Proceedings of the 2003 IEEE
International Conference on Robotics & Automation
Taipei, Taiwan, September 14-19, 2003

Control of Small Formations Using Shape Coordinates

Fumin Zhang, Michael Goldgeier and P. S. Krishnaprasad

Institute for Systems Research & Department of Electrical and Computer Engineering
University of Maryland at College Park
College Park, MD 20742
{fuminz, mbgold, krishna}@isr.umd.edu

Abstract— **Formations that contain a small number of robots are modeled as controlled Lagrangian systems on Jacobi shape space. This allows a blocked decoupled control for position, orientation and shape of the formation. Feedback control laws are derived using control Lyapunov functions. The controlled dynamics converges to the invariant set where desired shape is achieved. Controllers are implemented in a layered fashion via the extended motion description language(MDLe) system. Group MDLe plans are constructed to allow structured controller design for formations.**

I. Introduction

Robots in a formation can be viewed as physical objects with the control effort viewed as interaction. Hence a robot formation can be modeled as a controlled Lagrangian system of particles. What are the interactions which allow particles (robots) to form a meaningful (stable) formation is the problem we want to investigate.

The shape of a formation is invariant under translation and rotation. It is independent of the coordinate system we choose to study the whole formation. Jacobi proposed a special class of coordinates ([7], [2], [12]) which served as the starting point to what we call the Jacobi shape space. On this shape space, the global displacements (of translation and rotation) are not present. This shape space concept will be discussed in section II. It is important to robot formation control since without knowing robot coordinates in the laboratory-fixed coordinate system, only shape variables can be measured using on-board sensors of the robots. Ideally, the control laws to achieve desired shape should only depend on shape measurements. However, as suggested by the Lagrange equations derived in section III, it is impossible to achieve completely decoupled control of shape and orientation. In section IV, we design shape controllers using control Lyapunov functions. An estimate of the angular velocity of the whole formation will greatly simplify the task. We are able to control the formation to desired shape with fixed orientation.

To avoid implementing formation controller explicitly on each robot. Imagine the example of commanding a group of soldiers on drill. The commander will want to issue commands for the whole group instead of commanding each individual soldier. The decoupled control laws proposed in section IV allow us to take a hierarchical view. The control of overall position and orientation can be viewed as *group level* commands. This inspires us to add group features to an existing unified platform called *the extended motion description language (MDLe)*. This work is presented in section V.

II. Jacobi shape space

To describe the motion of a cluster of particles, we set up a fixed inertial coordinate frame first. Let $q_i \in \mathscr{R}^3$, $i = 1,2,...,N$, denote the coordinates of N particles with mass m_i. The kinetic energy of this cluster is

$$K^{tot} = \frac{1}{2}\sum_{i=1}^{N} m_i \| \dot{q}_i \|^2 \tag{1}$$

This kinetic energy is translation invariant. Let $M = \sum_{i=1}^{N} m_i$. Denote the center of mass as

$$q_c = \frac{\sum_{i=1}^{N} m_i q_i}{M} \tag{2}$$

Now one can define a new set of coordinates $\mathbf{c}_{fi}$ as $\mathbf{c}_{fi} = q_i - q_c$. Then the kinetic energy can be expressed as

$$K^{tot} = \frac{1}{2}M \| \dot{q}_c \|^2 + \frac{1}{2}\sum_{i=1}^{N} m_i \left\| \dot{\mathbf{c}}_{fi} \right\|^2 \tag{3}$$

But since $\sum_{i=1}^{N} m_i \mathbf{c}_{fi} = 0$, we seek $(N-1)$ independent vectors $(\rho_{fi}, i = 1,2,...,N-1)$ from $span(\mathbf{c}_{fi}, i = 1,2,...,N)$. We want ρ_{fi} to be chosen such that the kinetic energy has the form

$$K^{tot} = \frac{1}{2}M \| \dot{q}_c \|^2 + \frac{1}{2}\sum_{i=1}^{N-1} \left\| \dot{\rho}_{fi} \right\|^2 \tag{4}$$

