

On the Response of EMT-based Control to Interacting Targets and Models^{*}

Zinovi Rabinovich
nomad@cs.huji.ac.il

Jeffrey S. Rosenschein
jeff@cs.huji.ac.il

School of Computer Science and Engineering
Hebrew University
Jerusalem, Israel

ABSTRACT

A novel control mechanism was recently introduced based on Extended Markov Tracking (EMT) [9, 10]. In this paper, we present a study of its response to multiple interacting control goals. We show a simple extension that can be integrated into EMT-based control, and which provides it with the ability to handle several behavioral targets. Experimental support for the validity of this extension is provided. We also describe an experiment with a simulated robot, where EMT-based controllers interact and interfere indirectly via the environment. Experiments support the resilience of multi-agent EMT-based team control to potential conflicts that may appear within a team.

1. INTRODUCTION

Imagine a car moving quickly along a highway; a bend in the road approaches, and the car's autopilot considers possible actions. It faces two interacting and potentially interfering considerations: first, the car has to turn fast enough to fit the road's curvature, or it will run off the road; second, a car has to turn slowly enough to maintain its grip on the road, or it will spin out of control. Furthermore, not only its radial speed, but also its linear speed affects the car's successful negotiation of the bend in the road, and the autopilot has to consider (or at least respond to) the environment-enforced interaction between these two motion parameters. That is, the autopilot faces both explicit and implicit multi-objective control problems.

These can be solved using a variety of methods. For example, proper control strategy can be achieved through learning (as, for instance, in [5]). Depending on the controller representation, and if some basic control capabilities are available, behavior-based (e.g., [1, 2]) or expert-based (e.g., [4, 12, 8]) approaches can also be used to achieve the required balance. Both, however, have their limitations. For example, the behavior-based approach usually requires a hierarchy among basic, though potentially complex, behaviors, and at any single time only one behavior would actually be active. The

expert-based approach, on the other hand, provides a more fluent merger of basic advice from multiple sources, but concerns itself only with this merger and does not dictate how the basic advice is produced.

In this paper we consider the multi-criteria problem when basic control is performed using a control scheme based on Extended Markov Tracking (EMT) [9, 10]. EMT-based control provides a means to achieve a complex behavioral goal. We use a scaled additive mechanism directly integrated into EMT-based control to combine rationales for action with respect to multiple behavioral targets. This creates a fluent merger out of a well-defined underlying action choice mechanism, and we carry out experiments to test this new multi-target EMT control in an explicit multi-objective control scenario.

We then applied EMT-based control to implicit multi-objective control. We created a simulated robot setting using the Player/Stage simulator [6], in which an EMT-controlled robot had to follow a target. In this experiment, linear and radial speeds of the robot were controlled by two independent EMT-based controllers, which forced them to solve an implicit multi-objective problem, since the robot's motion correlates the two speeds, and thus the two controllers. In addition, we complicated the setting by providing controllers with inexact environment models. EMT-based control demonstrated strong resistance to multiple sources of system noise, and successfully handled the scenario.

The rest of the paper is organized as follows. Section 2 briefly describes the formal setting of Extended Markov tracking and a control scheme based on it, with Subsection 2.1 describing our extension of the additive combination mechanism to EMT-based control. We proceed in Section 3 to the experiments and experimental results of a simple multi-target problem. Section 4 describes our experiment with the simulated robot scenario of target following. We discuss in Section 5 our results and future work.

2. EXTENDED MARKOV TRACKING AND CONTROL

EMT-based control [9, 10] is a recently introduced framework for control of a stochastic environment. It relies on the fact that the system environment has a Markovian model of a form

$\langle S, s_0, A, T, O, \Omega \rangle$, where

- S is the set of all possible environment states;
- s_0 is the initial state of the environment (which can also be

^{*}First author is a student.

viewed as a distribution over S);

- A is the set of all possible actions applicable in the environment;
- T is the environment's probabilistic transition function: a mapping $T : S \times A \rightarrow \Pi(S)$. That is, $T(s'|a, s)$ is the probability that the environment will move from state s to state s' under action a ;
- O is the set of all possible observations. This is what the sensor input would look like for an outside observer;
- Ω is the observation probability function: a mapping $\Omega : S \times A \times S \rightarrow \Pi(O)$. That is, $\Omega(o|s', a, s)$ is the probability that one will observe o given that the environment has moved from state s to state s' under action a .

