JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 33, No. 4, July–August 2010

Explicit Dipole Trajectory Solution for Electromagnetically Controlled Spacecraft Clusters

Samuel A. Schweighart*
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
and
Raymond J. Sedwick†
University of Maryland, College Park, Maryland 20742

DOI: 10.2514/1.46363

Satellite formation flight is a much-anticipated technology for future space-science missions, such as high-resolution space-based telescopes. A potentially enabling technology under development allows clusters of spacecraft to maneuver without the use of propellant, instead using electromagnetic control of the relative degrees of freedom within the cluster. This electromagnetic formation-flight technology offers potentially limitless mission life, in exchange for a highly coupled, nonlinear control problem. Several methods have been explored to address this problem, and some have achieved considerable success at its optimal, or at least locally optimal, solution. This paper does not produce optimal trajectories for unconstrained maneuvers, but instead provides a method for determining all of the feasible control solutions that will produce a specified maneuver. An example is provided for the case of three spacecraft rotating in a planar, equilateral triangle configuration, while the plane of rotation is simultaneously changing direction, as might be encountered while retargeting a separated spacecraft interferometer. The approach may be used on its own to find feasible, albeit suboptimal, solutions, or in conjunction with an optimal-control solution, to probe its optimality, find alternative solutions, or plan for contingencies. The current research deals solely with free-space translational solutions, without consideration of torques that are developed or the resulting buildup or removal of angular momentum. Extension of the current approach to address these issues is the subject of a follow-on paper.

I. Introduction

OVER the past decade, there has been a significant amount of interest and research in the formation flight of satellites. The purpose of satellite formation flight is to perform a mission by using multiple spacecraft, operating from as close as a few meters to several hundred meters apart. A typical goal of such an approach is to increase the resolution of a telescope by synthesizing an aperture that is much larger than can be launched as a monolith. Whatever the mission, formation-flying multiple vehicles in close proximity comes with a variety of challenges, including sensing, control, and actuation.

The current work addresses a relatively new technology referred to as electromagnetic formation flight (EMFF). In brief, EMFF is the use of electrically generated magnetic fields, formed locally on all of the spacecraft, to control the relative degrees of freedom of a cluster of vehicles. With the addition of independent torque control devices, such as reaction wheels, it has previously been shown that all of the relative translational and rotational degrees of freedom of the cluster can be controlled [1]. An artist's concept of the vehicle is shown in Fig. 1. A number of papers addressing various aspects of this technology have been published or are in review by the current authors, as well as others [2,3], and a detailed description of hardware implementations [4–6], optimal-control strategies [7,8], and performance trades [9,10] can be found in the literature.

This paper is the first in a three-part series presenting an unconventional approach to finding the time history of the magnetic dipoles, which must be created on each vehicle in the cluster to produce a particular state-space trajectory. The method is referred to as unconventional, since a typical approach to such a problem would be to apply any of a number of optimal-control techniques to simultaneously solve for these dipoles as a dynamic constraint while minimizing a cost function that may include, for example, time, peak power, maximum torque, or total energy expended. A very effective example of such an approach can be found in [7,8]. In the first of these, Ahsun primarily addresses the issue of angular momentum management, which will be deferred until the second paper in this series. In [8], a potential-function method is employed to incorporate collision avoidance into a time-optimal reconfiguration maneuver and is shown to succeed with a relatively low computational burden.

The research presented here predates that of [7,8] and is actually a summary of the unpublished Ref. 9 of [7]. The current approach does not produce an optimal trajectory or solve the equations of motion, and in fact requires that the vehicle trajectories be provided as input. Used in conjunction with an optimal-control scheme, the current approach can find alternative dipole configurations that will achieve the same translational force profile while redistributing the torques among the vehicles of the cluster. Used with a feedback controller, the current approach can provide coordinated dipole solutions that will maintain the desired trajectory in the presence of disturbance forces. Only the translational force constraints are considered in this paper, the implicit assumption being that sufficient angular momentum storage exists to compensate for residual torques that are generated over the course of the maneuver. In the first of two follow-on papers, the current method is extended to demonstrate how the availability of multiple solutions can be used to redistribute the angular momentum between the vehicles. The current work also does not address the presence of external forces; however, the second follow-on paper shows how the method can again be extended to include the gravitational and magnetic field effects that would be present in an Earth-orbiting cluster.

II. Describing the Force-Constraint Equations

To control the translational motion of each of N spacecraft, any desired force must be achievable as a superposition of the forces from

Received 16 July 2009; revision received 26 October 2009; accepted for publication 27 October 2009.

*Doctoral Candidate, Space Systems Laboratory, Aeronautics and Astronautics, Room 37-350. Student Member AIAA.

†Assistant Professor, Space Power and Propulsion Laboratory, Aerospace Engineering, 3146 Martin Hall. Associate Fellow AIAA.

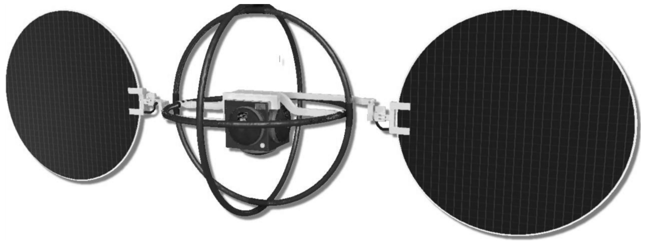

Fig. 1 Artist's concept of an EMFF vehicle, showing three orthogonal superconducting coils and a central reaction-wheel assembly.

the other $N-1$ spacecraft. The dipole on any given vehicle is assumed to be variable in both magnitude and direction, by implementing current coils spanning three orthogonal planes. A detailed description of the setup can be found in [10]. The force-constraint equations for the N spacecraft can be written generally as

$$\mathbf{F}_j = \sum_{\substack{i=1:N \\ i\neq j}} \mathbf{F}_{ij}(\mathbf{r}_{ij}, \boldsymbol{\mu}_i, \boldsymbol{\mu}_j) = \mathbf{f}_j \qquad j = 1{:}N \tag{1}$$

where the force between any two spacecraft is given in the far field by

$$\mathbf{F}_{ij} = \nabla\mathbf{B}_i \cdot \boldsymbol{\mu}_j = \frac{3\mu_0}{4\pi}\frac{\mu_i\mu_j}{r_{ij}^4}((\hat{\mu}_i\cdot\hat{\mu}_j)\hat{r}_{ij} + (\hat{\mu}_i\cdot\hat{r}_{ij})\hat{\mu}_j + (\hat{\mu}_j\cdot\hat{r}_{ij})\hat{\mu}_i - 5(\hat{\mu}_i\cdot\hat{r}_{ij})(\hat{\mu}_j\cdot\hat{r}_{ij})\hat{r}_{ij})\boldsymbol{\nabla} \tag{2}$$

Since for now we are assuming that the only forces are those generated between spacecraft, the sum of these internal forces over the entire cluster must equal zero by Newton's third law, or momentum conservation. From Eq. (1), we see that while there are still $3N$ independent control variables (the components of $\boldsymbol{\mu}_i$), we now have only $3(N-1)$ constraint equations. The reduction in constraints means that three components of the dipole moments can be chosen at will. The three components must belong to the same dipole, since specifying only two components of a vector is equivalent to indirectly choosing its third component to be zero. The ability to choose one of the dipoles will be shown to be an enabling feature of EMFF, and this unconstrained dipole will commonly be referred to as the *free dipole*.