Such a set of ρ_{fi} are called *Jacobi coordinates*. One way of constructing Jacobi coordinates is to let

$$\begin{aligned}
\rho_{f1} &= \sqrt{\mu_1}(\mathbf{c}_{f2}-\mathbf{c}_{f1}) \\
\rho_{f2} &= \sqrt{\mu_2}(\mathbf{c}_{f3}-\frac{m_1\mathbf{c}_{f1}+m_2\mathbf{c}_{f2}}{m_1+m_2}) \\
&\vdots \\
\rho_{fi} &= \sqrt{\mu_i}(\mathbf{c}_{f(i+1)}-\frac{\sum_{k=1}^{i} m_k\mathbf{c}_{fk}}{\sum_{k=1}^{i} m_k}) \\
&\vdots \\
\rho_{f(N-1)} &= \sqrt{\mu_{N-1}}(\mathbf{c}_{fN}-\frac{\sum_{k=1}^{N-1} m_k\mathbf{c}_{fk}}{\sum_{k=1}^{N-1} m_k}) \qquad (5)
\end{aligned}$$

where

$$\frac{1}{\mu_i}=\frac{1}{\sum_{k=1}^{i} m_k}+\frac{1}{m_{i+1}} \quad for\ i=1,2,...,N-1 \qquad (6)$$

A proof for this well-known construction satisfying (4) can be found in [22](see also discussion in [12]). As one can see from these equations, the vectors ρ_{fi} are constructed by finding the scaled relative displacement between the $(i+1)$th particle and the center of mass of the sub-cluster of first i particles. This process depends on how the particles are labeled. We can also change the way we sub-cluster particles as in [2], [11] and [12]. Hence Jacobi coordinates are not unique. However, between any two sets of Jacobi coordinates there exists an element $h \in O(N-1)$ s.t

$$[\rho_{f1}^1,\rho_{f2}^1,...,\rho_{f(N-1)}^1]=[\rho_{f1}^2,\rho_{f2}^2,...,\rho_{f(N-1)}^2]h \qquad (7)$$

This orthogonal group $O(N-1)$ is called *the democracy group* [12].

Let $Q=\mathscr{R}^{3N}$ be the total configuration space of the formation. The space of Jacobi coordinates is $\mathscr{R}^{3N-3}$. We define

$$K=\frac{1}{2}\sum_{i=1}^{N-1}\left\|\dot{\rho}_{fi}\right\|^2 \qquad (8)$$

This K is invariant under the diagonal left action on $\mathscr{R}^{3N-3}$ by the special orthogonal group $\mathbf{G}=SO(3)$. The action is:

$$\Phi_g(\rho_{fi})=g\rho_{fi} \quad for\ g\in\mathbf{G} \qquad (9)$$

This symmetry group $\mathbf{G}$ acts on $\mathscr{R}^{3N-3}$ properly and freely except for the shapes where all ρ_{fi} are collinear. We let the set F_0 be the set of all the Jacobi coordinates corresponding to collinear shapes. Let $F=\mathscr{R}^{3N-3}-F_0$ and call it the *Jacobi pre-shape space*. It is an open submanifold of the configuration space. Since $\mathbf{G}$ acts properly and freely on F, the base space $B=F/\mathbf{G}$ is a smooth manifold and the canonical projection $\pi: F\to B$ is differentiable. B is called the *Jacobi shape space*.

