[6] R. Lindemann, L. Reid, and C. Voorhees, "Mobility Sub-System for the Exploration Technology Rover," in *Proc. 33rd Aerospace Mechanisms Symp.*, 1999, pp. 115–130.

[7] E. T. Baumgartner, "In-Situ Exploration of Mars Using Rover Systems," in *Proc. AIAA Space 2000 Conf.*, 2000, AIAA Paper # 2000–5062.

[8] E. T. Baumgartner, H. Aghazarian, A. Trebi-Ollennu, T. L. Huntsberger, and M. S. Garrett, "State Estimation and Vehicle Localization for the FIDO Rover," in *Proc. SPIE Conf. Sensor Fusion and Decentralized Control in Autonomous Robotic Systems III*, vol. 4196, Nov. 2000, pp. 329–336.

[9] R. E. Arvidson, S. Squyres, E. T. Baumgartner, L. Dorsky, and P. Schenker, "Rover Trials for Mars Sample Return Mission Prove Successful," *EOS Trans. Amer. Geophys. Union*, vol. 81, no. 7, pp. 65–72, 2000.

[10] P. G. Backes and J. S. Norris, "Automated Rover Sequence Report Generation," in *Proc. IEEE Aerospace Conf.*, Big Sky, MT, March 2001.

[11] D. B. Gennery, "Camera Calibration Including Lens Distortion," JPL, Pasadena, CA, JPL Tech Report D-8580, 1991.

[12] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Trans. PAMI*, vol. 9, no. 4, pp. 532–550, 1987.

[13] M. D. Levine, *Vision in Man and Machine*. New York: McGraw-Hill, 1985.

[14] D. H. Titterton and J. L. Weston, "Strapdown inertial navigation technology," in *IEE Radar, Sonar, Navigation and Avionics Series 5*, E. Bradsell, Ed. London, U.K.: Peter Pergrinus Ltd, 1997, pp. 455–455.

[15] A. Brandt and J. F. Gardner, "Constrained Navigation Algorithms for Strapdown Inertial Navigation Systems with Reduced Set of Sensors," in *Proc. American Control Conf.*, vol. 3, 1998, pp. 1848–1852.

[16] E. Nebot and H. Durrant-Whyte, "Initial calibration and alignment of an inertial navigation," in *Proc. Fourth Annual Conf. on Mechatronics and Machine Vision in Practice*, 1997, pp. 175–180.

[17] S. L. Moshier, "Self-contained ephemeris calculator,", Astronomy and Numerical Software web site (http://people.ne.mediaone.net/moshier/index.html)..

[18] F. Cozman and E. Krotkov, "Robot Localization Using a Computer Vision Sextant," *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'95)*, pp. 106–111, 1995.

[19] E. T. Baumgartner, H. Aghazarian, and A. Trebi-Ollennu, "Rover Localization Results for the FIDO Rover," in *Sensor Fusion and Decentralized Control in Autonomous Robotic Systems IV*, vol. 4571, SPIE Proc, Newton, MA, 2001.

Formation Constrained Multi-Agent Control

Magnus Egerstedt and Xiaoming Hu

***Abstract*—We propose a model independent coordination strategy for multi-agent formation control. The main theorem states that under a bounded tracking error assumption our method stabilizes the formation error. We illustrate the usefulness of the method by applying it to rigid body constrained motions.**

***Index Terms*—Coordinated control, mobile robots, stability.**

I. Introduction

In the maturing field of mobile robot control, a natural extension to the traditional trajectory tracking problem [4], [7], [9], [15] is that of *coordinated tracking*. In its most general formulation, the problem is to find a coordinated control scheme for multiple robots that make them maintain some given, possibly time-varying, formation at the same time as the robots, viewed as a group, execute a given task. The possible tasks could range from exploration of unknown environments where an increase in numbers could potentially reduce the exploration time, navigation in hostile environments where multiple robots make the system redundant and thus robust [2], to coordinated path following [5]. The latter of these tasks is applicable in manufacturing or construction situations where multiple robots are asked to carry or push objects in a coordinated fashion [11], [13].

In this paper, we focus on a particular type of *path following*, and the idea is to specify a reference path for a given, nonphysical point. Then a multiple agent formation, defined with respect to the real robots as well as to the nonphysical *virtual leader*, should be maintained at the same time as the virtual leader tracks its reference trajectory.

