Intent inference and strategic escape in multi-robot games with physical limitations and uncertainty

Aris Valtazanos
School of Informatics, University of Edinburgh
a.valtazanos@sms.ed.ac.uk

Subramanian Ramamoorthy
School of Informatics, University of Edinburgh
s.ramamoorthy@ed.ac.uk

Abstract— **Many multi-robot decision problems present autonomous agents with a dual challenge: the accurate egocentric estimation of the *state* and *strategy* of their adversaries, in the face of physical limitations and sensory uncertainty. Although these are clearly difficult constraints on the capabilities of an autonomous robot, this is also an opportunity for exploiting the corresponding limitations of the adversary. In this paper, we propose a decision making framework for physically constrained multi-robot games, using a combination of probabilistic and game-theoretic tools. We first present the *Reachable Set Particle Filter*, an adversary state estimation algorithm combining data-driven approximation with dynamical constraints. Then, we use game-theoretic notions to formulate a strategy estimation framework that progressively learns and exploits the adversary's behaviour. We evaluate our framework in a series of robotic soccer games between robots with varying sensing and strategic capabilities. Our results demonstrate that the combination of probabilistic modeling and strategic reasoning leads to significant improvements in performance robustness, while flexibly adapting to dynamic adversaries.**

I. INTRODUCTION

Even as the Autonomous Robotics community pushes the frontier of what is possible by a robot requiring decreasingly less guidance from any external sources, we realise that some of the most exciting opportunities are to be found in a middle ground where an autonomous robot interacts with other agents - including people - in a mixed-initiative or social setting. However, we also find that such interactive behaviour - involving multiple objectives, constraints, ambiguity and incompleteness of knowledge - can be even more challenging than the fully autonomous scenario. Part of the reason for this is the difficulty of modeling sophisticated strategic interactions in a way that is both principled and practicable.

When these conceptual difficulties are coupled with more pragmatic considerations of hardware and processing power limitations, we find that even seemingly simple "intelligent" moves, such as passes and dribbles in a robotic soccer game, are scarce and often require explicit hand crafting of everything but a few open parameters. In such domains, autonomous robots must face uncertainty in their own egocentric beliefs, incompleteness and uncertainty in their knowledge of the strategies of their adversaries and physical limitations such as a very limited field of view using a mediocre camera. One could perhaps overcome hardware issues by investing in better equipment, but we believe that the research challenges are most pertinent at the intersection of these various concerns, which is surely what the personal and service robot of the near future must come to terms with.

Our focus in this paper is on the problem of robust strategic decision making in adversarial games with physical limitations in action and perception. We propose an approach to devising strategic interactive behaviours in autonomous robots, illustrated using the robotic soccer domain. We also argue that constraints need not be viewed only as a feature to be overcome or eliminated. In our domain, successful interactive behaviour often requires the exploitation of these constraints, leading to interesting forms of motion strategies. We develop a decision-making framework for physical multi-robot games, based on the following high level concepts:

- **Intent inference**: In the absence of a precise model of its adversary's strategic behaviour, a robot may approximate it using a finite collection of *intent templates*, each modeling a single *coarse behavioural class*. These templates are combined probabilistically into a single distribution that predicts the (re)actions of the adversary.
- **Escape strategies**: In a noisy, physically embodied, partially observable environment, robots may benefit from exploiting the observability limitations of their adversaries. We refer to such moves as *escape strategies*, as they seek to reduce the amount of information available to the adversaries and influence their decisions.
- **Probabilistic adversary state estimation**: Robots require a mechanism for 'filtering' their noisy observations of the adversary. We propose the *Reachable Set Particle Filter*, a probabilistic state estimation algorithm combining a formal characterisation of the dynamical constraints of a robotic system, with a data-driven estimation procedure. A reachable set characterises, *for all* instances of a class of strategies, the states that might be reached at *some* future point. Thus, it forms a powerful addition to the particle filter which, in the basic formulation, does not account for such constraints.
- **Regret minimisation**: In realistic games with uncertainty, robots can achieve the full benefit of strategic modeling only if they *adapt* to and *learn* from the actions of their adversaries. We use the notion of *regret minimisation* to infer online the effects of probabilistically selected intent templates, and adjust their distributions to reward retrospectively optimal strategies.

This work has taken place in the Robust Autonomy and Decisions group within the Institute of Perception, Action and Behaviour, School of Informatics. Research of the RAD Group is supported in part by grants from the UK Engineering and Physical Sciences Research Council (EP/H012338/1) and the European Commission (TOMSY). AV is supported by a doctoral studentship from the Research Consortium in Speckled Computing which is funded by the Scottish Funding Council as part of the Strategic Research Development Programme (R32329), and EPSRC as part of the Basic Technology Programme (C523881).

Our proposed framework brings together ideas from probabilistic modeling, game theory, and strategic reasoning, allowing for online decision making in adversarial robotic environments with physical constraints.

II. RELATED WORK

The particle filter [9] has become the popular tool for state estimation in uncertain environments, due to the ability to flexibly model arbitrary probability distributions. These algorithms have been used to estimate adversarial models from experience in strategic games such as poker [2]. A related concept is found in empirical games [10], where one attempts to extract strategic profiles in a data-driven fashion.

Interactive Partially Observable Markov Decision Processes (I-POMDPs) [8] were proposed as an extension to the popular POMDP framework [11] to incorporate strategic adversarial models. The Interactive Particle Filter [5] builds on this formalism by probabilistically estimating beliefs over other agents' intentions, whereas [15] considers adversarial models in complex problems such as money laundering. Extending such ideas to physically embodied robotics, a much more messy domain, is a current challenge.