Rather than directly specifying the components of the free dipole, three additional constraints could be levied on the system. This approach will be examined in the follow-on paper to this one, but for now we will just assume that the free-dipole components have been specified. While it might appear that the choice of the components of the free dipole is completely unconstrained, there are in fact degenerate cases that must be considered. From Eq. (2), the force on vehicle j as a result of the field generated by vehicle i is given by

$$\mathbf{F}_{ij} = \nabla\mathbf{B}_i \cdot \boldsymbol{\mu}_j \tag{3}$$

where $\nabla\mathbf{B}_i$ can be viewed as a matrix operator transforming the dipole vector into the force vector. Provided this matrix is full-rank, it will always be possible to find a dipole moment that will produce the desired force. Looking more closely at Eq. (2), it appears as though the unit vectors that span the space are given on the right side by $(\hat{\mu}_i, \hat{\mu}_j, \hat{r}_{ij})$. If the free dipole is chosen perpendicular to $\hat{r}_{ij}$, then from Eq. (2) the remaining subspace is spanned by $(\hat{\mu}_i, \hat{r}_{ij})$. If the problem is only two-dimensional, these unit vectors are sufficient to span the entire space. However, if the problem is three-dimensional, this two-dimensional basis cannot faithfully reproduce any arbitrary force that may be desired on the left-hand side of Eq. (3). In this case, we expect that $\nabla\mathbf{B}_i$ will not be full-rank. As the number of vehicles increases and the $\hat{r}_{ij}$ vectors become more diverse, it becomes less likely that any given choice of the free dipole will result in such a degenerate situation.

At this point, the current choice of notation becomes somewhat cumbersome. To try to make things more transparent, the following change in notation will be made:

$$F_i^m = \text{Component } i \text{ of magnetic force on vehicle } m \tag{4a}$$

$$m_j^m = \text{Component } j \text{ of dipole on vehicle } m \tag{4b}$$

$$f_i^m = \text{Component } i \text{ of required force on vehicle } m \tag{4c}$$

Under this redefinition, the force-constraint equations for the entire cluster can be written in the following compact representation:

$$F_i^m = m_j^m \, {}^{m}C_n^{jk} m_j^n = f_i^m \quad \text{with} \quad n \neq m \tag{5}$$

where i and m are free indices and the standard summation rule for repeated indices is used, with the exception that although m is repeated on the right, it is still free and not meant to be summed. The summation exception is indicated by placing the m label as a leading superscript. Also note that the summation over n is for $n \neq m$ or, alternatively, the values of ${}^{m}C_n^{jk}$, where $n = m$ could be defined to be zero. In this form, it can be seen that the force equations are essentially a system of multivariate bilinear forms, or bilinear polynomials. Solving systems of polynomial equations has been the focus of vast amounts of research which is still active to this day.

It is worth noting that the bilinear forms investigated in this approach are not unique to dipole force interactions. While the structure of the coefficients will differ, the interaction of electrostatic monopoles will also result in such forms, with each vehicle having a single-actuator degree of freedom instead of three. The approach discussed here can therefore be applied equally well to coulomb formation flight [11].

III. Characterization and Conditioning of the Force-Constraint Equations

Before solving the force-constraint equations, a brief discussion of their mathematical attributes is presented, followed by a series of steps to better condition them for solving.

A. Reduction of Degree and Removal of Solutions at Infinity

The degree of a polynomial system is a good indicator of the complexity of the system. Much the same as the degree of a univariate polynomial equation gives the anticipated difficulty in finding a solution, the degree of a system of polynomial equations gives an indication of the difficulty in solving the system. The degree of a polynomial system is given by [12]

$$d = \prod_{m=1}^{N-1} d_1^m d_2^m d_3^m \tag{6}$$

where d_j^m is the maximum degree of the monomials in the equation for vehicle m and component j. Looking at the force constraints given in Eq. (5), we can see that $d_m^j \leq 2$. If we define the components of the free dipole as being m_j^1 (the first vehicle), then the degrees of F_j^1 are each equal to one and the remaining $F_j^{m\neq 1}$ all have degree two. The overall degree of the system of Eq. (5) is therefore $d = 2^{3(N-1)}$, showing that the complexity of the system increases exponentially for each spacecraft added to the formation.

The number of solutions for a system of n polynomial equations of n variables is given by Bézout's theorem [12]. The system of polynomial equations has a degree d as defined in Eq. (6). Bézout's Theorem states that if there are a finite number of noninfinite solutions and a finite number of solutions at infinity, then there are a total of exactly d solutions, accounting for all multiplicities. A solution at infinity is a solution to the polynomial equation that can only be obtained as the variables approach infinity. A systematic method of finding the solutions at infinity is to define ${}^{H}F_i^m$ to be the *homogeneous* component of F_i^m. These can be found by

$$ {}^{H}F_k^m(m_i^j) = r^{d_k^m} F_k^m\left(\frac{m_i^j}{r}\right)\bigg|_{r=0} \tag{7}$$

where m and k are labels and are not meant to imply a summation. The solutions at infinity are not of use, so we would like to reduce the complexity of the system of polynomials by removing them through systematic reduction.

The process of reducing the system of polynomial equations is similar to row reduction for linear systems. A matrix with the coefficients of each monomial is first created, with each monomial associated with a column of the matrix and the columns ordered with higher degree terms to the left. Standard Gaussian elimination is then applied to the matrix, reducing the number of higher-order terms and it is hoped the overall degree of the system. As an example, consider the following system of polynomial equations of degree nine:

$$\begin{aligned} F_1(x,y) &= y^3 + 3xy^2 - 4x^3 + 2x^2 + 3y + 1 = 0 \\ F_2(x,y) &= 2y^3 + 6xy^2 - 8x^3 + y + 2 = 0 \end{aligned} \tag{8}$$

The coefficient matrix, both before and after Gaussian elimination to row echelon form, is then

$$\begin{array}{cc} & \begin{array}{cccccc} y^3 & xy^2 & x^3 & x^2 & y & 1 \end{array} \\ \begin{array}{c} F_1 \rightarrow \\ F_2 \rightarrow \end{array} & \begin{pmatrix} 1 & 3 & -4 & 2 & 3 & 1 \\ 2 & 6 & -8 & 0 & 1 & 2 \end{pmatrix} \\ \Rightarrow & \begin{pmatrix} 1 & 3 & -4 & 0 & 0.5 & 1 \\ 0 & 0 & 0 & 1 & 1.5 & 0 \end{pmatrix} \end{array} \tag{9}$$

Looking at the reduced matrix, all of the degree-three terms in the second equation now have zero coefficients. The reduced system of equations is now given by

$$\begin{aligned} F_1^R(x,y) &= y^3 + 3xy^2 - 4x^3 + 0.5y + 1 = 0 \\ F_2^R(x,y) &= x^2 + 1.25y + 2 = 0 \end{aligned} \tag{10}$$

and we see that the reduced system has degree six. The three missing solutions all lie at infinity.

