Reconfiguration of Identical Vehicles in 3D

M. E. Broucke[†]
Department of Electrical and Computer Engineering
University of Toronto, Toronto ON Canada M5S 3G4
broucke@control.utoronto.ca

***Abstract*— We present efficient linear programming based algorithms for reconfiguring identical point mass vehicles without collision in 3D space.**

I. Introduction

In [3] we initiated a study of efficient planning algorithms for reconfiguring a collection of identical point mass vehicles moving in the plane. In this paper we extend the results to 3D motion. Further we relax the general position assumption that provided a sufficient condition for reconfiguring the vehicles without collision. Our new general position assumption is: the initial and final positions of the vehicles must be distinct points. Consider the following idealized problem.

We are given N vehicles modeled as point masses with zero width moving in 3D space. Vehicle i has initial position $s_i = (x_i^0, y_i^0, z_i^0)$ and final position $t_i = (x_i^f, y_i^f, z_i^f)$. All positions are assumed to be distinct. At time t vehicle i is at location $p_i(t) = (x_i(t), y_i(t), z_i(t))$. The vector of positions of all the vehicles $p = (p_1, \dots, p_N)$ is called a *configuration*.

Reconfiguration problem: *Given N point mass vehicles with zero width, an initial configuration p^0, and a final configuration p^f, find a set of N paths connecting p_i^0 to p_i^f, $i = 1, \dots, N$, such that the sum of the distances travelled is minimized and no vehicles collide.*

It is too strict to require that identical vehicles move to prespecified target locations; further the requirement of disjoint paths is too strict to ensure there are no collisions. Nevertheless, disjoint paths are a desirable feature for safety. Hence, we propose a strategy that finds as many disjoint paths as possible, with the remainder of the paths arranged so that we can guarantee no collisions. This is done by relaxing the problem to allow vehicle i to go to any target location t_j. We have a problem of finding a matching between two sets $S = \{s_1, \dots, s_N\}$ and $T = \{t_1, \dots, t_N\}$. Let $G = (S, T, E)$ with $|S| = |T| = N$ be a bipartite graph with edge weights $w : E \to \mathbb{R}$. A *matching* is a subset $M \subseteq E$ such that no two edges in M share a vertex. A *perfect matching* touches all vertices exactly once, i.e. $|M| = N$. The *minimum weight matching problem* is to find a perfect matching M such that the sum of the weights of the edges in M is minimum over all possible perfect matchings. The bipartite weighted matching problem is also called the *assignment problem*. The classic solution of the assignment problem runs in time $O(N^3)$ and is called the *Hungarian Method* [13].

The solution of the matching problem does not guarantee that vehicles follow collision free paths. In [3] we exploited the underlying geometry of the problem, namely, that the weights are derived from a Euclidean metric, to show that a weighted matching algorithm always produces "cross" free (vertex disjoint) paths. For if two paths cross, then the targets can be swapped to yield two paths with strictly lower cost, a contradiction. This property is independent of the number of dimensions considered. The result is that any integer-valued solution to the assignment problem will solve a vertex disjoint paths problem, under suitable general position assumptions. A benefit of this approach is that we are able to use standard algorithms from graph theory and linear programming. The Integrality theorem ensures that an integer-valued solution to the linear program always exists [1]. Improvements can be obtained by using special features of the weighted bipartite matching problem to obtain algorithms with running time $O(N^{2.5} \log N)$ [19], [20]. This solution approach is extended to other metrics: the 3D Manhattan metric and the 3D diagonal Manhattan metric. These metrics are useful when considering vehicles with non-zero width.

There is a rapidly growing literature on multivehicle problems. Graph theoretic approaches to multivehicle planning problems, in particular, have recently been considered in [2], [4], [6], [9], [14], [16], [17], and [18].

The paper is organized as follows. In section II we give the basic idea when distances are defined using the Euclidean metric. In section III we develop the matching algorithm for 3D rectilinear grid graphs. In section IV we examine the problem for 3D diagonal grid graphs. Brief conclusions are stated in section V.