Extended Markov Tracking (or EMT for short) then uses this environment model to follow the system development, and maintains two estimators. First is the regular estimator for this type of environment — the state estimator, or state beliefs, expressed by a distribution vector over the system states $p_t \in \Pi(S)$. The second is the system dynamics estimator $PD : S \rightarrow \Pi(S)$, which can be seen as a stochastic function that “explains away” the changes in the system state beliefs seen so far.

To keep track of the actual system development, EMT dictates that its estimators be updated as follows:

- For state beliefs a Bayesian update is used. If at time t an action $a \in A$ was taken and then an observation $o \in O$ received, the new state beliefs p_{t+1} would be expressed by the formula:

$$p_{t+1}(s) \propto \Omega(o|s, a) \sum_{s'} T(s|a, s') p_t(s')$$

- For the system dynamics estimate EMT uses the Kullback-Leibler distance [3]

$$D_{KL}(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

to perform a conservative update based on the following convex optimization problem:

$$PD_{t+1}(\cdot|s) = \arg \min_{Q(\cdot|s)} \langle D_{KL}(Q(\cdot|s)||PD_t(\cdot|s)) \rangle_{p_t(s)}$$

$$\begin{aligned} & \text{s.t.} \\ & \forall s' \quad p_{t+1}(s') = \sum_s Q(s'|s) p_t(s) \\ & \forall s \quad \sum_{s'} Q(s'|s) = 1 \end{aligned}$$

That is, an explanation is sought for the change in system state beliefs from p_t to p_{t+1} , similar (as much as possible) to the old dynamics estimate PD_t . This update is denoted for short as $PD_{t+1} = H[p_{t+1}, p_t, PD_t]$.

Despite the fact that the optimization problem contained within the dynamics estimate update looks difficult, it has rather attractive properties. First, it is a convex optimization problem over a convex domain, which means that it can be solved efficiently. Second, a numerical iterative solution exists, tailored specifically for this type of optimization problem — iterative proportional fitting [7].

EMT-based control's approach to estimation has been embedded within a simple greedy control algorithm that is summarized below. The algorithm employs a reference system dynamics $r : S \rightarrow \Pi(S)$, termed the *ideal system dynamics* or *tactical target*; the control algorithm greedily selects actions so as to modify the estimated system dynamics to concur with the reference. This approach is similar to closed-loop differential control [11].

0. Initialize estimators:

- the system state estimator $p_0(s) = s_0 \in \Pi(S)$,
- system dynamics estimator

$$PD_0(\bar{s}|s) = \text{prior}(\bar{s}|s)$$

Set time to $t = 0$.

1. Select action a^* to apply using the following computation:

- For each action $a \in A$ predict the future state distribution $\bar{p}_{t+1}^a = T_a * p_t$;
- For each action, compute

$$D_a = H(\bar{p}_{t+1}^a, p_t, PD_t)$$

- Select $a^* = \arg \min_a \langle D_{KL}(D_a||r) \rangle_{p_t}$

2. Apply the selected action a^* and receive an observation $o \in O$.

3. Compute p_{t+1} due to the Bayesian update.

4. Compute $PD_{t+1} = H(p_{t+1}, p_t, PD_t)$.

5. Set $t := t + 1$, goto 1.

2.1 Multi-Target EMT-based Control

At times, there may be several behavioral preferences. For example, in the case of multi-robot movement in formation, two preferences on motion direction exist — one dictated by formation keeping, the other by obstacle collision avoidance. Successful formation navigation requires a robot to adhere to, and balance, both of these behaviors. For EMT-based control, this would mean facing several tactical targets $\{r_k\}_{k=1}^K$, and the question becomes how to merge and balance them. A balancing mechanism can be applied to resolve this issue.

Note that EMT-based control, while selecting an action, creates a preference vector over the set of actions based on their predicted performance with respect to a given target. If these preference vectors are normalized, they can be combined into a single unified preference. We thus replace the 1 stage of EMT-based control (action selection stage) by the following:

1. Given a set of tactical targets $\{r_k\}_{k=1}^K$, and their corresponding weights $w(k)$, select action a^* based on the following computations:

- For each action $a \in A$ predict the future state distribution $\bar{p}_{t+1}^a = T_a * p_t$;
- For each action, compute

$$D_a = H(\bar{p}_{t+1}^a, p_t, PD_t)$$

- For each $a \in A$ and r_k tactical target, denote

$$V(a, k) = \langle D_{KL}(D_a \| r_k) \rangle_{p_t}.$$

Let $V_k(a) = \frac{1}{Z_k} V(a, k)$, where $Z_k = \sum_{a \in A} V(a, k)$ is a normalization factor.