When the force-constraint equations were first introduced, we stated that as a result of momentum conservation the equations are not all independent. We used this fact to eliminate three of the equations, since the sum of the forces in each direction must be zero. Based on the discussion above, it is clear that by eliminating the equations in which the free dipole does not multiply all of the terms, a greater reduction in the overall degree of the system is achieved. The reduction method just described can be used to systematically accomplish this same goal. Let us assume that we have four spacecraft and the free dipole is on vehicle no. 1. We order the columns of the reduction matrix such that the terms containing the free dipole are in the rightmost columns. For simplicity, only one row is given for each vehicle, representing all three components, and the coupling coefficients have been relabeled. The matrices before and after reduction are then

$$\begin{array}{cc} & \begin{array}{cccccc} \boldsymbol{\mu}^3\boldsymbol{\mu}^4 & \boldsymbol{\mu}^2\boldsymbol{\mu}^4 & \boldsymbol{\mu}^2\boldsymbol{\mu}^3 & \boldsymbol{\mu}^1\boldsymbol{\mu}^4 & \boldsymbol{\mu}^1\boldsymbol{\mu}^3 & \boldsymbol{\mu}^1\boldsymbol{\mu}^2 \end{array} \\ \begin{array}{c} \mathbf{F}_1 \rightarrow \\ \mathbf{F}_2 \rightarrow \\ \mathbf{F}_3 \rightarrow \\ \mathbf{F}_4 \rightarrow \end{array} & \begin{pmatrix} 0 & 0 & 0 & \mathbf{C}_4 & \mathbf{C}_5 & \mathbf{C}_6 \\ 0 & \mathbf{C}_2 & \mathbf{C}_3 & 0 & 0 & -\mathbf{C}_6 \\ \mathbf{C}_1 & 0 & -\mathbf{C}_3 & 0 & -\mathbf{C}_5 & 0 \\ -\mathbf{C}_1 & -\mathbf{C}_2 & 0 & -\mathbf{C}_4 & 0 & 0 \end{pmatrix} \\ \Rightarrow & \begin{pmatrix} \mathbf{C}_1 & 0 & -\mathbf{C}_3 & 0 & -\mathbf{C}_5 & 0 \\ 0 & \mathbf{C}_2 & \mathbf{C}_3 & 0 & 0 & -\mathbf{C}_6 \\ 0 & 0 & 0 & \mathbf{C}_4 & \mathbf{C}_5 & \mathbf{C}_6 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \tag{11}$$

As expected, the number of equations has been reduced by three. Also, because the three rightmost columns are only of degree one, we have also ensured that one of the vector equations, $\mathbf{F}_3$, is only of degree one.

B. Rescaling the Equations

To numerically solve for the dipole components in a large system of equations, it is often beneficial to rescale the equations to bring the solution variables into roughly the same order of magnitude. The scaling does not affect the final solution, since an inverse transformation of the scaling is done after a solution is found. The approach below follows that of Morgan [12]. We will assume a set of equations given by $f_j(x_k) = 0$. The first step is to scale the variables and the equations as follows:

$$x_k = 10^{s_k} z_k \qquad 10^{v_i} f_i(x_k) = 0 \tag{12}$$

where s_k and v_i are vectors of scaling values and no summation is meant to be implied. Each monomial term in both the unscaled and scaled equations is given by

$$a_{ij} \prod_k x_k^{g_{ij}^k} \Rightarrow 10^{v_i} a_{ij} \prod_k (10^{s_k} z_k)^{g_{ij}^k} \tag{13}$$

where a_{ij} is the coefficient of the jth term in the ith equation, and g_{ij}^k is the degree of each variable in that term. The goal is now to pick the values for s_k and v_i so that the scaled equations are numerically well-behaved, meaning that the terms are all of the same order of magnitude. Once the new equations have been solved, we can obtain the solutions to the original equations applying the inverse scaling.

To find the appropriate scaling values, a least-squares method is used. We want to minimize both the differences in the values of the a_{ij} as well as their deviation from unity. To accomplish this goal, we note that if each term in the scaled equation of Eq. (13) is written in the form

$$10^{c_{ij}} \prod_k (z_k)^{g_{ij}^k} \quad \text{where } c_{ij} = v_j + \sum_{k=1}^{n} s_k g_{ij}^k + \log_{10}(|a_{ij}|) \tag{14}$$

then to minimize the differences in the constants and their deviations from unity, we must just minimize

$$J = \sum_{i=1}^{N} \sum_{j=1}^{l_i} c_{ij}^2 + \sum_{i=1}^{N} \sum_{1 \le i \le m \le l_i} (c_{ij} - c_{im})^2 \tag{15}$$

where l_i is the number of terms in equation i. This objective function is just a sum of quadratics, and the v_j and s_k can easily be found using any of a variety of quadratic solvers.

C. Singular Solutions

For single-variable equations, a solution is said to be singular when the equation and the derivative of the equation are both equal to zero. For example, x is a singular point if

$$g(x) = x^2 - 2x + 1 = 0 \qquad g'(x) = 2x - 2 = 0 \tag{16}$$

which in this case corresponds to $x = 1$. In this case, this solution has a multiplicity of two, because there are essentially two merged solutions. This can be seen by varying the parameters slightly,

$$\begin{aligned} x^2 - 1.99x + 1 = 0 &\rightarrow x \approx (0.99 + 0.099i, 0.99 - 0.09i) \\ x^2 - 2x + 1 = 0 &\rightarrow x = (1, 1) \\ x^2 - 2.01x + 1 = 0 &\rightarrow x \approx (1.11, 0.91) \end{aligned} \tag{17}$$

and as can be seen, the solution transitions from imaginary to real at the singularity, a useful result to be used later.