In dropping from F to B, we get rid of the $SO(3)$ symmetry from the Jacobi coordinates. After the reduction, the dimension of the shape space $\mathbf{B}$ is $(3N-6)$. On this shape space we can define *shape coordinates* s^j as

$$s^j=s^j(\rho_{f1},\rho_{f2},...,\rho_{f(N-1)}) \ for\ j=1,2,...(3N-6) \qquad (10)$$

s.t.

$$s^j(g\rho_{f1},g\rho_{f2},...,g\rho_{f(N-1)})=s^j(\rho_{f1},\rho_{f2},...,\rho_{f(N-1)}) \qquad (11)$$

for all $g\in SO(3)$. Candidates for s^j are functions of dot products $(\rho_{fi}\cdot\rho_{fj})$ and triple products $(\rho_{fi}\cdot(\rho_{fj}\times\rho_{fk}))$. Thus, mutual distances, mutual angles, areas and volumes formed by the line segments connecting the particles all serve as candidates for shape variables. There is a large statistical literature on the subject of shape space and shape coordinates([8] [9] [19]).

One can establish a body coordinate system on a formation with certain shape. The reference orientation of this formation can be defined as the orientation when the body coordinate frame and the lab coordinate frame coincide. Then the orientation of this formation with the *same* shape can be described by an element $g\in SO(3)$. The Jacobi coordinates in these two coordinate systems have the following relationship:

$$\rho_{fi}=g\rho_i(s) \qquad (12)$$

where

$$s=(s^1,s^2,...,s^{3N-6})^T. \qquad (13)$$

ρ_i are Jacobi coordinates in the body coordinate frame which only depend on shape coordinates.

Taking derivative on both sides of (12), we get

$$\dot{\rho}_{fi}=\dot{g}\rho_i+g\dot{\rho}_i \qquad (14)$$

On a matrix Lie group $\mathbf{G}$, $\dot{g}\in T_g\mathbf{G}$. There exists $\Omega\in\mathfrak{g}$ the Lie algebra s.t. $\dot{g}=g\Omega$. In our case, $\mathbf{G}=SO(3)$, so $\mathfrak{g}=(\mathscr{R}^3,\times)$. Thus the derivative of $\dot{\rho}_{fi}$ is

$$\dot{\rho}_{fi}=g(\Omega\times\rho_i+\sum_{j=1}^{3N-6}\frac{\partial\rho_i}{\partial s^j}\dot{s}^j) \qquad (15)$$

In the body coordinate frame, the *angular momentum* of the whole system J can be calculated as

$$\begin{aligned}
J &= g^{-1}\sum_{i=1}^{N-1}(\rho_{fi}\times\dot{\rho}_{fi}) \\
&= I(\Omega+\sum_{i=1}^{3N-6}A_j\dot{s}^j) \qquad (16)
\end{aligned}$$

where

$$I(s)=\sum_{i=1}^{N-1}(\left\|\rho_i\right\|^2\mathbf{e}-\rho_i\rho_i^T) \qquad (17)$$

is defined as the *locked inertia tensor* of the formation in the body coordinate frame and

$$A_j(s)=I^{-1}\sum_{i=1}^{N-1}\rho_i\times\frac{\partial\rho_i}{\partial s^j} \qquad (18)$$

are *vector potential functions*. These quantities are defined on the shape space because ρ_i only depend on shape coordinates.

Let

$$A = [A_1, A_2, ..., A_{3N-6}] \tag{19}$$

Then we can rewrite the kinetic energy in block diagonalized form as

$$K^{tot} = \frac{1}{2}M \| \dot{q}_c \|^2 + \frac{1}{2}(\Omega + A\dot{s})^T I(\Omega + A\dot{s}) + \frac{1}{2}\dot{s}^T G\dot{s} \tag{20}$$

where

$$G_{jk} = -A_j^T I A_k + \sum_{i=1}^{N-1} \frac{\partial \rho_i}{\partial s^j}\frac{\partial \rho_i}{\partial s^k} \tag{21}$$

In our paper in preparation [22], a geometrically intrinsic approach will be taken for the above construction following the theory of block diagonalization as in [16], [17], [18], [10], [21].