The formation problem for multiple robots has been extensively studied in the literature, and, for instance, in [2] a behavior-based, decentralized control architecture is exploited, where each individual platform makes sure that it is placed appropriately with respect to its neighbors. In [5] and [6], the situation is slightly different and the solution is based on letting one robot take on the role of the leader, meaning that all of the other robots position themselves relative to that robot. Furthermore, in [10], [16], and [18], an extensive line of work has been conducted with the dynamic model taken into account explicitly, while a very specific type of "string stability" is achieved for multiple autonomous vehicles.

In contrast to this, the approach suggested in this paper is platform-independent, proven successful, and general enough to support a number of different actual controllers. The idea that we capitalize on is that the tracking controllers can be designed independently of the coordination scheme, which provides us with additional control power.

Manuscript received April 20, 2000; revised August 3, 2001. This paper was recommended for publication by Associate Editor L. Kavraki and Editor V. Lumelsky upon evaluation of the reviewers' comments. The work of M. Egerstedt was supported in part by the Army Research Office under Grant DAAG 5597-1-0114, and in part by the Sweden–America Foundation 2000 Research Grant. The work of X. Hu was supported in part by the Swedish Foundation for Strategic Research through its Centre for Autonomous Systems at KTH, in part by TFR. This work was presented in part at ICRA, Seoul, Korea, May 2001.

M. Egerstedt is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: magnus@ece.gatech.edu).

X. Hu is with the Division of Optimization and Systems Theory, Royal Institute of Technology, Stockholm, Sweden (e-mail: hu@math.kth.se).

Publisher Item Identifier S 1042-296X(01)10916-X.

The outline of this paper is as follows. In Section II we discuss the coordinated tracking problem from a theoretical point of view, including our main stability theorem. In Section III, we show how our platform independent coordination scheme can be applied to the class of unicycle robots. We then conclude in Section IV with an illustration of how the proposed method can be used for generating rigid body motions.

II. Formation Control

The multi-agent system that we consider in this paper is given by m mobile robots, each of which is governed by its own set of system equations

$$\begin{aligned}\dot{\mathbf{z}}_i &= f_i(\mathbf{z}_i) + g_i(\mathbf{z}_i)\mathbf{u}_i \\ \mathbf{x}_i &= h(\mathbf{z}_i)\end{aligned}$$

where $\mathbf{z}_i \in \mathbb{R}^{p_i}$ is the state of the system, $\mathbf{u}_i \in \mathbb{R}^{k_i}$ is the control, and $\mathbf{x}_i \in \mathbb{R}^n$ are geometric variables used for defining the formation in $\mathbb{R}^n$. (Throughout this paper, we use boldface to distinguish vectors from scalars.) These geometric variables can either be just the state of the system or output projections onto some space on which one wants the formation to evolve. The m robots should keep a certain relative position and orientation, while moving along one given path, specified for the virtual leader, e.g., the geometric center of the formation.

Definition II.1 (Formation Constraint Function): Given a continuously differentiable, positive definite ($F = 0$ only at one point) map $F: \mathbb{R}^n \times \cdots \times \mathbb{R}^n \to \mathbb{R}^+$. If $F(\mathbf{x}_1, \ldots, \mathbf{x}_m)$ is strictly convex, then we say that $F(\mathbf{x}_1, \ldots, \mathbf{x}_m)$ is a *formation constraint function*. The shape and orientation of the robot formation is uniquely determined by $(\mathbf{x}_1, \ldots, \mathbf{x}_m) = F^{-1}(0)$.

The formation is thus given by the kernel of a formation constraint function, which is a mathematically appealing way of capturing the desired formation. It is obvious that, for a given formation, the corresponding formation constraint function is not unique. For example, for a given polygon in $\mathbb{R}^2$, one can choose either

$$F = \sum_{i=2}^{m} \left[(\|\mathbf{x}_{i-1} - \mathbf{x}_i\|^2 - d_i)^2 + (\|\mathbf{x}_i\|^2 - r_i)^2\right] + \|\mathbf{x}_1 - \mathbf{a}_1\|^2$$

or

$$F = \sum_{i=1}^{m} \|\mathbf{x}_i - \mathbf{a}_i\|^2.$$

From an implementation point of view, the former is preferable since the relative distance is coordinate-free and easier to measure than the absolute position.

