Optimal Commutation Laws by Real-Time Optimization for Multiple Motor Driven Systems

Shalom D. Ruben and Tsu-Chin Tsao

Abstract—This paper addresses commutation laws of brushless motor coils for multiple-degree-of-freedom motion. By exploiting the redundancy of motor coils and motor force generation that exist in over actuated motion platforms, we propose a commutation law, which generates the required forces while minimizing the effect of motor coil heat generation on thermal distortion on the motion platform work space. To realize the commutation laws in real-time, a real-time iterative solution, using an interior-point method, is described and shown to solve this problem in under 35 microseconds, within the update rate of our system running at over 9KHz.

I. INTRODUCTION

Precision motion control is becoming increasingly more important in the areas of nanotechnology. Lithography, for one, is requiring greater precision as transistor density continues to increase. Permanent magnet synchronous linear motors (PMSLM) are among the most prevalent in precision machines due to their long-range travel, high force density, and high precision and accuracy. Lack of mechanical coupling needed, versus rotary motors, and in conjunction with an air bearing eliminate nonlinearities such as backlash and friction [1].

A major source of dimensional inaccuracy for such ultra high-precision machines is thermally induced mechanical element deformation. Some techniques to reducing these errors are by decreasing the temperature variations and decreasing the sensitivity of the machine to temperature variations. Design symmetry and using material properties that have low thermal expansion coefficients will decrease the sensitivity to variations while cooling methods will reduce temperature variations themselves [2]. These approaches are costly and painstaking, as the holy grail of the field of precision engineering. Deviating drastically from these conventional approaches, this paper considers the mechanical motion control and the minimization of thermally induced deformation together through commutation laws of multiple linear motor driven system.

The Mechanical and Aerospace Engineering Department, University of California at Los Angeles, Los Angeles, CA 90095, USA, shalom@ucla.edu, ttsao@ucla.edu

A new optimal commutation algorithm is proposed where minimum power is desired subject to a power symmetric solution. By requiring all the motors to generate a symmetric power profile we reduce the temperature variations and thus minimize deformation across the system. The heat symmetric solution is not closed-form and a convex optimization problem needs to be solved at each servo cycle. We develop our own solver that will iterate to a solution within a phase cycle of 110μs and implement on a real-time control system.

II. COMMUTATION WITH COIL REDUNDANCY

In this section, focus is on commutation for a single linear motor with a Halbach magnet array, although the work that follows is applicable to standard linear motors. A PMSLM consists of two main parts, the stator and mover. The stator is the non-moving part containing n coils, and the moving part containing the permanent magnets is the mover. The force produced by the linear motor is proportional to the current supplied to the n coils and is defined as

\[ f = Bi \] (1)

The ideal Halbach motor law can be generalized for an n-phase motor (for \( n \geq 3 \)) by (2) where \( x \) is the linear position relative of the magnet array to the stator, \( \omega x \) is the electric angle, and \( \phi \) is a bias angle [3].

With the motor law defined, as the mapping from current to force (\( B: i \rightarrow f \)), the inverse motor law (or commutation) maps a desired force to current (\( T_{inv}: f \rightarrow i \)). As seen by the control loop in Fig. 1, the controller \( C \) generates a desired force and must be transformed, by \( T_{inv} \), into a desired current to be sent by the motor amplifier. The system of equations (2) (\( B \in \mathbb{R}^{2 \times n} \)) has two equations \( ([f_x f_z]^T) \) and \( n \) unknowns, giving us an
under-determined system with \( n - 2 \) vectors that span the
nullspace of \( B \) (\( i_N \in N(B) \)). Physically this means that
for \( n = 3 \) there is one, non-zero, current vector that will
produce zero force by the motor (\( Bi_N = 0 \)). An infinite
set of solutions can be found by taking a particular
solution, \( i_p \), for a desired force \( f_d \) (\( f_d = Bi_p \)), and
adding a constant current to each phase resulting in
\[
B(i_p + \alpha i_N) = Bi_p + \alpha Bi_N = Bi_p + 0 = f_d
\]
Having an infinite set of solutions gives the flexibility of
choosing a solution that best fits our needs. Specifically,
a solution with the minimum power (\( p = i_1^2 + i_2^2 + \cdots + i_n^2 = \|i\|^2 \)) is desired. This minimum power problem is
defined by (3).

\[
\begin{align*}
\text{minimize} & \quad \|i\|^2 \\
\text{subject to} & \quad f = Bi
\end{align*}
\]