II. Reconfiguration Problem

Consider a set of N vehicles with initial positions $s_i \in \mathbb{R}^3$ and final positions $t_i \in \mathbb{R}^3$, $i = 1, \dots, N$. The notation s_i or t_i is used both to represent points in $\mathbb{R}^3$ and to label vertices of a graph. Let $S = \{s_1, \dots, s_N\}$ and $T = \{t_1, \dots, t_N\}$. We assume that the collection of points $S \cup T$ satisfy the following general position assumption: *all points in S and T are distinct and if three or more points are colinear, then they are arranged in an alternating sequence of points from S and from T.* We define a weighted complete bipartite graph $G = (S, T, E)$. The edge weight $w(e)$ for $e = (s, t)$ is the Euclidean distance between s and t. Our goal is to find a matching $M \subset E$ such that:

†This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

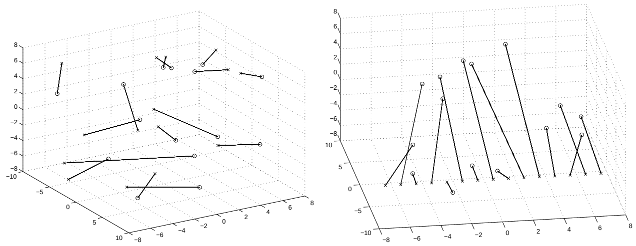

Fig. 1. Euclidean paths for random and linear arrangements of source and target positions.

1) The sum of the distances traveled $\sum_{e \in M} w(e)$ is minimized.
2) The straight line segments corresponding to edges included in the matching are disjoint. That is, for any $e = (s,t)$, $e' = (s',t')$, if $e, e' \in M$, then the segments $\overline{st}$ and $\overline{s't'}$ do not intersect. When two such segments intersect, we call it a *crossing*.

The following theorem for vehicles reconfiguring in $\mathbb{R}^3$ is proved by the same arguments as the proof for $\mathbb{R}^2$ found in [3].

Theorem 1: The solution of the minimum weight bipartite matching algorithm yields a matching with no crossings.

Results for 15 vehicles based on a linear programming implementation in Matlab are shown in Figure 1.

III. Reconfiguration on 3D Grid Graphs

In this section we consider a weighted complete bipartite graph $G = (S, T, E)$ where the positions of the vertices lie on the integer grid in $\mathbb{R}^3$ and the edge weight $w(e)$ for $e = (s_i, t_j)$ is the 3D Manhattan distance between s_i and t_j:

$$d(s_i, t_j) = |x_i - x_j| + |y_i - y_j| + |z_i - z_j| \, .$$

To directly extend the idea of using minimum weight bipartite matchings to achieve collision-free reconfiguration of vehicles, the following properties must be satisfied: (1) An appropriate general position assumption holds. (2) The weights of the bipartite graph satisfy the axioms of a metric. (3) If a matching M has a crossing, that is, two paths associated with two distinct edges in M intersect, then swapping the target vertices of the edges eliminates the crossing. (4) After eliminating a crossing by swapping target vertices, the cost of the new matching is strictly lower. The general position assumption for 3D grid graphs we adopt is: all points in S and T are distinct. The second property is satisfied because the weights are defined by the Manhattan metric. Property three does not hold for 3D grid graphs without introducing a more restrictive or unwieldy general position assumption. Property four also does not hold for 3D grid graphs. One can swap two target vertices to obtain paths that do not cross with the cost equal to the cost before the swap. We cannot argue as in Theorem 1 [3] that it is a contradiction to have a crossing in the solution of the minimum weight matching problem. Moreover, if we swap target vertices to eliminate a crossing we may introduce new crossings, and it is not evident whether this process will terminate. Hence, we have two problems to address:

Problem 1: *Given M a solution of a minimum weight bipartite matching problem with weights defined by the Manhattan metric, does there exist a sequence of swaps of target vertices of pairs of edges of M whose Manhattan paths cross, such that a minimum weight matching M' is obtained with a minimum number of crossings?*