If there are more than one variable and more than one equation, then the requirement for a point to be singular is that the equation and the determinant of the Jacobian must be zero. The Jacobian is defined by

$$J(x_i) = \frac{\partial F_j}{\partial x_i} \tag{18}$$

If we represent the polynomials as geometric curves or shapes, a singular solution is one where the tangents to the curves coincide. In other words, the curves are barely touching but do not completely intersect. It also marks the region where a solution transitions from being real to imaginary. A solution is singular if and only if it has a multiplicity of two or more. Consider, for example,

$$-x^4 + 2x^2 - y = 0 \qquad x^2 + (y - c)^2 - 1 = 0 \tag{19}$$

When $c = 1.05$, there are four real solutions with the associated determinant of the Jacobian:

$$\begin{aligned} (x, y) &\approx (\pm 0.19, 0.07) \qquad \det(J) \approx \mp 1.03 \\ (x, y) &\approx (\pm 0.99, 0.99) \qquad \det(J) \approx \pm 1.99 \end{aligned} \tag{20}$$

When $c = 1$, the solutions “merge” together at the origin, and the solution (0,0) has multiplicity two. The determinant of the Jacobian is also zero at this point. With $c < 1$, the solutions and the Jacobian become imaginary. These cases are shown in Fig. 2.

We can see the point at which a solution is approaching a singularity (while varying c) by tracking the Jacobian determinant. If the determinant approaches zero, then we know that the solution is becoming singular. However, this method is dependent on the scaling of the matrix. Instead of using the determinant to determine how close to being singular the matrix is getting, we can use the condition of the matrix. This can be found efficiently by use of singular-value decomposition (SVD) of the matrix

$$J = UDV^{-1} \qquad C(J) \equiv \frac{\max D_i}{\min D_i} \tag{21}$$

where D is a diagonal matrix of singular values and $C(J)$ is the condition number of the matrix. When there is a singular solution, the condition number goes to infinity. The condition matrix is useful because it always produces a real number and is unaffected by scaling. Continuing our example, we see in Fig. 3 that the condition of the matrix for the solutions that are singular at $c = 1$ approaches infinity at that point, while the condition of the other solutions remains around a value of two.

D. Solving Polynomial Equations Using Continuation Methods

Newton’s method is designed to find a local solution to a system of nonlinear equations given an initial guess. As long as the initial guess is sufficiently close to the final solution, Newton’s method will typically converge to the solution. However, from Bezout’s theorem, we know that there are multiple solutions to our system of equations, some of which may be more desirable than others. Using Newton’s method, one would have to randomly search the entire space in the hope of finding each solution. Because of solutions at infinity and singular points, one would never know if all possible solutions had been found. What is needed is a systematic method for finding all solutions to these polynomial equations, and continuation is just such a method.

Continuation methods use the idea of taking an equation that is difficult to solve and transforming it in a continuous manner into an equation that is simple to solve. After solving, this new equation is then continuously transformed back to the original equation, tracking the solutions along the way using Newton’s method. Because at each step the equation is very similar to the previous step, the solutions are also very close to the previous step and can be used as an initial guess in Newton’s method.

As an example, consider the equation $x^2 + 3x - 3 = 0$. While this equation can be readily solved using the quadratic formula, instead assume a slight modification to the equation yielding $x^2 + 3\delta x - 3 = 0$, where the parameter δ can take on any value from zero to one. When $\delta = 0$, this equation becomes $x^2 - 3 = 0$, which can be immediately solved to find $x = \pm\sqrt{3}$. However, when $\delta = 1$, this equation becomes the original equation. It stands to reason that as δ varies slowly from zero to one, the roots of the equation will

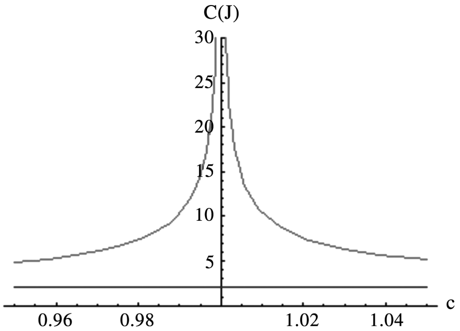

Fig. 3 Condition of the Jacobian matrix for both $c = 1$ (upper curve) and $c \neq 1$ (lower line).

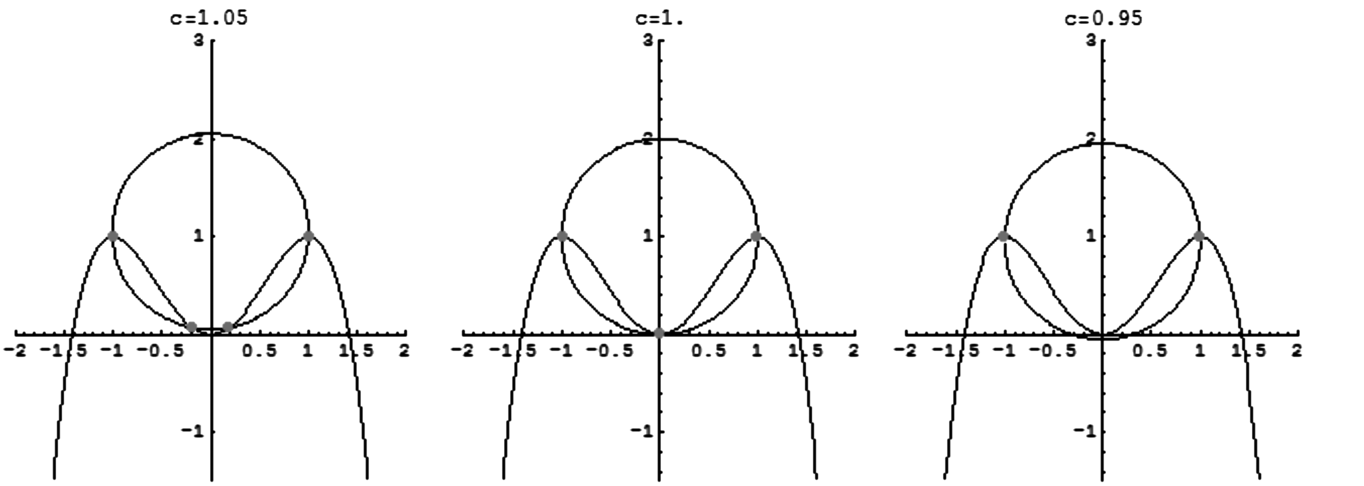

Fig. 2 Intersection of two curves at a singularity.

vary from $x = \pm\sqrt{3}$ to the solutions that we desire. This is indeed the case, and this technique forms the basis for the continuation method. This was a simple example of the continuation method; however, in practice there are a number of pitfalls and difficulties that a general method must overcome. For the discussion that follows, it is helpful to graphically represent the solution paths to the equations as they are varied.

