2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Hierarchical task allocation for multi-agent systems encoded by stochastic reachability specifications

Nikolaos Kariotoglou[1], Sean Summers[1] and Davide M. Raimondo[2]

Abstract— **We consider the problem of satisfying a set of objectives over a collection of agents. For a single agent, the optimal solution can be obtained via a stochastic reachability framework where optimal control policies come along with a performance metric, defined as the probability of successfully achieving a specified objective. As the number of agents increases, the approach quickly becomes computationally expensive and often intractable. We propose a method which includes an advisory controller that allocates tasks among agents based on their ability of handling individual objectives. This ability is encoded by the stochastic reachability performance metrics. The proposed method is tailored to an autonomous surveillance system composed of pan-tilt-zoom (PTZ) cameras and verified experimentally.**

I. INTRODUCTION

Nowadays, autonomous surveillance is often necessary to assist humans in critical tasks ranging from accident and crime prevention to systems monitoring. Intelligent PTZ camera-based surveillance systems have received growing attention [1]–[9]. In [5], surveillance tasks like target tracking and target acquisition were considered in form of a probabilistic pursuit-evasion game where pursuers and evaders are set-valued and governed by stochastic processes. In particular, a single framework for autonomous surveillance tasks based on the theory of stochastic reachability (for discrete-time stochastic hybrid systems (DTSHS)) and random sets [10], [11], was developed.

When large camera networks are considered, a centralized approach becomes computationally prohibitive and the use of hierarchical/distributed schemes becomes necessary. In [12], [13] the authors address the problem of camera network control and distributed synthesis of LTL formulas. The approach is deterministic and linear while the low level control of camera agents is not directly handled. In [14] a similar approach towards surveillance through LTL is provided with the same limitations. In [6], a decentralized task assignment approach, inspired by the Stable Married problem with Ties and Incomplete Lists, has been proposed for camera networks in order to alleviate the complexity of a centralized linear integer program. Camera assignment has been also considered by [15], where a distributed lookup table has been used in order to perform camera hand-offs.

Motivated by recent results in the application of stochastic reachability to autonomous surveillance [5], [1], we introduce an advisory controller for hierarchical task allocation encoded by stochastic reachability specifications. More precisely, tasks are assigned according to:

- the probability of successfully achieving the different objectives by the single cameras;
- a set of rewards (task and agent-based). We grade tasks according to priorities and reward agents on the base of agents-tasks preferences.

The resulting task allocation scheme is verified on a two camera, two evader experimental platform. Even though this set-up could appear not sufficient to test the potential of the proposed method, it actually fits the scope. The stochastic reachability framework we consider allows the presence of nonlinearities (e.g. cameras field of view projection on a plane) at the expense of a gridding-based approach (state and control space will be discretized). This makes the centralized approach suitable for one, maximum two cameras resulting in a 6D problem (in the case of a 2D environment and a 2D control space for each single camera), high storage requirements and slow online computation. Even though the decentralized approach results in a sub-optimal strategy (because tasks are considered individually), the experimental results are quite promising.

The paper is organized as follows: we first recall the theory of reachability for DTSHS and the connection to autonomous surveillance. We then augment the formulation to consider multi-agent systems and formulate reachability objectives for the augmented system by considering the single systems independently. The proposed advisory controller for task allocation is formulated for a multi-agent system expressed as a DTSHS. In the last section of the paper, we look at a specific case of camera surveillance with two robotic evaders and two pan-tilt cameras, formulate the problem in this context and present the results and performance of the developed advisory controller.

II. REACHABILITY ANALYSIS AND TASK ALLOCATION FOR MULTI-AGENT SYSTEMS

A. Mathematical Background

We briefly recall the theory of stochastic reachability for DTSHS [10], [16]. A DTSHS $\mathcal{H}$ can be described as a Markov control process with state space X, (compact) control space $\mathcal{A}$, and controlled transition probability function

[1]N. Kariotoglou and S. Summers are with the Automatic Control Laboratory, ETH, Zurich {karioto,ssummers,lygeros}@control.ee.ethz.ch
[2]D. M. Raimondo is with the Identification and Control of Dynamic Systems Laboratory, University of Pavia davide.raimondo@unipv.it
*This research was partially supported by the NanoTera-SSSCT Grant under the project NetCam, and by HYCON2, under the project Network of Excellence - FP7-ICT-257462. The work of Nikolaos Kariotoglou was supported by SNF grant number 2000211_37876

