Simultaneous Team Assignment and Behavior Recognition from Spatio-temporal Agent Traces

Gita Sukthankar and **Katia Sycara**
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
gitars@cs.cmu.edu and katia+@cs.cmu.edu

Abstract

This paper addresses the problem of activity recognition for physically-embodied agent teams. We define team activity recognition as the process of identifying team behaviors from traces of agent positions over time; for many physical domains, military or athletic, coordinated team behaviors create distinctive spatio-temporal patterns that can be used to identify low-level action sequences. This paper focuses on the novel problem of recovering *agent-to-team* assignments for complex team tasks where team composition, the mapping of agents into teams, changes over time. Without *a priori* knowledge of current team assignments, the behavior recognition problem is challenging since behaviors are characterized by the aggregate motion of the entire team and cannot generally be determined by observing the movements of a single agent in isolation. To handle this problem, we introduce a new algorithm, Simultaneous Team Assignment and Behavior Recognition (STABR), that generates behavior annotations from spatio-temporal agent traces. The proposed approach is able to perform accurate team behavior recognition without an exhaustive search over the combinatorial space of potential team assignments, as demonstrated on several scenarios of simulated military maneuvers.

Introduction

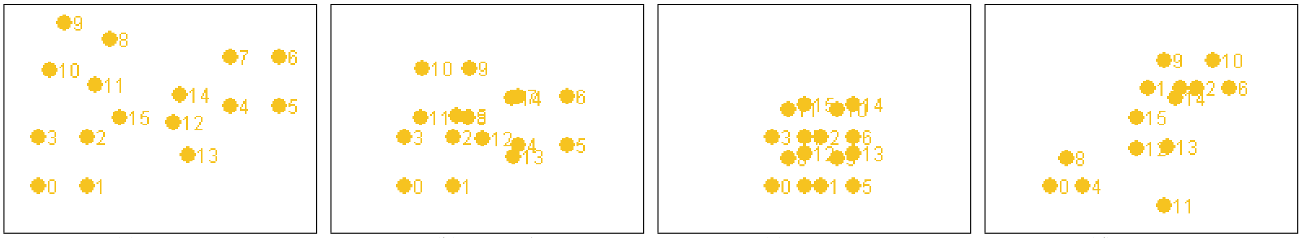
Although there has been considerable research on the problem of single-agent behavior recognition, there has been substantially less work on multi-agent behavior recognition. Most of the previous work makes one of two assumptions: (1) each agent is a decoupled entity that can be analyzed individually using a single-agent activity inferencing algorithm or (2) the agents are always working together and can be analyzed as a single cohesive entity represented by a high-dimensional feature vector. In either case, team composition is generally assumed to be *static*; this paper specifically addresses the problem of behavior recognition for teams with *dynamic* team composition. In scenarios with dynamic team composition, agents are assumed to be independent entities that coordinate with other agents to perform team behaviors; teams disband when the behavior is complete, merge with other teams to create larger formations, and occasionally split into subteams to perform multiple behaviors in parallel. Relaxing the assumption of static team assignment is desirable because it enables the analysis of more complex team tasks. However this makes the team behavior recognition problem substantially more difficult since behaviors are characterized by the aggregate motion of the entire team and cannot generally be determined by observing the movements of a single agent in isolation.

In this paper, we introduce a new algorithm, Simultaneous Team Assignment and Behavior Recognition (STABR), that recovers both team assignments and behavior annotations from traces of agent position over time. STABR leverages information from the spatial relationships of the team members to create sets of potential team assignments at selected time-steps. These spatial relationships are efficiently discovered using a randomized search technique, RANSAC (Fischler & Bolles 1981), to generate potential team assignment hypotheses. Sequences of team assignment hypotheses are evaluated using dynamic programming to derive a parsimonious explanation for the entire observed spatio-temporal trace. To prune the number of hypotheses, potential team assignments are fitted to a parameterized team behavior model; poorly-fitting hypotheses are eliminated before the dynamic programming phase. The proposed approach is able to perform accurate team behavior recognition without exhaustive search over the partition set of potential team assignments, as demonstrated on several scenarios of simulated military maneuvers.

The remainder of this paper is organized as follows. The next section discusses related work on practical applications of team behavior recognition. Then, we introduce our problem formulation and describe our algorithm, STABR (Simultaneous Team Assignment and Behavior Recognition). We present experiments in the domain of military formation recognition and discuss results.

