Safe Cooperative Robot Dynamics on Graphs [*]

Robert W. Ghrist
School of Mathematics
Georgia Institute of Technology
Atlanta, GA 30332-0160, USA
ghrist@math.gatech.edu

Daniel E. Koditschek
Department of Electrical Engineering and Computer Science
The University of Michigan
Ann Arbor, MI 48109-2110, USA
kod@eecs.umich.edu

Abstract

This paper initiates the use of vector fields to design, optimize, and implement reactive schedules for safe cooperative robot patterns on planar graphs. We consider Automated Guided Vehicles (AGV's) operating upon a predefined network of pathways. In contrast to the case of locally Euclidean configuration spaces, regularization of collisions is no longer a local procedure, and issues concerning the global topology of configuration spaces must be addressed. The focus of the present inquiry is the achievement of safe, efficient, cooperative patterns in the simplest nontrivial example (a pair of robots on a Y-network) by means of a state-event heirarchical controller.

1 Introduction

This paper, an extension of earlier work [9] explores the use of vector fields for reactive scheduling of safe cooperative robot patterns on graphs. The word "safe" means that obstacles — designated illegal portions of the configuration space — are avoided. The word "cooperative" connotes situations wherein physically distributed agents are collectively responsible for executing the schedule. The word "pattern" refers to tasks that cannot be encoded simply in terms of a point goal in the configuration space. The word "reactive" will be interpreted as requiring that the desired pattern reject perturbations.

An automated guided vehicle (AGV) is an unmanned powered cart "capable of following an external guidance signal to deliver a unit load from destination to destination" where, in most common applications, the guidepath signal is buried in the floor [5]. Thus, the AGV's workspace is a network of wires — a graph. The motivation to choose AGV based materials handling systems over more conventional fixed conveyors rests not simply in their ease of reconfigurability but in the potential they offer for graceful response to perturbations in normal plant operation. In real production facilities, the flow of work in process fluctuates constantly in the face of unanticipated workstation downtime, variations in process rate, and, indeed, variations in materials transport and delivery rates [6]. Of course, realizing their potential robustness against these fluctuations in work flow remains an only partially fulfilled goal of contemporary AGV systems.

Choreographing the interacting routes of multiple AGVs in a non-conflicting manner presents a novel, complicated, and necessarily on-line planning problem. Nominal routes might be designed offline but they can never truly be traversed with the nominal timing, for all the reasons described above. Even under normal operating conditions, no single nominal schedule can suffice to coordinate the workflow as the production volume or product mix changes over time: new vehicles need to be added or deleted and the routing scheme adapted. In any case, abnormal conditions — unscheduled process down times; blocked work stations; failed vehicles — continually arise, demanding altered routes.

The traffic control schemes deployed in contemporary AGV systems are designed to simplify the real-time route planning and adaptation process by "blocking zone control" strategies. The workspace is partitioned into a small number of cells and, regardless of the details of their source and destination tasks, no two AGVs are ever allowed into the same cell at the same time [5]. Clearly, this simplification results in significant loss of a network's traffic capacity.

The contemporary robotic motion planning literature does not seem to offer much in the way of an alterna-

[*] Copyright ©1999, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

tive. Starting with pioneering work of Alami [1] there has been a small literature on multiple coordinated robots, but almost all papers seem to be concerned with offline versions of the problem. Latombe, in his excellent monograph [10] distinguishes between "centralized" and "decoupled" approaches to this problem. In the latter case, motion planning proceeds using multiple copies of the configuration space within which to situate a set of non-interacting robot vehicles. For a recent example of what Latombe terms a "coordinated" view of the decoupled case, Svestka and Overmars [14] introduce a "supergraph" on which multiple vehicles can be stepped through their individually specified paths, and vehicle-vehicle collisions prohibited by detaining one or another vehicle. For a recent example of what Latombe terms a "prioritized" view of the decoupled case, Lee et al. [11] compute k-shortest paths for each vehicle's source-destination pair, and work their way down the list of prefences based upon a vehicle's priority. It should be clear that in neither of these approaches has the recourse to blocking zone control been eliminated. Thus, while the decentralized approach side steps the inevitable curse of dimensionality, passing to the underlying configuration space seems to be required if the rigidity and inefficiency of blocking zone control strategies is to be eliminated.