Q. Given a Markov control policy $\mu \in \mathcal{M}_m$ (where $\mathcal{M}_m$ denotes the set of all admissible Markov control policies) and initial state $x_0 \in X$, the execution $\{x_k, k = 0, ..., N\}$ is a time inhomogeneous stochastic process defined on the canonical sample space $\Omega = X^{N+1}$, endowed with its product $\sigma-$algebra $\mathcal{B}(\Omega)$ where $\mathcal{B}(\cdot)$ denotes the $\sigma-$algebra of Borel sets. The probability measure $P_{x_0}^{\mu}$ corresponds to the probability distribution of executions x_k and is uniquely defined by the transition kernel Q, the Markov policy $\mu \in \mathcal{M}_m$, and the initial condition $x_0 \in X$ (see [17]).

Let $K, K' \in \mathcal{B}(X)$ for all $k = 0, ..., N$, with $K \subseteq K'$ where K denotes a target set and K' denotes a safe set. In [10], [16] it was shown that the probability that the execution of the Markov control process associated with the Markov policy $\mu \in \mathcal{M}_m$ and the initial condition x_0 will hit K before hitting $X \setminus K'$ during the time horizon N can be expressed as the expectation of a sum-multiplicative cost function, i.e.

$$J = E_{x_0}^{\mu}\left[\sum_{j=0}^{N}\left(\prod_{i=0}^{j-1}\mathbf{1}_{K'\setminus K}(x_i)\right)\mathbf{1}_K(x_j)\right].$$

It was subsequently shown that the above probability (henceforth refered to as the reach-avoid (RA) probability) can be computed using a dynamic recursion. It was also proved that optimal decisions can be attained that maximize or minimize the above RA probability and that the solution is given by a dynamic program.

Recent extensions to the nominal RA problem for DTSHS include [11], [18], [19]. In [11], [19], the results of [10] were extended to include the case where the defined sets (safe and target) are time varying (also considered in [20]) and probabilistic. Importantly, a framework for autonomous surveillance was built upon this theory in [5]. For the mathematical details of stochastic reachability, in particular the formulation of the dynamic programming solutions, the reader is encouraged to visit the works [10], [11], [18], [20].

B. Multi-agent systems

Consider a (stochastic) multi-agent system consisting of n agents. Without control, we assume that the state evolution of each agent is independent in probability. That is, each agent is described by a Markov process with transition kernel only dependent on its own state (not influenced by the other $n-1$ agents). When optimizing the performance of the agents collectively, the optimal control policy for each agent is often explicitly a function of all agent states. When this happens, the system is coupled (conditionally dependent in probability) through each agents optimal control policy. Hence, the system becomes a single Markov process with a collective state space X, control space $\mathcal{A}$, and transition kernel Q. In theory, considering the n agent system as a single Markov decision process (or DTSHS), it is possible to synthesize optimal control policies for the agents using the reachability specifications described in Section II-A. In practice, however, solving a problem of any significant magnitude using dynamic programming (the most generic approach) is near impossible when gridding the state space (see Curse of Dimensionality [21]).

To account for the computational complexity of the above problem, we restrict the feasible set of feedback policies for each agent to those dependent only on its own state. More specifically, we consider a collection of n DTSHS $\mathcal{H}_i$, $i \in \{1, \ldots, n\}$. Each DTSHS $\mathcal{H}_i$, $i \in \{1, \ldots, n\}$, is defined according to its state space X_i, control space $\mathcal{A}_i$, and controlled transition probability function Q_i. The probability measure $P_{x_{0_i}}^{\mu_i}$ for each DTSHS is uniquely defined by the transition kernel Q_i, a Markov control policy $(\mu_i) \in \mathcal{M}_{m_i}$ (where $\mathcal{M}_{m_i}$ denotes the set of all admissible Markov control policies for the DTSHS $\mathcal{H}_i$) and its initial state $x_{0_i} \in X_i$. This is a rather strong assumption that leads to (in the general case) suboptimal control policies. The advisory controller proposed here re-introduces the coupling lost by the independence assumption. This happens through the final task assignment which is dependent on each agent's capabilities. For example, if the same objective can be achieved by two agents and the above assumption would result in conflicting control policies (agents trying to achieve the same objective), the advisory controller will make sure that the most suitable agent (the one with highest success probability) is assigned to the task. If these probabilities are equal, any solution can be chosen since they are exactly equivalent.

