SAND2000-0819C

TOPOLOGICAL CHARACTERIZATION OF SAFE COORDINATED VEHICLE MOTIONS

R. James Milgram
Department of Mathematics
Stanford University

Stephen G. Kaufman
Intelligent Systems and Robotics Center
Sandia National Laboratories

ABSTRACT

This paper characterizes the homotopy properties and the global topology of the space of positions of vehicles which are constrained to travel without intersecting on a network of paths. The space is determined by the number of vehicles and the network. Paths in the space correspond to simultaneous non-intersecting motions of all vehicles. We therefore focus on computing the homotopy type of the space, and show how to do so in the general case. Understanding the homotopy type of the space is the central issue in controlling the vehicles, as it gives a complete description of the distinct ways that vehicles may move safely on the network. We exhibit graphs, products of graphs, and amalgamations of products of graphs that are homotopy equivalent to the full configuration space, and are far simpler than might be expected. The results indicate how a control system for such a network of vehicles (such as a fleet of automatically guided vehicles guided by wires buried in a factory floor) may be implemented.

INTRODUCTION

The computational intractability of motion planning in high dimension configuration spaces is well known. Configuration spaces which incorporate the geometry of manipulators and the environment have both metric and topological properties.

In the present paper, we study configuration spaces of points, not geometric figures. Points represent position of vehicles. This allows us to concentrate on the topological properties of the space without the added complexity of metric properties. While it may seem that such spaces are not of practical interest, this is not true, since such spaces can be used to represent configurations of vehicles. There is growing interest in controlling large number of vehicles, and the methods herein may find application in this area. Also, the methods used in the sequel are of interest in their own right, since they lead to a very simple description of the topological properties of interest.

The particular application we study is that of vehicles constrained to follow fixed routes, such as automatically guided vehicles (AGVs) guided by energized wires in a factory floor. Our characterization of the homotopy type of the global configuration space is as a union of products of cubes and graphs. In the special case where the graph has a single node, the associated space is again a graph. In this space the homotopy classes of paths with given endpoints represent the possible safe (i.e., collision-free) motions of vehicles with given initial and final configurations. Hence, the motion planning problem for coordinated vehicles becomes a graph search. Furthermore, this characterization of safe can be used to implement reactive scheduling of vehicles.

Related Work

This paper was motivated by [GhKo98]. In that work, the authors consider the problem of safely coordinating the motions of mobile vehicles on fixed routes. "Safely" means avoiding collisions at route intersections. They focus on the problem of developing local collision avoidance strategies that can be integrated into the vehicles' controllers, so that a global specification of the required movements of the vehicles can be perturbed into safe movements. By using vector fields to specify the motions of the vehicles on the network of paths, the tools of dynamical systems theory are available to study the safety of the control laws realized by the vector fields. In particular, a "circulating" vector field on the configuration space of a three-way intersection is exhibited, and proven to guarantee that two vehicles will not cross the intersection simultaneously.

In contrast, we examine the structure of the global configuration space of an arbitrary number of vehicles on any network. At the conclusion of the paper, we will explain how the understanding of the global structure can be used in controllers to address the same problem of resolving potential collisions considered in [GhKo98].

Approach

The configuration space of the vehicles on the fixed network is replaced by a smaller, homotopy equivalent space, for which the fundamental group can be easily determined. The construction is in two stages. The network is represented as a graph in the obvious way, with nodes corresponding to path intersections and edges to non-intersecting portions of paths. The

This work was supported by the Laboratory Directed Research and Development program of Sandia National Laboratories, a multiprogram laboratory operated by Sandia Corporation for the United States Department of Energy under Contract DE-AC04-94AL85000.

DISCLAIMER

first step is to construct the configuration space for vehicles on a single "element" of the network, where an element is a single node and its incident edges. This space is determined by the number of incident edges and the number of vehicles in the element. In general, the dimension of the resulting configuration space is very large, so methods of constructing a simpler space are desirable. As homotopy is the critical property of the space in this application, the simpler space must be homotopy equivalent to the full space. We present methods for constructing such a space, and characterize its fundamental group.