In the previous example, the two solutions were never equal at any value of δ, and so their paths did not cross. However, consider another innocent looking quadratic equation:

$$\begin{gathered} x^2 + 5x + 4 = 0 \Rightarrow x^2 + \delta 5x + 4 = 0 \\ x = \{-1, -4\} \quad \text{for } \delta = 1 \end{gathered} \tag{22}$$

The solution paths are shown in Fig. 4 (left) as δ is varied from zero to one. We see that at $\delta = 0.8$ the solution paths cross. The crossing is a problem for two reasons. First, at $\delta = 0.8 + \Delta\delta$, Newton's method is seeded with the same initial conditions for both paths $[x(0.8) = -2]$, and as a result it will produce the same solution. One of the solution paths after the intersection will be lost. Second, when the solution paths cross, they are crossing at a singular solution to the continuation equation. Newton's method does not guarantee quadratic convergence at singular points and, in general, does not behave well. Since we are using real coefficients, it should be noted that when a singular point is reached (except for very specific cases), the solution passes from the real axis to the complex plane, and vice versa. The converse is also true—if a solution path switches from a real to a complex solution, or vice versa, then it must happen through a singular point.

One method of dealing with singular points is to choose a continuation equation that does not contain any singularities (except at $\delta = 1$ if the solution to the original equation contains singularities). An equation of one variable has a singularity when it and the first derivative are equal to zero. Therefore, for a quadratic continuation equation of the form

$$x^2 + \delta bx + c = 0 \tag{23}$$

the condition for singularity (and its location) can be found by setting the first derivative to zero, resulting in

$$\delta = \pm\frac{2\sqrt{c}}{b} \qquad x = \mp\sqrt{c} \tag{24}$$

Because δ is to vary from zero to one, if the value of δ given above falls within this range, the singularity will occur along the solution path. To remove this problem, we can instead try a different form of continuation equation:

$$x^2 + \delta bx + \delta c - (1 - \delta)q^2 = 0 \tag{25}$$

which by a similar calculation will have a singular point when

$$-q^2(1 - \delta) + c\delta\frac{b^2\delta^2}{4} = 0 \tag{26}$$

Since Eq. (26) is quadratic in δ, we know that there are two possible values. All we must do is find q such that Eq. (26) is not satisfied for $0 \le \delta < 1$. If we know b and c, it is straightforward to find an appropriate q. However, it is possible to find a generic q that almost always prevents Eq. (26) from being satisfied when $0 \le \delta < 1$. If we make q a complex number, then the δ that satisfies Eq. (26) will also be a complex number unless q, c, and b have a very specific relationship. As long as we pick q randomly, the chances that q, c, and b are in the required specific relation is quite small, and thus Eq. (26) cannot be generally satisfied with a real δ.

Referring back to Fig. 4 (right), the solution paths for Eq. (25) do not intersect, and both solutions are found. It should be noted that the size of q has an influence on the path shape. If q is very small, then the equations can be nearly dependent and the solutions pass near to each other but do not cross. If the paths come too close together, the solution obtained from Newton's method can jump from one path to another. Conversely, if we make q too large, then the initial solutions are very far away from the final solutions and the solution must change significantly as δ varies from zero to one. This situation can also cause problems with the solution jumps paths. A good rule of thumb is to choose q to be of the same order of magnitude as the constants in the equations. As long as we have appropriately scaled the equations (as described in the previous sections), the constants should all be close to unity. Therefore, the magnitude of q should be close to unity.

E. Change of Basis

Even though we may have attempted to reduce the system as much as possible, we may not have completely done so, and therefore there may be some solutions that remain at infinity. If a system has n solutions (some of which lie at infinity), and we start with n finite points, some paths must diverge to infinity. Diverging solutions are a problem, because the numeric solvers have difficulty finding these solutions as they increase to infinity. Unfortunately, we cannot tell a priori which solutions are bound for infinity. We could select bounds on the solution and terminate it if it crosses the bound, but it is difficult to know where to set the bounds. The preferred method of dealing with solutions at infinity is to use a projective representation of the equations, which is essentially a change of basis. We can change the basis in such a way that all of the solutions remain bounded, and once the solutions have been found we can simply change back to the original basis. A detailed explanation and proof of projective representations can be found in [12].

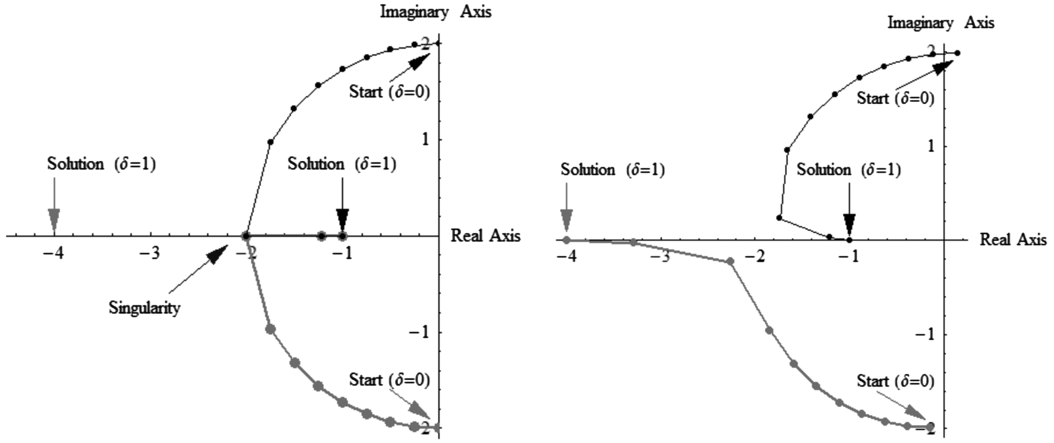

Fig. 4 Solution paths for Eq. (22) (left) and Eq. (25) (right).