C. Stochastic reachability objectives for multi-agent systems

Consider again a single stochastic reachability objective defined for the collection of n agents. As stated above, this problem is solved using dynamic programming. Therefore finding a solution becomes near impossible when gridding the state (and control) space. To overcome this limitation, we assume that the objectives specified on the collection of agents can be satisfied (albeit sub-optimally) by a combination of reachability objectives satisfied by the individual agents. For example, consider an autonomous surveillance task: two cameras (agents) have to track two evaders over a horizon of N steps. One possible way to satisfy this objective is to have one camera track one evader, and the other camera track the other evader. It might also be possible for one camera to track both.

Let us denote an objective for the overall system as a global objective and an objective for the individual agent as a local objective. The approach we propose can be summarized as follows:

1) each agent solves a collection of local stochastic reachability problems and reports the results to the advisory controller (as a probability of satisfying each local objective)
2) an advisory controller assigns a local objective to each agent based on 1) and a weighted hierarchy over both local and global objectives (according to the formulation in Section II-D)

A loose interpretation of this is that the final task allocation is chosen based on an estimate of how likely the local objectives (leading to the satisfaction of the global objective) can be satisfied. This is encoded by the probability of achieving each local task (objective) reported by each agent.

D. Advisory controller for task allocation

Briefly, the proposed advisory controller ranks the combinations of local and global objectives according to a weighted cost function. The cost function takes into consideration the probability of each agent achieving a local objective (solution of a stochastic reachability problem), a weight associated with each local objective (agent dependent), and a weight associated with the resulting global objective.

More specifically, defining $\mathcal{G}$ as the set of all possible global objectives, there exists a collection of objectives $G(x) \subseteq \mathcal{G}$ that can be achieved, where $x \in X$ denotes the state of the overall system . For each individual agent state $x_i \in X$, defining $\mathcal{L}_i$ as the set of all possible objectives that can be achieved by a single agent, there exists a collection of possible local objectives $L_i(x_i) \subseteq \mathcal{L}_i$ that can be achieved. Further, define $\mathcal{F} : \mathcal{L}_1 \times \cdots \times \mathcal{L}_n \to \mathcal{G}$ as a function that maps a collection of local objectives to a global objective, let the weight of each global objective be defined as $w_g \in \mathbb{R}$ for $g \in \mathcal{G}$, and let the weight of each local objective and agent be $w_{l_i,i} \in \mathbb{R}$. The advisory controller for task allocation using reachability objectives can therefore be posed as an optimization problem (over feasible local objectives (i.e. tasks)):

$$\max_{l_1,\ldots,l_n} w_g \prod_{i=1}^{n} w_{l_i,i} RA_{l_i,i}(x_i)$$

For each combination of feasible local objectives, $\mathcal{F}(l_1, ..., l_n)$ returns a corresponding global objective g that is satisfied. This global objective sets the reward weight w_g. Each l_i assignment (local objective for agent i) sets the reward weight $w_{l_i,i}$ for the success probability $RA_{l_i,i}(x_i)$ where $RA_i(x_i, l_i)$ denotes the (reach-avoid) probability of local objective l_i for agent i at state x_i. Note that the map $\mathcal{F}$ depends very much on the type of problem considered.

E. Scalability of the proposed method

The main drawback of the proposed allocation mechanism is that the number of possible combinations $(l_1, ..., l_n)$ grows exponentially with the number of possible local objectives and individual agents. However, in this way, we are able to solve problems that via gridding would be intractable due to huge space requirements associated with the transition probability involved in the reachability calculation. Moreover, the cost associated with each realization of $(l_1, ..., l_n)$ is a simple product calculation (given that each $RA_i(x_i, l_i)$ has already been solved independently and reported by each agent) and the maximization is a simple ordering of costs. These are very fast operations for up to several individual agents. Note that a centralized approach based on gridded Dynamic Programming methods would not be able to tackle these problems because of the excessive computational complexity. Furthermore, the independence in solving the specific $RA_i(x_i, l_i)$ problems by each agent allows full parallelization.