Problem 2: *Given M a solution of a minimum weight bipartite matching problem with weights defined by the Manhattan metric, does there exist a sequence of swaps of target vertices of pairs of edges of M whose Manhattan paths cross, such that a minimum weight matching M' is obtained for which we can guarantee there are no collisions?*

Our approach to these problems is to fix a set of paths that achieve the minimum cost solution of the weighted matching problem. Next, we identify among these paths the types of crossings that are feasible in a minimum cost matching. A crossing between particular path types is infeasible if swapping target vertices results in a cost that is strictly lower. Among feasible crossings, we identify two types: those for which swapping target vertices eliminates the crossing and those for which is does not. For those feasible crossings that cannot be eliminated by swapping target vertices, we show there is a sequence of swaps of target vertices of the associated edges such that the vehicles can move along these paths with no collisions. Finally, we prove that there exists a finite procedure that yields a collision free matching.

Let $\alpha \in \{\mathtt{r}, \mathtt{l}, \epsilon\}$, $\beta \in \{\mathtt{b}, \mathtt{f}, \epsilon\}$, and $\gamma \in \{\mathtt{u}, \mathtt{d}, \epsilon\}$ where `r` is "right", `l` is "left", `u` is "up", `d` is "down", `f` is "forward", `b` is "back", and ϵ stands for no motion. We consider paths in a 3D grid which are of the form $\alpha\beta\gamma$, where at least one symbol is not ϵ. Some examples are `ru`, `b`, and `lbd`. We claim that the paths can be organized into classes and that certain types of crossings between paths of a minimum weight matching can only occur between paths of the same class. The classes are:

1) { `r, b, u, rb, ru, bu, rbu` }
2) { `r, f, u, rf, ru, fu, rfu` }
3) { `r, b, d, rb, rd, bd, rbd` }
4) { `r, f, d, rf, rd, fd, rfd` }
5) { `l, b, u, lb, lu, bu, lbu` }
6) { `l, b, d, lb, ld, bd, lbd` }
7) { `l, f, u, lf, lu, fu, lfu` }
8) { `l, f, d, lf, ld, fd, lfd` }

We say that a crossing is *tranverse* if the two paths intersect at a single point only.

Lemma 1: Let M be a solution of the minimum weight bipartite matching problem with weights defined by the Manhattan metric. Then the only feasible transverse crossings of paths associated with edges in M are those between paths belonging to the same class.

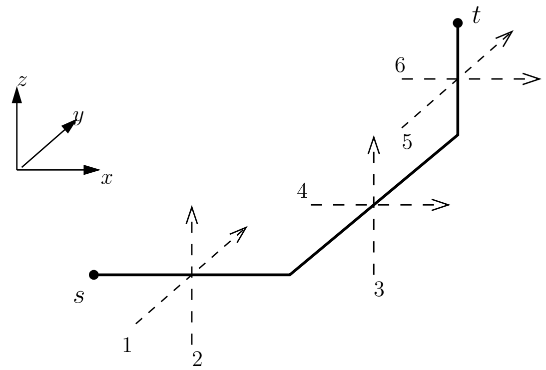

Fig. 2. Feasible transverse crossings for paths in the first class.

Proof: Suppose that $e = (s, t)$ and $e' = (s', t')$ in M have a crossing, and let the coordinates of the points be $s = (x, y, z)$, $t = (x_t, y_t, z_t)$, $s' = (x', y', z')$, and $t' = (x'_t, y'_t, z'_t)$. One can verify by direct calculation that the following conditions are necessary for e and e' to have a feasible transverse crossing, otherwise M is not optimal:

$$
\begin{aligned}
x_t - x > 0 \quad &\Longrightarrow \quad x'_t - x' \geq 0 \\
y_t - y > 0 \quad &\Longrightarrow \quad y'_t - y' \geq 0 \\
z_t - z > 0 \quad &\Longrightarrow \quad z'_t - z' \geq 0\,.
\end{aligned}
$$