- Select $a^* = \arg \min_a \sum_{k=1}^K w(k) V_k(a)$

The weights vector $\vec{w} = (w_1, \dots, w_K)$ allows the additional “tuning of importance” among tactical targets without the need to redesign the targets themselves. This balancing method is also seamlessly integrated into the EMT-based control flow of operation, and is compatible with its multi-agent extension, as explored in [10]. This compatibility makes possible the following experiment.

3. MULTIPLE TARGETS IN A SCALE BALANCING EXPERIMENT

Originally, EMT-based multi-agent control was tested for its team coordination capabilities over a small balancing problem involving two agents. In this problem, two agents of equal mass are standing on a long bar resting with its ends on two equal springs (Figure 1).

At each time step of the system, each agent has the choice of three actions: moving left one step, moving right one step, or staying put. Every movement of an agent has a non-zero probability of failing, and the probability is biased by the inclination of the bar. That is, an uphill motion will have less probability of succeeding than if the bar were level, and downhill motion will have greater probability of succeeding than if the bar were level. Notice that bar inclination depends on the current agent positions on the bar, thus creating a correlation between the effects of the agent actions, and provides for implicit information transfer between the agents.

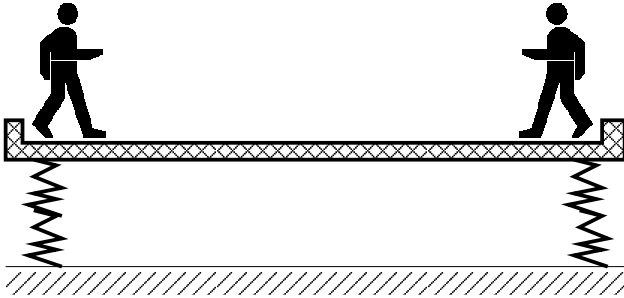


Figure 1: Springed bar setting

Formally, the system state is described by the positions of the two agents on the bar, $S = [1 : d_{max}]^2$, where d_{max} is the length of the bar in “steps”, and the initial state is an unbalanced one, so that $s_0 = (1, \frac{d_{max}}{2} + 1)$. The action sets are $A_i = \{left, stay, right\}$, and the transition probability is built according to the physics of motion as previously described.

In this paper, we consider only one observation scenario among the original two: both agents receive independent noisy observations about their joint position. That is, $O_i = S = \{\text{all positions of the two agents}\}$, $\Omega_1 = \Omega_2$ and creates uniform noise over the immediate neighborhood of the real joint position of agents.

However, unlike the original paper [10], we have modified action selection to accommodate multiple targets, as described in Sec-

tion 2.1. We then set the agents two conflicting targets. One is to balance the springed bar, while the other is to maintain a preset distance between themselves. Note that these targets are not requirements of the system state, but of the laws governing its behavior. For instance, distance maintenance is expressed by a dynamics matrix, that shifts any given state into one that possesses the correct distance property, and EMT-based control sets out to achieve this kind of law within the system.

Although the two behavioral targets we have chosen are compatible, that is, there exists a position of agents on the bar that satisfies both, the targets are indeed conflicting and interfering. For example, assume that we want agents to be at a distance of 4 from each other. Denote by 0 the coordinate line at the center of the bar, and assume that the system’s noisy response forced the agents into positions -2 and $+3$. In this case, the balancing target can encourage the motion of the left agent from -2 to -3 thus balancing the bar, but violating the distance constraint. On the other hand, the distancing target could be satisfied by the same agent moving right to -1 , violating bar balancing even further.

In our experiment sets, we tested both the reaction of EMT-control to multiple targets, and to their balancing by weight vector \vec{w} . The bar length was set to 14, so that the leftmost position on the bar would have coordinate 1 and the rightmost position would have coordinate 15, and the agents were also required by the second target to keep the distance between them at 4 steps.