Related Work

Multi-agent plan recognition has been developed for both athletic and military domains. To recognize athletic behaviors, researchers have exploited simple region-based (Intille & Bobick 1999) or distance-based (Riley & Veloso 2002) heuristics to build accurate, but domain-specific classifiers. For instance, based on the premise that all behaviors always occur on the same playing field with a known number of en-



(a) $t = 0$ (b) $t = 24$ (c) $t = 50$ (d) $t = 80$

Figure 1: (a) An example scenario with three teams of 4 agents, $(\{a_0, \ldots, a_3\}, \{a_4, \ldots, a_7\}, \{a_8, \ldots, a_{11}\})$ and four singleton agents $(a_{12}, \ldots, a_{15})$; (b) teams maneuver while maintaining formation and converge to central area; (c) the three teams disband and regroup into four teams of 3 agents; (d) the various teams scatter as units. The interleaving of agent formations, the presence of singletons and observation noise (suppressed here) makes the team assignment and behavior recognition challenging.

tities, it is often possible to divide the playing field into grids or typed regions (e.g., goal, scrimmage line) that can be used to classify player actions. Our algorithm does not rely on the presence of these external landmarks; however if such features exist, they can be incorporated into our framework both to reduce the number of team assignments considered and to potentially improve the behavior recognition accuracy. Previous work in this area typically assumes a known agent-team composition whereas our research focuses on behavior recognition for teams with dynamic composition.

There has also been work on extending single-agent plan recognition frameworks (Tambe & Rosenbloom 1995; Bui 2003), both to create symbolic (Tambe 1996) and probabilistic (Saria & Mahadevan 2004) multi-agent plan recognition frameworks. These efforts have focused on the use of temporal behavior models and do not extensively utilize spatial information; such models have also been employed to detect teamwork failures (Kaminka & Tambe 2000) and agent-coordination termination (Saria & Mahadevan 2004). Due to the difficulty of acquiring reliable location data for multiple entities, much of the research has been evaluated in simulation; however improvements in sensor technology such as the microwave position system described in (Beetz, Kirchlechner, & Lames 2005) should make real-world deployments possible in the future.

Problem Formulation

We formulate our problem as follows. Let $\mathcal{A} = \{a_0, a_1, \ldots, a_{N-1}\}$ be the set of agents in the scenario. A **team** consists of a subset of agents, and we require that an agent only participate in one team at any given time; thus a **team assignment** is a set partition on $\mathcal{A}$. An agent that is not currently a member of any team is known as a **singleton**, and is unrestricted in its motion. By contrast, the agents in a team are constrained to move according to a set of **team behaviors**, $\mathcal{B}$. The subset of behaviors available to a given team is specified by the domain and can depend on the number of agents in the formation and their relative configurations. For instance, the domain could specify that four agents in a square formation may execute a "wheel" (formation advances in an arc by rotating about a corner), but not a "pivot" (formation rotates about its center), which may be restricted to teams of three agents. In the course of a scenario, agents (either singletons or subsets of disbanding teams) can assemble into new teams; similarly, teams can disband to enable their members to form new teams or to operate as singletons. Thus the team assignment is expected to change over time during the course of a scenario. The team assignments over time and the behavior executed by each team are hidden from our system. We assume that our input consists only of a **spatio-temporal trace**, which is a sequence of noisy observations of the 2D position of each agent through time, $\mathbf{a}_i(t) \in \Re^2$. We illustrate this with an example: Figure 1 shows several frames from a scenario with 16 agents. In Figure 1(a), 12 of the agents are arrayed in three teams of four agents in a square formation, $(\{a_0, \ldots, a_3\}, \{a_4, \ldots, a_7\}, \{a_8, \ldots, a_{11}\})$, with the remaining four agents as singletons. In Figure 1(b), the squares are converging towards the central area and the formations are starting to interleave. In Figure 1(c), the squares are disbanding and those are regrouping into four groups of three, arrayed as triangles. Finally, in Figure 1(d), the triangles are moving away from the central area. For illustration purposes, observation noise is not shown in this figure.

Our goal is to recover a team and a behavior assignment for every agent $a_i \in \mathcal{A}$ at every time-step t. This can be succinctly expressed in the form of two tables: (1) an agent-to-team assignment $a_i(t) \mapsto \mathcal{S} \subset \mathcal{A}$; (2) a team-to-behavior assignment $S_j(t) \mapsto b \in \mathcal{B}$. It is important to note that one cannot, in general, infer the behavior of a team by examining the motion trace of any single agent. Similarly, one cannot assign an agent to a team without confirming that the behavior of the team is legal.