We now demonstrate the effectiveness of the proposed advisory controller on the problem of autonomous surveillance with PTZ cameras and verify the results (in closed loop) on an experimental testbed.

III. TASK ALLOCATION AND FEEDBACK CONTROL FOR AUTONOMOUS SURVEILLANCE: AN EXPERIMENTAL STUDY

The validation of the proposed advisory controller is experimentally performed on a two-camera two-evader system. In particular, we consider the experimental setup and stochastic reachability specifications consistent with [5]. In [5], autonomous surveillance tasks are posed as stochastic reachability problems and the evaders are treated as stochastic processes with independent transition kernels. The task for the surveilling agents (cameras) is to reach the state in their state space (pan-tilt space) that maximizes the integral (over their projected field of view) of the evader process distribution. The larger the integral, the higher the probability that the field of view overlaps with the evader. By solving this problem, an optimal control policy is obtained which attempts to track an evader over a finite time horizon.

The stochastic reachability framework lacks a patrolling algorithm, hence we use a global eye-in-the-sky camera that provides rough estimates of evader positions in the case a target is lost. These estimates are reported to the moving cameras that utilize stochastic reachability to get closer (zoomed-in) views of the evaders by solving combinations of tracking and acquiring objectives.

A. Local (camera) objectives

For camera $j \in \{1, 2\}$, the local state S_i^j, $i \in \{0, 1, 2, 3\}$, corresponds to which evaders can be seen with 0,1,2,3 corresponding to seeing none, evader 1, evader 2 or both respectively. For the specified set-up, each camera can solve the finite reachability problems of track, acquire and acquire while track for the two evaders (see [5]). The solution of each of these objectives is treated as a probability that a transition to a particular local state is successfully achieved. Hence, transitions can be encoded by stochastic reachability probabilities. Moreover, according to the current state that an agent is in (and consistent with the mathematical formalism above), only a subset of these objectives (equivalently transitions) will be considered. The objectives are given by a predefined map that discards all objectives that are trivially satisfied with probability 1. For example, if a camera has a non-zero probability of tracking evader 1, the probability of acquiring evader 1 will always be 1 (since this objective is trivially satisfied at time 0).

The set of all local objectives for camera j, i.e. $\mathcal{L}_j$, is

$$\mathcal{L}_j = \{T_1, T_2, A_1, A_2, T_1A_2, T_2A_1, T_1T_2, A_1A_2\}$$

where T_k, A_k denote tracking and acquiring evader k respectively . T_kA_m corresponds to acquire evader m while tracking evader k. T_kT_m corresponds to tracking both k, m and A_kA_m to acquiring both. The set of all local objectives for j as a function of state S_i^j are shown below. For example, when camera j is in state S_1^j, it can only stay at the same state by tracking evader 1 (T_1), transition to S_2^j by acquiring evader 2 (A_2) or transition to S_3^j by tracking evader 1 while acquiring evader 2 ($T_1 A_2$).

$$L_j(S_0^j) = \{A_1, A_2, A_1A_2\}, \quad L_j(S_1^j) = \{T_1, A_2, T_1A_2\}$$
$$L_j(S_2^j) = \{T_2, A_1, T_2A_1\}, \quad L_j(S_3^j) = \{T_1, T_2, T_1T_2\}$$

The camera state transition probabilities (again, equivalent to the successful completion of the local objective) are expressed by the solutions to the stochastic reachability objectives. Hence $Pr(A_k)$ for a camera j in state $i \neq k$ is the probability of a successful state transition from state i to state k.

With the above formulation, we have characterized a key ingredient necessary for the application of the advisory controller. We next design the local objective weights for the advisory controller. While any weight can in fact be considered, for the current experimental setup we consider the case that acquiring is fundamentally less desirable than tracking. Hence, we assign the weights for the local objectives as follows:

$$w_{\mathcal{L}_j} = \{1, 1, 1/N, 1/N, 1/N, 1/N, 1, 1/N^2\}$$
$$w_{L_j(S_0^j)} = \{1/N, 1/N, 1/N^2\}, \quad w_{L_j(S_1^j)} = \{1, 1/N, 1/N\}$$
$$w_{L_j(S_2^j)} = \{1, 1/N, 1/N\}, \quad w_{L_j(S_3^j)} = \{1, 1, 1\}$$

where N is the horizon length. The weighted contribution of each camera as a function of camera state and local objective is depicted in Figure 1.