The minimum power problem is a least-norm problem
and the solution is \( i^* = B^\dagger f \). Before solving for the
commutation law, let us assume that our motor is not
ideal and define \( B \) as:
\[
B = \begin{bmatrix} C_x \cdot & C_z \cdot \\ \end{bmatrix} = \begin{bmatrix} c_{x1}(x) & c_{x2}(x) & \cdots & c_{xn}(x) \\ c_{c1}(x) & c_{c2}(x) & \cdots & c_{cn}(x) \end{bmatrix}
\]

Using this notation, \( B^\dagger \) is reduced:
\[
B^\dagger = B^T(BB^T)^{-1}
\]
\[
= \begin{bmatrix} C_x \cdot & C_z \cdot \\ \end{bmatrix} \left( \begin{bmatrix} C_x \cdot & C_z \cdot \\ \end{bmatrix} \begin{bmatrix} C_x \cdot & C_z \cdot \\ \end{bmatrix} \right)^{-1}
\]
\[
= \begin{bmatrix} C_x \cdot & C_z \cdot \\ \end{bmatrix} \left( \begin{bmatrix} u & s \\ s & v \end{bmatrix} \right)^{-1}
\]
\[
= \begin{bmatrix} vC_x - sC_z & uC_z - sC_x \\ \end{bmatrix}
\]

Substituting \( B^\dagger \) back into \( i^* = B^\dagger f \) the generalized
minimum power commutation is found.
\[
i^* = \frac{1}{uv - s^2}\left[f_x(vC_x - sC_z) + f_z(uC_z - sC_x)\right]
\]

(4)

Where \( u = C_x \cdot C_x, v = C_z \cdot C_z, \) and \( s = C_z \cdot C_x = C_x \cdot C_z \).
Assuming the ideal case given by (2), the
constants \( u,v, \) and \( s \) reduce to \( u = v = \frac{na^2}{2} \)
and \( s = 0 \).

The term \( BB^T \) reduces to a constant multiple of the
identity:
\[
BB^T = \begin{bmatrix} u & s \\ s & v \end{bmatrix} = \frac{na^2}{2}I
\]

(5)

while the minimum power commutation reduces to
\[
i^* = \frac{2}{na^2}BB^T f = \frac{2}{na^2}(f_xC_x + f_zC_z)
\]

which is the conventional sinusoidal commutation. This
ideal case simplification, that \( BB^T = \frac{na^2}{2}I \), will be
crucial when deriving the commutation for a multiple
linear motor system.

### III. Commutation With Coil and Actuator Redundancy

A minimum power commutation law for multiple
linear motor system, such as The Sub-Atomic Measuring
Machine (SAMM) [4] having four 6-phase halbach lin-
ear motors and Multi-Alignment and Positioning System
(MAPS), shown in Fig. 2, [5] having four 3-phase
halbach motors, is derived in this section. The local force

![Fig. 2: 4 linear motors produce 8 forces to control 6 degrees of freedom](image)

vector \( f \) for a multiple motor system with \( m \) motors will be
defined as
\[
f = \begin{bmatrix} f_{x1} & f_{z1} & f_{x2} & f_{z2} & \cdots & f_{xm} & f_{zm} \end{bmatrix}^T
\]

\[
= \begin{bmatrix} f_1 & f_2 & \cdots & f_m \end{bmatrix}^T
\]

and the relationship between the local motor forces and
the global forces (\( F = [F_x F_y T_z T_x T_y]^T \)) is
\[
F = Af
\]

(7)

From Fig. 2 the relationship between \( F \) and \( f \), for
MAPS, can be shown to be

\[
A = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & -R & 0 & 0 & 0 & O & 0 & 0 \\ 0 & 0 & 0 & -R & 0 & 0 & 0 & R \end{bmatrix}
\]

(8)
where $R$ is the distance from the center of the platen to the center of each motor. The relationship between the current vector $i$ and the local forces for a multiple linear motor system is

$$B = \begin{bmatrix} B_1(x_1) & 0 & \cdots & 0 \\ B_2(x_2) & \cdots & \cdots & \cdots \\ \vdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & B_m(x_m) \end{bmatrix}$$ (9)

where

$$B_k(x_k) = \begin{bmatrix} C_{\pi}(x_k)^T \\ C_{\zeta}(x_k)^T \end{bmatrix}$$

and $x_k$ is the local position of the stator and mover of each motor. For MAPS, the local positions of each motor are

$$x_1 = x + R \sin(\theta_1); \quad x_2 = y + R \sin(\theta_2) \quad x_3 = -x + R \sin(\theta_2); \quad x_4 = -y + R \sin(\theta_2)$$