Ideally, one may wish to consider every legal agent-to-team assignment and team-to-behavior assignment at every time-step and then select the sequence that best matches the observed data. However, a straightforward implementation of this idea is computationally infeasible. The pool of potential agent-to-team assignments grows very quickly with the number of agents; this is equivalent to the number of partitions of a set, and is given by the *Bell number* of the set (Rota 1964):

$$B_n = \sum_{0}^{n} S(k, n),$$

where $S(k,n)$ denotes a Stirling number of the second kind,

$$S(k,n) = S(k-1,n-1) + kS(k,n-1) \qquad 1 \leq k \leq n$$
$$S(n,n) = S(n,1) = 1.$$

For instance, the number of team assignments in the 16-agent example shown in Figure 1, $B_{16} > 10^{10}$. Clearly, examining every potential team assignment at even a single time-step is infeasible. And naively evaluating all of the possible combinations of partitions over the entire spatio-temporal sequence further increases the complexity in an exponential manner.

Fortunately, a closer examination of the problem reveals structure that can be exploited to generate a computationally-feasible solution. The key observations behind our algorithm are summarized as follows. First, at each time-step, the relative positions of the agents in a team is constrained by the spatial configuration of the formation. Even though it may not be possible to unambiguously determine from a single time-step that an observed subset of agents is arrayed in a particular formation, one can profitably employ a static analysis of agent positions to generate hypotheses of valid team assignments and behaviors. Second, although an analysis of the motion of a single agent may not be sufficient to infer its behavior, an examination of the aggregate movement of several agents in isolation (i.e., a hypothesized team) generates significant information about team behavior. Third, by defining appropriate cost functions for the sequence, one can employ dynamic programming to dramatically reduce the time needed to find good sequences of team and behavior assignments through time. The next section details each of these ideas and describes how they contribute to the design of the STABR algorithm.

The STABR Algorithm

STABR analyzes spatio-temporal traces in three stages. First, it performs a static analysis of agent positions at each time-step to identify potential agent configurations that may correspond to known formations; these are used as an initial set of agent-to-team assignment hypotheses in later stages. STABR maintains multiple potentially-conflicting assignments for an agent, if there is spatial support. Second, STABR examines hypothesized team assignments in isolation and determines whether they have sufficient local spatio-temporal support. Pruning unlikely hypotheses at this stage is crucial since it greatly affects the performance of the last stage. This analysis also enables STABR to determine plausible behavior assignments for each of the surviving hypotheses. Third, these agent-to-team hypotheses are used to generate complete partitions over the agents. In the worst case, this state space could be exponential in the number of surviving hypotheses, underscoring the benefits of pruning. STABR then organizes the states (partitions) over the spatio-temporal sequence in the form of a lattice and employs dynamic programming to identify minimal cost solutions. These correspond to agent-to-team and team-to-behavior assignments that are a good fit to the observed sequence.

Static identification of agent formations

The first stage of the recognition process is to identify potential team assignments, based on static spatial cues. We do this by matching agent positions to pre-specified geometric formation templates; this enables the recovery of more complicated team relationships than the standard approach of clustering agents into teams based solely on proximity (see Results for a comparison of approaches). STABR employs a statistically-robust technique, Random Sampling and Consensus, RANSAC (Fischler & Bolles 1981), to generate and test potential team assignment hypotheses at selected time-steps. For each formation template, agents are drawn uniformly and at random from both the template and the scenario. These point correspondences are used to generate a transform hypothesis to project the remaining template points into the scenario coordinate frame.

We define the set of legal transforms to be the class of similarity transforms (rotation, translation, and scaling); these can be parameterized in homogeneous coordinates as follows:

$$\mathbf{T} = \begin{bmatrix} s\cos(\theta) & s\sin(\theta) & x \\ -s\sin(\theta) & s\cos(\theta) & y \\ 0 & 0 & 1 \end{bmatrix}.$$

Then we apply the static formation recognition scheme described in (Sukthankar & Sycara 2006) to efficiently identify matching transforms; this method is summarized below. The randomly sampled minimal set of point correspondences is expressed in homogeneous coordinates as the 3×3 matrices $\mathbf{A}$ and $\mathbf{B}$ respectively. Since $\mathbf{B} = \mathbf{TA}$, we can recover $\mathbf{T}$ directly using matrix inversion. The match quality of the transform hypothesis $\mathbf{T}$ is assessed by projecting the coordinates of the remaining agents, as given by the template, into the scenario coordinate frame using $\mathbf{T}$. If the predicted positions are sufficiently close to the observations, the template is accepted as valid and these agents are assigned to a team.