The Industrial Engineering AGV literature seems chiefly concerned with higher level issues of layout and capacity [12] or dispatching and more general scheduling [2]. One interesting approach to layout seeks to avoid the subsequent traffic control problem entirely by clustering pickup and delivery stops in decoupled single vehicle loops [3]. In general, modeling the real-time factory floor is challenging enough that even the most recent treatments of these higher level layout and dispatching problems seem to rely on simulation rather than analysis for understanding the implications of one or another policy [4]. Here, again, is an indication that a dynamical systems point of view might shed additional light. For recent years have witnessed increasingly successful efforts to characterize such ensemble properties as the "mean transit time" induced by a flow. Thus, it seems to us entirely possible that a dynamics based network traffic control strategy might yield more readily to statistical analysis than present practice affords.

In this paper, we will consider a centralized approach that employs dynamical systems theory to focus on real-time responsiveness and efficiency as opposed to computational complexity or average throughput. No doubt, beyond a certain maximum number of vehicles, the necessity to compute in the high dimensional configuration space will limit the applicability of any algorithms that arise. However, this point of view seems not to have been carefully explored in the literature. For the sake of concreteness we will work in the so-called "pickup and delivery" (as opposed to the "stop and go" [3]) paradigm of assembly or fabrication, and we will not be concerned with warehousing style AGV applications.

For additional background on this problem and the configuration space constructions associated to the Y-graph, see [9]. For details and proofs of all theorems stated here, see [8].

2 Notation and Background

A *graph*, Γ, consists of a finite collection of 0-dimensional vertices $\mathcal{V}:=\{v_i\}_1^N$, and 1-dimensional edges $\mathcal{E}:=\{e_j\}_1^M$ assembled as follows. Each edge is homeomorphic to the closed interval $[0,1]$ attached to $\mathcal{V}$ along its boundary points $\{0\}$ and $\{1\}$.

The configuration spaces we consider in the following are self-products of graphs. The topology of $\Gamma \times \Gamma$ is easily understood in terms of the topology of Γ as follows [13]. Let $(x,y) \in \Gamma \times \Gamma$ denote an ordered pair in the product. Then any small neighborhood of (x,y) within $\Gamma \times \Gamma$ is the union of neighborhoods of the form $\mathcal{N}(u) \times \mathcal{N}(v)$, where $\mathcal{N}(\cdot)$ denotes neighborhood within Γ. In other words, the products of neighborhoods form a *basis* of neighborhoods in the product space [13].

Given a graph, Γ, outfitted with a finite number n of non-colliding AGV's constrained to move on Γ, the configuration space of safe motions is defined as

$$\mathcal{C}:=(\Gamma \times \ldots \times \Gamma) - \mathcal{N}(\Delta),$$

where $\Delta:=\{(x_i) \in \Gamma \times \ldots \times \Gamma : x_j = x_k \text{ for some } j \neq k\}$ denotes the pairwise diagonal and $\mathcal{N}(\cdot)$ denotes (small) neighborhood.

We do not treat the general aspects of this problem comprehensively in this paper; rather, we restrict attention to the simplest nontrivial example which illustrates nicely the relevant features present in the more general situation.

3 The Y-Graph

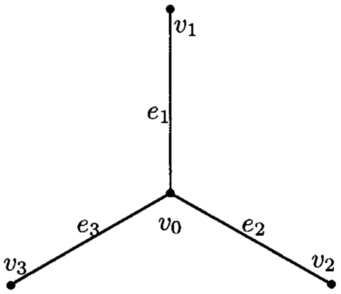

Figure 1: The Y-graph Υ.

For the remainder of this work, we consider the simplest example of a non-trivial configuration space: that associated to the *Y-graph*, Υ, having four vertices $\{v_i\}_0^3$ and three edges $\{e_i\}_1^3$, as illustrated in Figure 1. Although this is a simple scenario compared to what one finds in a typical setting, there are several reasons why this example is in many respects canonical.

1. **Simplicity:** Any graph may be constructed by gluing K-prong graphs together for various K. The $K = 3$ model we consider is the simplest nontrivial case and is instructive for understanding the richness and challenges of local cooperative dynamics on graphs.