We, of course, want to allow for the possibility of having a moving formation since we want the virtual leader to follow a given path. If the desired path that we want the virtual leader to follow is given by $\mathbf{p}_0(\cdot)$, we choose to parameterize the trajectory for the virtual leader, $\mathbf{x}_0 \in \mathbb{R}^n$, as

$$\mathbf{x}_0(t) = \mathbf{p}_0(s_0(t))$$

where $s_0 \in \mathbb{R}$ is a function of t (time), and where we assume that the trajectory is smooth, i.e., $\|(\partial \mathbf{p}_0(s_0)/\partial s_0)\| \neq 0$ for all s_0.

The reason for calling $\mathbf{x}_0$, together with its dynamics, a virtual leader is because it takes on the role of the leader for the formation. Using this terminology, our additional task is to design m new virtual robots for the individual robots to follow. We are thus free to design the evolution of these additional virtual vehicles, and we ignore the question concerning how to actually track these new virtual vehicles for the time being.

In light of the previous paragraph, it is more convenient to consider a moving frame with coordinates centered at $\mathbf{x}_0$. In the new coordinates we have $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_0$. Let the desired trajectories (subscript d), or virtual vehicles, be defined in the moving frame by

$$\begin{aligned}\tilde{\mathbf{x}}_{id} &= \tilde{\mathbf{p}}_i(s_i), \qquad i = 1, \ldots, m \\ \dot{\tilde{\mathbf{x}}}_{id} &= \frac{\partial \tilde{\mathbf{p}}_i(s_i)}{\partial s_i}\,\dot{s}_i\end{aligned}$$

where we have not yet specified what the desired trajectories should look like. In fact, $\partial \tilde{\mathbf{p}}_i(s_i)/\partial s_i$ and $\dot{s}_i \in \mathbb{R}$ can be designed by us, and they should be chosen in a systematic fashion so that the formation constraint is respected.

The solution we propose is to let the desired trajectories be given by the steepest descent direction to the desired formation, i.e., we set

$$\frac{\partial \tilde{\mathbf{p}}(\mathbf{s})}{\partial \mathbf{s}} = -\nabla F(\tilde{\mathbf{x}}_d)$$

where we have grouped together the contributions from the different robots as

$$\begin{aligned}\nabla F(\tilde{\mathbf{x}}_d)^T &= \left(\frac{\partial F(\tilde{\mathbf{x}}_d)}{\partial \tilde{\mathbf{x}}_{1d}}^T, \ldots, \frac{\partial F(\tilde{\mathbf{x}}_d)}{\partial \tilde{\mathbf{x}}_{md}}^T\right) \\ \tilde{\mathbf{p}}(\mathbf{s})^T &= \left(\tilde{\mathbf{p}}_1^T(s_1), \ldots, \tilde{\mathbf{p}}_m^T(s_m)\right) \\ \mathbf{s}^T &= (s_1, \ldots, s_m) \\ \tilde{\mathbf{x}}_d^T &= (\mathbf{x}_{1d}^T - \mathbf{x}_0^T, \ldots, \mathbf{x}_{md}^T - \mathbf{x}_0^T).\end{aligned}$$

The idea now is to let the evolution of the different virtual vehicles be governed by differential equations containing error feedback in order to make the control scheme robust. This can be viewed as a combination of the conventional trajectory tracking, where the reference trajectory is parameterized in time, and a dynamic path-following approach [15], where the criterion is to stay close to the geometric path, but not necessarily close to an *a priori* specified point at a given time.

In order to accomplish this, we define the evolution of the reference points as

$$\dot{s}_i = ce^{-\alpha_i \rho_i}, \qquad i = 1, \ldots, m$$

where $c, \alpha_i > 0$ and $\rho_i = \|\mathbf{x}_i - \mathbf{x}_{id}\| = \|\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_{id}\|$. We furthermore want the motion of s_0 to capture how well the formation is being respected. For this, we define

$$\rho_a = \sum_{i=1}^{m} \rho_i$$

and set

$$\dot{s}_0 = \frac{c_0}{\left\|\dfrac{\partial \mathbf{p}_0(s_0)}{\partial s_0}\right\|}\, e^{-\alpha_0 \rho_a}$$

where $c_0, \alpha_0 > 0$.

With these designs, we have the following stability theorem.

Theorem II.1 (Coordinated Tracking and Formation Control): Under the assumption that the real robots track their respective reference trajectories perfectly, it holds that

$$\lim_{t \to \infty} F(\tilde{\mathbf{x}}_d) = 0.$$

Remark II.1: This theorem shows that we have quite some freedom in initializing the virtual vehicles and that the algorithm is robust to measurement noises.