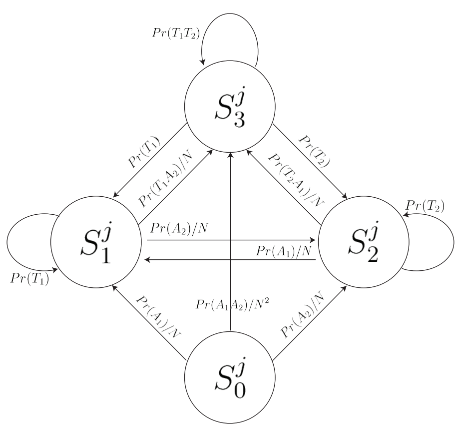


Fig. 1. Weighted contribution of each camera as a function of camera state and local objective.

B. Global (system) objectives

The number of global states is independent of the number of surveillance cameras and depends only on the number of evaders. In the considered system of two cameras and two evaders, the global state S_i^G of the surveillance system takes values $S_i^G \in \{S_0^G, S_1^G, S_2^G, S_3^G\}$ and again corresponds to which evaders can be seen. The function mapping a collection of local objectives to the global objective is given as:

$$S_0^G \leftarrow \{(S_0^1, S_0^2)\}$$
$$S_1^G \leftarrow \{(S_1^1, S_0^2), (S_0^1, S_1^2), (S_1^1, S_1^2)\}$$
$$S_2^G \leftarrow \{(S_2^1, S_0^2), (S_0^1, S_2^2), (S_2^1, S_2^2)\}$$
$$S_3^G \leftarrow \{(S_2^1, S_1^2), (S_1^1, S_2^2), (S_3^1, S_1^2), (S_1^1, S_3^2),$$
$$(S_3^1, S_2^2), (S_2^1, S_3^2), (S_0^1, S_3^2), (S_3^1, S_0^2), (S_3^1, S_3^2)\}$$

For example, if the state of the camera pair 1, 2 is (S_2^1, S_0^2), the combined system can only see evader 2. The weight assigned to the global objective $w_{\mathcal{G}}$ is defined as:

$$w_{\mathcal{G}} = [w_{S_0^G}, w_{S_1^G}, w_{S_2^G}, w_{S_3^G}]$$

For the time being, we leave these weights free. In the sequel (experimental validation), we will adjust these weights and evaluate the performance of the system as more weight is assigned to one evader than the other.

C. Receding horizon control

Given the objectives for the system defined in the preceding sections, we can apply the hierarchical task allocation controller encoded by stochastic reachability in a receding horizon. In particular, at each step we find the best allocation as proposed in Section II-D to assign the local objectives to each camera. Each camera then applies the first control decision associated with the satisfaction of the chosen local objective. The system is then propagated forward and the problem is solved again. We show experimentally that this approach performs very well over a significant time period.

D. Results

In this section we discuss the performance of the advisory controller on a testbed comprising two pan-tilt controllable cameras and two robotic evaders moving randomly on a table placed within the field of view of the surveillance cameras. To solve all reachability objectives for surveillance, we placed a camera on the room ceiling to provide (uncertain) measurements used to initialize distributions of reachable sets for the robotic evaders. The sampling time for the discrete time system is 0.25 seconds. All stochastic reachability dynamic programs were considered for a horizon of $N = 5$. For the remaining parameters and details of the experimental setup see [5].

We looked at the distribution among the global camera states for a sequence of long runs of the robotic evaders to evaluate the sensitivity of the task allocation and stochastic reachability algorithm to the global objective weights $w_{\mathcal{G}}$. In addition to the case where both cameras have full view of the table, we also looked at the performance of the task allocation in the case that the surveillance cameras have decoupled areas of operation. Camera window, placed at $(x = 0, y = 0)$ was restricted in these scenarios to cover all space for which $y \leq 2.8m$ while camera door, placed at $(x = 0, y = 4.55)$, the space for which $y \geq 2.4m$. More results illustrating the performance of the system can be found where links [1] to videos of the full working system are provided.