For instance, if $x_t - x > 0$, $y_t - y > 0$, and $z_t - z > 0$, meaning the path is of type `rbu`, then a crossing can occur with a path that satisfies $x_t - x \geq 0$, $y_t - y \geq 0$, and $z_t - z \geq 0$, which corresponds to paths `rbu`, `ru`, `bu`, `rb`, `u`, `b`, `r`. If $x_t = x$, $y_t = y$, and $z_t > z$, meaning the path is `u`, then the conditions yield feasible crossings with paths of type `r`, `l`, `f`, `b`, `u`, `rb`, `ru`, `bu`, `rbu`, `rf`, `fu`, `rfu`, `lb`, `lu`, `lbu`, `lf`, `fu`, or `lfu`. Carrying out this argument for each path type, one finds that feasible transverse crossings only occur between paths of the same class, with the classes as given above. ■

Lemma 2: Let M be a solution of the minimum weight bipartite matching problem with weights defined by the Manhattan metric. Suppose that e and e' in M have a transverse crossing. Then if we swap target vertices, the new paths belongs to the same class as that of e and e'.

Proof: By inspection. ■

Lemma 3: If e and e' in M have paths with a transverse crossing and the paths belong to the class C, then after swapping target vertices, if there is a new path (s, t') or (s', t) that belongs to a class $C' \neq C$ then that path cannot have any new crossings with other paths of type C'.

Proof: This follows from the observation that the paths that appear in more than one class consist of at most two segments. One can verify by examining all cases that after swapping target vertices, any new paths with at most two segments always consist of segments formed from intervals of segments of the origin paths (s, t) and (s', t'). Hence, no new crossings can appear on those segments. ■

Now we identify all those transverse crossings which can be eliminated by swapping target vertices. Considering for the moment the path type `rbu`, there are six types of transverse crossings depicted in Figure 2. For each, swapping target vertices eliminates the crossing. The dotted lines represent a segment of the path for e' that crosses e's path. The arrow indicates direction of vehicle motion. By inspection one can verify that if a new crossing is introduced after swapping target vertices, then the coordinates of the new crossing $(\tilde{x}, \tilde{y}, \tilde{z})$ satisfy:

1) $\tilde{x} \in (x'_t, x_t)$, $\tilde{y} \in (y', y)$, $\tilde{z} = z$.
2) $\tilde{x} \in (x'_t, x_t)$, $\tilde{y} \in (y', y_t)$, $\tilde{z} = (z', z)$.
3) $\tilde{x} = x'_t$, $\tilde{y} \in (y'_t, y_t)$, $\tilde{z} = (z', z)$.
4) $\tilde{x} = (x_t, x'_t)$, $\tilde{y} \in (y, y')$, $\tilde{z} = z$.
5) $\tilde{x} = x_t$, $\tilde{y} \in (y_t, y'_t)$, $\tilde{z} = (z, z')$.
6) $\tilde{x} = (x_t, x'_t)$, $\tilde{y} \in (y, y'_t)$, $\tilde{z} = (z, z')$.

Lemma 4: Let M be a solution of a minimum weight bipartite matching problem with weights defined by the Manhattan metric. There exists a finite sequence of swaps of target vertices such that a minimum weight matching M' is obtained which is free of transverse crossings between paths in the first class.

Proof: Suppose that the integer grid is defined by a rectangle $[-L_x, L_x] \times [-L_y, L_y] \times [-L_z, L_z]$ where $L_x, L_y, L_z \in \mathbb{Z}$ are bounded numbers such that all points of S and T are contained in the rectangle. Starting at the vertex $(-L_x, -L_y, L_z)$ we sweep the 3D grid first along the negative z-direction, then along the positive y-direction, and then along the positive x-direction, searching for crossings between paths in the first class. When one is found, we check if it is a transverse crossing, and if so, swap target vertices. By the discussion above, the current crossing is eliminated, and by Lemma 2 the new paths belong to the same class; hence they can have new crossings only with paths within the same class. By Lemma 3, for path types that belong to more than one class, new crossings with paths in another class cannot occur. Potential new crossings have coordinates satisfying conditions 1-6 above. In cases 1,2,4, and 6 above, the new crossings appear to the right of the current crossing, and hence will be encountered later in the 3D sweep. In cases 3 and 5, new crossings may appear along the x coordinate currently being scanned. But these new crossings appear behind (along the y axis) and below (along the z axis) so if we have inner loops sweeping in the positive y and negative z directions, these new crossings will be encountered later in the sweep as well. In this manner for each x, y, and z coordinate, successively, all transverse crossings between paths in the first class can be eliminated. ■