III. Lagrange equations for formations

By defining Jacobi vectors and then Jacobi shape variables we have gone through a sequence of changes of coordinates on the configuration space Q. In the first step, the transformation is a diffeomorphism between Q and R^{3N} mapping configuration variables q_i to the position of the center of mass q_c and the Jacobi vectors ρ_{fi} . In the second step, the transformation is a local diffeomorphism between $R^3 \times F$ and $R^3 \times \mathbf{G} \times B$ mapping (q_c, ρ_{fi}) where $i = 1,2,...,(N-1)$ to (q_c, g, s^j) where $j = 1,2,...(3N-6)$ s.t.

$$\rho_{fi} = g\rho_i(s^1, s^2, ...s^{3N-6}) \tag{22}$$

The Lagrangian on TQ is given by $L(q,\dot{q}) = K^{tot}(\dot{q}) - V(q)$.

Before the coordinate transformation, K^{tot} has the form in equation (1). The Lagrange equations for the system are:

$$m\ddot{q}_i = u_i - \frac{\partial V}{\partial q_i} \tag{23}$$

for $i = 1,2,...,N$ where u_i are control forces.

After the block diagonalization, K^{tot} is given by equation (20), and one can rewrite the Lagrangian in block diagonalized form using A and G as

$$\begin{aligned} L(q_c, g, s, \dot{q}_c, \dot{g}, \dot{s}) &= \frac{1}{2}M \| \dot{q}_c \|^2 + \frac{1}{2}(\Omega + A\dot{s})^T I(\Omega + A\dot{s}) \\ &\quad + \frac{1}{2}\dot{s}^T G\dot{s} - V(q_c, g, s, \dot{q}_c) \end{aligned} \tag{24}$$

Now

$$\begin{aligned} \frac{\partial L}{\partial \dot{q}_c} &= M\dot{q}_c \\ \frac{\partial L}{\partial \dot{g}} &= gI(\Omega + A\dot{s}) \\ \frac{\partial L}{\partial \dot{s}} &= A^T I(\Omega + A\dot{s}) + G\dot{s} \end{aligned} \tag{25}$$

Then the set of Lagrange equations takes the form

$$M\ddot{q}_c = -\frac{\partial V}{\partial q_c} + u_c \tag{26}$$

$$\frac{d}{dt}(I(\Omega + A\dot{s})) = -\Omega \times I(\Omega + A\dot{s}) - g^{-1}\frac{\partial V}{\partial g} + u_g \tag{27}$$

$$\begin{aligned} \frac{d}{dt}(G\dot{s}) \quad + \quad A^T \frac{d}{dt}(I(\Omega + A\dot{s})) &= \frac{1}{2}[\frac{\partial I}{\partial s}]^* : (\Omega, \Omega) \\ &\quad + \frac{1}{2}[\frac{\partial G}{\partial s}]^*(\dot{s}, \dot{s}) - \frac{\partial V}{\partial s} + u_s \end{aligned} \tag{28}$$

In these equations, $[\frac{\partial I}{\partial s}]$ and $[\frac{\partial G}{\partial s}]$ are three-tensors obtained by taking the Frechet derivatives of the matrix I,G with respect to vector s. $[\frac{\partial I}{\partial s}]^*$ and $[\frac{\partial G}{\partial s}]^*$ are the cyclic transpose of these three tensors c.f. [20] and [1].

The relationship between the control forces u_i and (u_c, u_g, u_s), comes from a well known fact for controlled Lagrangian system. If the coordinate transformation $r = r(q)$ is a local diffeomorphism, then

$$u_r = \frac{\partial r}{\partial q}u_q \tag{29}$$

Thus

$$u_c = \frac{\sum_{i=1}^N m_i u_i}{M} \tag{30}$$

We define

$$u_{fi} = \sqrt{\mu_i}(u_{(i+1)} - \frac{\sum_{k=1}^i m_k u_k}{\sum_{k=1}^i m_k}) \tag{31}$$

for $i = 1,2,...,(N-1)$. Then u_g and u_s can be solved from

$$u_{fi} = g(u_g \times \rho_i + \sum_{j=1}^{3N-6} \frac{\partial \rho_i}{\partial s^j} u_{sj}) \tag{32}$$

because ρ_i are not all collinear.