The motor law for a 6-DOF multiple linear motor system is defined as

$$F = ABi$$ (10)

Where the current vector $i$ is defined as

$$i = [i_{i1} \ i_{i2} \ \ldots \ i_{in} \ i_{m1} \ i_{m2} \ \ldots \ i_{mn}]^T$$

The minimum power commutation is found by solving

$$\min_{i} \ ||i||^2 \text{ subject to } F = ABi$$ (11)

The solution to this problem is given as $i^* = (AB)^\dagger F$. It is possible to attack this problem in two stages, a kinematic stage where the desired local force vector is found $f = A^\dagger F$ and an electro-magnetic stage, where the minimum power current vector, given the kinematic stage, is found $i = B^\dagger f$ giving us $i = B^\dagger A^\dagger F$. In general, this two-stage method does not give the same solution since $(AB)^\dagger \neq B^\dagger A^\dagger$ [6]. Therefore separating the kinematic stage and the electro-magnetic stage when minimizing the power is not, in general, minimal. Due to the structure of the ideal linear motors, namely that $BB^T = (na^2/2)I$ as in (5), the one stage solution turns out to be equal to the two stage solution $||AB)^\dagger F|| = ||B^\dagger A^\dagger F||$:

$$B^\dagger A^\dagger = B^T(BB^T)^{-1}A^T(AA^T)^{-1}$$

$$= B^T(\frac{na^2}{2}I)^{-1}A^T(AA^T)^{-1}$$

$$= \frac{2}{na^2} B^T A^T(AA^T)^{-1}$$ (13)

The two-stage equals the one-stage solution independent of the kinematic arrangement of linear motors (the structure of $A$).

IV. POWER SYMMETRY COMMUTATION

It is important to minimize thermally induced mechanical distortion to render dimensional accuracy. To satisfy this requirement for geometrically symmetric stage like MAPS, we add the constraint that each motor power ($\|i_k\|^2$) is equal to one another.

$$\min_{i} \ ||i||^2 \text{ subject to } F = ABi \quad ||i_k||^2 = p;$$ (14)

This problem is motivated by the temperature gradient, and subsequent non-symmetric deformation of the platen, produced by the motors generating different power profiles. By each motor generating the same power we force heat symmetry on the platen and therefore symmetric deformation about the center of the platen. This problem is non-convex due to the added quadratic equality constraint [7].

We propose to solve this problem by a 2-stage method. The first stage of this method involves solving a convex problem that minimizes the maximum power that any of the four motors can have $(\min_{i} \max_{k} ||i_k||^2)$. The power for the motor with the greatest power is defined as $P$. The second stage is to project the solution of the min max problem onto the set of solutions that each motor has power equal to $P$. This projection is possible by adding a non-zero current in the nullspace of $B$ (see table I). The projection onto the space of equal power motors is defined by

$$\text{Proj}(i_o, P) = \begin{bmatrix} i_{o1} + \alpha_1 i_{N1} \\ i_{o2} + \alpha_2 i_{N2} \\ \vdots \\ i_{om} + \alpha_m i_{Nm} \end{bmatrix}$$ (15)

By setting the desired power $P$ equal to the power given by new current vector $(i_{o} + \alpha i_{N})^T_k (i_{o} + \alpha i_{N})_k = P$, $\alpha_k$ can be found to be

$$\alpha_k = -\frac{\delta_{i_{Nk}}^T i_{Nk} i_{Nk} + \delta_{i_{Nk}}^T i_{Nk} (i_{o} + \alpha i_{N})_k - P}{\delta_{i_{Nk}}^T i_{Nk}}$$ (16)

Remember that $i_{Nk} \in N(B_k)$, which means that any added current in this direction can increase the
TABLE I: Convex and Algebraic solution to the non-convex problem (14)

| Method          | ||z_1||^2 | ||z_2||^2 | ||z_3||^2 | ||z_4||^2 | ||z_5||^2 |
|-----------------|---------|---------|---------|---------|---------|
| Min Power       | 0.5     | 0.5     | 0.5     | 0.5     | 0.5     |
| Pow Sym (Stage 1)| 0.5     | 0.5     | 0.5     | 0.5     | 0.5     |
| Pow Sym (Stage 2)| 0.5     | 0.5     | 0.5     | 0.5     | 0.5     |

Proof: Let the solution of (14) give \( P^* \) for each motor and the solution for the min-max problem of table I be \( P_{hs} \). By the problem in (14), the total power satisfies \( 4P^* \leq 4P_{hs} \). On the other hand, by the problem in table I, \( P_{hs} \leq P^* \). Therefore, \( P^* = P_{hs} \).