We are left with considering crossings that cannot be eliminated by swapping target vertices, depicted in Figure 3 for the first path class and called *tangential crossings*. The dotted lines show segments of paths that cross tangentially with path (s, t). One can verify by direct calculation that the following conditions are necessary for e and e' to have

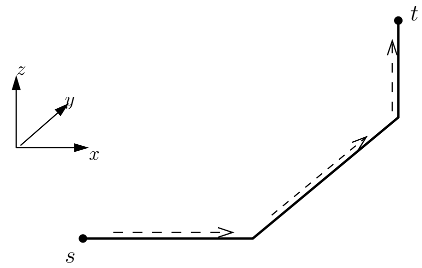

Fig. 3. Feasible tangential crossings for paths in the first class.

a feasible tangential crossing, otherwise M is not optimal:

$$\begin{aligned} y = y', z = z' &\implies sgn(x'_t - x') = sgn(x_t - x) \\ x_t = x'_t, z = z' &\implies sgn(y'_t - y') = sgn(y_t - y) \\ y_t = y'_t, x_t = x'_t &\implies sgn(z'_t - z') = sgn(z_t - z)\,. \end{aligned}$$

It should be noted that these conditions permit tangential crossings between paths in different classes, and particularly between paths in different classes that only belong to one path class. The following lemma provides a procedure to reassign target vertices associated with paths with tangential crossings so that vehicles moving on these paths have no collisions.

Lemma 5: Let M be a solution of a minimum weight bipartite matching problem with weights defined by the Manhattan metric and such that there are no transverse crossings of paths. There exists a finite sequence of swaps of target vertices such that a minimum weight matching M' is obtained such that vehicles taking paths with tangential crossings do not collide.

Proof: We consider the first path class. The other classes can be dealt using the same procedure after an appropriate linear transformation. First observe that if e and e' have a tangential crossing, then after swapping target vertices the new paths consist of segments of the original paths; hence the procedure below introduces no new crossings.

First we group the source/target pairs such that two pairs are equivalent if their sources have the same y and z coordinates and their paths are tangentially crossing. Such paths necessarily intersect along a line parallel to the x axis. Consider each equivalence class. We sort the sources in order of increasing x coordinate. Also sort the targets in order of increasing x coordinate. If two targets have the same x coordinate, sort them in order of increasing y coordinate. If two targets have the same x and y coordinates, sort them in order of increasing z coordinate. Then we match the sources and targets in the order they are sorted. The effect of this sorting is that a source with a smaller x coordinate is matched with a target with a smaller x coordinate relative to other sources whose paths tangentially cross in the same line parallel to the x-axis.

Next, we group the source/target pairs such that two pairs are equivalent if their targets have the same x coordinate and their paths are tangentially crossing. Such paths necessarily intersect along a line parallel to the y axis. Consider each equivalence class. We sort the targets in order of increasing y coordinate. Targets with the same y coordinate are sorted by increasing z coordinate. We sort the sources in order of increasing y coordinate. Sources with the same y coordinate are sorted by increasing x coordinate. Then we match the sources and targets in the order they are sorted. This sorting has the effect that a source with a smaller y coordinate is matched with a target with a smaller y coordinate relative to other sources whose paths tangentially cross in the same line parallel to the y axis. This matching does not disturb the previous matching in terms of x coordinates, because the reassignment of targets is among sources that have targets with the same y coordinate.