Plan recognition is concerned with the classification of an agent's actions into a pre-defined library of plans. In [1], a decision tree is used to process multi-featured observations for a simulated soccer game. Recognition algorithms based on particle filtering [4] and velocity tracking [13] have been proposed, although they do not consider strategic uncertainty. Probabilistic plan recognition [6] deals with actions for hierarchically decomposed plans in adversarial domains.

Our work shares a similar motivation with particle filters, I-POMDPs, and plan recognition, but seeks to extend them in two ways. First, we *decouple* the uncertainty arising from noisy sensing from *strategic* uncertainty. Sensing uncertainty is handled by the Reachable Set Particle Filter, which uses the adversary's dynamics as a prior for data-driven estimation, whereas strategic uncertainty is modeled through a combination of *intent filtering* and *regret minimisation*. Second, we augment intent filtering with strategic concepts suitable for physical adversarial games, such as escape strategies.

Hybrid systems provide a close link between optimal control and game theory. A notable line of work models challenging problems, such as aircraft collision avoidance, as pursuit-evasion (PE) games between adversaries [18]. Probabilistic PE games have also been considered to model reactive adversarial games between unmanned vehicles [19]. Visibility-based PE with limited field of view [7] is compatible with our notion of escape strategies, though the focus there is mostly on space coverage against non-strategic adversaries. More recent approaches propose tractable sampling-based solutions [12] to formally defined PE games, or seek Nash equilibria in such contexts [3].

Many multi-robot games require agents to vary their strategies over time, so they cannot be reduced to PE games that assume fixed strategies. Furthermore, full modeling of sensing limitations is often beyond the scope of the above mentioned PE literature.

Regret minimisation [16] is a game-theoretic method to determine the utility of actions against adversaries with unknown strategies. A popular application of regret minimisation is the bandit problem [17], which models decision making as a set of actions of initially unknown utility. We extend this concept for multi-robot games, where uncertainty and observability limitations pose a major constraint.

III. METHOD

We are interested in the problem of decision making by an autonomous robot with *physical limitations*. Such limitations include: limited velocities (including, perhaps, nonholonomy), noisy locomotion, noisy perception with limited sensing resources, limited methods of object manipulation (e.g. kicking a ball to a specific point). We develop a framework for the *robotic soccer* domain, as it features the above types of uncertainty, together with strategic adversaries of unknown capabilities. However, the underlying ideas could be extended to other general forms of interactions.

A. Preliminaries

1) Notation: We consider a game involving a total of N autonomous robots; let r_i refer to the i-th robot, $i = 1..N$. Furthermore, let $s \sim D$ be an abbreviation for drawing a sample s from a set or distribution D, let `rand` be a randomly generated number between 0 and 1, and let $\texttt{dist}(P_1, P_2)$ denote the Euclidean distance between points P_1 and P_2 . We consider **discrete time, continuous space** decisions (at time instants t), though in principle our framework can be extended to accommodate continuous time.

2) State Estimation: As most autonomous robots are restricted to egocentric sensing, they compute the states of the other robots *relative* to their own coordinate frame. For r_j, the collection of relative states of all other robots at time t gives the set of **robot beliefs**:

$$\mathcal{RB}_{j,t} = \{\ \langle x^i_j,\ y^i_j,\ \theta^i_j,\ c^i_j \rangle_t \ \mid\ i = 1..N, i \neq j\ \} \qquad (1)$$

where $\langle x^i_j,\ y^i_j,\ \theta^i_j \rangle_t$ denotes the relative state of r_i as computed by robot j, in terms of planar coordinates x, y and orientation θ, and $0 \leq c^i_j \leq 1$ is a weight representing the robot's *confidence* on the belief. At the simplest level, the position component $\langle x^i_j,\ y^i_j \rangle$ of a belief is equal to a raw sensor reading, whereas the orientation is inferred from a history of positions (see Section III-A.3). Correspondingly, relative soccer ball beliefs are given by $\mathcal{BB}_{j,t} = \langle x^B_j,\ y^B_j,\ c^B_j \rangle_t$. If the ball or a robot is visible at time t, its confidence weight is set to 1, otherwise it is set to the weight of time $t-1$ multiplied by a decay constant δ_c, $0 \leq \delta_c \leq 1$.

3) Orientation Estimation: Robots endowed with some sensing mechanism (vision and/or sonar) may approximate the relative planar positions of their adversaries. Unfortunately, this approach does not extend to the relative orientations[1]. Instead, we use the autoregressive procedure INFERORIENTATION (Appendix A) to compute orientations based on past beliefs. The algorithm infers an adversary's orientation by relating the flow of its motion to the position of the ball and the other robots.

[1] Unless sophisticated visual recognition algorithms are used, whose complexity would be prohibitive for the real-time decision making problems we are considering, and the kinds of robots we are targeting this at.

B. The Reachable Set Particle Filter

We have developed a variant of the original particle filter algorithm [9] for autonomous robots, which we term **Reachable Set Particle Filter (RSPF)**. The main innovation is the definition of the proposal distribution for particle updates in terms of *backward reachable sets* [18]. If the dynamics of a system of robots are known, together with their corresponding velocity bounds, then it is possible to compute future sets of states up to a - potentially infinite - time horizon. The worst-case backward reachable set $\mathcal{BRS}$ for r_i relative to r_j (assuming both robots are moving with their maximum linear velocities v_i and v_j) is obtained through the Hamilton-Jacobi-Isaacs Partial Differential Equation:

$$\frac{\partial v(q,t)}{\partial t} + \min[0, H(q, \nabla v(q,t))] = 0,\ v(q,0) = g(q), \quad (2)$$

with Hamiltonian

$$H(q,p) = \max_{a \in \mathcal{U}_i} \min_{b \in \mathcal{U}_j} p \cdot f(q,a,b,v_i,v_j), \quad (3)$$

where $q = \langle x^i_j, y^i_j, \theta^i_j \rangle$, $f(q,a,b,v_i,v_j) = \dot{q}$ denotes the relative system dynamics, $g(q)$ is a scalar function representing the reachable set at $t = 0$ (e.g. $g(q) = \sqrt{{x^i_j}^2 + {y^i_j}^2} - C$, with C constant), and $\mathcal{U}_i$,$\mathcal{U}_j$ are the sets of permissible angular velocities. The HJI PDE is solved backwards in time until convergence. For a more detailed discussion on the convergence properties of this method, see [18].