1) Average behavior over long periods for normal and high value targets: As already mentioned, in Figure 2 we show the percentage of time that the combined system of cameras spends on each state. In all cases states $0, 1, 2, 3$ correspond to seeing none, evader 1, evader 2, and both respectively. Each color graph corresponds to a different reward weight hierarchy, indicated by the legend. In all tests, the robotic evaders were moving randomly on the observation table avoiding collisions with each other and with the table borders. The blue bar depicts the time percentages for state weight choices $w_{\mathcal{G}}$ where $w_{S_3^G} >> w_{S_2^G} = w_{S_1^G} > w_{S_0^G} = 0$. The immediate result is that the advisory controller allocates surveillance tasks among the cameras with preference given to the coverage of both evaders simultaneously.

Changing the state weights (green bar) such that $w_{S_3^G} = w_{S_2^G} >> w_{S_1^G} > w_{S_0^G} = 0$ we notice an increase in the tracking of evader 2 with respect to evader 1. However, the total coverage (time in state S_3 plus time in S_2 or S_1) is lower for both evaders because of cases where evaders are spatially close and allocating tasks that achieve tracking of evader 2 fail, affecting both the percentage of seeing both (state 3) and just evader 2 (state 2). This highlights the need to clearly distinguish the state weights whenever a high value target is part of the evader group. The red bar shows the limit case where $w_{S_2^G} >> w_{S_1^G} = w_{S_3^G} > w_{S_0^G} = 0$. Clearly the high value target (evader 2) is well tracked.

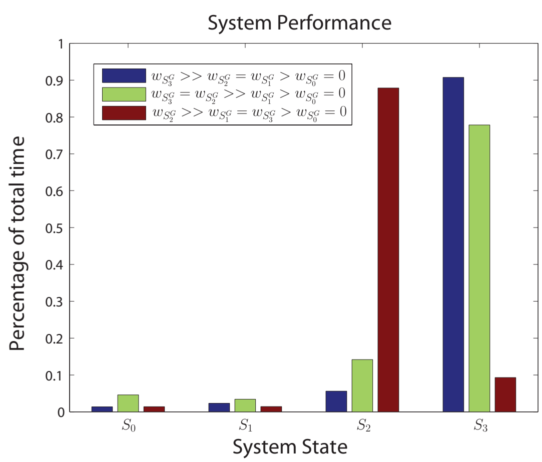


Fig. 2. Performance results over 30 minute experiments of the autonomous allocation system. Each bar color represents a different hierarchy of between global states weights, indicated by the legend. As expected, the percentage of time spent by the system in any of the states is directly controlled by the hierarchy choice.

2) Switching analysis for cameras with partial table coverage: With this experiment we illustrate the performance of the supervising task allocation controller when each camera can only cover a portion of the table area while robotic evaders move from one area to the other. In this situation, the task allocation manages to keep track of both robots at all times, successfully handing off the evader to the other camera when the limit is reached (with the only cooperation coming in the form of task assignment from the advisory controller).

Figure 3(a) shows the trajectories of both evaders and cameras until the point of the transition. Camera door is tracking evader 1 and camera window evader 2 since $Pr(T_1)$ and respectively $Pr(T_2)$ are satisfied with high probability (Figure 3(b)). At the time before the transition (Figure 3(c)) the probability for camera door to acquire evader 2 ($Pr(A_2)$) and the probability to do it while keeping track of evader 1 ($Pr(T_1 A_2)$) start rising. The equivalent situation with evader 1 is observed for camera window. At the transition point, camera window cannot track evader 2 over the entire time horizon anymore ($Pr(T_2) = 0$) and the advisory controller switches the task allocation automatically. Camera window is commanded to acquire evader 1 and camera door to acquire evader 2 since $Pr(A_1)$ and $Pr(A_2)$ are both high (Figure 3(d)). After the switch, Figure 3(e) confirms that both cameras can track the new captured evaders with very high probability. Figure 3(f) shows the camera and evader trajectories after the switch. Note that the top evader (evader 1) of Figure 3(a) is now the bottom one and vice versa. Several similar transition tests were carried out and similar analysis illustrated that the proposed task allocation scheme handles the case of partial coverage camera systems successfully.