The second step is to glue together the fundamental groups so obtained, each representing a network element, to obtain the fundamental group for the entire network. We describe each construction in turn, after summarizing the relevant topological concepts and facts.

The rest of this paper is organized as follows. The necessary topological concepts are introduced in the next section. The construction of a space homotopy equivalent to the configuration space of a single network element is given in the third section. The fourth section describes how these spaces are glued together. The final section discusses applications to controllers.

TOPOLOGICAL PRELIMINARIES

The following concepts and facts are fundamental to the sequel, since we are concerned with paths through the space of vehicle configurations. In particular, homotopy is the basic topological concept describing when paths are equivalent. More detail can be found in any number of books; good references are [Ma91] and [Ma96].

The *configuration space* of k-tuples of points in X, $C_k(X)$, consists of all ordered k-tuples of points $\langle x_1, x_2, \ldots, x_k \rangle$ $x_i \in X$ with *no two the same*, i.e., $x_i \neq x_j$ if $i \neq j$.

The *deleted symmetric product* of k-tuples of points in X, $DP^k(X)$, consists of all *unordered* k-tuples of points $\langle x_1, x_2, \ldots, x_k \rangle$ $x_i \in X$ with *no two the same*, i.e., $x_i \neq x_j$ if $i \neq j$.

Two continuous maps $f, g : X \to Y$ are said to be *homotopic* if they can be continuously deformed into one another; i.e., if there exists a continuous function $F : I \times X \to Y$ such that $F(0, x) = f(x)$ and $F(1, x) = g(x)$ for all $x \in X$.

If $x \in X$, a *loop* based at x is a continuous map $u : I \to X$ such that $u(0) = u(1) = x$. Loops at a fixed base point break up into equivalence classes under the relation of being homotopic by homotopies which fix the base point.

These equivalence classes of loops form a group, the *fundamental group* of X based at x_0, $\pi_1(X, x_0)$. The group structure is obtained by defining the product of two loops as their concatenation, and a loop's inverse as "running it backward". Note the sensitivity to base point in the definition of fundamental group.

The fundamental group of a graph is a free group and the number of generators is one minus the Euler class of the graph.

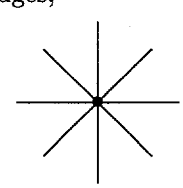
A subspace Y of X is a *deformation retract* of X if there is a homotopy $H: I \times X \to Y$ so that (1), $H(t, y) = y$ for all $y \in Y$, (2) $H(0, x) = x$ and $H(1, x) \in Y$ for all $x \in X$. If there is a deformation retraction from X to Y then Y has the same homotopy type as X, so $\pi_1(X, y_0) = \pi_1(Y, y_0)$.

CONFIGURATION SPACE OF A NETWORK ELEMENT

The main result of this section is that the configuration space of a network element is homotopy equivalent to a bipartite graph. We will show how to compute the number of generators of this free group as an explicit function of the number of edges incident on the node and the number of vehicles in the network element, as well as describing the generators themselves.

We construct the space of k vehicles on a network X by taking the deleted symmetric product $DP^k(X)$. This effectively makes each vehicle anonymous, and reduces the combinatorics of the analysis by a factor of $k!$ as distinguished from the case where the k-tuples are *ordered*. Certainly the number of individual configurations is reduced by this factor; but also, the fundamental group of the configuration space of *ordered* k-tuples is a normal subgroup of the fundamental group of $DP^k(X)$, with index $k!$ and there are explicit algorithms for constructing its generators from those of $\pi_1(DP^k(X))$.

We begin with the basic case where $X = X_m$ is a node, A, with m edges,



Here the center is A. (Formally, X_m is given as the identification space made out of the disjoint union of m copies of the unit interval, where we identify all the origins with a single base point $*$:

$$X_m = \coprod_1^m I_i / (0_i \sim * \mid 1 \leq i \leq m).$$

In the literature, this construction is also called the *wedge product* or *one-point union*).