Finally we group the source/target pairs such that two pairs are equivalent if their targets have the same x and y coordinates and their paths are tangentially crossing. Such paths necessarily intersect along a line parallel to the z axis. Consider each equivalence class. We sort the targets in order of increasing z coordinate. We sort the sources in order of increasing z coordinate. Sources with the same z coordinate are sorted by increasing y coordinate. Sources with the same z and y coordinate are sorted by increasing x coordinate. Then we match the sources and targets in the order they are sorted. This sorting has the effect that a source with a smaller z coordinate is matched with a target with a smaller z coordinate relative to other sources whose paths tangentially cross in the same line parallel to the z axis. This matching does not disturb the previous two matchings in terms of x and y coordinates, because the reassignment of targets is among sources that have targets with the same x and y coordinates.

Let $[e]$ be the set of edges with tangentially crossing paths containing edge e. That is, $[e]$ is the transitive closure of edges with tangentially crossing paths starting with edge e. It is not difficult to see that the proposed matching of sources and targets allows vehicles to move along paths without collision, by using a distance-velocity decomposition, as in [10]. ■

Theorem 2: Given M a solution of a minimum weight bipartite matching problem with weights defined by the Manhattan metric, there exists a sequence of swaps of target vertices such that a minimum weight collision-free matching M' is obtained.

Proof: For each class of paths, we apply a 3D sweep algorithm as described in Lemma 4. The same sweep algorithm can be applied to the other classes by applying an appropriate linear transformation to the 3D graph. This eliminates transverse crossings within each class. By Lemma 3 elimination of transverse crossings for one class of paths does not introduce new crossings with another class. Thus, we obtain a matching M'' that is free of transverse crossings for all path classes. We are left with tangential crossings. Using Lemma 5 we can apply a procedure that swaps target vertices to obtain a matching M' in which vehicles on paths with tangential crossings do

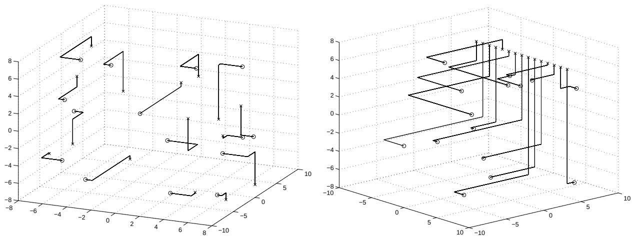

Fig. 4. Manhattan paths for random and linear arrangements of source and target positions.

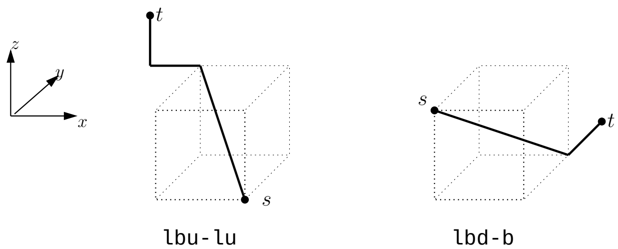

Fig. 5. Sample diagonal Manhattan paths.

not collide. ■

Results for 15 vehicles are shown in Figure 4.

IV. Reconfiguration on Diagonal Grid Graphs

In this section we consider a weighted complete bipartite graph $G = (S, T, E)$ where the positions of the vertices lie on the integer grid in $\mathbb{R}^3$ and the edge weight $w(e)$ for $e = (s_i, t_j)$ is the 3D diagonal Manhattan distance between s_i and t_j. Let $m = \min\{|x_i - x_j|, |y_i - y_j|, |z_i - z_j|\}$. Then

$$d(s_i, t_j) = \begin{cases} \sqrt{3}m + [|y_i - y_j| - m] + [|z_i - z_j| - m], \\ \qquad m = |x_i - x_j| \\ \sqrt{3}m + [|x_i - x_j| - m] + [|z_i - z_j| - m], \\ \qquad m = |y_i - y_j| \\ \sqrt{3}m + [|x_i - x_j| - m] + [|y_i - y_j| - m], \\ \qquad m = |z_i - z_j| \end{cases} .$$

The same four properties as in Section III must be satisfied. The general position assumption for diagonal grid graphs we adopt is: all points in S and T are distinct. For the second property, one can show, as in the planar case, that d is a metric. Neither property three nor property four hold for diagonal grid graphs. As before we fix a set of paths that achieve the minimum weight matching, identify classes of paths that can have feasible crossings, isolate the types of feasible crossings for which crossings can be eliminated by swapping target vertices, and show there is a finite procedure to eliminate those crossings. For the remainder of the crossings, we apply a procedure of swapping target vertices to allow vehicles to move on those paths without collision.