E. Timing

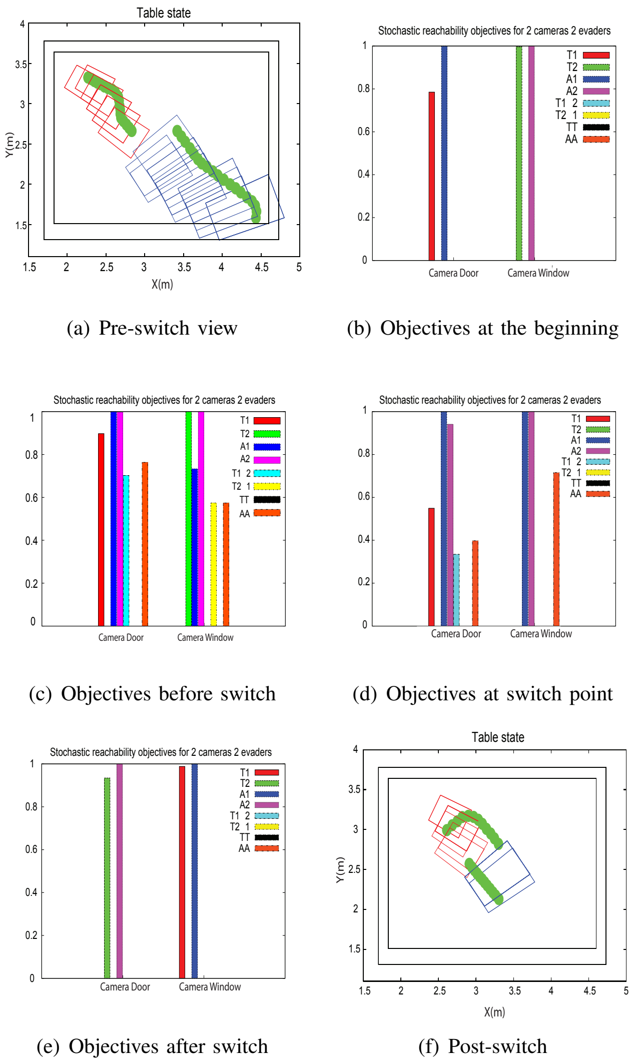
In the described setup the sampling time is $T_s = 0.25s$ during which we were able to solve up to 50 reach-avoid problems. In the context of automatic surveillance for n_e evaders, n_c cameras and n_o objectives, the number of possible reach-avoid problems is given by $n_e n_c n_o$ while the number of possible allocations is given by $n_o^{n_c}$. Using the proposed method, we can easily deal with 5 cameras, 3 evaders and the objectives of track, acquire and track while acquire (45 reach-avoid problems and 243 possible allocations). The cost calculation associated with each possible allocation is a multiplication of scalar values. An equivalent solution on the augmented system using full space gridding would have to deal with a 20-dimensional system (2 inputs per camera).

IV. Conclusion

In this work we introduced a simple advisory controller that allocates tasks to multi-agent systems whose objectives can be expressed by stochastic reachability. The enumeration carried out to determine the optimal allocation utilizes reach-avoid success probabilities that are assumed to correspond to state transitions in the local agent systems. This is done through simple preference weights applied on the global states of the combined system, assuming particular allocations among the individual agents.

The main observation of this paper is the possibility to deal with intractable problems (space gridding and curse

[1] http://control.ee.ethz.ch/~nkarioto/



(a) Pre-switch view
(b) Objectives at the beginning
(c) Objectives before switch
(d) Objectives at switch point
(e) Objectives after switch
(f) Post-switch

Fig. 3. Switching performance

of dimensionality) of multi-agent DTSHS control (in the form of Markov control processes). The presented idea is built upon specific independence assumptions and requires an agent task allocator based on stochastic reachability. The proposed controller is shown to work exceptionally well on a surveillance system of two movable pan-tilt cameras and two robotic evaders. More results and system analysis are provided in the referenced videos. Future work includes a thorough analysis of the optimality of the task allocation scheme (in comparison to alternate distributed control algorithms) and a rigorous examination of the complexity growth as more cameras and evaders are considered.

References

[1] P. Natarajan, T. Hoang, K. Low, and M. Kankanhalli, "Decision-theoretic approach to maximizing observation of multiple targets in multi-camera surveillance," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pp. 155–162, International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[2] B. Dieber, C. Micheloni, and B. Rinner, "Resource-aware coverage and task assignment in visual sensor networks," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 21, no. 10, pp. 1424–1437, 2011.

