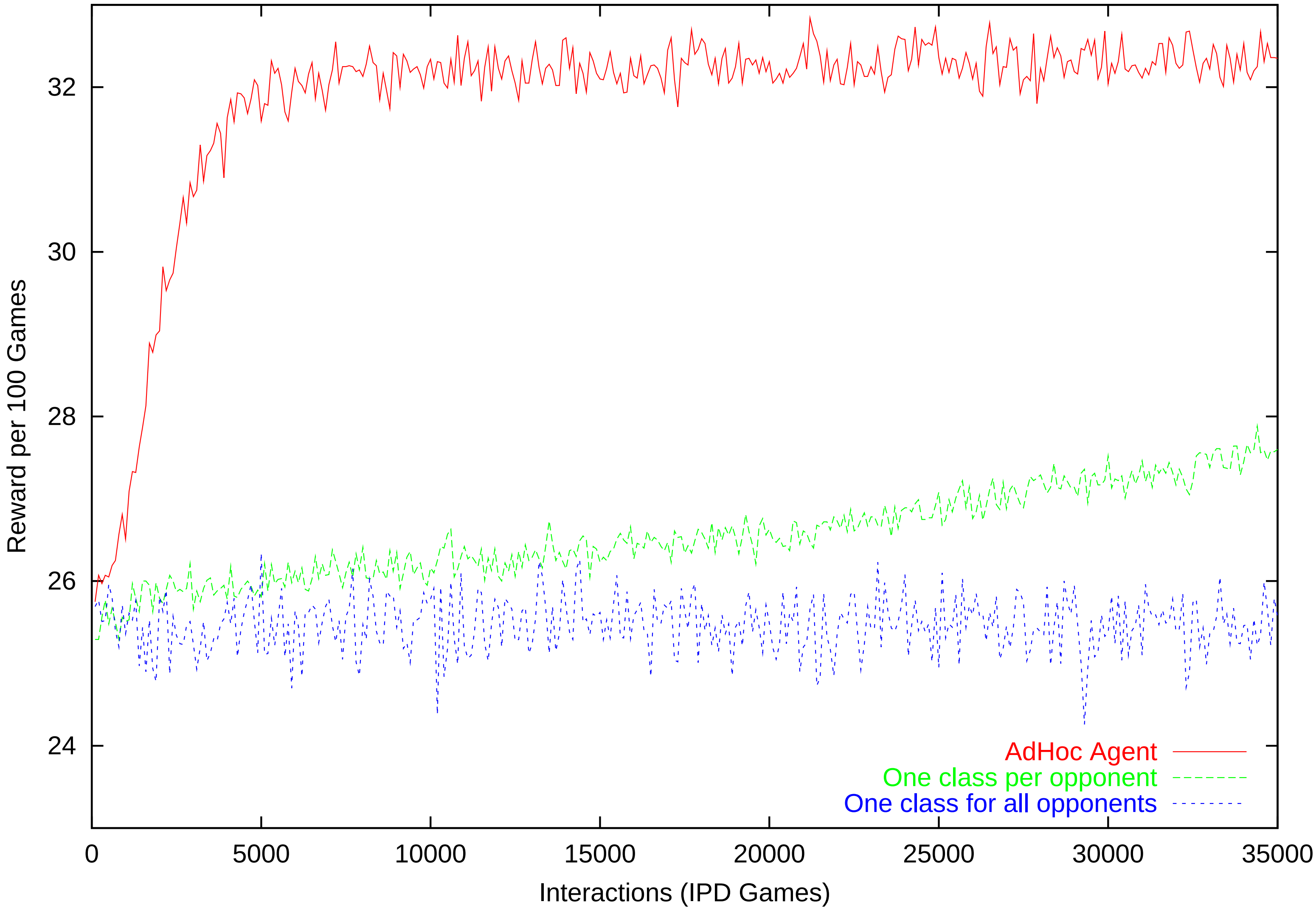
Simulation settings:

- ADHOC agents play against fixed-strategy opponents with the following (and random DFA) strategies:

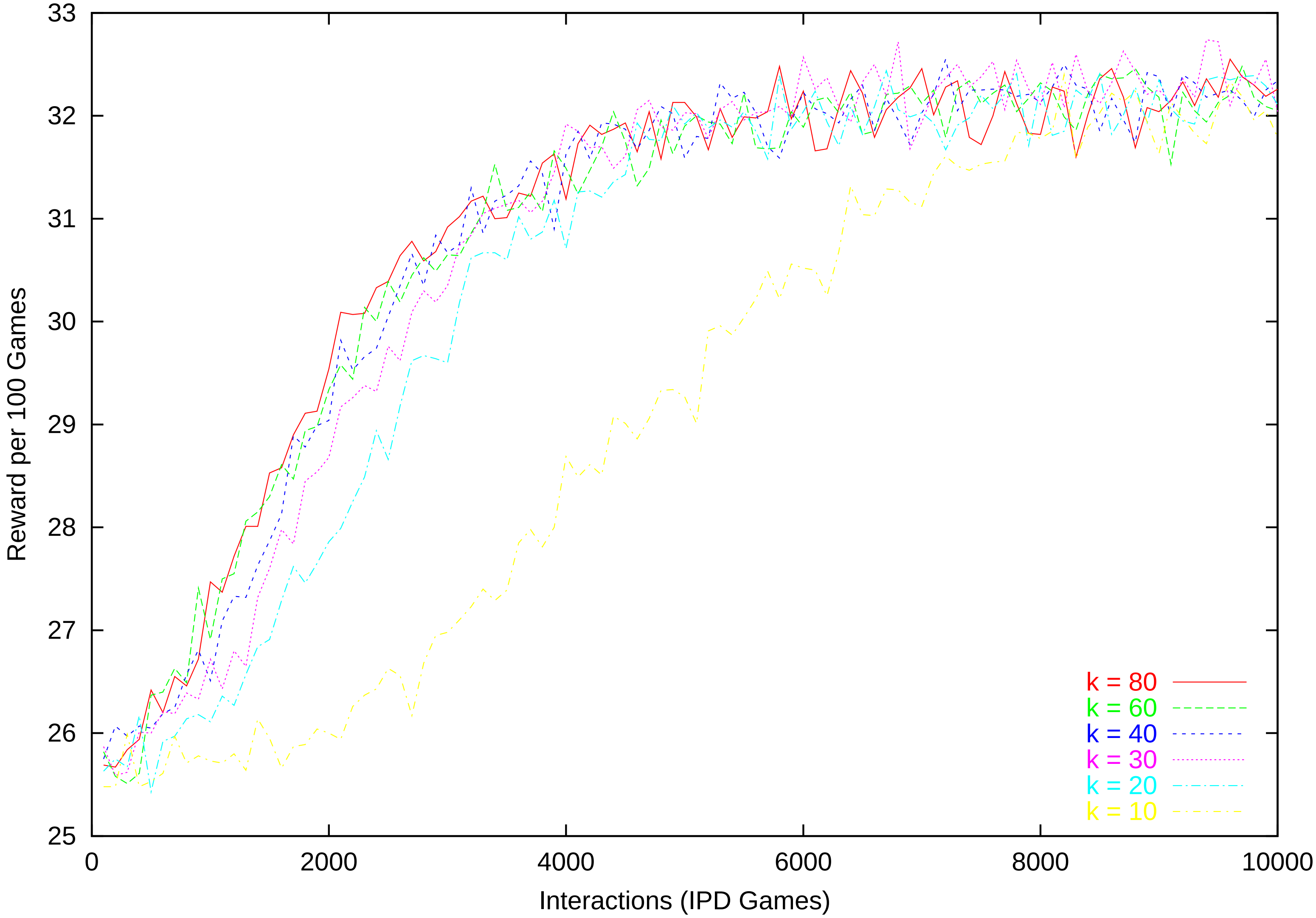



- Typically, we increase population from 80 to 200 agents over time (tested populations up to 1000)
- Store 6 encounter samples per class for DFA learning