Let ϵ stands for "no vehicle motion". Then let $\alpha \in \{$`r`, `l`$\}$, $\beta \in \{$`b`, `f`$\}$, $\gamma \in \{$`u`, `d`$\}$, $\zeta \in \{$`r`, `l`, `f`, `b`, `u`, `d`, `rf`, `rb`, `ru`, `rd`, `lf`, `lb`, `lu`, `ld`, `fu`, `fd`, `bu`, `bd`, $\epsilon\}$. We consider paths in a 3D grid which are of the form $\alpha\beta\gamma - \zeta$. Some examples are `ru`, `rbd-d`, and `lbu-lu`. Figure 5 shows some sample paths. We claim that the paths can be organized into classes and that a crossing between paths of a minimum weight matching can only occur between paths of the same class. The classes are:

1) {`r, b, u, rb, ru, bu, rbu, rbu-r, rbu-b, rbu-u, rbu-rb, rbu-ru, rbu-bu` }
2) { `r, f, u, rf, ru, fu, rfu, rfu-r, rfu-f, rfu-u, rfu-rf, rfu-ru, rfu-fu` }
3) { `r, b, d, rb, rd, bd, rbd, rbd-r, rbd-b, rbd-d, rbd-rb, rbd-rd, rbd-bd` }
4) { `r, f, d, rf, rd, fd, rfd, rfd-r, rfd-f, rfd-d, rfd-rf, rfd-rd, rfd-fd` }
5) { `l, b, u, lb, lu, bu, lbu, lbu-l, lbu-b, lbu-u, lbu-lb, lbu-lu, lbu-bu` }
6) { `l, b, d, lb, ld, bd, lbd, lbd-l, lbd-b, lbd-d, lbd-lb, lbd-ld, lbd-bd` }
7) { `l, f, u, lf, lu, fu, lfu, lfu-l, lfu-f, lfu-u, lfu-lf, lfu-lu, lfu-fu` }
8) { `l, f, d, lf, ld, fd, lfd, lfd-l, lfd-f, lfd-d, lfd-lf, lfd-ld, lfd-fd` }

The following sequence of lemmas and the theorem are proved in analogy with the previous section.

Lemma 6: Let M be a solution of the minimum weight bipartite matching problem with weights defined by the diagonal Manhattan metric. Then the only feasible crossings of paths associated with edges in M are those between paths belonging to the same class.

Lemma 7: Let M be a solution of the minimum weight bipartite matching problem with weights defined by the diagonal Manhattan metric. Suppose that e and e' in M have a crossing. Then if we swap target vertices, the new paths belongs to the same class as those of e and e'.

Lemma 8: If e and e' in M have paths with a crossing and the paths belong to the class C, then after swapping target vertices, if there is a new path (s, t') or (s', t) that belongs to at least one other class C' then the path cannot have any new crossings with a path in C'.

Lemma 9: Let M be a solution of a minimum weight bipartite matching problem with weights defined by the diagonal Manhattan metric. There exists a finite sequence of swaps of target vertices such that a minimum weight matching M' is obtained which is free of transverse crossings between paths in the first class.

Lemma 10: Let M be a solution of a minimum weight bipartite matching problem with weights defined by the Manhattan metric. There exists a finite sequence of swaps of target vertices such that a minimum weight matching M'

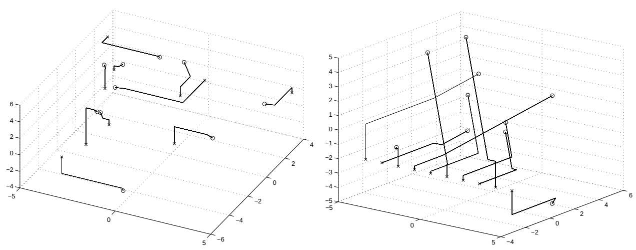

Fig. 6. Diagonal Manhattan paths for random and linear arrangements of source and target positions.

is obtained such that vehicles taking paths with tangential crossings do not collide.