To characterize the number of generators of $\pi_1(DP^n(X_m))$, we need to define the following coefficients:

Notation: Let $(a_1, a_2, \ldots, a_m)$ be an ordered partition of k by m elements. By this we mean, first, that the $a_i \geq 0$ for each i and second $\sum_1^m a_i = k$. (Note particularly that 0's are allowed.) We say that the *length* of the ordered partition $(a_1, \ldots, a_m)$ is s if and only if exactly s of the a_i are greater than 0. Then we set $P(m, k, s)$ equal to the *number of partitions of k by m elements of length exactly s*. Note that $P(m, k, s) = 0$ if $s \geq min(k, m)$. We now put

$$M(m,k) = \sum_{l \leq m} P(m,k,l)(l-1),$$

$$P(m,k) = \sum_{l \leq k} P(m,k,l)$$

We can now state the main result of this section:

Theorem: *The fundamental group $\pi_1(DP^k(X_m))$ is the free group on $M(m,k) - P(m,k-1) + 1$ generators.*

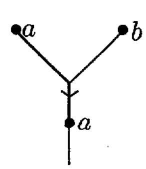
Before proving this theorem, we give two examples of its application. Three lemmas needed in the proof are also given.

Example 1: Fix $k = 2$, and let m vary. Then $P(m,k,1) = m$, $P(m,k,2) = \binom{m}{2} = \frac{m(m-1)}{2}$ so $\pi_1(DP^2(X_m))$ is the free group on

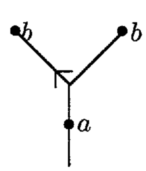
$$\frac{m(m-1)}{2} - m + 1 = \frac{(m-1)(m-2)}{2}$$

generators. It turns out that we can also describe the generators themselves. Assume that the base point in X_3 is given as the pair of points, $\langle a, b \rangle$ where a is the vertex of the first edge and b is the vertex of the second. A generator for $\pi_1(DP^2(X_3)) = \mathbf{Z}$ can be given as the following sequence of moves:

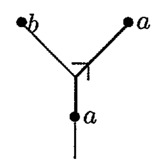
(1) the point on the first edge moves down it, through the node, and into the third edge:



(2) the point in the second edge moves down to the node and the up to the position formerly occupied by the first point in the first edge.



(3) the first point, now on the third edge moves up the second edge to the position formerly occupied by the second point.



More generally, this type of move, using any one of the the $m - 2$ free edges for the place to park a while b moves to the position originally occupied by a gives $m - 2$ of the generators, (assuming the same kind of basing condition). The remaining generators are obtained by moves where first we move the point in the first edge into the s-edge, $s \neq 1, 2$, and then move the point in the second edge into the t-edge, $t \neq 1, 2, s$, and from there, move the point in s to 2 and the point in t to 1.

Example 2: Fix $k = 3$. Then $\pi_1(DP^3(X_m))$ is free on

$$2\binom{m}{3} + m(m-1) - \binom{m}{2} - m + 1 = \frac{m(m-1)(2m+3)}{6}$$

generators. We will see, however, in the proof of the theorem, that the genesis of these generators is not essentially different from those in $\pi_1(DP^2(X_m))$.

We now state the needed lemmas.

Lemma 1: *Let I be the half-open unit interval. Then $DP^m(I)$ is piecewise linearly homeomorphic to the simplex σ^m with all but one face deleted.*

Proof: The simplex σ^m is given as the set of m-tuples of points $\sigma^m = \{(t_1, t_2, \ldots, t_m) \mid 0 \leq t_1 \leq t_2 \leq \cdots \leq t_m \leq 1\}$. This is clearly identical to a specification of the configuration of m points on I, which is $DP^m(I)$. The deleted faces are those containing a vertex corresponding to the missing endpoint, or points describing configurations where two points are at identical positions. ∎