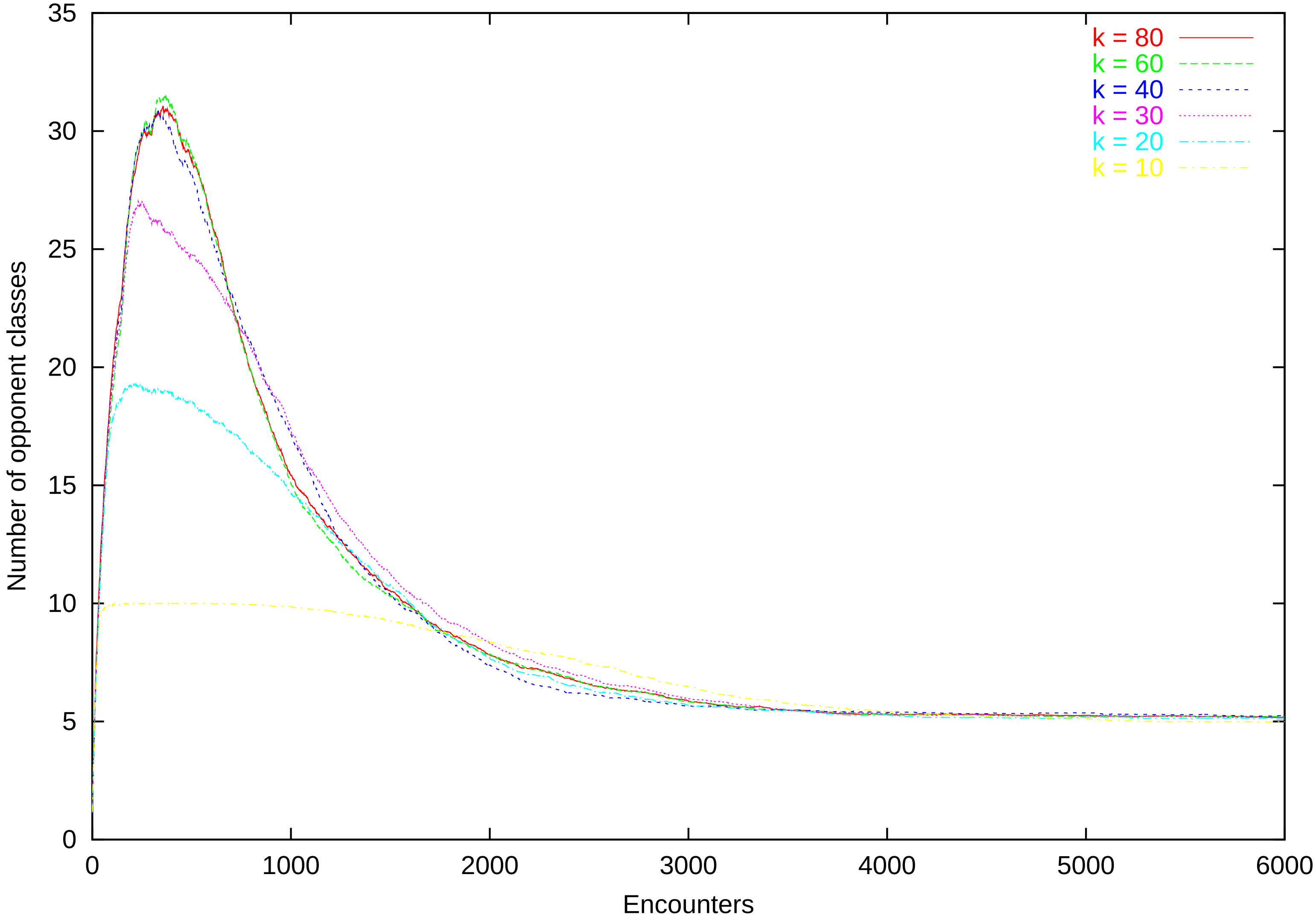
# AdHoc **Performance**



# Different $k$ values – Rewards



# Different $k$ values – Classes



# AdHoc vs. AdHoc

- AdHoc agents do well against fixed-strategy opponents
- They learn faster than “unboundedly rational” agents
- They can manage with low values of $k$