2. **Genericity with respect to graphs:** Graphs which consist of copies of Υ glued together, the *trivalent graphs*, are generic in the sense that any nontrivial graph may be perturbed in a neighborhood of the vertex set so as to be trivalent. For example, the 4-valent graph resembling the letter 'X' may be perturbed slightly to resemble the letter 'H' — a trivalent graph. An induction argument shows that this is true for all graphs. Hence, the dynamics on an arbitrary graph are approximated by patching together dynamics on copies of Υ.

3. **Genericity with respect to local dynamics:** Finally, pairwise local AGV interactions on an arbitrary graph restrict precisely to the dynamics of two agents on Υ as follows. Given a vertex v of a graph Γ, assume that two AGV's, x and y, are on different edges e_1 and e_2 incident to v and moving towards v with the goal of switching positions. A collision is immanent unless one AGV "moves out of the way" onto some other edge e_3 incident to v. The local interactions thus restrict to dynamics of a pair of AGV's on the subgraph defined by $\{v; e_1, e_2, e_3\}$. Hence, the case we treat in this paper is the generic scenario for the local resolution of collision singularities in cooperative dynamics on graphs.

The configuration space of N points on Υ is a subset of the N-fold cartesian product $\Upsilon \times \Upsilon \times \cdots \times \Upsilon$. Since each graph which is physically relevant to the setting of this paper is embedded in a factory floor or ceiling and thus planar, the configuration space $\mathcal{C}^N(\Upsilon)$ embeds naturally in R^{2N}. We wish to modify this embedding to facilitate both analysis on and visualization of the configuration space. We will present alternate embeddings in both higher and lower dimensional euclidean spaces for these purposes.

We begin with representing the configuration space within a higher-dimensional euclidean space via a coordinate system which is best described as *intrinsic*: it is independent of how the graph is embedded in space. We illustrate this coordinate system with the Y-graph Υ, noting that a few simple modifications yields a coordinate scheme for more general graphs.

Let $\{e_i\}_1^3$ denote the three edges in Υ, parametrized so that e_i is identified with $[0, 1]$ with each $\{0\}$ at the center v_0 of Υ. Any point $x \in \Upsilon$ is thus given by a vector (x_1, x_2, x_3) in the $\{e_i\}$ frame, where $x_i \in [0, 1]$ and at least two of these coordinates is zero. In other words, we are embedding Υ as the positive unit axis frame in R^3.

Likewise, a point in $\mathcal{C}$ is given as a pair of distinct vectors (x, y), i.e., as the positive unit axis frame in R^3 cross itself sitting inside of $R^3 \times R^3 \cong R^6$. We have thus embedded the configuration space of two distinct points on Υ in the positive orthant of R^6. It is clear that one can embed the more general configuration space of N points on Υ in R^{3N} in this manner.

This coordinate system is particularly well-suited to describing vector fields on $\mathcal{C}$ and in implementing numerical simulations of dynamics, as the coordinates explicitly keep track of the physical position of each point on the graph.

More useful for visualization purposes, however, is the following construction which embeds $\mathcal{C}$ within R^3.

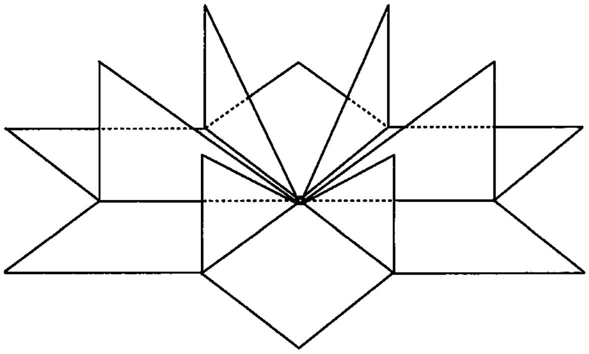

Figure 2: The configuration space $\mathcal{C}$ embedded in R^3.

Theorem 1 *The configuration space $\mathcal{C}$ associated to a pair of AGV's restricted to the Y-graph Υ is homeomorphic to a punctured disc with six 2-simplices attached as in Figure 2.*

Denote by $\mathcal{D}$ that portion of the configuration space which corresponds to the AGV's being on distinct edges of the graph: as proven earlier, $\mathcal{D}$ is homeomorphic to a punctured disc. The intrinsic coordinates on the configuration space $\mathcal{C}$ is illustrated in Figure 3, where only $\mathcal{D}$ is shown for illustration purposes. The reader should think of this as a collection of six square coordinate planes, attached together pairwise along axes with the origin removed. The six triangular fins are then attached as per Figure 2.