The results of a simulated example, where the desired global forces are \( F_x = 1, T_z = 4R \) and all the rest are zero, are shown in Table II. Two solutions, a minimum power (11) and both stages of the solution to power symmetry (14) are presented. Both solutions produce the same force, while the power output per motor is quite different.

V. REAL-TIME OPTIMIZATION SOLVER

Many convex optimization problems involve the solution of a sequence of problems until an optimal solution has been reached. Unless the update rate is slow, as in the chemical process industry [8], then embedded convex optimization methods are not viable. Control of mechanical systems with update rates in microseconds, such as MAPS with an update rate of 110 \( \mu \)s, have traditionally not been able to use embedded convex optimization techniques. The following section describes the development of a solver using interior-point method, specifically for the power symmetry problem. We wrote special code to take advantage of every exploitable feature that a general solver might not be able to find. First, a reduction of our problem size is possible. Assuming that (5) holds, the min-max problem in table I becomes equivalent to solving the following problem:

\[
\begin{align*}
\text{minimize} & \quad \max_k (||f_k||) \\
\text{subject to} & \quad F = Af
\end{align*}
\]

and the desired current \( i \) is found by solving the least-norm problem \( i = B^Tf^* \). In this section, we will only deal with solving the convex optimization problem of (17), giving \( f^* \), but keep in mind that to get the total solution \( i^* = \text{Proj}(B^Tf^*), P = \max_k(B^Tf^*)_k \) must be calculated. In words, the solution \( (f^*) \) from (17) is found, then the current \( (i) \) that minimizes the power subject to \( f^* = Bi \) is solved. Finally the motor currents are projected so that their powers are equal.

The problem of (17) can be reformulated as an SOCP (18).

\[
\begin{align*}
\text{minimize} & \quad q \\
\text{subject to} & \quad F = Af \\
& \quad ||f_k|| \leq q
\end{align*}
\]

Both equality and inequality constraints need to be dealt with in (18). It is possible to eliminate the equality constraint by defining \( f \) as

\[
f = f_o + Zy
\]

where \( f_o = [f_{o1}, f_{o2}, \cdots, f_{om}]^T \) is a feasible solution (\( Af_o = F \)) and \( Z = [Z_1, Z_2, \cdots, Z_m]^T \) spans the nullspace of \( A \). This might not always be beneficial since a costly QR factorization may be necessary to find \( Z \), if \( Z \) changes every iteration. In the case of multiple linear motor systems, \( A \) is constant and therefore the nullspace is invariant, so \( Z \) can be predefined and stored in memory. Note that depending on the structure of \( A \), such as sparsity, methods that deal directly with equality constraints may exploit that structure and be faster.

By eliminating the equality constraint from (18) we get

\[
\begin{align*}
\text{minimize} & \quad q \\
\text{subject to} & \quad ||(f_o + Zy)||_\infty \leq q
\end{align*}
\]

and reduce the number states in (18), \( \{q, f \in \mathbb{R}^8\} \), from nine to three, \( \{q, y = [y_1, y_2]^T\} \). This leaves an inequality constrained problem. Primal-Dual methods, advanced Interior point methods [7], are used to solve inequality constrained problems and are prevalent in software such as SeDuMi, and SDPT3. A simpler interior-point method, known as the primal log-barrier method, is is used to solve our SOCP problem. Although primal-dual methods are more advanced than the primal barrier method, the details shown here can easily be extended to the primal-dual method. The idea behind the log-barrier method is to remove the inequality constraints to obtain the unconstrained optimization problem

\[
\begin{align*}
\text{minimize} & \quad tq + \sum_k \phi_k(q, y),
\end{align*}
\]
where \( \phi_k(x) = -\log(q^2 - (f_o + Zy)^T(f_o + Zy)k) \), is convex and differentiable everywhere, can be thought of as a weight that increases exponentially the closer you get to having an infeasible solution (a barrier). The barrier method is an iterative method that solves (20) many times for different values of \( t \). Each solution, starting with a small value of \( t \) and updating \( (t = \mu t) \) at each iteration, is known as the central path. As \( t \) increases the affect from the log-barrier function decreases so the solution approaches the solution of the original primal problem (19).