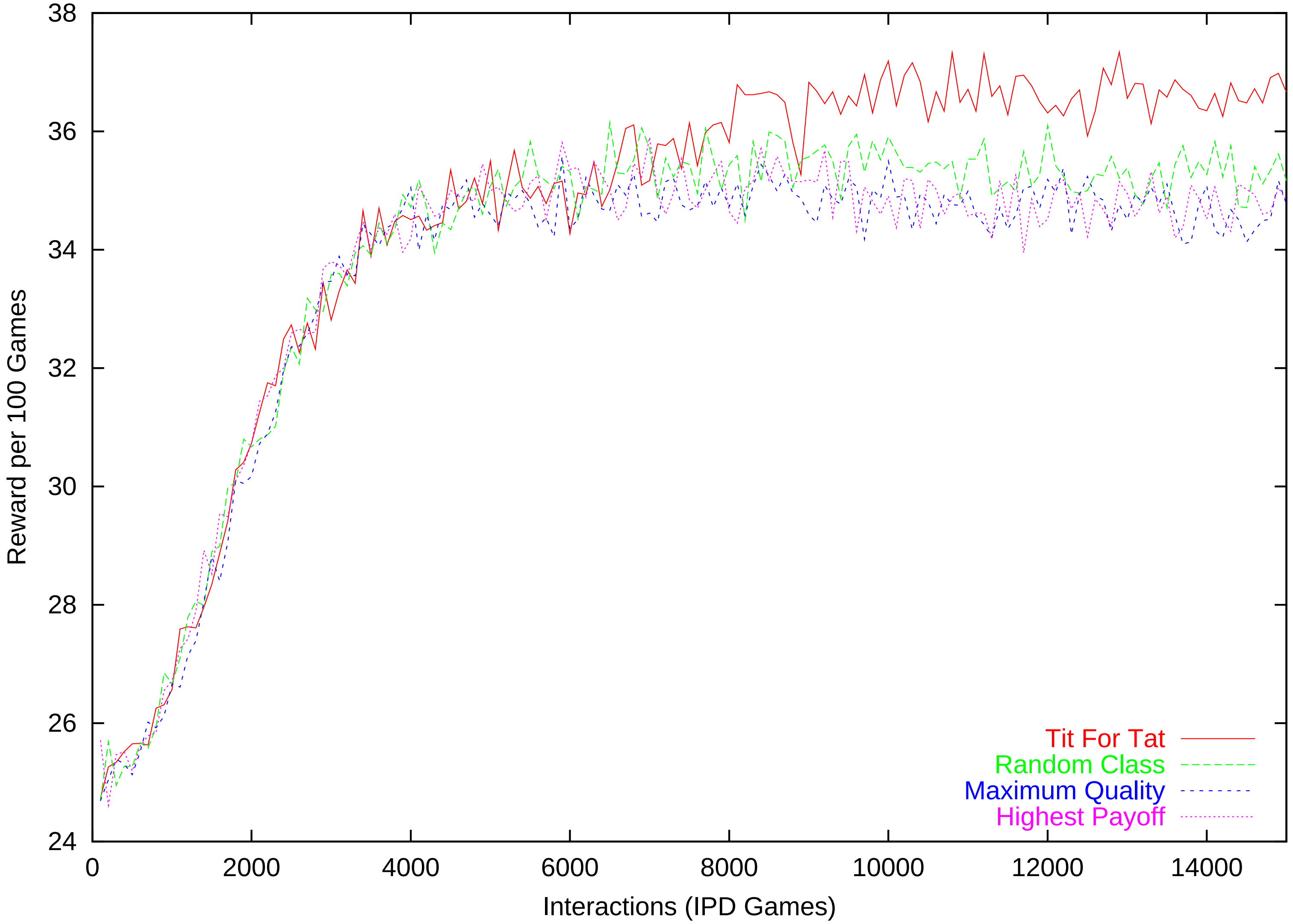
***but***

- Playing against other AdHoc agents causes random behaviour
- Reason: learning agents cannot be represented by DFA

