Limitations Imposed by Single DOF Actuators on Discrete Actuator Arrays

Mark Bedillion, William Messner, Howie Choset
Department of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
capn@andrew.cmu.edu, bmessner@andrew.cmu.edu, choset@andrew.cmu.edu

Abstract

The Modular Distributed Manipulator System (MDMS) is an array of actuators that is capable of manipulating objects in the plane [1]. Each actuator in the array consists of two closely spaced, orthogonally mounted motorized roller wheels whose combined action approximates a programmable vector force. This paper first derives the equations of motion for transported objects by explicitly considering the traction force from each wheel of the actuator. This contrasts with prior work, which assumed an ideal single point of contact for each actuator [2]. A translational velocity field is developed based on these dynamics and is compared with the discrete elliptic field derived in [2]. Errors in translational dynamics under an elliptic velocity field are discussed. This paper then presents simulation results that demonstrate some of the difficulties encountered when using non-ideal actuators.

1 Introduction

The Modular Distributed Manipulator System (MDMS) was first introduced in 1995 [1] as a system for positioning and orienting objects in the plane. The MDMS consists of many small actuators in a fixed array that cooperate to manipulate objects that lie above them. Each unit of the array, or cell, consists of an actuator capable of approximating a vector force, a sensor capable of detecting an object, and a microcontroller that communicates with neighboring cells and controls the actuator. The system is capable of operating at a completely distributed level such that sensing, planning, and actuation are accomplished without a centralized controller.

The MDMS has several advantages over conveyor belts and conventional robotic manipulators. Actuator arrays can orient objects as well as transport them over large distances and have potential applications in package sorting operations where multi-object manipulation is needed and in flexible manufacturing settings where manipulating many different types of objects is important. If the cells are designed to easily snap together, the array can be easily reconfigured to meet demand in a flexible manufacturing environment. The array’s redundancy also allows for good fault tolerance because objects can be redirected around or passed over broken cells. Furthermore, the system design is scalable so that the primary difference between manipulating parcels and large, heavy objects is the choice of actuator.

Previous work on the MDMS has assumed ideal actuators capable of generating a vector force from a single point [2], but experimentally observed behavior not predicted by this model has motivated the authors to employ a more realistic model with two points of contact at each cell, one for each motor. Each cell consists of two orthogonally mounted motors fitted with roller wheels as shown in Figure 2. This design was chosen over finger-
like actuators and inverted trackballs for simplicity, and over steerable wheels to avoid nonholonomic constraints and delay in directional changes. The separation between the two points of contact at each cell affects object dynamics and equilibrium orientations under open loop control. The distance between the wheels is not negligible relative to the cell spacing so that there are many instances in which an object lies on only a single roller wheel at a given cell.

This paper is organized as follows. Section 2 describes previous research into actuator arrays and prior work on the MDMS. Section 3 derives the equations of motion for a manipulated object with two points of contact at each cell. Section 4 discusses velocity field design under open loop control. Section 5 presents simulation results that compare the behavior of ideal actuators with that of single DOF motors. Finally, Section 6 presents conclusions and discusses open issues.

2 Prior Work

Prior work on actuator arrays has largely focused on MEMS scale systems whose action can be described by continuous force fields. This research area has applied the sensorless manipulation approach of Mason and Erdman [3,4] and Goldberg’s use of a sequence of squeezes to orient parts [5] to actuator arrays. Bohringer and Donald [6] created a system with cilia-like actuators and analyzed object motion using continuous field approximations. Their work with Kavraki [7] developed fields that could position and orient any non-symmetric part. The modeling of the MDMS has differed from the models of [6] and [7] because spacing between actuators is not considered negligible relative to object size, and thus continuous force field approximations are inappropriate.

Luntz et. al. [2] derived the equations of motion for the MDMS assuming a single point of contact at each cell. These dynamics were exploited to develop open-loop and closed-loop control laws for positioning and orienting objects on the array [8]. It was shown in [8] that open-loop manipulation with elliptic fields aligned only a subset of mirror symmetric rectangular objects to the field axes due to array discreteness. Murphey and Burdick examined contact modes and open-loop instability under elliptic fields for actuator arrays [9].