4 Tuning Limit Cycles

In order to proceed with the construction of vector fields which realize monotone cycles, we work with vector fields on the smooth unit disc in R^2 and map these to the annular

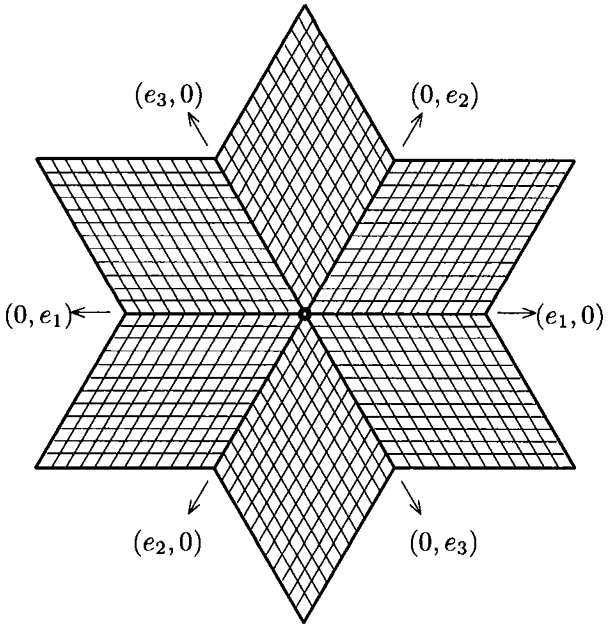

Figure 3: The coordinate system on the unfinned region $\mathcal{D}$ of $\mathcal{C}$.

region $\mathcal{D}$ of the configuration space via the push-forward induced by the natural homeomorphism. It will be convenient to keep track of which "wedge" of the annular region a point (r,θ) is. To do so, we introduce a parity function

$$P(\theta) := (-1)^{\{\lfloor 3\theta/\pi\rfloor+\lfloor 6\theta/\pi\rfloor\}}, \qquad (1)$$

where $\lfloor t\rfloor$ is the integer-valued floor function. Recall the notation for the intrinsic coordinates for a point x on the graph Υ: $x = |x|\hat{e}_{\iota(x)}$, where $|x| \in [0,1]$ is the distance from x to the central vertex, and $\hat{e}_{\iota(x)}$ is the unit tangent vector pointing along the direction of the $\iota(x)$-edge. Here the index, $\iota(x)$ is an integer (defined modulo 3) and will be undefined in the case when $|x| = 0$, i.e., x is at the central vertex.

Lemma 1 *The following is a piecewise-linear homeomorphism from the punctured unit disc in R^2 to the subset $\mathcal{D}$ given by the collection of points which are not on (the interior of) identical edges:*

$$\begin{aligned} F(r,\theta) = \quad & (x,y) \ \textit{where} \\ & |x| = \begin{cases} r & \mathcal{P}(\theta)=+1 \\ r\left|\cot\frac{3}{2}\theta\right| & \mathcal{P}(\theta)=-1 \end{cases} \\ & |y| = \begin{cases} r\left|\tan\frac{3}{2}\theta\right| & \mathcal{P}(\theta)=+1 \\ r & \mathcal{P}(\theta)=-1 \end{cases} \\ & \iota(x) = \left\lfloor -\tfrac{3}{2\pi}(\theta-\pi)\right\rfloor \\ & \iota(y) = \left\lfloor -\tfrac{3}{2\pi}\theta\right\rfloor \end{aligned} \quad . \qquad (2)$$