We assume that all robots have similar velocity constraints, so we compute a single reachable set $\mathcal{BRS}$ up to a horizon of 1s. We now show how $\mathcal{BRS}$ can be used in particle filtering.

Each robot r_j maintains a separate particle filter for every other robot r_i. In each case, a set of P particles and weights:

$$\mathcal{RP}^i_j = \{\langle p_k, pw_k \rangle \mid k = 1..P\} \quad (4)$$

is maintained, where every $p_k = \langle \tilde{x}^i_j, \tilde{y}^i_j, \tilde{\theta}^i_j \rangle$ is a state hypothesis and pw_k is its associated weight, such that $\sum_{k=1}^{P} pw_k = 1$. Furthermore, we define $\mathcal{RM}^i_j$ as a second set of Q particles over the potential *one-step reactions* of r_i:

$$\mathcal{RM}^i_j = \{\langle m_k, mw_k \rangle \,|\, m_k = \langle \tilde{dx},\ \tilde{dy},\ \tilde{d\theta} \rangle,\ k = 1..Q\}, \quad (5)$$

which are the potential moves r_i can take in a single discrete time step. Each set $\mathcal{RM}^i_j$ is initialised randomly. The *candidate* move at time t is $\bar{m} = \langle \langle x^i_{j,t}, y^i_{j,t} \rangle - \langle x^i_{j,t-1}, y^i_{j,t-1} \rangle,\ \textsc{InferOrientation}(\mathcal{RB}^i_j, \mathcal{BB}_j) \rangle$. The weight $\bar{mw}$ of a candidate is defined using $\mathcal{BRS}$, so that:

$$\bar{mw} = \begin{cases} 1/Q, & \bar{m} \in \mathcal{BRS} \\ 0, & \bar{m} \notin \mathcal{BRS} \end{cases} \quad (6)$$

The oldest particle $\langle m_o, mw_o \rangle$ in $\mathcal{RM}^i_j$ is replaced by $\langle \bar{m}, \bar{mw} \rangle$. Following the replacement, all weights are normalised so that they add to 1. Thus, $\mathcal{RM}^i_j$ essentially acts as a predictive distribution for r_i, by combining both egocentric estimates and ground truth dynamics from the reachable set.

To complete our formulation, we set $\mathcal{RM}^i_j$ as the proposal distribution for the prediction step of $\mathcal{RP}^i_j$. Then, at time t:

$$p_k \leftarrow p_k + \tilde{m},\ \langle \tilde{m}, \tilde{mw} \rangle \sim \mathcal{RM}^i_j,\ k = 1..P \quad (7)$$

The remaining steps are similar to the algorithm described in [9]. The likelihood distributions for the correction step should be suited to the physical limitations of the robot's sensors (e.g. sonar range). The state component of a belief is revised as the weighted sum of its associated particles:

$$\langle x^i_j,\ y^i_j,\ \theta^i_j \rangle = \sum_{k=0}^{P} p_k \cdot p_{wk} \quad (8)$$

C. Action Types, Actions, and Strategic Modes

Every robot has access to the following set of parameterisable **action types**:

$$\mathbf{AT} = \{\textsc{Move}(dx, dy, d\theta),\ \textsc{Kick}(kt, ks),\ \textsc{Scan}(dy, dp)\} \quad (9)$$

$\textsc{Move}(dx, dy, d\theta)$ corresponds to a desired displacement and turn; $\textsc{Kick}(kt, ks)$ executes a kick of a given type $kt \in \{left\_straight, right\_straight, left\_side, right\_side\}$ and speed factor $0 < ks \leq 1$, where $ks = 1$ corresponds to full speed; and $\textsc{Scan}(dy, dp)$ alters the robot's head yaw and pitch by dy and dp respectively, to allow scanning of a different region of the environment. An **action** α is an *instantiation* of an action type α_τ, e.g. $\textsc{Move}(0.1, 0.0, 0.0)$.

In order to cluster similar behaviours together, we also define a set of roles, or **strategic modes**:

$$\mathbf{M}^- = \{\textsc{Kicker},\ \textsc{Defender}\} \quad (10)$$

A KICKER tries to go to the ball and kick it, with additional constraints for adversary avoidance. The DEFENDER mode is triggered when a robot cannot see the ball but is near an adversary, so it instead attempts to block its path.

The mode μ and action type $\alpha\tau$ for r_j at time t is chosen **deterministically** using a decision tree, based on the actual beliefs $\mathcal{RB}_{j,t}$ and $\mathcal{BB}_{j,t}$. We label this procedure:

$$\langle \alpha\tau_t, \mu_t \rangle \leftarrow \textsc{SelectActTMode}(\mathcal{RB}_{j,t}, \mathcal{BB}_{j,t}). \quad (11)$$

D. Intent Inference

Robots should be able to distinguish between "intelligent" and "non-intelligent" adversaries, and adapt their behaviour accordingly. An exhaustive search over all strategies available to a robot and its adversary would be both intractable and inflexible. Instead, we propose and define an **intent filter**, which is used to classify the observed movements of the adversary into coarse classes of strategic behaviours. The intent filter for the adversary r_i with respect to r_j is a set

$$\mathcal{I}^i_j = \{\ \langle I_k,\ im_k,\ iw_k \rangle \mid k = 1..K\ \} \quad (12)$$

where I_k is one of K predefined **intent templates**, im_k is the next move currently predicted by I_k for r_j, and iw_k is its associated weight. In our robotic soccer model, we define the following coarse intent templates:

$$\mathbf{I}^- = \{\textsc{Static}, \textsc{Ball}, \textsc{Pursue}, \textsc{PredMove}\} \quad (13)$$

where STATIC predicts that r_i will not move at the next time step, BALL predicts a movement towards the ball, PURSUE predicts a movement towards r_j, and PREDMOVE sets the next move to a random weighted sample from $\mathcal{RM}^i_j$. Note that these templates are both *logic-based* (e.g. STATIC) and *data-driven* (e.g. PREDMOVE). For every logic-based template I_k, the next move im_k is predicted deterministically (e.g. for BALL, the relative position of the ball to r_i determines how the adversary will move towards it).