3 Derivation of Equations of Motion

This section reviews the derivation followed by Luntz et. al. in [2] but makes changes to account for single degree of freedom actuators. The object rests on \( n \) motors that act in the \( x \) direction and \( n \) motors that act in the \( y \) direction. Figure 3 shows several wheels supporting an object. A subscript \( x \) or \( y \) denotes the orientation of a given actuator. For instance, \( x_{xy} \) denotes the \( x \) coordinate of the position of the \( i \)th \( y \)-acting actuator. Vectors are defined with elements referencing \( x \)-actuators given first for convenience, as in

\[
X = \begin{bmatrix}
\vec{x}_1 \\
\vdots \\
\vec{x}_n
\end{bmatrix} = \begin{bmatrix}
x_{x1} & \cdots & x_{xn} \\
y_{x1} & \cdots & y_{xn}
\end{bmatrix},
\]

It can be shown through the same manipulations presented in [2], using a linear spring model for each suspension, that the normal forces are given by the same expression

\[
\vec{N} = W\mathbf{B}^T(\mathbf{B}\mathbf{B}^T)^{-1}\begin{bmatrix}1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \vec{X}_{cm},
\]

where

\[
\mathbf{B} = \begin{bmatrix}
x_{x1} & \cdots & x_{xn} \\
y_{x1} & \cdots & y_{xn}
\end{bmatrix},
\]

\( W \) is the weight of the object, and \( \vec{X}_{cm} \) is a vector of the object’s center of mass coordinates. Assuming viscous friction between the object and the MDMS, the force generated by each wheel is the product of the coefficient of friction, \( \mu \), the normal force, and the difference between the wheel’s velocity and the velocity of the object at the point of contact, i.e.,

\[
f_{xi} = \mu N_{x_i} (v_{xi} - \dot{x}_{cm} + \dot{\theta}(y_{xi} - y_{cm})),
\]

\[
f_{yi} = \mu N_{y_i} (v_{yi} - \dot{y}_{cm} - \dot{\theta}(x_{yi} - x_{cm})),
\]

where \( N_{x_i} \) and \( v_{xi} \) are the normal force and velocity at \( k \)th \( x \)-acting wheel, \( \dot{x}_{cm} \) is the \( x \) component of velocity of the object’s center of mass, and \( \dot{\theta} \) is the object’s angular velocity. The viscous friction model was chosen because the wheel speeds are generally much greater than the

![Figure 3 Object Supported by Actuators](image-url)
object speed and it is supported by qualitative experimental observations. A discussion of other contact modes and conditions under which rolling contact occurs appears in [10].

The total force on the object is found by summing the forces due to each motor. By rewriting the force at each motor as $\vec{f}_j = \begin{bmatrix} f_{xj} \\ 0 \end{bmatrix}$ for x-acting motors and $\vec{f}_j = \begin{bmatrix} 0 \\ f_{yj} \end{bmatrix}$ for y-acting motors and by defining

$$\mathbf{V} = \begin{bmatrix} v_{x1} & \cdots & v_{xn} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & v_{y1} & \cdots & v_{yn} \end{bmatrix},$$

(5)

it follows from the analysis in [2] that

$$\mathbf{f} = \mu \mathbf{WV}^T (\mathbf{BB})^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} - \mu \mathbf{WX}^T \mathbf{X}_{cm} ^T,$$

(6)

or

$$\ddot{\mathbf{X}}_{cm} + \mu \mathbf{WX}^T \mathbf{X}_{cm} - \mathbf{K}_s \mathbf{X}_{cm} = \vec{f}_o,$$

(7)

where $\mathbf{K}_s$ is a 2x2 constant matrix and $\vec{f}_o$ is a constant 2x1 vector.

The torque on the object is similarly identical to that given in [2], i.e.,

$$\tau = \mu \mathbf{WR}^T (\mathbf{BB})^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \mu \mathbf{WR}^T (\mathbf{BB})^{-1} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \times \mathbf{X}_{cm}$$

$$+ \mathbf{X}_{cm} \times (\vec{f}_o + \mathbf{K}_s \mathbf{X}_{cm}) - \mu \theta (\mathbf{X}_{cm}^T \mathbf{WX}^T \mathbf{X}_{cm}) + \mathbf{X}_{cm} \times (\vec{f}_o + \mathbf{K}_s \mathbf{X}_{cm}) - \mathbf{X}_{cm} \times (\vec{f}_o + \mathbf{K}_s \mathbf{X}_{cm})$$

$$- \mu \theta (\mathbf{X}_{cm}^T \mathbf{WX}^T \mathbf{X}_{cm})$$

(8)

or

$$\ddot{\mathbf{X}}_{cm} + \mu \theta (\mathbf{X}_{cm}^T \mathbf{WX}^T \mathbf{X}_{cm}) - \mathbf{K}_{sx} \mathbf{X}_{cm} = \tau_o$$

(9)

where $\ddot{\mathbf{X}}_{cm}$ is a row vector with components $R_i = \ddot{X}_i \times \dddot{V}_i$, $\mathbf{X}_{cm}$ is a row vector with components $X_i = \dddot{X}_i \mathbf{X}_{cm}$, $\dot{X}_i$ is a 1x2 constant vector, and $\tau_o$ is a constant torque.