Lemma 2: *Suppose that X is the disjoint union of m copies of the (half-open) interval. Then $DP^k(X)$ is the disjoint union of products of the form $DP^{k_1}(I_1) \times DP^{k_2}(I_2) \times \cdots \times DP^{k_m}(I_m)$, such that $\sum k_i = m$, and $k_i \geq 0$.*

(Evident.) ∎

Example: If $k = 2$ and $m = 3$, the portion of $DP^2(X)$ that does not include configurations with one vehicle at A is the disjoint union of three copies of $DP^2(I)$, one for each edge, and three copies of

$DP^1(I) \times DP^1(I)$, one for each *unordered* pair of distinct edges. $DP^2(I)$ is the simplex σ^2 (a triangle) missing one edge, and the closure of $DP^1(I) \times DP^1(I)$ in $DP^2(X)$ is a rectangle missing two edges and the point $(0,0)$ common to the closures of the two edges present.

Because of these lemmas, we may conclude that the configuration space of a network element having no vehicle at the central node A is the disjoint union of products of simplices. Products of simplices are homeomorphic to balls. To incorporate configurations having a vehicle at A, we must consider what happens when a vehicle leaves an edge to go to A. The answer is very simple; there is now one less vehicle on that edge. Therefore the ball corresponding to a configuration with m vehicles on the edge has on its boundary a ball corresponding to the configuration with $m-1$ vehicles on this edge and the same number of vehicles on each of the other edges as was originally the case.

In one of the balls of $DP^k(X_m - \{A\})$, there will be several such boundary pieces, one for each of the edges with one or more vehicles on it.

The next lemma gives us the key step required to transform $DP^k(X_m)$ into a graph while preserving homotopy.

Lemma 3: *Suppose that $D_1^{n-1}, \ldots, D_j^{n-1}$ are disjoint balls on the boundary of the ball D^n. Let M be the mid-point of D^n, M_l be the mid-point of D_l^{n-1}, $1 \leq l \leq j$, and suppose that deformation-retractions from the identity to M_l are given on each D_l^{n-1}. Let V be the graph which consists of the lines connecting the mid-point of D^n, M, to the M_l. Then the deformation-retractions on the D_l^{n-1} extend to a deformation retraction from the identity on D^n to V.*

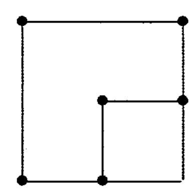
Proof: Define a deformation retraction $f : I \times (D^n, V) \to (D^n, V)$ as follows. f agrees with each given deformation retraction of the D_l^{n-1}, (which are identified with subsets of the boundary of D^n). It is then extended to the *cones from the center M to these* D_l^{n-1} in the obvious way by just mapping the line between $d \in D_l^{n-1}$ and M linearly onto the line between $f(t,d)$ and M. Finally, the points in the remainder of D^n move towards the line between the center of the *nearest* D_l^{n-1} at the same time that they move towards the center M. ∎

We now prove Theorem 1 in the special case of $DP^2(X_m)$. The proof of the general case is not essentially different, and we will discuss the minor changes needed after the proof.

Proof of Theorem 1: The configuration represented by a point in $DP^2(X_m)$ either contains the node A or it doesn't. Therefore $DP^2(X_m)$ consists of two sets: the first of points containing A, which we denote $DP^2(X_m)_A$; and its complement, consisting of all the points of $DP^2(X_m)$ which do not contain A. Now, $DP^2(X_m)_A$ is clearly just the disjoint union of copies of the half-open intervals $I_j - \{0\}$, and the second of them is $DP^2(X_m - \{A\})$. As discussed in the example following Lemma 2, this set consists of the disjoint union of rectangles indexed by pairs of disjoint sides, and one simplex σ^2 in case both points are in a single half open edge.

In order to apply lemma 3 in the present case, we must observe that each edge consisting of points containing A is in the boundary of $m-1$ rectangles and one simplex (corresponding to a vehicle transiting from one arm to another). Therefore we must ensure that the presence of these edge identifications, which amount to taking the closures of the rectangles and the simplices, do not interfere with the deformation retraction indicated in the proof of lemma 3.