Proof:

$$\frac{d}{dt}F(\tilde{\mathbf{x}}_d)=\nabla F(\tilde{\mathbf{x}}_d)^T\dot{\tilde{\mathbf{x}}}_d=-\sum_{i=1}^{m}\left\|\frac{\partial F(\tilde{\mathbf{x}}_d)}{\partial \tilde{\mathbf{x}}_{id}}\right\|^2 ce^{-\alpha_i\rho_i}.$$

Now assume that we have perfect tracking, i.e., $\rho_i=0, i=1,\ldots,m$. This assumption, combined with the assumption that F is positive definite and convex, implies that $(d/dt)F(\tilde{\mathbf{x}}_d)$ is negative definite since otherwise F would have a local minimum. This concludes the proof.■

Corollary II.1: If all the tracking errors are bounded, i.e., it holds that $\rho_i\le\rho<\infty, i=1,\ldots,m$, then

$$\lim_{t\to\infty}F(\tilde{\mathbf{x}}_d)=0.$$

The proof of this corollary is just a straightforward extension of the proof of the previous theorem. This corollary is furthermore very useful since one typically does not want $\rho=0$ due to the potential chattering that such a control strategy might give rise to [7] and [8]. Instead it is desirable to let $\rho>0$ be the look-ahead distance at which the robots should track their respective reference trajectories.

III. Control of Mobile Robots

In this section we shift our focus to the actual tracking of the virtual reference points in the workspace of $\mathbb{R}^2$ (i.e., $n=2$). Our solution to this problem is based on position and orientation error feedback. The solution is largely model-independent because it provides only the rotational and translational velocity controls. In other words, they are higher-level controls. Naturally, for platforms that do not support direct control over these velocities, one needs to be somewhat more careful when designing the actuator controllers.

Under the assumption that we can control the rotational and translational velocities, we model the robots as unicycles of the form

$$\begin{aligned}\dot{x}&=v\cos\phi\\ \dot{y}&=v\sin\phi\\ \dot{\phi}&=\omega\end{aligned}$$

where (x, y) is the center of gravity of the robot in the inertially fixed coordinate system, and ϕ is its orientation. The two controlled inputs (v, ω) correspond to the longitudinal and angular velocities, respectively.

It should be noted that we, throughout this section, choose to drop the subscript $i\in\{1,\ldots,m\}$ since we assume that all robots have the same dynamics. The evolution of the reference points are moreover still given by the coordination algorithm from the previous section.

Let $\Delta x=x_d-x$, $\Delta y=y_d-y$, and $\Delta\phi=\phi_d-\phi$, where $x_d=p_x(s_0, s)$, $y_d=p_y(s_0, s)$, and $\phi_d=\operatorname{atan2}(\Delta y, \Delta x)$. Here $\mathbf{p}(s_0, s)=(p_x(s_0, s), p_y(s_0, s))^T$ is the desired trajectory, and $p_x(s_0, s)=p_{0x}(s_0)+\tilde{p}_x(s)$ and $p_y(s_0, s)=p_{0y}(s_0)+\tilde{p}_y(s)$ for each of the m agents, where s_0 and s are as defined before. We propose the following simple, intuitive control algorithm for the actual robots.

Algorithm III.1:

$$\begin{aligned}v&=\gamma\rho\cos\Delta\phi\\ \omega&=k\Delta\phi+\dot{\phi}_d\end{aligned}$$

where $\rho=\sqrt{\Delta x^2+\Delta y^2}$ and $k, \gamma>0$.

We should point out that $\Delta\phi$ is not defined at $\rho=0$ since ϕ_d is not defined. In implementation, one can replace ϕ_d by the equation shown at the bottom of the page where ϵ is a small positive number. It is easy to see that $\hat{\phi}_d$ is well defined at $\rho=0$ since $\lim_{\rho\to 0}\phi_d(-2\rho^3+3\epsilon\rho^2)=0$.