Recall that the initial set of equations we are trying to solve is

$$\mathbf{F}_j(m_i) = 0 \tag{27}$$

The first step is a change of variables and a rescaling of the equations:

$$m_i \to \frac{y_i}{y_{n+1}} \qquad \hat{F}_j(y_1, y_2, \ldots, y_{n+1}) = y_{n+1}^{d_j}\mathbf{F}_j\left(\frac{y_1}{y_{n+1}}, \frac{y_2}{y_{n+1}}, \ldots, \frac{y_n}{y_{n+1}}\right) \tag{28}$$

where d_j is the degree of the equation. $\hat{F}_j$ is simply the original equation with an added variable multiplied onto each term, so that each term has the same degree as the original equation. For example,

$$F_1(\mathbf{m}) = 3m_1^2m_2^2 + m_3^2 + m_1m_3^2 \Rightarrow \hat{F}_1(\mathbf{y}) = 3y_1^2y_2^2 + y_3^2y_4^2 + y_1y_3^2y_4 \tag{29}$$

It should be noted that solutions to $\hat{F}(y_1, y_2, \ldots, y_n, 0) = 0$ produce the solutions at infinity for $\mathbf{F}$, because Eq. (29) produces the homogenous equations of $\mathbf{F}$. We now define

$$l(\mathbf{y}) = a_1y_1 + a_2y_2 + \ldots + a_ny_n \qquad L(\mathbf{y}) = l(\mathbf{y}) + a_{n+1} \tag{30}$$

where the constants a_j are real or complex numbers chosen at random, and $a_{n+1} \neq 0$. The projective representation of $\mathbf{F}$ is given as

$$\hat{F}(y_1, \ldots, y_n, L(\mathbf{y})) = 0 \tag{31}$$

the solutions of which can be converted to solutions of $\mathbf{F}(\mathbf{m})$ by

$$\mathbf{m}_j = \frac{\mathbf{y}_j}{l(\mathbf{y}_j)} \tag{32}$$

where j is the solution number and not the vector component of m or y. Solutions at infinity are recovered when $l(\mathbf{y}) = 0$. To recap, some solutions of $\mathbf{F}$ may lie at infinity. These solutions are problematic to the numeric solver, because the solution grows without bound. To avoid this situation, $\mathbf{F}$ is changed into $\hat{F}$ through a change of basis. Random constants are chosen, and $L(x)$ and $l(x)$ are created. $\hat{F}^{L,n+1}$ has no solutions at infinity, and can be solved using a numeric solver. These solutions are finally mapped exactly back to solutions to the original equation by dividing by $l(\mathbf{y})$.

IV. General Solution Applied to EMFF Force-Constraint Equations

Having discussed all of the difficulties that can happen when trying to find the solutions to a set of polynomial equations, we finally have all the pieces to develop a method to solve the EMFF force-constraint equations. Starting off with the problem statement, our goal is to solve the following equation system of equations:

$$\mathbf{f}(\mathbf{m}) = 0 \tag{33}$$

where each component of $\mathbf{f}$ is a polynomial equation in the variables $\mathbf{m}\boldsymbol{\mu}$. The number of equations is initially the same as the number of variables. Before we attempt to solve the system, we must reduce and scale the equations. Since we cannot guarantee that we have completely reduced the equations, we must also use the projective representation to account for solutions at infinity. The entire process is

$$\mathbf{f}(\mathbf{m}) = 0 \xrightarrow{\text{Reduce}} \mathbf{f}'(\mathbf{m}) = 0 \xrightarrow{\text{Scale}} \mathbf{f}''(\mathbf{m}'') = 0 \xrightarrow{\text{Project}} \mathbf{f}'''(\mathbf{m}''') = 0 \tag{34}$$

For simplicity, we will use the notation $\mathbf{F}(\mathbf{x}) \equiv \mathbf{f}'''(\mathbf{m}''')$. Because the form and degree of the equations vary, we need a continuation equation that will handle all the different types. The following continuation equation essentially starts with a completely different equation at $\delta = 0$, and each individual equation in $\mathbf{g}(\mathbf{x})$ has the same degree as the associated equation in $\mathbf{F}(\mathbf{x})$:

$$\mathbf{h}(\mathbf{x}, \delta) \equiv (1 - \delta)\mathbf{g}(\mathbf{x}) + \delta\mathbf{F}(\mathbf{x}) = 0 \tag{35}$$

Because we do not want to allow solution paths to cross, we must add randomness (independence) to the continuation equation. This addition can be accomplished by the choice of $\mathbf{g}(\mathbf{x})$. Let $\mathbf{p}$ and $\mathbf{q}$ be vectors of random complex numbers, with magnitude of order unity:

$$g_j(\mathbf{x}) = p_j^{d_j}x_j^{d_j} - q_j^{d_j} \tag{36}$$

where d_j is the degree of the jth equation. The initial solutions $\mathbf{h}(\mathbf{x}, 0) = 0$ are given by

$$x_j|_{\delta=0} = \frac{q_j}{p_j}\exp\left(i\frac{2\pi k_j}{d_j}\right) \quad \text{where } 1 \leq k_j \leq d_j \quad \text{and} \quad k_j \in \mathbb{Z} \tag{37}$$

The final step is to propagate each initial solution, as δ varies from zero to one. Once all solutions have been found at $\delta = 1$, they must be changed back to the original basis and then the values must be rescaled. The result is that all possible solutions (real, imaginary, and infinite) have been systematically found.

With the general framework in place, we can now apply the method to an EMFF example. At a specific point in time, we are given the desired forces to be applied on each spacecraft. The free dipole has not been assigned, and we will choose it arbitrarily. Using Bezout's theorem, we know that for two vehicles there will be only one solution, and the use of the continuation method is pointless since we can easily use Newton's method to seek out the one real solution to the problem. Therefore, for our example we will use three vehicles in three dimensions, which will provide eight solutions. Assume that there are three vehicles in an equilateral triangle configuration, with the center of mass corresponding to the origin of an inertial frame, as shown in Fig. 5. Each vehicle is 10 meters from the center of mass, and there are no outside forces or fields present. The plane in which the spacecraft initially reside is the x-y plane, and their positions and desired forces at a given instant are

$$\begin{aligned} \mathbf{r}_1 &= [10, \quad 0, \quad 0]^T & \mathbf{f}_1 &= [-30.4, \quad 0, \quad 7.62]^T \text{ mN} \\ \mathbf{r}_2 &= [-5, \quad 8.66, \quad 0]^T & \mathbf{f}_2 &= [15.2, \quad -26.8, \quad -3.81]^T \text{ mN} \\ \mathbf{r}_3 &= [-5, \quad -8.66, \quad 0]^T & \mathbf{f}_3 &= [15.2, \quad 26.8, \quad -3.81]^T \text{ mN} \end{aligned} \tag{38}$$

It should be noted that the sum of the desired forces is equal to zero, as required. Since the free dipole can be chosen at will, it is selected to be on spacecraft no. 1, and the full component set is given by

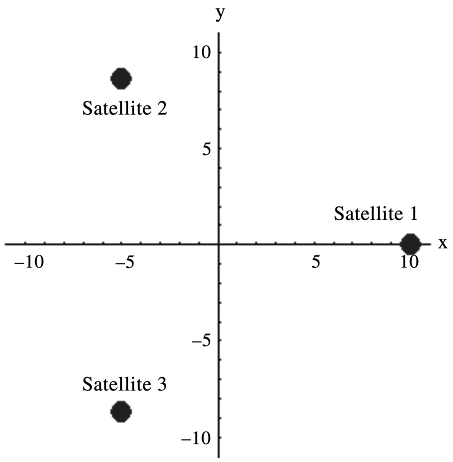

Fig. 5 Position of spacecraft.