IV. Feedback control using shape measurements

Suppose the potential function V is invariant under translation and rotation i.e. V is only a function of the shape variables. Then in the system equations (26), (27) and (28), the equation for q_c is decoupled from the other two. Then we can define a function on the tangent bundle of the pre-shape space as

$$V_L = \frac{1}{2}\| s - s^0 \|^2 + \frac{1}{2}(\Omega + A\dot{s})^T I(\Omega + A\dot{s}) + \frac{1}{2}\dot{s}^T G\dot{s} \tag{33}$$

where s^0 specifies a desired shape. The derivative of this function along the reduced dynamics (27) and (28) is

$$\begin{aligned} \dot{V}_L &= < (s - s^0), \dot{s} > + < \Omega, u_g > + < \dot{s}, u_s - \frac{\partial V}{\partial s} > \\ &= < \dot{s}, u_s - \frac{\partial V}{\partial s} + (s - s^0) > + < \Omega, u_g > \end{aligned} \tag{34}$$

where we use $<,>$ to denote the inner product. Hence by letting the control law be

$$\begin{aligned} u_g &= -k_1\Omega \\ u_s &= \frac{\partial V}{\partial s} - (s - s^0) - \dot{s} \end{aligned} \tag{35}$$

where $k_1 > 0$, we have

$$\dot{V}_L = -\|\dot{s}\|^2 - k_1\|\Omega\|^2 \leq 0 \tag{36}$$

We know that on the tangent bundle TF of Jacobi preshape space F, V_L is radially unbounded. So we can apply LaSalle's invariance principle to argue that the controlled dynamics converge to the maximal invariant set C_2 within the set M_2 where $\dot{V}_L = 0$. Hence

$$M_2 = \{(g, s, \Omega, \dot{s}) \in TF | \Omega = 0, \dot{s} = 0\} \tag{37}$$

and

$$C_2 = \{(g, s, \Omega, \dot{s}) \in M_2 | \dot{\Omega} = 0, \ddot{s} = 0\} \tag{38}$$

In the system equations (27) and (28), letting $\Omega = 0$ and $\dot{s} = 0$, we have

$$\begin{aligned} I\dot{\Omega} &= u_g = 0 \\ G\ddot{s} + I\dot{\Omega} &= u_s - \frac{\partial V}{\partial s} = 0 \end{aligned} \tag{39}$$

Thus on the set C_2, in order for $\dot{\Omega} = 0$ and $\ddot{s} = 0$, we must have $s - s^0 = 0$. Thus we have proved the following theorem

Theorem 4.1: Suppose the potential V is rigid motion invariant. By using the feedback control law (35), the Jacobi shape s^0 is locally asymptotically stablized.

However, by letting $u_g = -k_1\Omega$ we already made the assumption that Ω can be measured. In fact, Ω need not to be measured accurately. All we need is an estimate of the direction of Ω which will ensure $\|\Omega\|$ be decreasing. This estimate can not be obtained by only measuring s and $\dot{s}$. Some extra sensors need to be employed which will observe the relative movements of fixed landmarks with respect to the formation.

V. Group MDLe

Motion description language (MDL) was first developed as a setting for robot programming in [4],[3],[5]. In the enhanced form MDLe, it provided a formal basis for robot programming using behaviors and at the same time permitted incorporation of kinematic and dynamic models of robots in the form of differential equations. In the work of Manikonda, Krishnaprasad and Hendler ([14],[13]), the idea of introducing sensor-triggered interrupts as elements of an enhanced MDL was introduced. In the paper [15], a comprehensive overview of these developments on MDLe is given. A recent implementation of MDLe is described in [6].