Since RANSAC stochastically searches the space of possible transforms it is not guaranteed to find the best match. However the following formula can be used to determine the number of iterations that are required to find the best match with a specified probability of success (Xu & Zhang 1996):

$$m = \left\lceil \frac{\log(1-P)}{\log[1-(1-\epsilon)^s]} \right\rceil.$$

P is the target probability (e.g., $P = 0.99$ means the best match is found 99% of the time). s is the number of elements required to define the minimal set ($s = 2$ since a similarity transform requires 2 pairs of point correspondences). ϵ is the expected fraction of outliers in the data set. RANSAC is highly efficient at detecting templates that consist of many agents, since the number of iterations needed to achieve the desired probability is independent of team size. Detecting small teams in a scenario with many distracting agents is a harder problem since the other agents function as outliers. In the example scenario, where 75% of the points are effectively outliers for the square formation, a reliable detection of that template only requires 71 iterations, whereas an exhaustive search through the space would need ${}^{16}C_4 = 1820$ iterations.

Spatio-temporal analysis of individual teams

The first stage of the algorithm identified, independently for each time-step, a set of hypothetical team assignments. The second stage identifies those team assignments that have significant temporal support, and generates behavior hypotheses for each such team that are consistent with the observed positions. The inability to find a plausible behavior to explain the motion of a hypothesized team is a strong indicator that the hypothesis does not correspond to a real team, but is rather a visual illusion caused by a coincidental configuration of agents.

The behavior recognition proceeds on a team-by-team basis. Each team is independently evaluated over the temporal intervals during which it was detected against a set of parameterized team behavior models. Although STABR can employ arbitrary motion models, in this paper, we model team behaviors as a set of constrained rigid-body transforms, such as "advance" (pure translation in the forward direction), "wheel" (rotation about a forward corner), and "pivot" (rotation about team centroid). If the team's movement does not fit any known behavior, the assignment is discarded since it is likely to have been the result of a spurious detection. This is best illustrated by an example. Consider a team hypothesis that assigns three agents, (a_1, a_2, a_3) to a "triangle" formation. Domain knowledge specifies the set of behaviors that are available to each formation, and this imposes constraints on their observed motion. For instance, a triangle formation may only be allowed to pivot about its centroid or advance as a wedge. The algorithm evaluates the likelihood of explaining the observed team positions $\{\mathbf{a}_1(t), \mathbf{a}_2(t), \mathbf{a}_3(t)\}$, using each behavior, over a sliding time window. Each behavior is characterized by a small number of parameters; given these parameters and an initial agent configuration, the behavior specifies the position of each agent in the formation over the time interval. For example, the pivot behavior for a triangle formation is parameterized by the angular velocity, ω. $\forall i \in \{1, 2, 3\}$:

$$\mathbf{a}_i(t) = \begin{bmatrix} \cos\omega(t-t_0) & \sin\omega(t-t_0) \\ -\sin\omega(t-t_0) & \cos\omega(t-t_0) \end{bmatrix} [\mathbf{a}_i(t_0) - \bar{\mathbf{a}}] + \bar{\mathbf{a}}$$

where $\bar{\mathbf{a}}$ denotes the (stationary) centroid of the triangle formation. Our algorithm attempts to fit the model to the observed data. In this example, if a_1, a_2 and a_3 were really executing a pivot, the algorithm should recover an ω that matches the observations well (in a least-squares sense). On the other hand, if a_1, a_2 and a_3 were actually translating, then the pivot behavior would fail to match (for any choice of ω). Thus, we iterate through each behavior and prune those behaviors that fail to match and (most importantly) prune those team hypotheses that cannot be explained by any legal behavior. The computational benefits of pruning team hypotheses are discussed in the next section.