Moreover, the strategy of r_i may vary depending on the mode μ_t chosen by r_j at t (e.g. r_i may be more aggressive if it determines that r_j is playing defensively). We define a separate intent filter $\mathcal{I}^i_{j,\mu}$ for every mode μ, each with its own distribution over the intent templates. Through such a decomposition, intent inference becomes a probabilistic game where r_j selects a behavioural mode and action, in response to an action selected independently by r_i. Thus, even if the modes and intent templates do not exactly correspond to the true strategy of r_i, they can help achieve a good approximation that can be refined over time.

E. Strategic escape

Modes and intent templates can be extended to include strategies that *exploit* observability constraints. We call this class of behaviours **escape strategies**, as they strive to move information out of the adversary's sensing range. To support the selection of such strategies, we first compute the *observability bounds* of r_i with respect to r_j as the set:

$$\mathcal{OB}^i_j \equiv \{\, vbs^i_j, sbs^i_j \,\} \leftarrow \textsc{ObservBds}(\mathcal{RB}^i_{j,t}, \mathcal{BB}^i_{j,t}) \quad (14)$$

where vbs^i_j and sbs^i_j are trapezoidal approximations to the vision and sonar sensing ranges of r_i, respectively, and $\overline{vbs^i_j}$ and $\overline{sbs^i_j}$ are their corresponding barycentres.

Examples of escape actions in robotic soccer would be:

- Kick the ball so that the resulting trajectory maximises the distance from the adversary's field of view. Given a set of m candidate ball trajectories (of varying sizes), $\mathcal{BT} \doteq \{\, \beta_m \equiv \{\beta_{mk} \equiv \langle x^b_{mk}, y^b_{mk}\rangle \mid k = 1..|\beta_m|\} \,\}$, the optimal ball escape trajectory $\widehat{\beta}$ is given by:

$$\widehat{\beta} = \operatorname*{argmax}_{\beta_m \in \mathcal{BT}} \frac{1}{|\beta_m|} \sum_{k=1}^{|\beta_m|} \mathtt{dist}(\beta_{mk}, \overline{vbs^i_j}) \quad (15)$$

- Move so that the resulting path trajectory maximises the distance from the adversary's sonar sensing range. As above, given a set of n candidate robot trajectories $\mathcal{RT}$, the optimal robot escape trajectory $\widehat{\rho}$ is:

$$\widehat{\rho} = \operatorname*{argmax}_{\rho_n \in \mathcal{RT}} \frac{1}{|\rho_n|} \sum_{k=1}^{|\rho_n|} \mathtt{dist}(\rho_{nk}, \overline{sbs^i_j}) \quad (16)$$

Finally, (10) and (13) can be augmented to become:

$$\mathbf{M}^+ = \{\textsc{Kicker}, \textsc{Defender}, \textsc{Exploiter}\}, \quad (17)$$

$$\mathbf{I}^+ = \{\textsc{Static}, \textsc{Ball}, \textsc{Pursue}, \textsc{PredMove}, \textsc{Escape}\}. \quad (18)$$

An EXPLOITER is endowed with the capability of probabilistically selecting escape trajectories. This is modeled in the ESCAPE template, which represents the utility of choosing an escape strategy at a given time. This feature is incorporated into the overall action selection procedure (Algorithm 1).

Algorithm 1 Optimal Action Selection

```
OPTACTION($j, \mathcal{RB}_{j,t}, \mathcal{BB}_{j,t}, \alpha\tau_t, \mu_t, \mathcal{I}_j$)
Input: Robot $j$, robot/ball beliefs $\mathcal{RB}_{j,t}/\mathcal{BB}_{j,t}$, action type $\alpha\tau_t$, strategic mode $\mu_t$, current intent filters $\mathcal{I}_j$
$i \leftarrow$ FINDNEARESTADVERSARYINDEX
$\langle I^i, im^i\rangle \sim \mathcal{I}^i_j$ {sample template and predicted move of $r_i$ from intent filter $\mathcal{I}^i_j$, based on weights $iw$}
$\mathcal{RB}^i_{j,t} \leftarrow \mathcal{RB}^i_{j,t} + im^i$ {incorporate prediction}
$\mathcal{OB}^i_{j,t} \leftarrow$ OBSERVBDS($\mathcal{RB}^i_{j,t}, \mathcal{BB}_{j,t}$) {Eq. 14}
if $\alpha\tau_t ==$ MOVE($\cdot,\cdot,\cdot$) or $\alpha\tau_t ==$ KICK($\cdot,\cdot$) then
    if $I^i ==$ ESCAPE then
        $\mathcal{T} \leftarrow$ ESCAPETS($\mathcal{RB}^i_{j,t}, \mathcal{BB}_{j,t}$) {find candidate escape trajectories for current beliefs}
        $\widehat{\tau} =$ OPTESCAPE($\mathcal{T}, \mathcal{OB}^i_{j,t}$) {Eq. 16 or 15}
    else
        $\mathcal{T} \leftarrow$ NORMTRAJ($\mathcal{RB}^i_{j,t}, \mathcal{BB}_{j,t}$)
        $\widehat{\tau} =$ OPTNORM($\mathcal{RT}$) {no escape heuristic}
    end if
    $\alpha_t =$ MOVE($dx, dy, d\theta$) $\leftarrow$ CHOOSEMOVE($\widehat{\tau}$) or $\alpha_t =$ KICK($kt, ks$) $\leftarrow$ CHOOSEKICK($\widehat{\tau}$) {find path/move or kick type/speed for chosen trajectory}
else
    $\alpha_t =$ SCAN($dy, dp$) $\leftarrow$ CHOOSESCAN($\mathcal{BB}_{j,t}$) {ball not visible, attempt to retrack}
end if
return $\alpha_t$
```