Despite the constant conflict between the two targets, multiagent EMT-based control equipped with our multi-target action selection approach managed to maintain both targets quite closely. Although relentless system noise caused fluctuations, as seen from an example run at Figure 2, these fluctuations occurred around the only common position that satisfies the demands of both targets. In fact, the mean values of distance between agents and the position of the center of mass almost perfectly matches the ideal, as can be seen from the value distributions in Figure 3, which in our experiments would be 4 and 8 respectively.

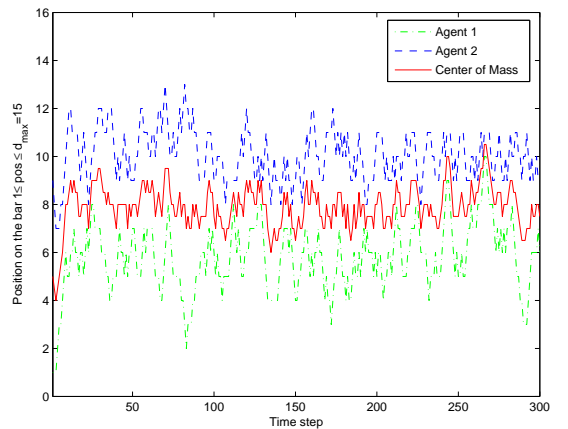


Figure 2: Example Run of Dual-Target Springed-Bar Problem

As was mentioned above, our experiments included different balancing vectors \vec{w} , e.g., the distribution in Figure 3 was obtained from the weight vector $\vec{w} = (0.2, 0.8)$, that is, setting the balancing target at 0.2 and the distancing target at 0.8. Changing the

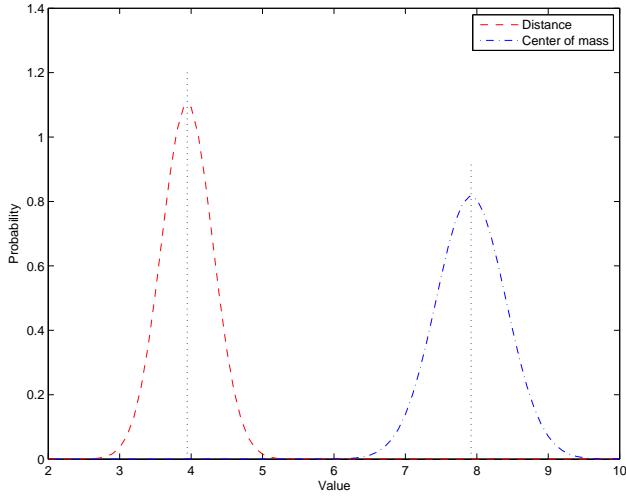


Figure 3: (Fitted normal) Distribution of distance and center of mass in dual-target springed-bar problem with (0.2, 0.8) balancing

weights vector to be $\vec{w} = (0.4, 0.6)$ exposes an interesting property of multi-target EMT-based control. Since the targets were weakly compatible, the algorithm maintained both targets with the new balancing vector, as it did with the old one. The difference occurred when the control algorithm had to correct system behavior in response to noise — the algorithm was more ready to briefly deviate from the distancing target than the balancing one, as shown by the distributions in Figure 4.

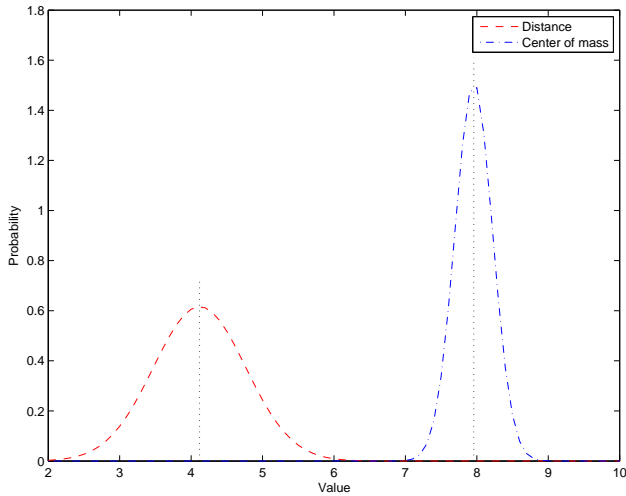


Figure 4: (Fitted normal) Distribution of distance and center of mass in dual-target springed-bar problem with (0.4, 0.6) balancing

However, as the weight of the the balancing target increased, not only the variance of the distance between the agents changed, but also the mean. EMT-based control began to lean strongly towards the balancing target, almost abandoning the distancing target. This can be seen clearly from Figure 5, which depicts changes of the distance and center of mass distributions mean, as a function of weight of the target.