The error dynamics then becomes

$$\begin{aligned}\dot{\Delta}x&=-\frac{\partial p_{0x}(s_0)}{\partial s_0}\dot{s}_0+\frac{\partial\tilde{p}_x(s)}{\partial s}\dot{s}-\gamma\rho\cos\Delta\phi\cos\phi\\ \dot{\Delta}y&=\frac{\partial p_{0y}(s_0)}{\partial s_0}\dot{s}_0+\frac{\partial\tilde{p}_y(s)}{\partial s}\dot{s}-\gamma\rho\cos\Delta\phi\sin\phi\\ \dot{\Delta}\phi&=-k\Delta\phi.\end{aligned}$$

Assumption III.1: Along trajectories, the formation satisfies $\|F(\tilde{\mathbf{x}}_d)\|<M<\infty$ for some $M\in\mathbb{R}^+$.

We can now formulate the following stability theorem.

Theorem III.1 (Stability): Under the control action given in Algorithm III.1, it holds that

$$\limsup_{t\to\infty}\rho(t)\le d$$
$$\limsup_{t\to\infty}\|\Delta\phi\|\le\delta$$

for some $d, \delta>0$ that can be made arbitrarily small with an appropriate choice of the control parameters k and γ.

Proof: Since $\dot{\Delta}\phi=-k\Delta\phi$, the second of the two control objectives clearly holds. Furthermore, differentiating ρ gives

$$\dot{\rho}=c_0e^{-\alpha_0\rho_a}\cos(\phi_d-\phi_{r_0})+ce^{-\alpha\rho}\left\|\frac{\partial\tilde{p}(s)}{\partial s}\right\| \cdot\cos(\phi_d-\phi_r)-\gamma\rho\cos^2\Delta\phi$$

where

$$\phi_r=\operatorname{atan2}\left(\frac{\partial\tilde{p}_y(s)}{\partial s}, \frac{\partial\tilde{p}_x(s)}{\partial s}\right)$$

and

$$\phi_{r_0}=\operatorname{atan2}\left(\frac{\partial p_{0y}(s_0)}{\partial s_0}, \frac{\partial p_{0x}(s_0)}{\partial s_0}\right).$$

Now let $a(t)=-\gamma\cos^2\Delta\phi$, and let $\Phi(t, s)$ be the transition matrix of $a(t)$. Then

$$\begin{aligned}\|\Phi(t, s)\|&=\exp\left(\int_s^t a(\sigma)\,d\sigma\right)\\ &=\exp\left(-\int_s^t\gamma(1-\sin^2\Delta\phi)\,d\sigma\right)\\ &\le e^{\gamma(\Delta\phi^2(0)/k-(t-s))}, \qquad \forall t\ge s\ge 0\end{aligned}$$

which gives

$$|\rho(t)|\le|\Phi(t, 0)|\rho(0)+\int_0^t|\Phi(t, \sigma)|\left(c_0+c\left\|\frac{\partial\tilde{\mathbf{p}}(s(\sigma))}{\partial s}\right\|\right)d\sigma.$$

However, since the constraint function satisfies $F(\tilde{\mathbf{x}}_d)<M$ according to Assumption III.1, and since it is continuously differentiable,

$$\hat{\phi}_d=\begin{cases}\phi_d, & \text{if } \rho>\epsilon\\ \dfrac{\phi_d(-2\rho^3+3\epsilon\rho^2)+\theta_r(-2(\epsilon-\rho)^3+3\epsilon(\epsilon-\rho)^2)}{\epsilon^3}, & \text{if } \rho\le\epsilon\end{cases}$$

there exists a positive constant $K < \infty$ such that

$$
\begin{aligned}
|\rho(t)| &\leq |\Phi(t, 0)|\rho(0) + \int_0^t |\Phi(t, \sigma)|(cK + c_0)\, d\sigma \\
&\leq e^{\gamma(\Delta\phi^2(0)/k - t)}\rho(0) + \frac{cK + c_0}{\gamma}\, e^{\gamma\Delta\phi^2(0)/k}
\end{aligned}
$$

where the first term decays exponentially, and the second term can be made arbitrarily small with an appropriate choice of k and γ. The theorem thus follows. ■

IV. Rigid Body Motions

In this section, we show how our coordination method can be used for executing translational rigid body motions. With such a motion, we understand a formation constraint that specifies a desired distance between the different robots, as well as distances between the robots and the virtual leader. The term "rigid body" is somewhat misleading since we have no guarantee that the right distances are maintained for all times. On the contrary, the introduction of flexibility into the system is crucial, as we will see further on, when reactive obstacle avoidance terms are added to the controller. In that case, we both want to maintain formation and avoid obstacles at the same time, which calls for a certain amount of flexibility.