F. Regret minimisation

We now consider *online learning* of the intent filter weights as a means of adapting to the adversary (Algorithm 2). At time t, each intent template I_k predicts a move im_k (Eq. 12); however, a robot probabilistically picks just one template and acts based on its prediction. Then, at $t{+}1$, regret minimisation assesses the correctness of *all* predictions, and modifies their weights accordingly.

Algorithm 2 Intent Regret Minimisation

```
REGMIN($\mathbf{I}$,$t$,$\mu_{t-1}$,$j$)
Input: Intent templates $\mathbf{I}$, time $t$, strategic mode $\mu \equiv \mu_{t-1}$ at time $t-1$, estimating robot index $j$; $\epsilon \leftarrow 0.05$
for $i = 1$ to $N$ ; $i \neq j$ do
    $WA = \{\, \epsilon - 2(k-1)\epsilon/(|\mathbf{I}|-1) \mid k = 1..|\mathbf{I}| \,\}$ {weight adjustments, $+\epsilon ... -\epsilon$}
    $Rs \leftarrow \emptyset$ {regrets}
    for $k = 1$ to $|\mathbf{I}|$ do
        $\langle I,\ im,\ iw\rangle \leftarrow \mathcal{I}^i_{j,\mu}[k]$
        $PP \leftarrow im + \langle x^i_j, y^i_j\rangle_{t-1}$ {predicted position}
        $Rs[k] \leftarrow \mathtt{dist}(PP, \langle x^i_j, y^i_j\rangle_t)$ {regret $\propto$ |predicted position - actual position|}
    end for
    $Rs \leftarrow$ SORT($Rs$) {in ascending order}
    for $k = 1$ to $|\mathbf{I}|$ do
        $\mathcal{I}^i_{j,\mu}[Rs[k]].iw \leftarrow \mathcal{I}^i_{j,\mu}[Rs[k]].iw + WA[k]$
    end for
end for
NORMALISEWEIGHTS {so they add to 1}
return $\mathcal{I}_{j,\mu}$
```

G. Summary

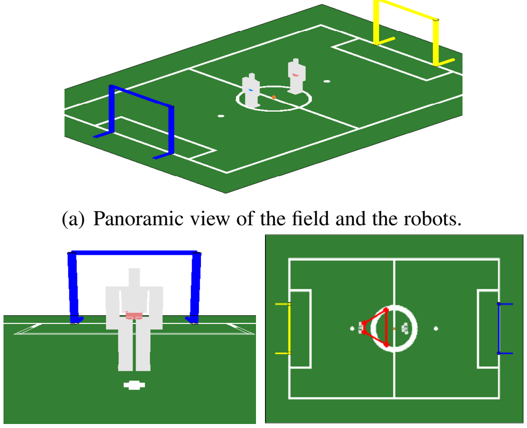
Algorithm 3 summarises the overall decision making procedure, unifying all components and ideas described so far.

Algorithm 3 Complete Decision Making Algorithm

```
DECMAKER($\mathbf{I}$,$\mathbf{M}$,$rm$,$\mathcal{IW}$,$j$)
Input: Intent templates $\mathbf{I}$, strategic modes $\mathbf{M}$, boolean $rm$ for regret minimisation, initial intent template weight distributions $\mathcal{IW}$, estimating robot index $j$ ; $t \leftarrow 0$
$\mathcal{I}_j \leftarrow$ INITIALISEIFS($\mathbf{I}$,$\mathcal{IW}$) {Initialise intent filters}
while TRUE do
    SENSEWORLD {get latest sensor data}
    $\langle \mathcal{RB}_{j,t}, \mathcal{RP}_j, \mathcal{RM}_j \rangle \leftarrow$ RSPF {c.f. Sec. III-B}
    if $rm$ == TRUE and $t > 0$ then
        $\mathcal{I}_{j,\mu_{t-1}} \leftarrow$ REGMIN($\mathbf{I}$,$ld$,$t$,$\mu_{t-1}$,$j$) {Alg. 2}
    end if
    $\langle \alpha\mathcal{T}_t, \mu_t \rangle \leftarrow$ SELECTACTTMODE($\mathcal{RB}_{j,t}, \mathcal{BB}_{j,t}$)
    $\mathbb{I}_t \sim \{\langle I_k, im_k, iw_k \rangle \leftarrow \mathcal{I}_{j,\mu_t}[k] \,|\, k = 1..|\mathbf{I}|\}$ {Select intent filters based on current weights $iw_k$}
    $\alpha_t \leftarrow$ OPTACTION($j, \mathcal{RB}_{j,t}, \mathcal{BB}_{j,t}, \alpha\mathcal{T}_t, \mu_t, \mathbb{I}_t$)
    EXECUTEACTION($\alpha_t$)
    $t \leftarrow t + 1$
end while
```

IV. RESULTS

We evaluate our algorithms on a simulated robotic soccer environment with realistic physical constraints. Figure 1 shows a panoramic view of the soccer field, along with the associated field of view and sonar range. Constraints have been placed on the allowed magnitudes of MOVE and KICK commands. Distances to objects falling in the sonar range are subjected to Gaussian noise with a mean equal to its true magnitude and a standard deviation of 1. False positives occur when a robot is near field edges or goal posts.