The equations presented here are identical to those in [2] other than a redefinition of quantities to account for the two motors at each actuator. It is these redefinitions, however, that create problems during velocity field design. In particular, the form of $\mathbf{V}$ given in (5) requires a complex field for open-loop object translation.

### 4 Velocity Field Design

#### 4.1 Translational Velocity Field

Equation 7 shows that the object has simple mass-spring-damper translational dynamics when resting on a given set of supports. The translational open-loop control problem involves finding a set of wheel velocities that impart dynamics specified by the elements of the stiffness matrix $\mathbf{K}_s$ and the components of the offset force $\vec{f}_o$. Without loss of generality, the desired object position is defined to be the coordinate origin. By definition,

$$\begin{bmatrix} f_{ox} \\ f_{oy} \\ k_{sx} \\ k_{sy} \\ k_{gx} \\ k_{gy} \end{bmatrix} = \mu \mathbf{WV}^T (\mathbf{BB})^{-1}.$$

(10)

Due to the symmetry of $\mathbf{BB}^T$, the transpose of Equation 10 is

$$\begin{bmatrix} f_{ox} \\ f_{oy} \\ k_{sx} \\ k_{sy} \\ k_{gx} \\ k_{gy} \end{bmatrix} = \mu \mathbf{W} (\mathbf{BB})^{-1} \begin{bmatrix} v_{x1} & 0 \\ \vdots & \vdots \\ 0 & v_{yn} \end{bmatrix},$$

(11)

which may be stacked and rewritten as

$$\begin{bmatrix} f_{ox} \\ k_{sx} \\ k_{gx} \\ f_{oy} \\ k_{sy} \\ k_{gy} \end{bmatrix} = \mu \mathbf{W} (\mathbf{BB})^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}.$$

(12)

A solution in the separated case is given by
where

\[
0 = 1 / \mu W \begin{bmatrix} B_1^T (B_1 B_1^T)^{-1} B B^2 & 0 \\ 0 & B_2^T (B_2 B_2^T)^{-1} B B^2 \end{bmatrix}
\]

and

\[
\begin{bmatrix} f_{o_x} \\ k_{s_x} \\ k_{s_y} \end{bmatrix} = \begin{bmatrix} f_{o_x} \\ k_{s_x} \\ k_{s_y} \end{bmatrix}
\]

This problem was solved in [2] for single-point-of-contact actuators and resulted in a velocity field that was invariant with respect to the set of supporting cells. Assuming that \( k_{s_x} \) and \( k_{s_y} \) are zero to decouple the x and y equations of motion, the translational velocity field for ideal actuators is the discrete elliptic field

\[
\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} 1/2 \mu W B_1^T (B_1 B_1^T)^{-1} B \\ 1/2 \mu W B_2^T (B_2 B_2^T)^{-1} B \end{bmatrix}
\]

so that the velocities have the form

\[
v_{x_0} = f_{o_x} + k_{s_x} x_{x_0},
\]

\[
v_{y_0} = f_{o_y} + k_{s_y} y_{y_0}
\]

The field given in (15) is attractive because the only information each cell needs is its location relative to the goal, and the wheel velocities do not change when the object changes supports. However, the field in (13) for separated actuators requires each cell to know the supporting set of cells and to change the velocity of its wheels when the object changes supports. The communication and computation involved in disseminating information limits the speed with which the object can accurately be manipulated.

Note that as \( B_1 B_1^T \rightarrow B_2 B_2^T \), or as the two points of contact at each cell approach each other, Equation 13 approaches Equation 15. Thus, for cases in which the actuator separation is small relative to the cell spacing, the error in using a standard elliptic field is small. In the MDMS prototype wheel separation relative to cell spacing is not negligible; cells are approximately 13 cm apart and the two wheels at each cell are approximately 4 cm apart.

### 4.2 Error Under Elliptic Field

Actual object dynamics under the elliptic velocity field are slightly different from those specified in the velocity field derivation. The error between the exact velocity field and the approximate field is given by

\[
\begin{bmatrix} f_{o_x} \\ k_{s_x} \\ k_{s_y} \end{bmatrix} = \begin{bmatrix} 1 / \mu W B_1^T (B_1 B_1^T)^{-1} (B_2 B_2^T - B_1) \\ 0 \end{bmatrix}
\]

\[
\begin{bmatrix} f_{o_x} \\ k_{s_x} \\ k_{s_y} \end{bmatrix}
\]