$$\boldsymbol{\mu}_1 = [0, \quad 40000, \quad 0]^T \text{ A-m}^2$$

$$\boldsymbol{\mu}_2 = [m_1, \quad m_2, \quad m_3]^T \text{ A-m}^2 \qquad \boldsymbol{\mu}_3 = [m_4, \quad m_5, \quad m_6]^T \text{ A-m}^2 \tag{39}$$

Substituting Eqs. (38) and (39) in Eq. (1) results in the following nine force-constraint equations:

$$\begin{aligned}
&-1.83(10^{-7})m_1 + 2.89(10^{-8})m_2 + 3.33(10^{-12})m_2m_4 \\
&\quad + 3.33(10^{-12})m_1m_5 - 0.0152 = 0 \\
&2.89(10^{-8})m_1 + 1.17(10^{-7})m_2 + 3.33(10^{-12})m_1m_4 \\
&\quad - 6.67(10^{-12})m_2m_5 + 3.33(10^{-12})m_3m_6 + 0.0268 = 0 \\
&6.67(10^{-8})m_3 + 3.33(10^{-12})m_3m_5 + 3.33(10^{-12})m_2m_6 \\
&\quad + 0.00381 = 0 \\
&1.83(10^{-7})m_4 - 3.33(10^{-12})m_2m_4 + 2.89(10^{-8})m_5 \\
&\quad - 3.33(10^{-12})m_1m_5 - 0.0152 = 0 \\
&2.89(10^{-8})m_4 - 3.33(10^{-12})m_1m_4 - 1.17(10^{-7})m_5 \\
&\quad + 6.67(10^{-12})m_2m_5 - 3.33(10^{-12})m_3m_6 - 0.0268 = 0 \\
&-6.67(10^{-8})m_6 - 3.33(10^{-12})m_3m_5 - 3.33(10^{-12})m_2m_6 \\
&\quad + 0.00381 = 0 \\
&1.83(10^{-7})m_1 - 2.89(10^{-8})m_2 - 1.83(10^{-7})m_4 \\
&\quad - 2.89(10^{-8})m_5 + 0.0304 = 0 \\
&-2.89(10^{-8})m_1 - 1.17(10^{-7})m_2 - 2.89(10^{-8})m_4 \\
&\quad + 1.17(10^{-7})m_5 = 0 \\
&-6.67(10^{-8})m_3 + 6.67(10^{-8})m_6 - 0.00762 = 0
\end{aligned} \tag{40}$$

Because the sum of the forces is constrained to be zero, we know that three of the equations are dependent and can be simply removed. From inspection, the first six equations have degree two, while the last three equations have degree one. To reduce the overall degree of the system, it would be prudent to remove the equations that have degree two, reducing the overall degree to eight. However, to demonstrate the reduction step, all nine equations are retained. We can also see that we could easily scale each equation by 10^8, but we once again allow the algorithms to do the work, and the equations are rescaled in the scaling step.

After reducing the equations and rescaling, we are left with the following set:

$$\begin{aligned}
&0.28m_1' + 1.00m_2' + 1.24m_1'm_4' - 1.50m_2'm_5' + 0.91m_3'm_6' \\
&\quad + 2.11 = 0 \\
&0.86m_3' + 1.12m_3'm_5' + 1.94m_2'm_6' + 0.54 = 0 \\
&-2.05m_1' + 0.29m_2' + 1.27m_2'm_4' + 0.97m_1'm_5' - 1.38 = 0 \\
&-1.25m_3' + 0.52m_6' - 1.56 = 0 \\
&-1.18m_1' - 2.04m_2' + 0.53m_5' + 0.80 = 0 \\
&-1.64m_1' + 0.48m_2' + 0.55m_4' - 2.30 = 0
\end{aligned}$$

$$\mathbf{s} = \begin{bmatrix} 5.09 \\ 5.04 \\ 4.96 \\ 4.58 \\ 4.42 \\ 4.58 \end{bmatrix} \tag{41}$$

where the scaling vector **s** is shown. As we can see, the algorithm reduced what was initially a system of nine equations, with a total degree of 64, to a system of six equations, with a total degree of only 8. In addition, the coefficients of the monomial terms are all near unity, and the system is ready for the continuation solver.

Setting up the solver, we need to pick random constants, and as stated earlier we want p, q, and a to have a magnitude near unity. While **q** can be chosen to be either real or complex, for this example **q** is chosen to be real. The random-number generator produced the following values for Eqs. (30) and (36):

$$\mathbf{p} = \begin{bmatrix} 1.15 + 1.01i \\ 1.19 + 1.05i \\ 1.04 + 1.14i \\ 1.14 + 1.02i \\ 1.19 + 1.13i \\ 1.09 + 1.14i \end{bmatrix} \qquad \mathbf{q} = \begin{bmatrix} 0.864 \\ 0.837 \\ 0.975 \\ 0.970 \\ 0.899 \\ 0.891 \end{bmatrix} \qquad \mathbf{a} = \begin{bmatrix} 1.227 \\ 0.760 \\ 0.990 \\ 1.479 \\ 1.013 \\ 1.046 \\ 1.289 \end{bmatrix} \tag{42}$$

We must now use the projection method to ensure that the system has no solutions at infinity, which is accomplished by forming **L** and **l** from Eq. (30) and then from Eq. (31). Because the equation for $\hat{F}$ is fairly long, it has not been included.

The next step is to create the "simple" equations $\mathbf{g}(\mathbf{x})$, with which the continuation solver starts initially:

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} -0.747 + (0.322 + 2.320i)x_1^2 \\ -0.701 + (0.330 + 2.497i)x_2^2 \\ -0.951 - (0.219 - 2.384i)x_3^2 \\ -0.970 + (1.140 + 1.023i)x_4 \\ -0.899 + (1.195 + 1.128i)x_5 \\ -0.891 + (1.086 + 1.137i)x_6 \end{bmatrix} = 0$$

$$\mathbf{x}|_{\delta=0} = \begin{bmatrix} \pm 0.426 \mp 0.371i \\ \pm 0.397 \mp 0.348i \\ \pm 0.425 \mp 0.466i \\ 0.471 - 0.423i \\ 0.398 - 0.376i \\ 0.392 - 0.410i \end{bmatrix} \tag{43}$$

where the initial solutions were determined from Eq. (37) and are made from all possible combinations [(8)] of the vector shown on the right.