Notice that the closure of a rectangle in $DP^2(X_m)$ *does not contain the point* $(0,0)$ since this would require two points to be at the vertex A simultaneously, and this is not allowed. Thus, in the closure, the two half open edges that are added on to the rectangle are disjoint. Consequently, we can apply Lemma 3, and replace the rectangle by a single line segment from the middle of the first added edge to the midpoint of the rectangle, together with a line segment from the midpoint of the second added edge to the midpoint of the rectangle:



Similarly, when we consider the simplexes, they only add one edge in the closure, and hence can simply be replaced by the mid-point of this edge up to homotopy equivalence.

Since in each closure the deformation retraction to the graph can obviously be made to agree on the common edges from Lemma 3, we find that the entire space $DP^2(X_m)$ deformation retracts to a bipartite graph given in the following way.

(1) There are $\binom{m}{2}$ vertices corresponding to the rectangles,

(2) m vertices corresponding to the common edges, and

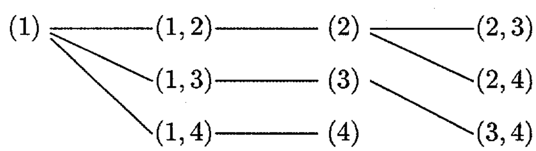
(3) from each vertex corresponding to a rectangle

there are exactly two edges going to two distinct vertices corresponding to the common edges.

The Euler class of the resulting graph is thus $m - \binom{m}{2}$ and the proof for $k = 2$ is complete. ∎

The proof in the general case is not essentially different. The lemmas above guarantee that we can deformation retract in the general case in exactly the same way in which we did it above with the rectangles. The major difference being that here there are as many as m disjoint faces in each product of simplices which are in the closure. ∎

It remains to describe the structure of the generators of the fundamental groups for one of the spaces $DP^n(X_m)$. Again, let us start with the case where $n = 2$. We begin by constructing a maximal tree in the graph above, which is homotopy equivalent to $DP^2(X_m)$. We start with the vertex (1). This connects with the vertices $(1,2)$, $(1,3)$ to $(1,m)$. Next $(1,i)$ connects to i, $2 \leq i \leq m$. Then (2) connects with $(2,i)$, $i \geq 3$, while (3) connects to $(3,i)$ $i \geq 4$ and so on. This gives the maximal tree. The remaining edges are in one to one correspondence with the generators of the fundamental group.



The tree for $DP^2(X_4)$

For example, consider the first edge not in this list – the edge connecting (3) to $(2,3)$. Since we assumed that our base point was in $(1,2)$ we have that $(2,3)$ represents the following edge path:

$$(1,2) \rightarrow (1) \rightarrow (1,3) \rightarrow (3) \rightarrow (2,3) \rightarrow (2) \rightarrow (1,2).$$

Clearly, this is represented by first moving the particle in (2) to (3), then moving the particle in (1) to (2), and finally moving the particle which is now in (3) to (1). This is the sequence of moves given in our first example.

Next, consider the generator associated to the edge connecting (4) to $(3,4)$. This is represented by the sequence of edges:

$$\begin{aligned}(1,2) \rightarrow (1) \rightarrow (1,4) \rightarrow (4) \rightarrow (3,4) \rightarrow (3)\\ \rightarrow (1,3) \rightarrow (1) \rightarrow (1,2)\end{aligned} \qquad (4 - \text{loop})$$

which means first move the particle in (2) to (4), then the particle in (1) to (3), then the particle in (4) to (1) and finally the particle in (3) to (2) – the sequence of moves given in the second type of example.

If we change the base point to $(1,4)$, then the associated loop is

$$(1,4) \rightarrow (4) \rightarrow (3,4) \rightarrow (3) \rightarrow (1,3) \rightarrow (1,4)$$

which is a loop of the three move type in our first example, and the loop of 4–loop is obtained by base point change, using the edge path $(1,2) \rightarrow (1) \rightarrow (1,4)$. An easy generalization of this construction shows that for $DP^2(X_m)$, and more generally $DP^n(X_m)$, when we allow base point changes every generating loop can be chosen to be of the three move type.