The inverse of this homeomorphism is given by

$$\begin{aligned} & F^{-1}(x,y) = (r,\theta) \ \textit{where} \\ & \theta = \begin{cases} \frac{2}{3}\tan^{-1}\frac{|y|}{|x|} - \frac{2\pi}{3}(\iota(y)+1) & \begin{matrix}\iota(y)=\iota(x)+1 \\ \textit{or}\ |x|=0\end{matrix} \\ -\frac{2}{3}\tan^{-1}\frac{|y|}{|x|} - \frac{2\pi}{3}(\iota(x)-1) & \begin{matrix}\iota(x)=\iota(y)+1 \\ \textit{or}\ |y|=0\end{matrix} \end{cases} \\ & r = \begin{cases} |x| & \mathcal{P}(\theta)=+1 \\ |y| & \mathcal{P}(\theta)=-1 \end{cases} \end{aligned} \quad . \qquad (3)$$

Note that all θ values are defined modulo 2π, all index values are integers defined modulo 3, and $\lfloor\cdot\rfloor$ denotes the integer-valued floor function.

Proof: Begin by working on the region $\mathcal{D}_{1,2}$ of $\mathcal{D}$ where $\iota(x) = 1$ and $\iota(y) = 2$. We need to map this to the subset $\{(r,\theta) : r \in (0,1], \theta \in [0,\pi/3]\}$. The simplest such homeomorphism is to first shrink along radial lines, leaving the angle invariant; hence

$$r = \begin{cases} |x| & : |x| \leq |y| \\ |y| & : |y| \leq |x| \end{cases} \qquad (4)$$

Next, we squeeze the quarter-circle into a sixth of a circle by multiplying the angle by 2/3, leaving the radial coordinate invariant:

$$\theta = \frac{2}{3}\tan^{-1}\frac{|y|}{|x|}. \qquad (5)$$

This gives the basic form of F^{-1} as per Equation (3). To extend this to the remainder of $\mathcal{D}$, it is necessary to carefully keep track of $\iota(x)$ and $\iota(y)$ and subtract off the appropriate angle from the computation of θ. Also, the condition of $|x| \leq |y|$, etc., in Equation (4) is incorrect on other domains of $\mathcal{D}$, since the inequalities flip as one traverses from square to square: the parity function $\mathcal{P}(\theta)$ keeps track of which "wedge" one is working on.

To determine F from F^{-1} is a tedious but unenlightening calculation, made more unpleasant by the various indices to be kept track of. Briefly, given r and θ on the first sixth of the unit disc, one knows from Equation (4) that either $|x| = r$ or $|y| = r$, depending on whether θ is above or below $\pi/4$. To solve for the other magnitude, one inverts Equation (5) to obtain $|y| = r\left|\tan\frac{3}{2}\theta\right|$ or $|x| = r\left|\cot\frac{3}{2}\theta\right|$ respectively. To generalize this to the other $\mathcal{D}_{i,j}$ domains of $\mathcal{D}$, it is necessary to take absolute values and to use the parity function $\mathcal{P}(\theta)$ as before. Finally, the computation of the index is obtainable from the combinatorics of the coordinate system as illustrated in Figure 3. □

Hence, by taking the push-forward of a vector field $X = (\dot{r},\dot{\theta})$ with respect to F, one obtains the piecewise smooth vector field,

$$\left\{ \begin{array}{ll} \left(\begin{array}{l} \dot{|x|} = \dot{r} \\ \dot{|y|} = \dot{r} \left|\tan(\frac{3}{2}\theta)\right| + \frac{3}{2} r\dot{\theta} \sec^2(\frac{3}{2}\theta) \end{array} \right) & \mathcal{P}(\theta) = +1 \\ \left(\begin{array}{l} \dot{|x|} = \dot{r} \left|\cot(\frac{3}{2}\theta)\right| + \frac{3}{2} r\dot{\theta} \csc^2(\frac{3}{2}\theta) \\ \dot{|y|} = \dot{r} \end{array} \right) & \mathcal{P}(\theta) = -1 \end{array} \right. , \tag{6}$$

which simplifies to yield:

$$\left\{ \begin{array}{ll} \left(\begin{array}{l} \dot{|x|} = \dot{r} \\ \dot{|y|} = \dot{r}\frac{|y|}{|x|} + \frac{3}{2}\dot{\theta}\frac{|x|}{1+\left(\frac{|y|}{|x|}\right)^2} \end{array} \right) & \mathcal{P}(\theta) = +1 \\ \left(\begin{array}{l} \dot{|x|} = \dot{r}\frac{|x|}{|y|} + \frac{3}{2}\dot{\theta}\frac{|y|}{1+\left(\frac{|x|}{|y|}\right)^2} \\ \dot{|y|} = \dot{r} \end{array} \right) & \mathcal{P}(\theta) = -1 \end{array} \right. . \tag{7}$$