(a) Panoramic view of the field and the robots.

(b) Perspective field of view

(c) Sonar sensing range

Fig. 1. Soccer simulator environment.

A. Reachable Set Particle Filter

We first compare the Reachable Set Particle Filter (RSPF) against a number of other state estimation variants:

- **No filtering (NF)**: the extracted (noisy) sensory observations are converted directly to beliefs, without additional motion models or observation distributions.
- **Simple Particle Filter (SPF)**: This is a particle filter algorithm without the additional reachable set constraint. In effect, Eq. 6 is modified so that all candidate particles are assigned a probability of $1/Q$.
- **Intent-based state estimation (IBSE)**: This procedure estimates both a robot's state and intent *in one pass*. The probabilistic motion model of Section III-B is replaced with an intention-based distribution similar to Eq. 12. The algorithm attempts to map intents to adversary observations directly, without explicitly taking into account *motion constraints* and *observation likelihoods*.

We evaluate the algorithms on soccer games between two robots, each using its sonar to estimate the adversary's state. In the initial configuration (Figure 1(a)), robots are outwith the sensing range of their adversaries. Agents execute a simple algorithm to move towards the ball, though their exact strategy is not important at this point. For each filtering algorithm $\mathbf{f}$, we compute the error as the mean distance of r_j's egocentric estimates, $\mathbf{x}_{j,t}^{i,\mathbf{f}}$, to the true location of r_i, $\bar{\mathbf{x}}_t^i$:

$$\mu\mathbf{DTL}(\mathbf{f}) = \frac{1}{T}\sum_{t=1}^{T}\sqrt{\mathbf{x}_{j,t}^{i,\mathbf{f}} - \bar{\mathbf{x}}_t^i}^{\,2} . \tag{19}$$

with $\sigma\mathbf{DTL}$ being the corresponding standard deviation. For the RSPF algorithm, the backward reachable set $\mathcal{BRS}$ (Section III-B) was computed with respect to the relative dynamics of the two robots, using Mitchell's level set toolbox [14]. Figure 2 provides a visualisation of this set, along with the corresponding reaction and state particle distributions, illustrating the utility of the reachable set as a filtering tool.

Method (f)	μ**DTL(f)**	σ**DTL(f)**	**Error gain wrt. NF**
NF	14.79 cm	1.846	-
SPF	17.63 cm	1.953	-19.2%
RSPF	13.76 cm	1.996	+6.96%
IBSE	15.16 cm	2.544	-2.5%

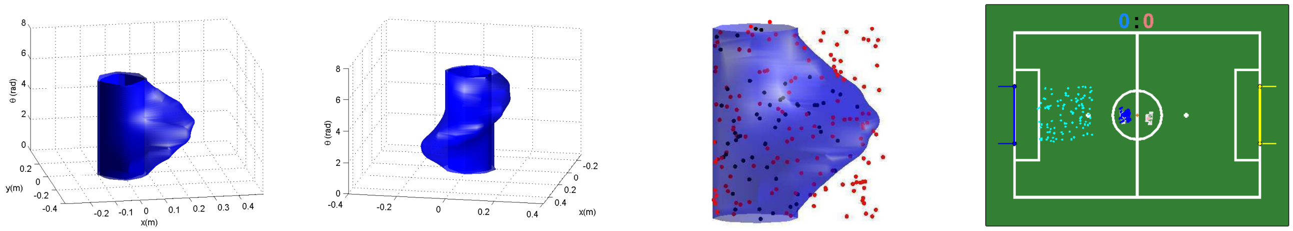
TABLE I

MEAN ERROR PER FILTERING METHOD

Table I summarises the results, as averaged over 20 trials. At a first glance, the 6.96% gain obtained when using RSPF instead of no filtering (NF) may seem small, but one must acknowledge the complexity of the task: robots must estimate the state of dynamic adversaries whose exact behavioural model is unknown, in the face of noisy sensor data. Moreover, any improvement in rejecting spurious trajectories has a substantial impact on the following steps that attempt to learn responses on top of this information. The performance of RSPF relative to IBSE supports our claim that in games characterised by strategic and sensory noise, state and strategy estimation should be decoupled.

B. Strategic decision making

1) Preliminaries: We now fix the RSPF as the state estimation algorithm and we evaluate different permutations of



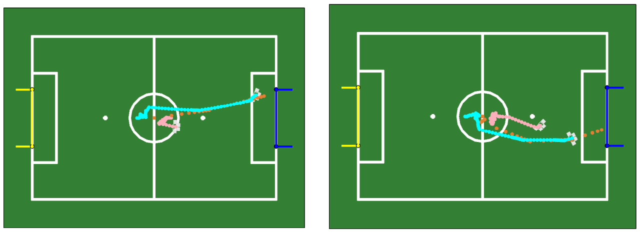
(a) Reachable set - side view.

(b) Reachable set - front view.

(c) **One-step reaction particles**: valid (*black*) and discarded particles (*red*)

(d) **State particles**: initial (*cyan*) and converged distribution at t=15 (*blue*).

Fig. 2. (a)-(b): Three dimensional (x,y,θ) reachable set computed for a time horizon of 1s based on the relative dynamics of the two robots - maximum linear velocity = 0.2m/s, maximum angular velocity = 0.2rad/s. (c): One-step reaction particle filtering - observations outwith the reachable set are discarded. (d): State particles for blue robot (left) as computed by pink robot (right) - initial uniform-random distribution and convergence after 15 time steps.