The barrier method has two loops, the outer loop described by the central path and the inner which solves the newton system (20). Solving the newton system, a set of linear equations, is the most time intensive part of the algorithm. Defining our state vector \( u = [y \; q]^T \) we rewrite (20) as

\[
\text{minimize} \quad te^T u + \sum_k \phi_k(u)
\]

and define the newton system as

\[
H \Delta u = -g
\]

where

\[
g = tc + \nabla_u \phi
\]

\[
H = \nabla^2_u \phi
\]

Each newton iteration, the newton direction \( \Delta u = -H^{-1}g \) is solved and used to update the state vector by

\[
u^{(j+1)} = u^{(j)} + s\Delta u
\]

where \( s \) is chosen such that the new cost is less than the previous cost,

\[
c^T u^{(j+1)} + \sum_k \phi_k(u^{(j+1)}) < c^T u^{(j)} + \sum_k \phi_k(u^{(j)})
\]

(26)

Iterations of the inner loop continue until the stopping condition of \( \lambda^2/2 \leq \epsilon \) \((\lambda^2 = g^T H^{-1} g)\) is met or it is forcibly stopped by defining the max number of inner iterations.

Depending how \( s \) is chosen can increase complexity in each inner iteration. Backtracking line search is an iterative process where \( s \) is initialized at one and updated \((s = \beta s)\) until (26) is met. A very simple, closed-form, way to calculate \( s \) [9], but not as aggressive as backtracking line-search, is

\[
s = \frac{1}{1 + \lambda}
\]

(27)

The price of the simplicity and speed of calculation, is the larger number of newton iterations it takes to converge. Therefore, backtracking line-search will be more costly to perform but will reduce the number of newton iterations and therefore possibly the total time to solve the problem. In the results section, we will compare these two ways of calculating \( s \).

VI. NUMERICAL EXAMPLES AND EXPERIMENTAL RESULTS

Table III compares optimality gap, number of newton iterations, and time to solve, of the SOCP solver. To calculate optimality gap, we first solved a specific problem \((F = [1 \; 2 \; 3 \; 4 \; 5 \; 6]^T)\) using CVX [10] (0.5s to solve) and used this solution to compare optimality gap with our SOCP solver running on an 800MHz laptop PC. The computation time was estimated by solving the same problem \(10^6 \) times.

The terms \( \mu, \beta \), and \( \alpha \), used in line-search and outer update were all kept constant. The results show that it is possible to use convex optimization in real-time applications. MAPS PMAC controller, running an 800MHz power PC, updates the phase every 110\(\mu s\). We can use any of the adaptations of the SOCP solver. The fastest solves were forced to stop before converging and not using backtracking line-search or the Cholesky factorization. This makes sense since all of these take time to calculate. Notice that using the line-search algorithm will reduce the number of newton iterations, as assumed.

The methods were implemented on the MAPS for demonstration. Fig. 3 shows the power of each motor while MAPS was under closed-loop regulation and minimum power commutation. An external disturbance
was added at motor #1 (see Fig. 2), making motor #3 a pivot point where power is the least of all the motors. The implementation of the power symmetry commutation of table I, is shown in Fig. 4 where disturbance was also added at motor #1. Notice from Fig. 5 that implementing either commutation scheme has no effect on the following error.

![Graphs showing motor power over time](image1)

**Fig. 3: minimum power implementation**

![Graphs showing motor power over time](image2)

**Fig. 4: minimum power with symmetry constraint implementation**

**VII. CONCLUSION**

In this paper we developed two optimal commutation schemes for multiple linear-motor precision systems. The first was a minimum power commutation which resulted in a closed-form solution, that is easily implemented in real-time. The second was a power symmetry commutation which resulted in needing to solve an SOCP problem. We wrote our own SOCP solver that was able to solve the problem in less than 35 µs, which is within one phasing cycle of 110 µs. Finally, both commutation methods were implemented under a 440 µs PID servo control loop and 110 µs phase loop to demonstrate their effectiveness were presented.

**ACKNOWLEDGMENTS**

The authors wish to acknowledge Prof. L. Vandenberghe (UCLA, Electrical Engineering Department) for his invaluable advice, Delta Tau Data Systems, Inc., for donating the Power PMAC real-time control hardware and software, and partial support by the National Science Foundation under grant no. DMI 0327077 and CMMI 0751621.

**REFERENCES**