CONFIGURATION SPACE OF THE ENTIRE NETWORK

Let X be a general graph corresponding to the entire network, with nodes $A_1, \ldots, A_m$ These nodes and their incident edges correspond to the network elements. Some edges are connected to a single node only; these correspond to paths at the edge of the network. Suppose that there are exactly b_i edges incident on A_i, with $b_i \geq 3$ for all $1 \leq i \leq m$ (if $b = 2$, then there is only one path into and out of the node, so it is not an intersection). A certain number, s, of the edges of X have *both endpoints* at nodes. A vehicle transits from one network element to another by crossing the midpoint of one of these edges. For this reason, we distinguish the midpoints of these s edges calling them $B_1, \ldots, B_s$, respectively (in the case where X is a tree we can renumber so that B_i lies on the line connecting A_i with A_{i+1}, and there will be exactly $m-1$ of the B_i). The basic idea of the construction is to link configurations of network elements (whose structure we now understand) with the configurations resulting from a vehicle crossing one of the B_i. We do this by defining a series of subsets of $DP^n(X)$ as follows.

Definition: $F_l(DP^n(X)) \subset DP^n(X)$ is the subspace where at least l of the n unordered coordinates are contained in the set $\{B_1, \ldots, B_s\}$.

This sequence of subsets (also called a *filtration*) has the following properties:

$$F_s(DP^n(X)) \subset F_{s-1}(DP^n(X)) \subset \cdots \subset F_0(DP^n(X))$$

$$\bigcup_{0 \leq l \leq s} F_l(DP^n(X)) = DP^n(X)$$

Furthermore, $F_l - F_{l+1}$ consists of those configurations having *exactly* l vehicles in the set $\{B_1, \ldots, B_s\}$. It consists of disjoint unions of products of configuration spaces of single nodes:

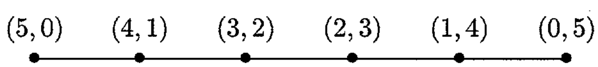
$$DP^{i_1}(A_1) \times DP^{i_2}(A_2) \times \cdots \times DP^{i_m}(A_m)$$

where $\sum_1^m i_j = n - l$, and the disjoint union is taken over all partitions of $n - l$ into m subsets, with 0's allowed. By the second property of the filtration, we know that

$$\bigcup_{0 \leq l < s} (F_l(DP^n(X)) - F_{l+1}(DP^n(X))) = DP^n(X).$$

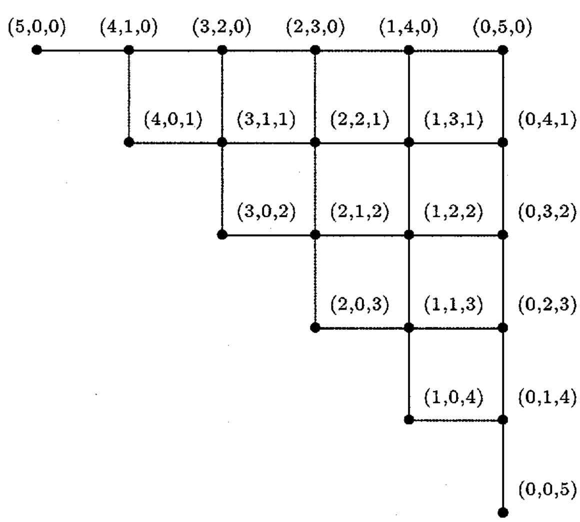
So we need only take the union of these sets, whose structure we already know, to determine $DP^n(X)$.