(a) $\langle E, I\rangle$ (cyan) vs. $\langle N, N\rangle$ (pink)

(b) $\langle E, I\rangle$ (cyan) vs. $\langle N, I\rangle$ (pink)

Fig. 3. Trajectory traces from selected game episodes.

decision making strategies, based on the concepts described in Section III. Each strategy is a tuple $\langle \mathbf{e}, \mathbf{r}\rangle$, where:

- $\mathbf{e} \in \{(N)\text{one}, (E)\text{scape}\}$ denotes the escape strategy used. In other words, $\mathbf{e} \leftarrow N$ uses $\mathbf{M}^-$ and I^- (10-13), whereas $\mathbf{e} \leftarrow E$ uses $\mathbf{M}^+$ and I^+ (17-18) as their strategic modes and intent templates, respectively,
- $\mathbf{r} \in \{(N)\text{one}, (I)\text{ntent regret minimisation}\}$ is the type of regret minimisation used (parameter rm in Alg. 3).

This formulation leads to a total of 4 valid permutations, namely $\langle N, N\rangle$, $\langle N, I\rangle$, $\langle E, N\rangle$ and $\langle E, I\rangle$. We compare these strategies in a round-robin one-versus-one soccer tournament, where all strategies are played in three pairs of games (**10a**-**10b**, **20a**-**20b**, **50a**-**50b**) consisting of 10, 20, and 50 *episode* games, respectively. An episode terminates if a robot scores a goal, if the ball leaves the field bounds, if the robots collide, or if the maximum episode time (set to 100 time steps for each robot) elapses. The initial configuration (Figure 1(a)) introduces additional *strategic* constraints:

- The ball is too far from the goal mouths, so robots require a *sequence* of actions in order to score.
- The robots require dexterous manoeuvres to evade their opponents and kick the ball past them.

2) Results and statistics: In addition to the final scores of the games, we recorded other relevant statistics, such as the mean time to score a goal **(MTS)** and to evade the adversary **(MTE)**, and the mean time taken by the adversary to score **(A-MTS)** and to evade **(A-MTE)**. Table II(a) summarises these statistics, together with the total goals scored for **(GF)** and against **(GA)**. The entries are sorted with respect to the total points **(P)**, which are determined through the standard soccer point system (3 points for victory, 1 for draw, 0 for defeat). The game results are summarised in Table II(b).

Traces from two game episodes are given in Figures 3(a) (episode between a "good" and "bad" strategy) and Figure 3(b) (the two best strategies). In the latter case, the scoring robot takes more time to escape, as indicated by the larger concentration of movements around the centre of the field.

3) Partial orderings: We summarise observations from Tables II(a) and II(b) as **partial orderings** among templates. We use the notation $\mathcal{T}_1 \geq \mathcal{T}_2$ to denote that template $\mathcal{T}_1$ performs at least as well as $\mathcal{T}_2$, '$\cdot$' for the wildcard symbol, and $\neg A$ for any instantiation of a template except A.

- $\langle \cdot, I\rangle \geq \langle \cdot, \neg I\rangle$: Regret minimisation seems to be by far the most prevalent strategy, both on its own and when combined with escape strategies, as indicated by both the overall scores and the statistics.
- $\langle E, \cdot\rangle \geq \neg\langle N, I\rangle$: The use of escape strategies is also highly beneficial when compared to the benchmark $\langle N, N\rangle$ that makes no assumptions about the adversary.
- $\neg\langle N, N\rangle \geq \langle N, N\rangle$: All strategies using at least one heuristic out outperform the benchmark $\langle N, N\rangle$.

4) Convergence of regret minimisation: Finally, we test regret minimisation against a stationary adversary. The regret minimising robot is not aware of this, so it initialises all intent weights uniformly. Figure 4 illustrates how regret minimisation converges to the "true" template distribution. Note that the PREDMOVE template is also representative of a static adversary; if that robot is never observed to move, the distribution $\mathcal{RM}$ will assign high weights to null moves, thus also predicting a static reaction. This similarity partly explains the spikes in the STATIC and PREDMOVE curves. Nonetheless, the joint weights of the two templates (Static + PredMove curve) are correctly observed to converge to 1.

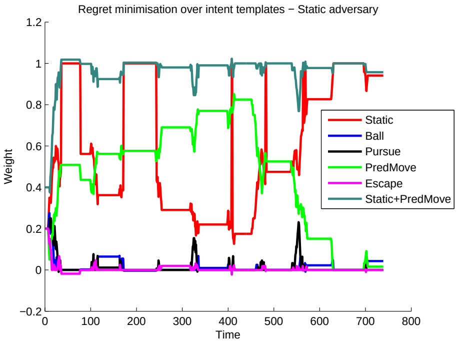


Fig. 4. Regret minimisation over the intent templates of a static adversary

(a) Summary of scores and strategy statistics -best entries in boldface.

Strategy	P	GF	GA	MTS	MTE	A-MTS	A-MTE
$\langle N, I\rangle$	**35**	**143**	129	**68.77**	54.12	**70.65**	54.55
$\langle E, I\rangle$	28	130	**119**	69.15	**53.37**	67.65	54.85
$\langle E, N\rangle$	24	121	125	71.67	55.19	68.45	53.30
$\langle N, N\rangle$	15	119	140	71.87	55.31	70.16	**53.13**

(b) Soccer tournament results

	10a	10b	20a	20b	50a	50b
$\langle N, N\rangle$-$\langle N, I\rangle$	3-4	2-4	6-8	4-7	12-13	13-19
$\langle N, N\rangle$-$\langle E, N\rangle$	1-4	4-4	5-4	5-5	18-11	10-12
$\langle N, N\rangle$-$\langle E, I\rangle$	1-2	1-2	5-5	4-3	16-11	9-22
$\langle N, I\rangle$-$\langle E, N\rangle$	3-0	4-1	9-5	6-4	10-15	14-15
$\langle N, I\rangle$-$\langle E, I\rangle$	4-2	3-3	4-7	4-7	13-13	14-17
$\langle E, I\rangle$-$\langle E, N\rangle$	1-2	4-1	5-5	3-5	8-16	15-12