In MDLe, the physical system is modeled as a so-called *kinetic state machine*. As defined in [15], a kinetic state machine is governed by a differential equation of the form:

$$\dot{x} = f(x) + G(x)U; \quad y = h(x) \in \mathscr{R}^p \tag{40}$$

where $x(\cdot) : \mathscr{R}^+ \to \mathscr{R}^n$, $U(\cdot) : \mathscr{R}^+ \times \mathscr{R}^n \to \mathscr{R}^m$ is a time dependent control law and G is a matrix whose columns g_i are vector fields in $\mathscr{R}^n$. The smallest building block of MDLe is called an **atom**. An **atom** is a triple of the form $\sigma = (U, \xi, T)$, where U is the control defined before, $\xi : \mathscr{R}^p \to \{0, 1\}$ is a boolean interrupt function defined on the space of outputs from p sensors, and $T \in \mathscr{R}^+$ denotes the time (measured from the instance an atom is activated) at which the atom will "time out". To evaluate the atom σ means to apply the input U to the kinetic state machine until the interrupt function ξ is triggered or until T units of time elapse, whichever occurs first. We note that T is allowed to be ∞. The input u could be an open loop command or could be given by a feedback law of the type $u = u(t, x)$. In our implementation, atoms can be assembled using **quarks**. A quark is a piece of code which is reusable by different atoms. For instance,

$$(Atom\ (bumper)\ (go\ 30\ 30))$$

defines an atom to drive a robot forward at speed 30 until it bumps into an obstacle. This will trigger the bumper sensors to interrupt the atom. Here, "bumper" and "go" are quarks which can be shared by other atoms.

To implement the formation control law, we define a new concept called the *group kinetic state machine* by modifying the form of equation (40) as

$$\begin{aligned} \dot{x}_i &= f_i(x_i) + G_{fi}(x_i)U_{fi} + G_{gi}(x_1, x_2, ..., x_N)U_g; \\ y_i &= h(x_i) \in \mathscr{R}^p \end{aligned} \tag{41}$$

where the index i denotes the i^{th} robot. U_{fi} represents the control laws that will achieve the formation stablization or bring about changes in shape and size of the formation. U_g represents the group level control which will treat the whole group as one robot. As is obvious, we separated controls achieving formations from controls that treat the formation as a unity. With respect to this separation, we define a **group atom** to be the triple (U_g, ξ_g, T) where ξ_g is defined as the group level interrupt which depends on the sensor information of all the robots. A **shadow atom** is a modification of a group atom. It is defined as (U_i, ξ_i, T) where

$$U_i = U_{fi} + \eta_i; \ G_{fi}(x_i)\eta_i = G_{gi}(x_1, x_2, ..., x_N)U_g \tag{42}$$

and ξ_i is a function of (y_i, ξ_g). This will result in

$$\dot{x}_i = f_i(x_i) + G_{fi}(x_i)U_i; \quad y_i = h(x_i) \in \mathscr{R}^p \tag{43}$$

which agree with the kinetic state machine representation of a single robot. The mapping from group atoms to

shadow atoms will allow formation keeping controllers to take effect. Mappings can be written a priori by designing proper quarks. Thus all shadow atoms can share a small set of mapping quarks. For instance,

$$(GroupAtom\ (Bumper)\ (go\ 30\ 30))$$

is a group atom which will drive a group of robots forward until any of them bumps into an obstacle. In order to keep all the robots in formation, assume one of the robots is lagging behind its nominal position, the shadow atom on this robot might be

$$(Atom\ (Bumper)\ (go\ 31\ 31))$$

where the number "31" is obtained by a mapping quark where formation control is implemented.

A set of atoms can be composed into a string with its own interrupt function and timer. Such strings are called **behaviors**. Behaviors themselves can be used to form higher-level structures (partial plans) which in turn can be nested into **plans**. MDLe allows for arbitrarily many levels of nested atoms, behaviors and plans. The behaviors and plans consisting purely of group atoms are called **group behaviors** and **group plans**. Group plans are shared by all the robots in a formation.