Explaining sequences of hypotheses

The final stage of STABR searches the space of team assignment and behavior recognition hypotheses generated by earlier stages for a consistent explanation over the entire spatio-temporal trace. In general, there may be several consistent explanations for the given observed agent movements; for instance, it is always possible to explain any trace as a coincidental convergence of uncoordinated singleton movement (though this would be highly improbable). We employ a cost function that encodes the intuition that an explanation that requires fewer changes in team assignment and behavior is preferable (Sukthankar & Sycara 2005). Given this cost function, we apply dynamic programming over the sequence to efficiently find the minimal-cost solution.

For every time slice, STABR generates a list of potential set partitions from the team assignment labels returned by the RANSAC template matching and validated by spatio-temporal behavior analysis. This list of set partitions represents a potential world state for that time slice; each world state contains a team assignment for every agent such that no agent is assigned to multiple teams (see Discussion for ramifications of relaxing this requirement). Generating a list of consistent world states is exponential in the number of team assignment hypotheses but is dramatically faster than considering the Bell number of total set partitions at that time step. Thus, effective pruning of team assignment hypotheses using spatio-temporal behavior analysis in earlier stages can greatly reduce running time.

Any sequence through this set of partitions is both consistent (all agents are assigned to teams and no agent is assigned to multiple teams) and supported by local spatio-temporal evidence. We use a cost function to select the solution that most parsimoniously explains the scenario. This can be formulated as a shortest path problem through the space of consistent team and behavior assignments and solved efficiently using dynamic programming. The cost, C_{T-1} of explaining the entire sequence can be computed using:

$$C^q_{t+1} = \min_{\forall p}\{C^p_t + D_{p,q}\} + \gamma|q|$$

where p and q are potential world states at time-steps t and $t+1$ respectively, $|q|$ denotes the number of teams in the partition, and γ is a domain-specific parameter that controls the degree to which STABR favors sequences with large teams. D is a "distance" between states that captures differences in both team membership and behavior:

$$D_{p,q} = \sum_{\forall a_i \in \mathcal{A}} \beta I[\mathrm{b}_p(a_i), \mathrm{b}_q(a_i)] + \tau I[\mathrm{t}_p(a_i), \mathrm{t}_q(a_i)]$$

where $I[.,.]$ is an indicator variable; $b_p(a_i)$ and $b_q(a_i)$ return the behavior label for agent a_i under states p and q respectively; $t_p(a_i)$ and $t_q(a_i)$ return the team assignment for agent a_i under states p and q respectively; β and τ are domain-specific parameters which can be tuned to improve recognition accuracy.

Experiments

We evaluate STABR on a set of scenarios of simulated military formations. The simulator generates traces for the position of each agent, corrupted with iid Gaussian observation noise and emits ground-truth data of the correct team assignments and behavior for the scenario. STABR processes this data and generates a team-assignment and a behavior for each agent, at every time-step. Our evaluation metrics are summarized below:

1. **team assignment accuracy**: We score, at each time-step, whether the team assignment for each agent is correct. We employ a conservative metric and require the team memberships to match exactly; e.g., the absence of a single agent in a k-member team counts as k errors — one for each of the incorrectly-labeled agents — rather than a single assignment error. Team assignment accuracy is plotted over time (Figure 2(a)) to show results on a particular scenario and averaged over the scenario to generate aggregate results (Table 1).
2. **behavior recognition accuracy**: This measures the quality of behavior recognition and is computed in an analogous manner as team assignment accuracy, using the same conservative metric.
3. **hypothesis set size**: We examine the number of hypotheses that are considered by STABR during various stages. This enables us to assess the contribution of spatio-temporal pruning.

Each of the following experiments examines a particular aspect of STABR to better understand its contributions. In all of these experiments, the STABR parameters were set to $\beta = \tau = 1$ and γ=0.1, unless otherwise specified.

The first experiment evaluates the benefits of employing the RANSAC-based formation template approach to identifying teams against a standard proximity-based clustering. K-means and agglomerative clustering are two popular unsupervised clustering methods (Duda, Hart, & Stork 2001) that are frequently employed to group agents into teams. Since the former requires that the number of clusters be externally-specified, we chose to compare STABR against the latter. In this experiment, the first stage of STABR is replaced with agglomerative clustering, where groups of proximal agents were aggregated into teams. Figure 2(a) presents the team assignment accuracy for both algorithms on the scenario shown in Figure 1. Agglomerative clustering and RANSAC both perform well when the agent teams are well-separated. However, as the formations begin to interleave, the accuracy of agglomerative clustering deteriorates rapidly. This is because agents that are proximal should frequently be assigned to different teams. The transient drop in accuracy near $t = 50$ corresponds to frames where 12 agents simultaneously transition from three groups of 4 agents to four groups of 3 agents over the span of a few frames; although either assignment would be correct during this interval, the ground truth file arbitrarily selects a single transition point, and STABR's explanation is marked as incorrect. Results on behavior recognition (not shown) mirror those for team assignment, since correctly identifying an agent's behavior generally requires the algorithm to also group it into the correct team.