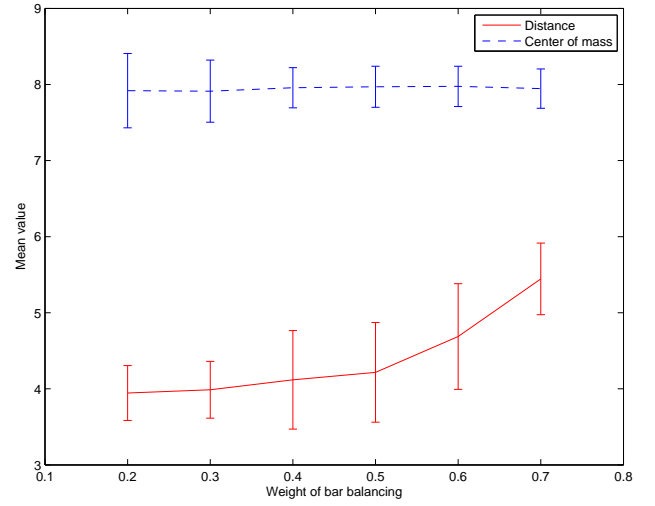


Figure 5: Means of distributions with respect to the weight of the balancing target. Error bars depict variance.

Note that the balancing target had a very strong presence at the weight 0.4, as expressed by the mean and variance of the center of mass distribution. On the other hand, the distancing target at the weight of 0.4 had much less attention from EMT-based control. This inequality in attention to targets with respect to symmetric weight vectors was at first surprising, but can be explained. It has to do with the strength of preference as expressed by the two targets used.

While constructing the targets for the EMT-based controller, a simple inverse exponential scheme was selected to produce the preferences. The balancing target was constructed to have inverse exponential preference over the center of mass deviation from an ideal, while the distancing target was composed based on deviation from the distance ideal. However, simple parameter selection for this scheme produced two targets with different *strength* of preference. While the balancing target included preferential differences of dozens of orders of magnitude, the distancing target featured preference differences of only 2–4 orders of magnitude.

As a result, any change in system dynamics had much higher impact with respect to the balancing behavioral preference, which thus had a stronger effect on proper system behavior. The distance tactical target was very mild in comparison, and thus had less influence on action choices.

4. SIMULATED OBJECT TRACKING EXPERIMENTS

The experiments above with multiple targets were the first step in testing EMT-based controllers for inter-controller interference. It is possible and even necessary to consider situations where separate controllers interact via the environment, interfering with one another. This situation often occurs naturally in robotics, and we chose to test EMT-based control in that domain.

4.1 Robot Motion with EMT Control

Computerized algorithmic control of a robot, unlike low-level reactive control, requires a discretization of sensory information received by the robot, and later the carrying out, by the robot's ac-

tuators, of a discrete action description. In effect, the actual robot serves as an interface between the real world and a discrete controller, transforming sensory information to fit the internal environment model used to make decisions.

Two adjustments are required, in order to apply EMT control to robotic motion.

If the internal model is chosen to be a Markovian model, $\langle S, A, T, O, \Omega, s_0 \rangle$, then sensory discretization requires mapping real world sensations onto the observation set O . This makes the set of states S abstract — it is only indirectly connected to the real world, and all $s \in S$ gain meaning through the observation probability function Ω .

Abstraction of S has several consequences. First, a control algorithm based on a Markovian model would usually require the transition function $T : S \times A \rightarrow \Pi(S)$, also called the “action model,” to be strongly coherent with the environmental dynamics of the system state’s *meanings*, determined by O and Ω . However, the EMT Controller manages to achieve reasonable, though not optimal, results even with an inaccurate action model (Section 4.2).

Second, if EMT Control is to be applied, its strategic layer has to create a tactical target by “reverse engineering” — a high-level target is described by limitations on *observations*, and the strategic layer uses Ω to convert it into system state dynamics, $r : S \rightarrow \Pi(S)$.