TABLE II

RESULTS AND STATISTICS

V. CONCLUSION

We present a strategic decision making framework for a relatively complex class of multi-robot games, characterised by both sensory and strategic uncertainty. The specific contributions of this paper are twofold. On the one hand, we present a novel probabilistic adversarial state estimation algorithm, featuring both data-driven approximation and reasoning about a dynamical constraint. Our evaluation shows a performance improvement, in simulation, compared to general purpose filtering algorithms, which supports our argument in favour of decoupling estimation of a noisy state from estimation of strategy. This gain is likely to be even higher in interactions between physical robots, where sensory data is both sparser and much more spurious. On the other hand, we have adapted game theoretic concepts, which had previously been studied primarily in an abstract theoretical setting without physical constraints, into a unified intent inference framework for multi-robot games. Our results favour the use of regret minimisation as an adaptive learning mechanism, while showing promise for careful use of escape strategies that exploit the adversary, as part of a decision making system with a diverse set of intent templates.

The concepts presented in this paper can be extended to allow robots to act strategically over greater time horizons and using a larger repertoire of templates. One approach would be the incorporation of *deception* in the decision making framework, whereby robots could execute actions that conceal their true intentions. Developing such sophisticated behaviours requires a deeper understanding of the properties of various concepts described in this paper. We are currently working on evaluation of our algorithms in adversarial games involving physical robots, while also considering the interesting and important special case of *human*-robot games.

REFERENCES

[1] D. Avrahami-Zilberbrand and G. A. Kaminka. Fast and complete symbolic plan recognition. In *Proceedings of the 19th international joint conference on Artificial intelligence*, pages 653–658, 2005.

[2] N. Bard and M. Bowling. Particle filtering for dynamic agent modelling in simplified poker. In *Proc. AAAI*, pages 515–521, 2007.

[3] S. Bhattacharya and S. Hutchinson. On the existence of Nash equilibrium for a two-player pursuitevasion game with visibility constraints. *International Journal of Robotics Research*, 29(7):831–839, 2010.

[4] H. H. Bui. A general model for online probabilistic plan recognition. In *Proceedings of the 18th international joint conference on Artificial intelligence*, pages 1309–1315, 2003.

[5] P. Doshi and P. J. Gmytrasiewicz. Monte carlo sampling methods for approximating interactive pomdps. *Journal of Artificial Intelligence Research*, 34:297–337, 2009.

[6] C. Geib and S. Harp. Empirical analysis of a probabilistic task tracking algorithm. In *Proceedings of Workshop on Agent Tracking, Autonomous Agents and MultiAgent Systems (AAMAS)*, 2004.

[7] B. P. Gerkey, S. Thrun, and G. Gordon. Visibility-based pursuit-evasion with limited field of view. In *International Journal of Robotics Research*, pages 20–27, 2004.

[8] P. J. Gmytrasiewicz and P. Doshi. A framework for sequential planning in multi-agent settings. *Journal of Artificial Intelligence Research*, 24:24–49, 2005.

[9] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing*, 140(2):107–113, August 1993.

[10] P. R. Jordan and M. P. Wellman. Generalization risk minimization in empirical game models. In *AAMAS*, 2009.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(CS-96-08):99–134, 1998.

[12] S. Karaman and E. Frazzoli. Sampling-based algorithms for a class of pursuit-evasion games. In *WAFR*, 2010.

[13] R. Messing, C. Pal, and H. A. Kautz. Activity recognition using the velocity histories of tracked keypoints. In *ICCV*, pages 104–111, 2009.

[14] I. M. Mitchell and J. A. Templeton. A toolbox of hamilton-jacobi solvers for analysis of nondeterministic continuous and hybrid systems. In *HSCC 2005*, pages 480–494. Springer-Verlag, 2005.

[15] B. Ng, C. Meyers, K. Boakye, and J. Nitao. Towards applying interactive pomdps to real-world adversary modeling, 2010.

[16] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, 2007.

[17] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–35, 1952.

[18] C. J. Tomlin, I. Mitchell, A. M. Bayen, and M. Oishi. Computational techniques for the verification of hybrid systems. In *Proc. IEEE*, pages 986–1001, 2003.

[19] R. Vidal, O. Shakernia, H. Kim, D. Shim, and S. Sastry. Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. *Trans. on Robotics and Automation*, 18(5):662 – 669, 2002.

APPENDIX

A. Relative orientation estimation

Algorithm 4 Relative Orientation Inference

```
INFERORIENTATION($\mathcal{RB}_j^i$,$\mathcal{BB}_j$)
Input: Robot beliefs $\mathcal{RB}_j^i$ for $r_i$, ball beliefs $\mathcal{BB}_j$
$distTh \leftarrow 0.7m$ {distance threshold}
$\langle cdr_i, cdb, cdbr_i, ldbr_i\rangle \leftarrow$ {current distance of $r_i$ and ball, current/last distance of ball from $r_i$}
if $\texttt{rand} < 0.7$ and $(cdr_i > cdb$ or $cdbr_i < ldbr_i)$ and $cdr_i > distTh$ then
    return $\texttt{atan2}(y^B_{j,t} - y^i_{j,t}, x^B_{j,t} - x^i_{j,t})$ {$r_i$ has moved closer to the ball $\rightarrow$ infer that it is facing towards it}
else
    if $\texttt{rand} < 0.5$ and $cdr_i < distTh$ then
        return $\texttt{atan2}(y^i_{j,t}, x^i_{j,t}) + \pi$
    else
        return $\pi$ {$r_i$ is facing in the direction of $r_j$}
    end if
end if
```