Table 1 summarizes the agent team assignment accuracy for STABR over several scenarios. While proximity-based clustering can handle the simplest scenario, it copes poorly with the interleaved formations in more complex scenarios.

The second experiment studies the contribution of the spatio-temporal behavior recognition, not in terms of accuracy but rather in terms of reducing the number of hypotheses from which world-states need to be generated. Since the execution time of STABR's last stage can grow exponentially with the size of this hypothesis set, it is important to reduce the set of team assignment hypotheses (without jeopardizing accuracy). Figure 2(b) shows (in semi-log scale) the size of the hypothesis set before and after spatio-temporal pruning along with the actual size of the consistent set (which is not actually known until stage 3). As can be seen, the spatio-temporal behavior recognition dramatically reduces the number of hypotheses that need to be considered by the third stage — without adversely affecting accuracy.

Table 1: Agent team assignment accuracy, averaged over all agents and time for a variety of scenarios. The benefits of RANSAC over proximity-based clustering are clearly evident. The scenario illustrated in Figure 1 is Scenario D.

	Cluster	RANSAC
Scenario A	95.8%	97.8%
Scenario B	57.0%	99.3%
Scenario C	36.0%	99.5%
Scenario D	18.3%	98.5%
Scenario E	0.0%	95.0%

Discussion

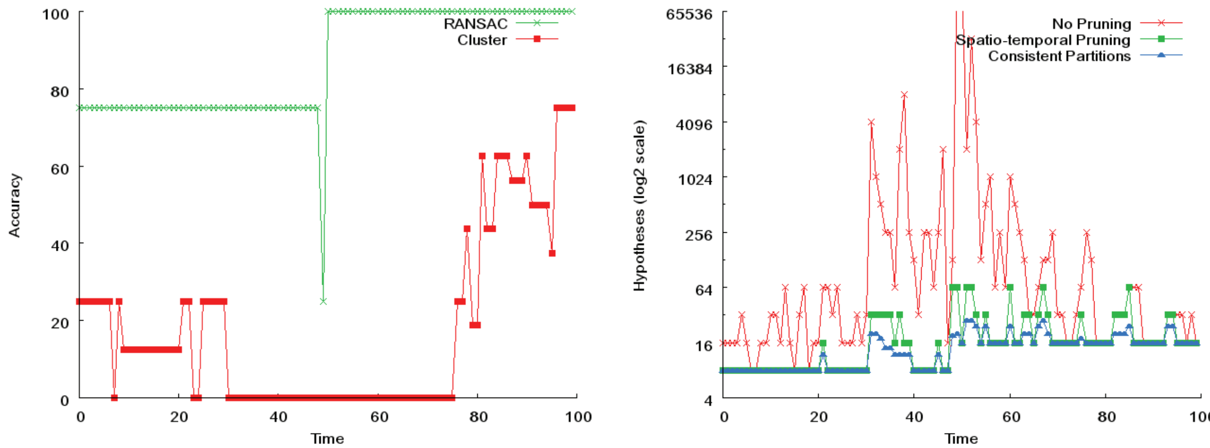
In military scenarios, group assignment can be quite challenging because modern forces often split into multiple disconnected parts (e.g., far-ranging scouts, small diversion groups, and flanking elements). Recognizing what the force is doing is often possible once it is clear which units are involved. STABR correctly recovers team assignments even in cases of non-spatially contiguous divisions that foil standard clustering approaches to team assignment.

The STABR approach is based on the following intuitions:

1. Initial agent-to-team assignments can be made on the basis of static spatial cues.
2. The aggregate agent movement for an incorrect team assignment will generally fail to match any behavior model; this can be exploited to prune poor team assignments thus speeding computation.
3. Desirable team and behavior assignments "explain" the activities of a large number of agents over long chunks of the spatio-temporal sequence.