[3] P. Salvagnini, M. Cristani, A. Del Bue, and V. Murino, "An experimental framework for evaluating ptz tracking algorithms," *Computer Vision Systems*, pp. 81–90, 2011.

[4] D. Raimondo, N. Kariotoglou, S. Summers, and J. Lygeros, "Probabilistic certification of pan-tilt-zoom camera surveillance systems," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 2064–2069, IEEE, 2011.

[5] N. Kariotoglou, D. M. Raimondo, S. Summers, and J. Lygeros, "A stochastic reachability framework for autonomous surveillance with pan-tilt-zoom cameras," in *Conference on Decision and Control (CDC)*, Dec. 2011.

[6] A. Cenedese, F. Cerruti, M. Fabbro, C. Masiero, and L. Schenato, "Decentralized task assignment in camera networks," in *Decision and Control (CDC), 2010 49th IEEE Conference on*, pp. 126–131, IEEE, 2010.

[7] O. Avni, F. Borrelli, G. Katzir, E. Rivlin, and H. Rotstein, "Scanning and tracking with independent cameras–a biologically motivated approach based on model predictive control," *Autonomous Robots*, vol. 24, no. 3, pp. 285–302, 2008.

[8] I. Everts, N. Sebe, and G. Jones, "Cooperative object tracking with multiple ptz cameras," in *the 14th Intl Conf. on Image Analysis and Processing*, Citeseer, 2007.

[9] S. M. Huck, N. Kariotoglou, S. Summers, D. M. Raimondo, and J. Lygeros, "Design of importance-map based randomized patrolling strategies," in *Complexity in Engineering (COMPENG), 2012*, pp. 1–6, IEEE, 2012.

[10] S. Summers and J. Lygeros, "Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem," *Automatica*, vol. 46, no. 12, pp. 1951 – 1961, 2010.

[11] S. Summers, M. Kamgarpour, C. Tomlin, and J. Lygeros, "A stochastic reach-avoid decision problem with random sets," 2011. Submitted.

[12] N. Ozay, U. Topcu, R. Murray, and T. Wongpiromsarn, "Distributed synthesis of control protocols for smart camera networks," in *Cyber-Physical Systems (ICCPS), 2011 IEEE/ACM International Conference on*, pp. 45–54, IEEE, 2011.

[13] T. Wongpiromsarn, U. Topcu, and R. Murray, "Receding horizon control for temporal logic specifications," in *Proceedings of the 13th ACM international conference on Hybrid systems: computation and control*, pp. 101–110, ACM, 2010.

[14] X. Ding, C. Belta, and C. Cassandras, "Receding horizon surveillance with temporal logic specifications," in *Decision and Control (CDC), 2010 49th IEEE Conference on*, pp. 256–261, IEEE, 2010.

[15] J. Park, P. Bhat, and A. Kak, "A look-up table based approach for solving the camera selection problem in large camera networks," in *Proceedings of the International Workshop on Distributed Smart Cameras (DCS06)*, 2006.

[16] S. Summers and J. Lygeros, "A Probabilistic Reach-Avoid Problem for Controlled Discrete Time Stochastic Hybrid Systems," in *IFAC Conference on Analysis and Design of Hybrid Systems, ADHS*, (Zaragoza, Spain), Sept. 2009.

[17] D. P. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete-Time Case*. Athena Scientific, February 2007.

[18] F. Ramponi, D. Chatterjee, S. Summers, and J. Lygeros, "On the connections between PCTL and dynamic programming," in *HSCC '10: Proceedings of the 13th ACM international conference on Hybrid systems: computation and control*, (New York, NY, USA), pp. 253–262, ACM, 2010.

[19] S. Summers, M. Kamgarpour, C. Tomlin, and J. Lygeros, "A Stochastic Reach-Avoid Problem with Random Obstacles," in *HSCC (Hybrid Systems: Computation and Control)*, (Chicago, USA), Apr. 2011.

[20] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, "Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems," *Automatica*, vol. 44, pp. 2724–2734, November 2008.

[21] R. Bellman, *Dynamic Programming*. Princeton, New Jersey, USA: Princeton University Press, 1957.