Let the formation constraint be given by

$$
F(\tilde{\mathbf{x}}) = \sum_{i=1}^{m}\sum_{j\neq i} \tau_{ij}\left(\|\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j\|^2 - d_{ij}^2\right)^2 + \sum_{i=1}^{m} \tau_i \left(\|\tilde{\mathbf{x}}_i\|^2 - d_i^2\right)^2
$$

where $\tau_{ij} = \tau_{ji} \geq 0$ are the weights that determine how important it is that a particular distance $d_{ij} = d_{ji}$ is maintained between $\tilde{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_j$ and similarly τ_i and d_i determine how close $\mathbf{x}_i$ should be to the virtual leader $\mathbf{x}_0$. Of course, the d_{ij}s and d_is need to be chosen in such a way that a physically feasible formation is being defined by $F^{-1}(0)$.

In this case, no orientation of the formation is specified. Thus, F does not meet the condition that $|F^{-1}(0)| = 1$ in Definition II.1. In fact, F has a continuum of global minima, which each corresponds to a given orientation of the formation. However, since each of these solutions are acceptable, our method is still applicable.

Example IV.1 (Triangular Formations): We consider a triangular formation without the orientation fixed

$$
\begin{aligned}
F(\tilde{\mathbf{x}}) = &(\|\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_2\|^2 - 1)^2 + (\|\tilde{\mathbf{x}}_2 - \tilde{\mathbf{x}}_3\|^2 - 1)^2 \\
&+ (\|\tilde{\mathbf{x}}_3 - \tilde{\mathbf{x}}_1\|^2 - 1)^2 + \left(\|\tilde{\mathbf{x}}_1\|^2 - \tfrac{1}{3}\right)^2 \\
&+ \left(\|\tilde{\mathbf{x}}_2\|^2 - \tfrac{1}{3}\right)^2 + \left(\|\tilde{\mathbf{x}}_3\|^2 - \tfrac{1}{3}\right)^2
\end{aligned}
$$

which corresponds to maintaining an equilateral triangular shape (side lengths equal to one) between the different robots. (One of the terms in the function is actually redundant for defining the shape.) The midpoint of the triangle is the virtual leader in this case. An example of this can be seen in Fig. 1.

As pointed out in [11], [13], [14], and [17], such rigid body formations are useful in a number of applications where groups of robots are asked to carry or push objects in a coordinated manner.

A. Obstacle Avoidance

If we assume that the robots we are controlling are of the unicycle type, we can add a standard, reactive obstacle avoidance term (see, for example, [1]) to the individual control algorithms in Algorithm III.1. We choose to keep the longitudinal velocity from Section III, i.e., $v = \gamma\rho\cos\Delta\phi$, but augment the angular velocity with an avoidance term

$$
\omega = w_{OA}(d)(\phi_{OA} - \phi) + k\Delta\phi + \dot{\phi}_d \quad (1)
$$

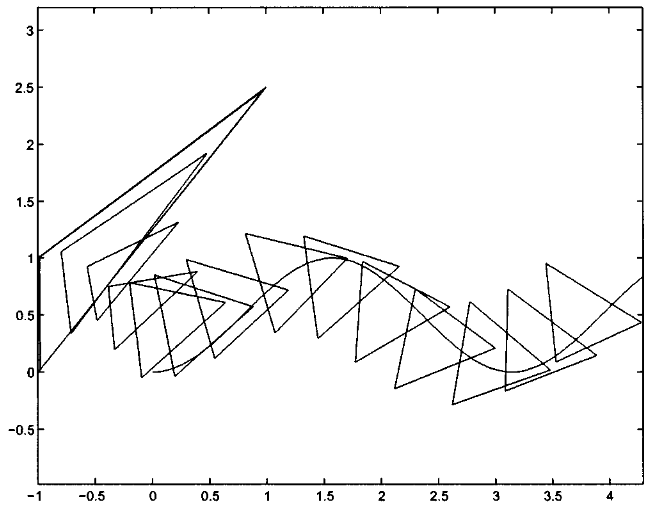

Fig. 1. The evolution of a triangular formation under a perfect tracking assumption. The triangular formation and the reference path for the mid-point of the triangle are shown.