Other than the two adjustments above, the basic EMT Controller remains unchanged for robot control.

4.2 Target-Following Experiment

We designed a straightforward simulation experiment, within the Player/Stage simulation environment [6], to see whether EMT Control would be effective for robot control, and whether it would be able to cope with implicit controller interference and an inaccurate environment model.

In the experiment, two independent EMT Controllers, $EMTC_1$ and $EMTC_2$, were applied to linear and rotation speed modulation of a single (simulated) Pioneer-2X robot, with the task of following another robot’s motion. The sensory information was received through a *blob finder* — an on-robot camera with basic image analysis that makes possible the detection of color blobs within the picture. Camera information was approximately mapped onto the observation sets: color blob relative area and centering within the picture. Thus, the observation distributions provided state meanings of linear distance for $EMTC_1$, and angular distance for $EMTC_2$.

To follow (but not capture) an object, a robot must solve two balancing problems: stay within a certain distance from the object, and stay directed straight at it.

A well-studied formal balancing problem exists that can easily be adopted for robotic controllers: the Drunk Man Walk (DMW). This problem concerns a weakly-controlled random walk over a linear graph, as depicted in Figure 6. The actions in this model tilt the probability of left and right steps, while the observation distribution blurs the current position.

Since EMT-based control was shown to be successful at solving this problem [9], we chose to use the DMW as the internal model for the

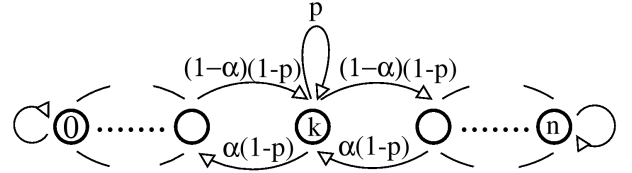


Figure 6: Drunk Man state transition diagram. α is subject to action effect.

$EMTC_1$ and $EMTC_2$ controllers. Their balancing acts translated into a tactical target in a natural way, essentially projecting all states in S_i onto $s = \frac{|S_i|}{2}$.

Notice, however, that the transition function of a DMW (our action model) is not entirely coherent with the real world’s reactions, and with the meaning of system states. This incoherence actually serves our purposes, since we want to verify the performance of EMT-control with respect to an imperfect model. For example, $EMTC_1$ uses system states S_1 to represent distance from a followed object, and since their meaning is based on the camera picture, they represent an irregular set of distances. This lack of regularity would dictate uneven transition probabilities from different states under the same action, but this is not the case in the DMW model, where all system states are symmetric. A similar problem exists within $EMTC_2$, because change in the visible angle of an object may depend on its linear speed as well as on its angular speed. However, this is not accounted for by the DMW model within $EMTC_2$.

This incoherence indeed influences the EMT Controller’s performance; however, it was still able to successfully perform the tracking task. A sample run of the EMT-controlled robot can be seen in Figure 7, depicting three positions of the robots at different times. In this run, the followed robot (which we shall call the *prey*) performed a constant loop, and the EMT-controlled robot that followed it (the *predator*) managed to capture this motion.¹ The predator traces a smaller loop, concentric with the one traced by the prey. However, due to action model incoherence, the predator did not perform optimally and reacted to the change in the prey’s position with insufficient correcting actions. Effectively, it resulted in the predator being significantly further away from the prey than was required by the tactical target.

5. DISCUSSION AND FUTURE WORK

In this research we explored the performance of EMT-based control in response to multiple interacting targets, as well as interacting environment models. In the latter, EMT-based control was applied in an experimental setting with the additional complication of an incoherent environment model.

For EMT to handle explicit multiple objectives, we introduced an additive weighing scheme that integrated directly into EMT operations, and allowed it to select actions in a well-balanced way. In fact, in our experiments EMT control managed in many situations to achieve both targets, despite their interference and, at times, conflict with each other.

Our results also showed that EMT-based control is sensitive to the strength of its tactical targets, and actions are selected in corre-

¹Predator/Prey terminology is used here for naming convenience only, since there is no actual ‘capture’ intended.