To illustrate the usage of MDLe for formation control, we show an MDLe script that will command a group of two mobile robots to wander around without hitting obstacles while keeping their initial relative positions.

```
main = (ExecPlan 1 (nop)
robotOn
(GroupConnect 2 12346 12356 sulu:12345 sulu:12355)
(GroupPlan -1 (or (not (areclientsconnected 1)) (bumper))
(GroupAtom frontObstacle (go 30 30))
(GroupAtom (GroupClear 20 15 10) (go -50 50)))
robotOff);
```

Here, "robotOn" and "robotOff" are quarks that will turn on and off the robot. "GroupConnect" is a quark which will terminate after it connects to every other robot in the group. Its arguments include the ports this robot should listen on for connections and the addresses of the robots it will connect to. The "2" is the number of robots in the group and "sulu" is the name of the other robot. Note that in this example, we are using TCP/IP sockets for inter-robot communications.

The group plan specifies a plan level interrupt that will stop the entire plan from executing if either the bumper is hit, or if one of the robots that is supposed to be in the group gets disconnected. The parameter "-1" indicates that this plan will run forever unless interrupted. It will run each of the two group atoms, one after the other . The first atom moves straight ahead until the front is not clear, i.e. obstacles are detected. The next one turns the robot until the front is clear, i.e. obstacles are avoided.

Experiments are done using a pair of Hilare-type mobile robots to show that the mapping from group atoms to shadow atoms is necessary. In the first experiments, the two robots will run the sample script without mapping the group atoms into shadow atoms. The result is shown in Figure (1). In this figure, the thick lines indicate the

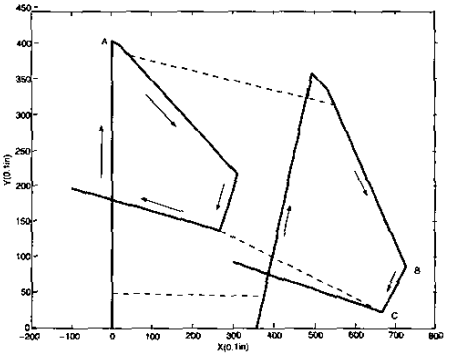

Fig. 1. Executing the Group Plan without Formation Control

ground trajectories of the two robots. The dashed lines connected the two robots at specific time to show their formation. Positions A, B, C indicates where one of the robot detects obstacles. We can see that the robots failed to keep in the starting formation after a short while. We find that the main reason for this is that by using TCP/IP sockets there is a delay of on average about 200ms. In the second experiment, the mappings are added. A simple PI controller is implemented within the mapping process.

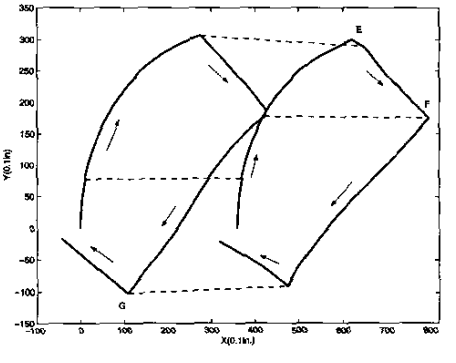

Fig. 2. Executing the Closed Group Plan with PI Controllers

As one can see in Figure (2), the controllers have caused the robots to no longer track straight lines. The formation is successfully kept. However, the performance is limited. Also at position E,F and G one can observe errors caused by communication delays.

VI. SUMMARY

In this paper we have outlined an approach to the problem of formation shape control using the theory of Jacobi coordinates and associated block-diagonalization of the Lagrangian dynamics of a system of robots modeled as point masses, A feedback law for locally stabilizing a shape of interest is given. The approach also permits overall advection of the formation in the Euclidean group. Details of this as well as an intrinsic geometric treatment of our ideas will appear in a work in preparation. We have also investigated the problem of devising formation controllers for mobile robots in software. Proceeding from a motion description language framework, we have shown how certain specific constructs in the language MDLe that support group behavior together with certain communication primitives that support effective sharing of sensor data enable stable coordination of a set of robots. These experimental results are preliminary and work is under way to devise light-weight protocols for robot communication that reduce the effects of latency in the feedback loop.