Theorem 3: Given M a solution of a minimum weight bipartite matching problem with weights defined by the diagonal Manhattan metric, there exists a sequence of swaps of target vertices such that a minimum weight collision-free matching M' is obtained.

Results for 10 vehicles are shown in Figure 6.

V. Conclusion

In this paper we study collision-free reconfiguration of coordinated autonomous vehicles that move freely in 3D space or on a 3D rectilinear grid or 3D diagonal grid. The vehicles are required to have appropriate sensors to form and stay on a common coordinate system. Further the vehicles are assumed to have sensors to detect vehicles in their vicinity in order to adjust their speeds appropriately while remaining on their designated paths. Our future work will involve experimental validation of the algorithms.

VI. References

[1] R Ahuja, T. Magnanti, and J. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, 1993.

[2] M. Branicky. Nonlinear and hybrid control via RRT's. *Proc. of Mathematical Theory of Networks and Systems* (MTNS '02) May 2002.

[3] M. Broucke. Disjoint path algorithms for planar reconfiguration of identical vehicles. *Proc. of the American Control Conference*(ACC '03), June 2003.

[4] J. Desai, J. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 17(6):905-908, 2001.

[5] S. Even and R. Tarjan. Network flow and testing graph connectivity. *SIAM Journal on Computing*, 4, 1975, pp. 507-518.

[6] J. Fax and R. Murray. Graph Laplacians and stabilization of vehicle formations. *Proc. of the 15th IFAC World Congress*, July 2002.

[7] M. Garey and D. Johnson. *Computers and Intractibility*. W.H. Freeman Co. 1979.

[8] J. Goodman and J. O'Rourke, eds. *Handbook of Discrete and Computational Geometry*. CRC Press, 1997.

[9] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *Proc. of the IEEE Conference on Decision and Control* (CDC '02), December 2002.

[10] K. Kant and S. Zucker. Toward efficient trajectory planning: the path-velocity decomposition. *International Journal of Robotics Research*. vol. 5, no. 3, 1986.

[11] R. M. Karp, F.T. Leighton, R.L. Rivest, C.D. Thompson, U.V. Vazirani, and V.V. Vazirani. Global wire routing in two-dimensional arrays. *Algorithmica*, vol. 2, pp. 113-129, 1987.

[12] M.R. Kramer and J. van Leeuwen. The complexity of wire routing and finding minimum area layouts for arbitrary VLSI circuits. VLSI Theory. F.P. Preparata, ed. JAI Press, Advances in Computing Research, vol. 2, pp.129-146, 1984.

[13] H.W. Kuhn. The Hungarian method for the assignment problem. *Naval Res. Logist. Quart.*, 2, pp. 83-97, 1955.

[14] N. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. *Proc. of the IEEE Conference on Decision and Control* (CDC '01), 2001.

[15] J.F. Lynch. The equivalence of theorem-proving and the interconnection problem. *ACM SIGDA Newsletter*, 5, pp. 31-65, 1975.

[16] M. Mesbahi and F.Y Hadegh, Formation flying of multiple spacecraft via graphs, matrix inequalities, and switching. *AIAA Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, pp. 369-377, March 2000.

[17] R. Olfati-Saber and R. Murray. Graph rigidity and distributed formation stabilization of multi-vehicle systems. *Proc. of the IEEE Conference on Decision and Control* (CDC '02), December 2002.

[18] R. Olfati-Saber and R. Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. *Proc. of the 15th IFAC World Congress*, July 2002.

[19] J.W. Suurballe. Disjoint paths in a network. *Networks*, no. 4, pp. 125-145, 1974.

[20] P. Vaidya. Geometry helps in matching. *Siam Journal on Computing*, vol. 18, no. 6, pp. 1201-1225, December 1989.