To make the resulting structure reflect the paths between various configurations, we "thicken" the F_l by allowing the points at the B_j to move slightly away towards either node. This replaces $F_l - F_{l+1}$ by the product $I^l \times (F_l - F_{l+1})$, $0 \leq l \leq s$, and allows us to build up $DP^n(X)$ as a union of l-cubes producted with disjoint unions of products of the $DP^j(X_w)$. For example, if we have a tree with two nodes (so $s = 1$), one gets the following picture for $n = 5$:

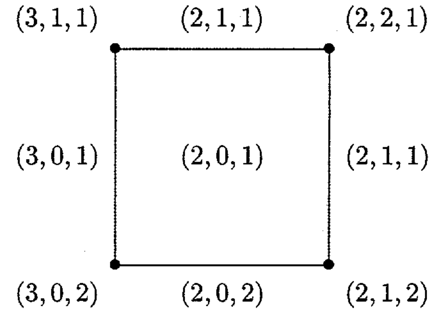


(in this and subsequent figures, $(x_1, \ldots, x_k)$ means that x_j vehicles are in the jth network element, $1 \leq j \leq k$).

Likewise, if X is a tree with three nodes (so $s = 2$), and again 5 points then one gets the following picture:



The interior of each rectangle corresponds to a single component of the level two filtration (which is $F_2(DP^5(X))$), where one vehicle is at B_1 and the other is at B_2. The remaining points are distributed among the three network elements. A closer look at one of the rectangles, pictured below, shows what each node, edge, and rectangle describe:



The vertical edges represent the single components in the filtration that have one vehicle at B_2 and the remaining four points distributed as follows: $(3, 0, 1)$ and $(2, 1, 1)$ while the horizontal lines represent the single components in the filtration having one vehicle at B_1 with the remaining four vehicles distributed as $(2, 1, 1)$ and $(2, 0, 2)$.

When the graph is not a tree we first consider a maximal tree contained in it, and then adjoin, one at a time, the remaining edges with both ends at nodes. This allows us to construct the fundamental group one step at a time, first the group for $DP^k(X)$ with X a tree, and then the groups as we add successive edges. The key tool in this construction is the Van Kampen theorem, which states that the fundamental group of two intersecting spaces is given as the *amalgamated product* of the two spaces. This theorem and its applications are the subject of Chapter 4 of [Ma91].

This is applied more or less directly as we build up the maximal tree. The result here is a group generated by the 3-moves at the various nodes, with the relations generated by the requirement that it does not matter in which order we do moves at *different* nodes.

The groups that result when we adjoin the remaining edges are somewhat more complex, because of the loops they introduce. A single new generator is adjoined for each such edge, and the resulting group is of a type called a Higman, Neumann, Neumann (or HNN) extension. See ([MKS76], p. 403) for details and references about HNN extensions.

CONTROL IMPLEMENTATION

The natural question to ask now is how to exploit the characterization developed above in a control system for vehicles on a fixed network. We sketch an answer in the following.

There are two levels of control: the global and the per-node. They correspond directly to the two levels of homotopy characterization given in the previous section. A path through the global homotopy graph corresponds to the global control. Each node in that graph represents a placement of vehicles on the various paths in the network. Moving from one edge to another in the global homotopy graph requires that one or more vehicles go through intersections in the network, and how this is done at each intersection is determined by the per-node controllers. These use the fundamental group computed as indicated above to determine how the intersection crossings should take place. Whether per-node control is implemented as a stand-alone system, or distributed among the vehicles, is a question we do not consider here.

REFERENCES

[GhKo98] Ghrist, R.W., and Koditschek, D.E., 1998, "Safe Cooperative Robot Patterns via Dynamics on Graphs", *Robotics Research: the Eighth International Symposium*, Springer-Verlag. pp. 81 — 92

[MKS76] Magnus, W., Karrass, A., and Solitar, D. *Combinatorial Group Theory*, Dover (originally published: Interscience Publishers, 1966)

[Ma91] Massey, W.S., *A Basic Course in Algebraic Topology*, Graduate Texts in Mathematics, 127, Springer

[Ma96] Maunder, C.R.F., 1996, *Algebraic Topology*, Dover (originally published: London: Van Nostrand Reinhold Co., 1970)