VII. REFERENCES

[1] S. S. Antman. *Nonlinear problems of elasticity*. Springer-Verlag, New York, 1995.

[2] V. Aquilanti and S. Cavalli. Coordinates for molecular dynamics: Orthogonal local system. *Journal of Chemical Physics*, 85(3):1355–1361, 1986.

[3] R. W. Brockett. On the computer control of movement. In *Proceedings of the 1988 IEEE Conference on Robotics and Automation*, pages 534–540, April 1988.

[4] R. W. Brockett. Formal languages for motion description and map making. In *Robotics*, pages 181–93. American Mathematical Society, 1990.

[5] R. W. Brockett. Hybrid systems in classical mechanics. In *Proc. of the 1996 IFAC Congress*, volume J, pages 473–477, 1996.

[6] D. Hristu, P. S. Krishnaprasad, S. Andersson, F. Zhang, P. Sodre, and L. D'Anna. The mdle engine: a software tool for hybrid motion control. *Technical Report, ISR, UMCP*, 2000.

[7] C. G. J. Jacobi. *Vorlesungen über Dynamik*. Reimer, Berlin, 1866.

[8] D. G. Kendall. Shape manifolds, procrustean metrics, and complex projective spaces. *Bulletin of London Math. Society*, 16:81–121, 1984.

[9] D. G. Kendall, D. Barden, T. Karne, and H. Le. *Shape and shape theory*. Chichester, New York, 1999.

[10] D. R. Lewis. Lagrangian block diagonalization. *Journal of Dynamical Systems and Differential Equations*, 4:1–42, 1992.

[11] R. Littlejohn and M. Reinsch. Internal or shape coordinates in the n-body problem. *Physical Review A*, 52(3):2035–2051, 1995.

[12] R. Littlejohn and M. Reinsch. Gauge fields in the separation of rotation and internal motions in the n-body problem. *Reviews of Modern Physics*, 69(1):213–275, 1997.

[13] V. Manikonda, J. Hendler, and P. S. Krishnaprasad. Formalizing behavior-based planning for nonholonomic robots. In *Proceedings of 1995 International Joint Conference on Artificial Intelligence*, pages 142–149, 1995.

[14] V. Manikonda, P. S. Krishnaprasad, and J. Hendler. A motion description language and a hybrid architecture for motion planning with nonholonomic robots. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, pages 2021–2028, 1995.

[15] V. Manikonda, P. S. Krishnaprasad, and J. Hendler. Languages, behaviors, hybrid architectures and motion control. In J.C. Willems J. Baillieul, editor, *Mathematical Control Theory*, pages 199–226. Springer, 1998.

[16] J. E. Marsden, J. C. Simo, D. R. Lewis, and T.A. Posbergh. A block diagonalization theorem in the energy-momentum method. *Contemporary Mathematics*, 97:297–313, 1989.

[17] J. C. Simo, D. R. Lewis, and J. E. Marsden. Stability of relative equilibria I: The reduced energy momentum method. *Archive for Rational Mechanics and Analysis*, 115:15–59, 1991.

[18] J. C. Simo, D. R. Lewis, and J. E. Marsden. Stability of relative equilibria II: Three dimensional elasticity. *Archive for Rational Mechanics and Analysis*, 115:61–100, 1991.

[19] C. G. Small. *The Statistical Theory of Shape*. Springer, New York, 1996.

[20] L.-S. Wang. *Geometry,Dynamics and Control of Coupled Systems*. PhD thesis, ISR Technical Report PhD90-5, University of Maryland, 1990.

[21] L.-S. Wang and P. S. Krishnaprasad. Gyroscopic control and stabilization. *Journal of Nonlinear Science*, 2:367–415, 1992.

[22] F. Zhang and P. S. Krishnaprasad. Geometric methods in formation control. *In preparation*, 2002.