Given a simple closed curve γ in R^2 which has nonzero winding number with respect to the origin, γ may be parametrized as $\{(r, \theta) : r = f(\theta)\}$ for some periodic positive function f. To construct a vector field on R^2 whose limit sets consist of the origin as a source and γ as an attracting limit cycle, it suffices to take the push-forward of the vector field

$$\dot{r} = r(1-r) \qquad \dot{\theta} = \omega$$

under the planar homeomorphism

$$\phi : (r, \theta) \mapsto (f(\theta)r, \theta),$$

which rescales linearly in the angular component. The calculations follow:

$$\begin{aligned} \phi_* \left(\begin{array}{c} \dot{r} \\ \dot{\theta} \end{array} \right) &= D\phi \left(\begin{array}{c} \dot{r} \\ \dot{\theta} \end{array} \right) \Bigg|_{r \mapsto \frac{r}{f}} \\ &= \left[\begin{array}{cc} f & rf' \\ 0 & 1 \end{array} \right] \left(\begin{array}{c} r(1-r) \\ \omega \end{array} \right) \Bigg|_{r \mapsto \frac{r}{f}} \\ &= \left(\begin{array}{c} r\left(1 - \frac{r - f'\omega}{f}\right) \\ \omega \end{array} \right) \end{aligned} . \tag{8}$$

Hence, given $f(\theta)$, we may tune a vector field to trace out the desired limit cycle and then use Equations (2) and (3) to map it into intrinsic coordinates.

To design optimal cycles with winding number zero, then, we turn to constructing customized portions of limit cycles, or *chords* which can be pieced together via a state-actuated hybrid controller. In other words, instead of building a simple fixed vector field with a limit cycle, we will use a set of vector fields which vary discretely in time and which may be pieced together so as to tune a limit cycle to the desired specifications.

Let G denote a word representing a desired monotone limit cycle on the configurations space $\mathcal{C}$. Choose points $\{q_i\}$ on the boundary of $\mathcal{D}$ which correspond to the attaching zones for the cycle given by G. Choose arcs α_i on $\mathcal{D}$ which connect q_i to q_{i+1} (using cyclic index notation). The arcs α_i are assumed given in the intrinsic coordinates on $\mathcal{D}$, as would be the case if one were determining a length-minimizing curve.

In the case where the limit cycle $\alpha := \cup_i \alpha_i$ is an embedded curve of nonzero index, the procedure of the previous subsection determines a vector field X_α on $\mathcal{C}$ which realizes α as an attracting limit cycle with the appropriate dynamics on the complementary region. Recall: one translates α to a curve on the disc model via the homeomorphism of Equation (3). Then, representing the limit cycle α as a function $f_\alpha(\theta)$, one takes the vector field of Equation (8) and, if desired, takes the image of this vector field under Equation (7).

If, however, this is not the case, consider the arc α_j for a fixed j and construct an index ± 1 cycle $\beta^j = \cup_i \beta^j_i$ which has attaching zone $\{q_i\}$ such that $\beta^j_j = \alpha_j$. Then the vector field X^j as constructed above has β as an attracting limit cycle. Denote by Φ^j the Lyapunov function which measures proximity to β: $\Phi^j(p) := \left\|p - \beta^j\right\|$ (with distance measured in say the product metric on $\mathcal{C}$). Then, consider the modified Lyapunov function

$$\Psi^j(p) := \Phi^j(p) + \|p - q_{j+1}\|, \tag{9}$$

which measures the distance to the endpoint of the arc β^j_j in addition to the proximity to β^j.

Repeat this procedure for each j, yielding the vector fields $\{X^j\}$ which attract respectively to limit cycles β^j. It follows that X^j prepares X^{j+1} since the goal point of X^j, q_{j+1} lies on the attracting set of X^{j+1}. The Lyapunov functions $\{\Psi^j\}$ serve as a set of funnels which channel the orbit into the sequence of arcs α_j, forming α. One scales the Ψ^j so that a $\Psi^j < \epsilon$ event triggers the switching in the hybrid controller from X^j to X^{j+1}:

$$X := \left\{ \begin{array}{lll} X^1 & : & \Phi^j > \epsilon \forall j \\ X^j & : & \Phi^j < \epsilon \text{ and } \Psi^j > \epsilon \end{array} \right. \tag{10}$$

By construction, the hybrid controller (10) realizes a limit cycle within ϵ of α as the attracting set.