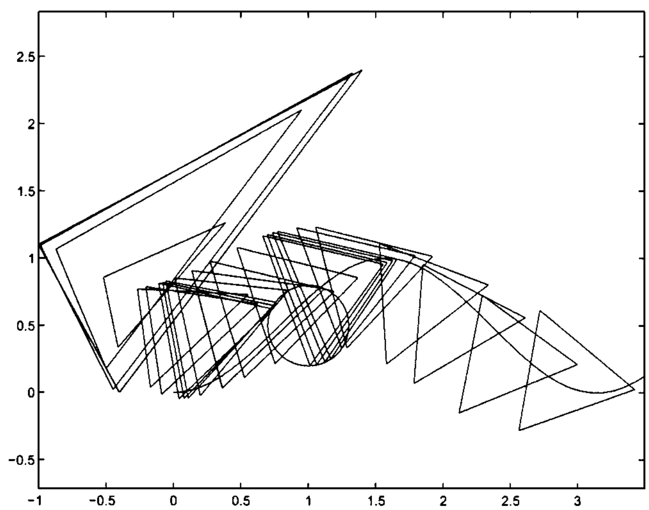

Fig. 2. Obstacle avoidance. In this case, the robots are negotiating the circular obstacle by moving around it on different sides while the virtual leader passes through the obstacle.

where $d = \sqrt{(x - x_{\text{ob}})^2 + (y - y_{\text{ob}})^2}$, $w_{OA}(d) = 1/d^2$ if $d < d_{OA}$, $w_{OA}(d) = 0$ otherwise, and $\phi_{OA} = \pi + \text{atan2}(y_{ob} - y, x_{\text{ob}} - x)$. Here, the subscript OA stands for obstacle avoidance, and d_{OA} is the fixed distance from an obstacle, located at $(x_{\text{ob}}, y_{\text{ob}})$, where the behavior becomes active. (If more than one obstacle is present, the contributions from the different obstacles are just summed up in a straight forward manner.) We thus have a method for controlling the individual robots so that they drive toward the reference points, at the same time as they avoid obstacles, as seen in Fig. 2.

V. Conclusion

In this paper, we propose a model-independent coordination strategy for multi-agent formation control. The problem is defined by a formation constraint in combination with a desired reference path for a nonphysical, so-called virtual leader. We show that if the robots track their respective reference points perfectly, or if the tracking errors are bounded, our method stabilizes the formation error. This is a very useful fact since it allows us to decouple the coordination problem into one planning problem, with proven features as long as the tracking is good enough, and one tracking problem.

The tracking problem is solved for a class of nonholonomic robots of the unicycle type, and we illustrate the soundness of our method by applying it to rigid body constrained motions.

ACKNOWLEDGMENT

The authors would like to thank the three anonymous reviewers for their insightful comments.

REFERENCES

[1] R. C. Arkin, *Behavior-Based Robotics*. Cambridge, MA: MIT Press, 1998.

[2] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 926–939, Dec. 1998.

[3] B. E. Bishop and M. W. Spong, "Control of redundant manipulators using logic-based switching," in *Proc. 37th IEEE Conf. Decision and Control*, Tampa, FL, Dec. 1998.

[4] C. Canudas de Wit, B. Siciliano, and G. Bastin, *Theory of Robot Control*. Berlin: Springer-Verlag, 1996.

[5] J. Desai, J. Ostrowski, and V. Kumar, "Control of formations for multiple robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998.

[6] J. Desai, "Motion planning and control of cooperative robotic systems," Ph.D. dissertation, Univ. of Pennsylvania, Philadelphia, 1998.

[7] M. Egerstedt, X. Hu, and A. Stotsky, "Control of a car-like robot using a virtual vehicle approach," in *Proc. 37th IEEE Conf. Decision and Control*, Tampa, FL, Dec. 1998, pp. 1502–1507.

[8] ——, "Control of mobile platforms using a virtual vehicle approach," IEEE Trans. Automat. Contr., to be published.

[9] R. Frezza, G. Picci, and S. Soatto, "Nonholonomic model-based predictive output tracking of an unknown three-dimensional trajectory," in *Proc. 37th IEEE Conf. Decision and Control*, Tampa, FL, Dec. 1998.

[10] J. K. Hedrick, D. McMahon, V. Narendran, and D. Swaroop, "Longitudinal vehicle controller design for IVHS systems," in *Proc. American Control Conf.*, 1991, pp. 3107–3112.

[11] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, A. Casal, and A. Baader, "Force strategies for cooperative tasks in multiple mobile manipulation systems," in *Int. Symp. Robotics Research*, Munich, Germany, Oct. 1995.