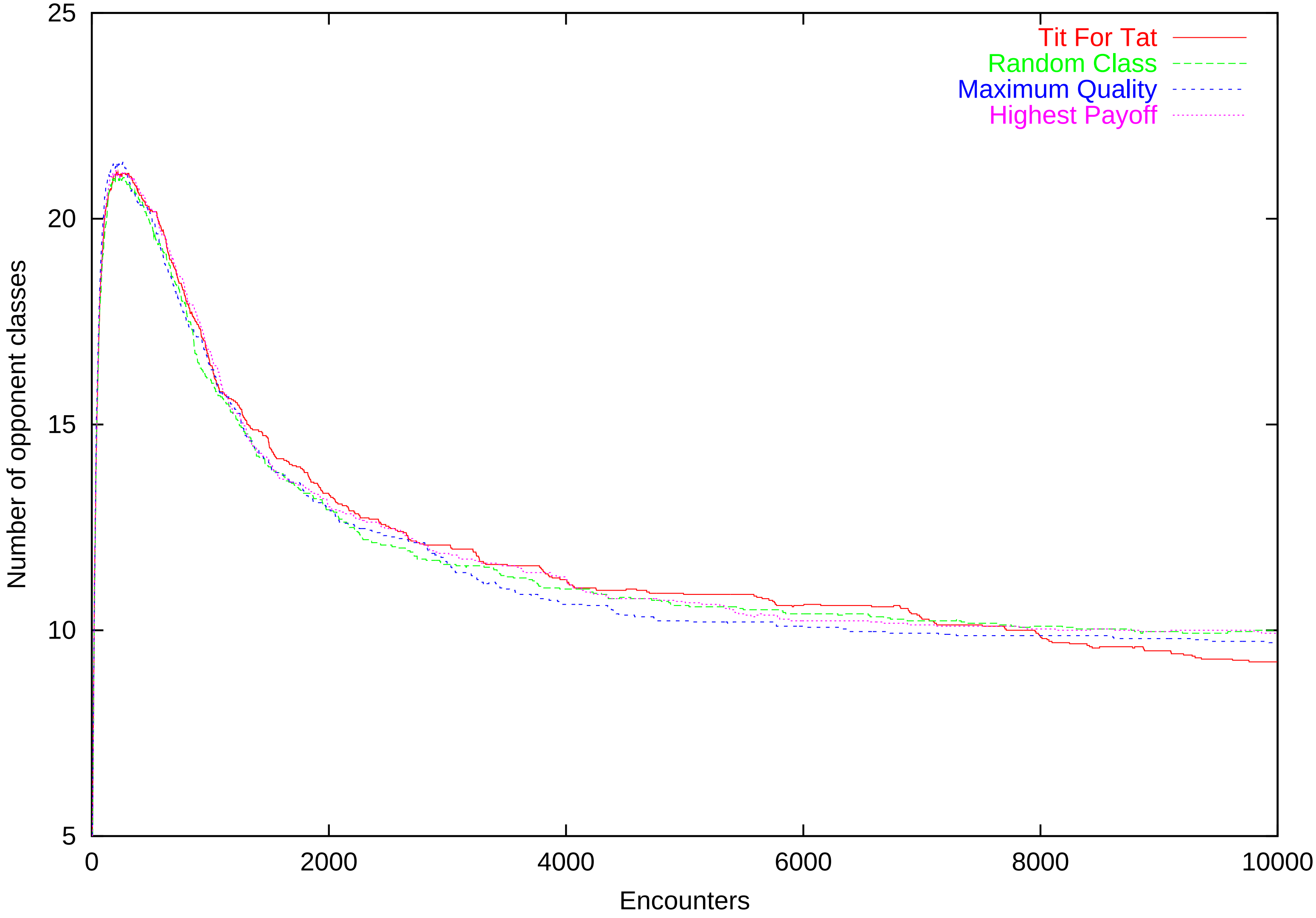
# AdHoc vs. AdHoc

- Extend the heuristic to detect random opponent behaviour
- In that case, switch to a fixed DFA according to which the agent will play (for a limited period of time)
- Strategies for selecting this DFA:
  - use a hard-coded strategy (e.g. TIT FOR TAT),
  - choosing a random automaton from $\mathcal{C}$,
  - choosing the DFA with maximum expected payoff/quality.
- Results rather unsatisfying

# AdHoc vs. AdHoc – Rewards



# ADHOC vs. ADHOC – Classes



# Conclusions

# Conclusions

- Achieved *social complexity reduction* by applying a simple classification heuristic
- Leads to considerable speed-up in the convergence of models
- Seamless integration of learning, classification and strategic reasoning
- Paves the way for boundedly rational yet effective adaptation in large-scale open multiagent systems
- Problem: adaptive, classifying agents cannot be represented with the opponent modelling method used here

# Outlook

# Outlook

- Model adaptation *during* encounters
- Richer interaction scenarios (esp. partner selection and context-dependent strategy choice)
- Explore possibilities mixed or fuzzy class membership functions
- Use of explicit communication for higher-level coordination
- Alternative opponent modelling methods to cope with ADHOC vs. ADHOC (incremental, probabilistic)

# Thank you for your attention!