5 Future Directions

A point of primary concern is the adaptibility of the global topological approach to systems which increase in complexity, either through more intricate graphs or through increased numbers of AGV's. The latter is of greater difficulty than the former, since the dimension of the resulting configuration space is equal to the number of AGV's. Hence, no matter how simple the underlying graph is, a system with ten independent AGV's will require a dynamical controller on a (topologically complicated) ten-dimensional space: a formidable problem both from the topological, dynamical, and computational viewpoints.

However, there are some approaches which may facilitate working with such spaces. Consider the model space C with which this paper is concerned: although a two-dimensional space, C can be realized as the product of a graph (a circle with six radial edges attached) with the interval $(0, 1]$.

A similar approach may work for arbitrary graphs. The following result has recently been proven [7]:

Theorem 2 *Given a graph Γ, the configuration space of N distinct points on Γ can be deformation retracted to a subcomplex whose dimension is bounded above by the number of vertices of Γ of valency greater than two.*

This theorem implies the existence of low-dimensional *spines* which carry all of the topology of the configuration space. For example, the above theorem implies that the configuration space of N points on the Y-graph can be continuously deformed to a one-dimensional graph, regardless of the size of N. Since the full space can be deformation retracted onto the spine, a vector field defined on the spine can be (in theory) pulled back continuously to the full configuration space, thus opening up the possibility of reducing the control problem to that on a simpler space.

We believe that the benefits associated with using the full configuration space to tune optimal dynamical cycles justifies an exploration into the feasability of these challenging spaces.

References

[1] R. Alami, T. Simeon, and J.-P. Laumond. A geometrical approach to planning manipulation tasks. the case of discrete placements and grasps. In H. Miura and S. Arimoto, editors, *Robotics Research*, pages 453–463. MIT Press, 1990.

[2] J. J. Bartholdi and L. K. Platzman. Decentralized control of automated guided vehicles on a simple loop. *IIE Transactions*, 21(1):76–81, 1989.

[3] Y. A. Bozer and M. M. Srinivasan. Tandem configurations for automated guided vehicle systems and the analysis of single vehicle loops. *IIE Transactions*, 23(1):72–82, 1991.

[4] Y. A. Bozer and C.-K. Yen. Intelligent dispatching rules for trip-based material handling systems. *J. Manufacturing Systems*, 15(4):226–239, 1996.

[5] G. A. Castleberry. *The AGV Handbook*. Braun-Brumfield, Ann Arbor, MI, 1991.

[6] S. B. Gershwin. *Manufacturing Systems Engineering*. Prentice Hall, Englewood Cliffs, NJ, 1994.

[7] R. Ghrist. Configuration spaces of graphs in robotics. To appear in *Proc. in Honor of Joan Birman's 70th Birthday*, 1999.

[8] R. Ghrist and D. Koditschek. Safe cooperative robot dynamics on graphs. In preparation.

[9] R. Ghrist and D. Koditschek. Safe cooperative robotic motions via dynamics on graphs. In Y. Nakayama, editor, *8th Intl. Symp. on Robotic Research*. Springer-Verlag, 1998. To appear.

[10] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Press, Boston, MA, 1991.

[11] J.-H. Lee, B. H. Lee, M. H. Choi, J. D. Kim, K.-T. Joo, and H. Park. A real time traffic control scheme for a multiple agv system. In *IEEE Int. Conf. Rob. Aut.*, pages 1625–1630, Nagoya, Japan, 1995.

[12] W. L. Maxwell and J. A. Muckstadt. Design of automatic guided vehicle systems. *IIE Transactions*, 14(2):114–124, 1981.

[13] J. R. Munkres. *Topology, A First Course*. Prentice Hall, 1975.

[14] P. Svestka and M. H. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. In *IEEE Int. Conf. Rob. Aut.*, pages 1631–1636, 1995.