[12] M. Mataric, M. Nilsson, and K. Simsarian, "Cooperative multi-robot box-pushing," in *Proc. IROS*, Pittsburgh, PA, 1995.

[13] P. Ögren, M. Egerstedt, and X. Hu, "A control Lyapunov function approach to multi-agent coordination," in *IEEE Conf. on Decision and Control*, Orlando, FL, Dec. 2001.

[14] N. Sarkar, X. Yun, and V. Kumar, "Dynamic path following: A new control algorithm for mobile robots," in *Proc. 32nd Conf. Decision and Control*, San Antonio, TX, Dec. 1993.

[15] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE Trans. Automat. Contr.*, vol. 41, pp. 349–357, Mar. 1996.

[16] J. P. Tabuada, G. J. Pappas, and P. Lima, "Feasible formations of multi-agent systems," in *Proc. American Control Conf.*, Arlington, VA, June 2001.

[17] D. Yanakiev and I. Kanellakopoulos, "A simplified framework for string stability in AHS," in *Proc. 13th IFAC World Congress*, vol. Q, 1996, pp. 177–182.

Randomized Path Planning for Linkages With Closed Kinematic Chains

Jeffery H. Yakey, Steven M. LaValle, and Lydia E. Kavraki

***Abstract*—We extend randomized path planning algorithms to the case of articulated robots that have closed kinematic chains. This is an important class of problems, which includes applications such as manipulation planning using multiple open-chain manipulators that cooperatively grasp an object and planning for reconfigurable robots in which links might be arranged in a loop to ease manipulation or locomotion. Applications also exist in areas beyond robotics, including computer graphics, computational chemistry, and virtual prototyping. Such applications typically involve high degrees of freedom and a parameterization of the configurations that satisfy closure constraints is usually not available. We show how to implement key primitive operations of randomized path planners for general closed kinematics chains. These primitives include the generation of random free configurations and the generation of local paths. To demonstrate the feasibility of our primitives for general chains, we show their application to recently developed randomized planners and present computed results for high-dimensional problems.**

***Index Terms*—Closed linkages, kinematic chains, randomized path planning.**

I. INTRODUCTION

This paper addresses the problem of path planning for general linkages that have closed kinematic chains with redundant degrees of freedom (DOF), in an environment that contains obstacles, as shown in Fig. 1. In general, the constraints imposed by a closed linkage form an algebraic variety and in principle complete planners such as [6] and [3] could be used; however, the high computational complexity and implementation difficulty of all of these algorithms for problems with high degree of freedom makes them too prohibitive for practical use. This motivates our approach in this paper, which extends randomized planning techniques that were developed for open-chain systems [13], [19] to general closed-chain systems.

Planning for linkages with closed kinematic chains has applications both in and beyond robotics. Parallel manipulators involve closed kinematic constraints [22]. In manipulation planning, when multiple robots grasp a single object, they form a closed loop containing the object as a link of the chain [1], [15]. Many of the existing methods for manipulation planning require inverse kinematics solutions for the robots [15] which can be a limitation. Regrasping is also an important issue as one or more of the manipulators often attain a singular configuration [23]. The ability to plan for linkages with closed kinematics chains eliminates the need of inverse kinematics solutions and could reduce the number of regrasps needed during manipulation tasks, as the linkage

Manuscript received August 25, 2000; revised September 5, 2001. This paper was recommended for publication by Associate Editors N. Overmars and N. Amato and Editor A. De Luca upon evaluation of the reviewers' comments. The work of S. LaValle is supported in part by the National Science Foundation through a CAREER award IRI-9875304 and by Honda Research through a grant. The work of L. Kavraki is supported in part by the National Science Foundation through a CAREER Award IRI-970228, a Whitaker Award, an ATP Award, and a Sloan Fellowship.

J. H. Yakey is with the Department of Computer Science, Iowa State University, Ames, IA 50011 USA (e-mail: yakeyj@cs.iastate.edu).

S. M. LaValle is with the Department of Computer Science, University of Illinois, Urbana, IL 61801 USA (e-mail: lavalle@cs.uiuc.edu).

L. E. Kavraki is with the Department of Computer Science, Rice University, Houston, TX 77005 USA (e-mail: kavraki@cs.rice.edu).

Publisher Item Identifier S 1042-296X(01)11163-8.