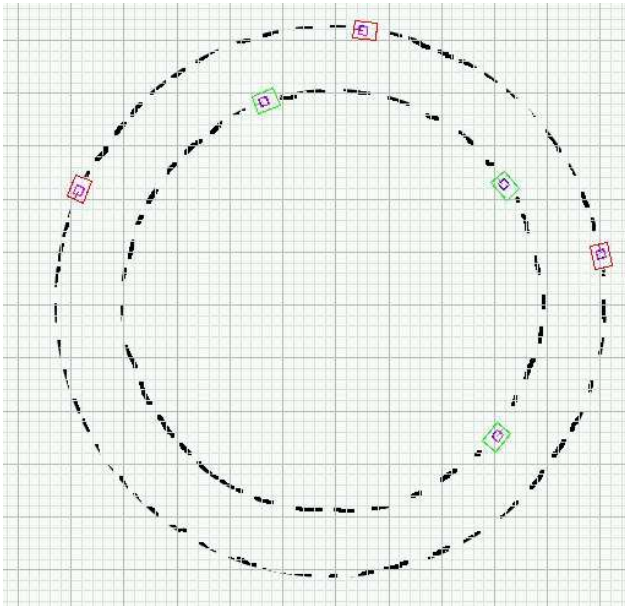


Figure 7: Target Following with a Weakly Coherent Model

spondence to that strength. As the proposed weighing scheme affects EMT's attention to targets as well, we conclude that it can and should become a part of an overall target design process, allowing for finer tuning of the combined behavior.

The simplicity with which tactical behavioral targets can be balanced within EMT-based control suggests further development of dynamic balancing schemes, guided, perhaps, by some external global system behavior evaluation. For example, in our car driving scenario above, it might be the successful passing of a bend in the road.

As our weighted balancing scheme is closely related to other multi-objective algorithms, it is important to continue research into the possibility of their integration. Since EMT-based control includes a prediction mechanism, expert-based balancing approaches (such as [4, 12, 8]) would be of particular interest. Insights from multi-objective reinforcement learning (e.g., [5, 2]) could also be used.

EMT sensitivity to preference strength can be further exploited by introducing non-linear balancing and weight schemes, in effect amplifying preferences as needed, without complicating tactical target design.

The proposed multi-target approach, combined with the natural abilities of EMT control with respect to implicit interference of multiple controllers, can be readily used in a multi-robot setting, quickly providing a sustainable behavior of a team. Formation restricted navigation, multi-featured manipulator control, multi-robot sweeping, and many other tasks become susceptible to the use of EMT with the aid of our multi-target approach, and we plan to explore these applications in the future.

6. REFERENCES

- [1] R. C. Arkin. *Behavior-Based Robotics*. MIT Press, 1998.
- [2] O. Buffet, A. Dutech, and F. Charpillet. Learning to weigh basic behaviors in scalable agents. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, volume 3, pages 1264–1265, 2002.
- [3] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley, 1991.
- [4] Y. Freund, R. E. Schapire, Y. Singer, and M. K. Warmuth. Using and combining predictors that specialize. In *Proceedings of the 12th Annual ACM Symposium on the Theory of Computing (STOC-97)*, pages 334–343, 1997.
- [5] Z. Gabor, Z. Kalmar, and C. Szepesvari. Multi-criteria reinforcement learning. In *Proceedings of the 15th International Conference on Machine Learning (ICML-98)*, pages 197–205, 1998.
- [6] B. Gerkey, R. T. Vaughan, and A. Howard. The player/stage project: Tools for multi-robot and distributed sensor systems. In *Proceedings of the 11th International Conference on Advanced Robotics (ICAR-03)*, pages 317–323, 2003.
- [7] S. Kullback. Probability densities with given marginals. *The Annals of Mathematical Statistics*, 39(4):1236–1243, 1968.
- [8] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108:212–261, 1994.
- [9] Z. Rabinovich and J. S. Rosenschein. Extended Markov Tracking with an application to control. In *The Workshop on Agent Tracking: Modeling Other Agents from Observations, at the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 95–100, New York, July 2004.
- [10] Z. Rabinovich and J. S. Rosenschein. Multiagent coordination by Extended Markov Tracking. In *The Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 431–438, Utrecht, The Netherlands, July 2005.
- [11] R. F. Stengel. *Optimal Control and Estimation*. Dover Publications, 1994.
- [12] V. Vovk and C. Watkins. Universal portfolio selection. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98)*, pages 12–23, 1998.