The scenarios presented in this paper illustrate the operation of STABR in environments that lack the external cues used by other multi-agent plan recognition approaches, such as landmarks, cleanly-clustered agent teams, and extensive domain knowledge. We believe that when such cues are available they can be directly incorporated into STABR, both to improve accuracy and to prune hypotheses. STABR provides a principled framework for reasoning about dynamic team assignments in spatial domains.

The chicken-and-egg problem of simultaneous team assignment and behavior recognition is conceptually similar to other AI problems, such as image segmentation/object recognition in computer vision. During the image segmen-



(a) team assignment accuracy (b) #hypotheses

Figure 2: (a) Team assignment accuracy for STABR comparing agglomerative clustering with RANSAC on the scenario shown in Figure 1. Clearly proximity-based clustering is ineffective when agent formations are in close proximity. (b) Pruning team assignment hypotheses based on spatio-temporal behavior recognition drastically reduces the number of hypotheses that STABR considers. The number of hypotheses that remain after pruning closely follows the actual size of the consistent partitions.

tation phase, pixels are assigned to objects that are then classified by an object recognition algorithm. The choices made by segmentation affect the quality of the object recognition; thus one can favor segmentations that generate recognizable objects. In the same way, STABR favors team assignments that produce recognizable behaviors.

Although STABR was designed specifically for the analysis of spatio-temporal traces, we believe that STABR can also be applied to a broader class of problems, where spatial information does not govern team structure. For instance, agents could be assigned to teams based on observed inter-agent communication patterns in conjunction with role templates that represent functional relationship between agents. In such domains, it may be necessary to relax the restriction on team membership to allow an agent to simultaneously belong to multiple teams. This change would simplify the process of generating valid world states since it removes the need for consistency checking at the expense of increasing the number of potential hypotheses that need to be considered.

Conclusion

This paper introduces a new algorithm, STABR, that generates both behavior annotations and team assignments from spatio-temporal agent traces. The proposed approach performs accurate team behavior recognition without an exhaustive search over the combinatorial space of potential team assignments. Experiments on several simulated military maneuvers demonstrate that STABR is accurate at both team assignment and behavior recognition. In future work, we will integrate symbolic plan recognition into STABR to analyze hierarchically-structured observation sequences acquired from military urban training operations.

Acknowledgments

The authors would like to thank Rahul Sukthankar for his thoughtful comments and proof-reading. This work has been supported by AFOSR grant F49620-01-1-0542.

References

Beetz, M.; Kirchlechner, B.; and Lames, M. 2005. Computerized real-time analysis of football games. *Pervasive Computing* 4.

Bui, H. 2003. A general model for online probabilistic plan recognition. In *Proc. IJCAI*.

Duda, R.; Hart, P.; and Stork, D. 2001. *Pattern Classification*. Wiley & Sons, Inc.

Fischler, M., and Bolles, R. 1981. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM* 24(6).

Intille, S., and Bobick, A. 1999. A framework for recognizing multi-agent action from visual evidence. In *Proc. AAAI*.

Kaminka, G., and Tambe, M. 2000. Robust agent teams via socially attentive monitoring. *Journal of Artificial Intelligence Research* 12:pp.105–147.

Riley, P., and Veloso, M. 2002. Recognizing probabilistic opponent movement models. In Birk, A.; Coradeschi, S.; and Tadorokoro, S., eds., *RoboCup-2001: Robot Soccer World Cup V*. Springer Verlag.

Rota, G. 1964. The number of partitions of a set. *American Mathematical Monthly* 71.

Saria, S., and Mahadevan, S. 2004. Probabilistic plan recognition in multiagent systems. In *Proceedings of International Conference on AI and Planning Systems*.

Sukthankar, G., and Sycara, K. 2005. A cost minimization approach to human behavior recognition. In *Proc. AAMAS*.

Sukthankar, G., and Sycara, K. 2006. Robust recognition of physical team behaviors using spatio-temporal models. In *Proceedings of Workshop on Modeling Others from Observations (MOO 2005)*.

Tambe, M., and Rosenbloom, P. 1995. RESC: An approach to agent tracking in a real-time dynamic environment. In *Proc. IJCAI*.

Tambe, M. 1996. Tracking dynamic team activity. In *Proc. AAAI*.

Xu, G., and Zhang, Z. 1996. *Epipolar Geometry in Stereo, Motion, and Object Recognition*. Kluwer.