When the velocity error expression is substituted into Equation 10, the errors in the dynamic parameters are found to be

\[
\begin{bmatrix} \Delta f_{o_x} \\ \Delta k_{s_x} \\ \Delta k_{s_y} \end{bmatrix} = \mu W \begin{bmatrix} f_{o_x} \\ k_{s_x} \\ k_{s_y} \end{bmatrix} \begin{bmatrix} B_2 B_2^T - B_1 B_1^T \end{bmatrix}^{-1}
\]

\[
\begin{bmatrix} \Delta f_{o_x} \\ \Delta k_{s_x} \\ \Delta k_{s_y} \end{bmatrix} = \mu W \begin{bmatrix} f_{o_x} \\ k_{s_x} \\ k_{s_y} \end{bmatrix} \begin{bmatrix} B_2 B_2^T - B_1 B_1^T \end{bmatrix}^{-1}
\]

It should be noted from (18) that the actual \( k_{s_x} \) and \( k_{s_y} \) may be nonzero even when the design values are set to zero. The error in the x-direction dynamic properties involves the term

\[
B_2 B_2^T - B_1 B_1^T
\]

\[
\begin{bmatrix} n-m & \sum_{i=1}^{n} x_{i} & \sum_{i=1}^{n} y_{i} \\ \sum_{i=1}^{m} x_{i} & \sum_{i=1}^{m} x_{i} & \sum_{i=1}^{m} y_{i} \\ \sum_{i=1}^{m} y_{i} & \sum_{i=1}^{m} y_{i} & \sum_{i=1}^{m} y_{i} \\ \sum_{i=m+1}^{n} x_{i} & \sum_{i=m+1}^{n} x_{i} & \sum_{i=m+1}^{n} y_{i} \\ \sum_{i=m+1}^{n} y_{i} & \sum_{i=m+1}^{n} y_{i} & \sum_{i=m+1}^{n} y_{i} \end{bmatrix}
\]

This matrix is equal to the zero matrix when the object rests on an equal number of x-acting and y-acting wheels.
and when the x-acting and y-acting wheels are arranged with mirror symmetry about the origin, as in Figure 4. MDMS actuators, however, are arranged as shown in Figure 5 so that no destination on the array features mirror symmetry for a symmetric object. This lack of symmetry causes objects to experience a small torque even when the object’s axes are aligned with those of the elliptic field.

5 Simulations

We contrast in simulation the motion of a box resting on one-point-of-contact and two-point-of-contact actuators. The simulations are run in a Simulink model that tracks cell transitions and object dynamics. Figure 6 shows the simulation results for a rectangular object that is 5.1d long and 4.1d wide, where d is the distance between adjacent cells. The analysis of [2] predicts that this object will not align its axes with the elliptic field axes. When both points of contact are considered, the object does align its axes with the field.

Figure 7 shows the torque on the object as a function of its orientation when it rests at the coordinate origin. Zero crossings indicate equilibrium points. If the slope is negative, then the equilibrium point is stable, because rotation in either direction from the point generates a
restoring torque. The existence of two points of contact at each cell moves the locations of stable equilibria and generates additional stable points, thus making open loop manipulation with elliptic fields difficult. The existence of many possible equilibrium orientations makes predicting final object orientation difficult. This type of effect may be seen in even very large objects relative to cell spacing, as shown in Figure 8. This object covers approximately 300 full cells and yet does not align itself with the array when using two-point-of-contact actuators. Note that in this plot and in Figure 7, torques at ± 90° are zero for single point of contact actuators and small but nonzero for two point of contact actuators.

6 Conclusions

This paper derived dynamics for objects on discrete actuator arrays with two points of contact considered at each actuator. Under a viscous friction contact assumption, it was shown that each cell must know the entire supporting set of cells to accurately translate an object. If a standard elliptic field is used, the velocity error can cause errors in object dynamics and cause the object axes to fail to align with the field axes.

Simulation studies showed that stable object orientations under elliptic field manipulation change when two points of contact are explicitly considered at each cell. This effect was evident even in objects that were very large with respect to the cell spacing. Depending on design, this effect may be significant even in MEMS applications. Two points of contact may create many possible stable equilibria so that initial conditions play a large role in the orientation that an object ultimately assumes.

While wheel separation causes problems in open loop manipulation, its effect may be eliminated using closed-loop control. If object position and orientation are available through a vision system or local sensing, feedback control can be applied through the same methodology as in [8]. This is a topic for future research. Another possible approach is to use no-slip rolling contact between the object and the wheels to precisely position and orient an object while elliptic field-based open loop manipulation is used to transport objects over large distances. One way to avoid the problem altogether is to use near-ideal actuators such as the spherical motor [11] that allow single point of contact manipulation. However, these actuators are typically expensive when compared to the cost of simple motors.

Acknowledgements

The authors gratefully acknowledge NSF support IIS-9872255 for this work. The authors would also like to acknowledge Dr. Jonathan Luntz for creating Figures 1 and 2. They would also like to thank their reviewers for many thoughtful suggestions.

References