The next step is to create the continuation-Eq. (35) and begin solving. In this example, the step value used was $\Delta\delta = 0.002$. Once again, for brevity, all eight solutions to the continuation equation are not shown. Once all eight solutions have been found, we must change back to the original basis using Eq. (32) and the vector

$$\boldsymbol{l}(\mathbf{x}) = \begin{bmatrix} -0.355 - 0.169i \\ -4.997 \\ 3.108 \\ 0 \\ -0.304 \\ -0.155 \\ 0 \\ -0.355 + 0.169i \end{bmatrix} \tag{44}$$

From Eq. (44) we see that there are two solutions at infinity, because $\boldsymbol{l}(\mathbf{x})$ is zero for solution nos. 4 and 7. The final step is to rescale the variables, resulting in the eight solutions given in the columns of the following matrix:

$$[m_{j1},\ldots,m_{j8}]=\begin{bmatrix} -80600-5380i & -13700 & -84700 & 1.00 & -84700 & -14500 & 1.00 & -80600+5380i \\ 15400-34200i & 62400 & -10000 & -0.247 & -10000 & 32200 & -0.247 & 15400+34200i \\ -57100 & -72200 & -118000 & 1.06i & 3490 & -420000 & -1.06i & -57100 \\ 80600+5380i & 14500 & 84700 & 1.00 & 84700 & 137000 & 1.00 & 80600-5380i \\ 15400-34200i & 32200 & -10000 & 0.247 & -10000 & 62400 & 0.247 & 15400+34200i \\ 57100 & 42000 & -3490 & 1.06i & 118000 & 72200 & 1.06i & 57100 \end{bmatrix} \quad (45)$$

We can see that there are two infinite solutions (nos. 4 and 7), two complex solutions (nos. 1 and 8), and four real solutions. Since we are looking for real solutions, we see that there are *four* viable solutions: four different ways of arranging the dipoles to produce the desired force profile we specified in Eq. (38). Recall that the six rows correspond to the six dipole components: m_x, m_y, and m_z for vehicle no. 2, and m_x, m_y, and m_z for vehicle no. 3.

It is informative to look at the solution paths of the variables, and the path for the second solution for m_1 is shown in Fig. 6. On the left is the scaled/projected solution, so that constants and the solution are of order unity. On the right are the unscaled values of the component. Starting off at $m'_1 = 0.426 - 0.371i$, the scaled component path wanders quite smoothly until it ends up on the real axis at the end. Early in the solution path, the solution varies slowly, which is desirable, because each successive solution is near the last and the probability of path jumping is reduced. Towards the end of the solution path, however, the spacing is increased. While the spacing between the points is not so large as to cause path jumping, it should be noted that the change in δ between points is already very small at 0.002. If the spacing were larger, $\Delta\delta$ would have to be reduced even further. Ideally, we would have a solver that varies $\Delta\delta$ dynamically to prevent large changes in the solution while at the same time keeping the change in δ from being prohibitively small in other places. We can find the change in solution based on the change in δ, by taking the derivative of the solutions with respect to delta and then implementing a rule that a larger derivative would result in a smaller δ. This method has been implemented with some success.

V. Continuation with Respect to Time

Having successfully found every solution at one point in time, our focus now turns to finding the dipole solution at other points in time. Specifically, the dipole solution $\mathbf{m}(t)$ from an initial time to a final time is found:

$$\mathbf{F}(\mathbf{m}(t), t) - \mathbf{f}(t) = 0 \quad \forall\ t \in [t_0, t_1, \ldots, t_f] \qquad (46)$$

Because we are using numeric solvers, there is no way to find a continuous solution, so we will have to find the dipole solution at the discrete points in time and interpolate solutions in-between. One way of finding the solution at every point in time is to repeat the continuation method described above at each point. While this approach would work, it would be time-consuming and neglect important information already gathered, specifically the solution at the previous time step. Since we can make Δt arbitrarily small, we can infer that $\Delta\mathbf{m}$ can also be made small; the solutions at one point in time can be made very near the solutions at the previous point in time:

$$\mathbf{m}\,(t_k) - \mathbf{m}(t_{k-1}) \to 0 \quad \text{as } t_k - t_{k-1} \to 0 \qquad (47)$$

Newton's method could just use the old solution to seed the numeric solver. Essentially, what we are doing is creating a continuation equation that instead of varying in δ now varies in t.

While this approach seems very straightforward, we can use the knowledge about continuation equations to see the pitfalls we will encounter. In the continuation method we used a specific continuation equation that included independent constants, so that there would not be any singularities in the continuation method. Remember, at singularities, solution paths cross and cause trouble for Newton's method. Our continuation equation no longer has those independent equations, and the solutions paths will cross should the equations become singular. Another difficulty is that we scaled and reduced the equations based on these constants. The constants are based on the geometry of the formation, because they contain the separation distances of the vehicles, but now the geometry of the formation is changing. We could rescale the equation initially, carrying this scaling through in time, but if the geometry changes significantly the scaling could be useless. On the other hand, rescaling at every point in time could be prohibitive in terms of processing. One could also imagine that the geometry may change, such that the equations would reduce in degree: for example, if the spacecraft were to become coplanar or colinear. In this case, a solution could move to infinity, or move from infinity, if the vehicles moved into a more complex geometry. Since these are the actual solutions, we cannot use projective representations to get around these solutions at infinity.

A. Addressing Singularities in the Solution Path

As stated above, because the continuation equation does not contain independent constants, it is quite possible that the solution has singular points at some point in time along the trajectory. Since a singularity only happens at distinct points in time, and we are only solving at discrete points in time, it is possible that the singularity may also lie between two points in time for which we are solving. Even though we do not solve at the singularity, it will still cause difficulty for the continuation solver. Because we are working with real coefficients, every time a solution passes into the complex plane there is a singularity, and vice versa. These complex solutions must come in pairs (with their conjugates), so that there will always be two solution paths that intersect at the singularity point. Even though we may not be sampling at the singularity, the paths will be close

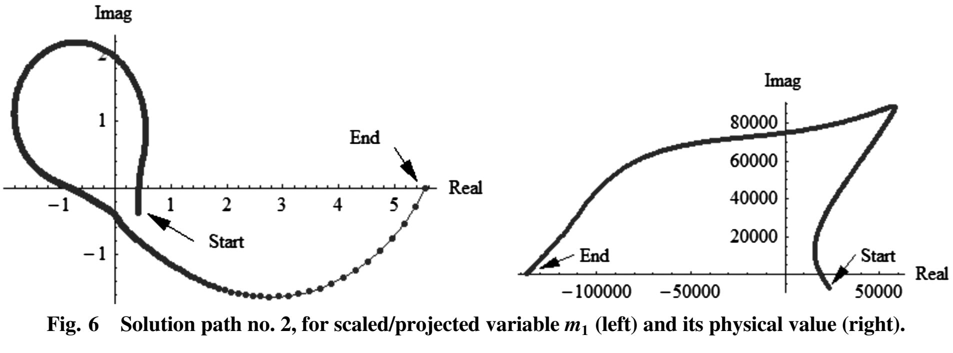

Fig. 6 Solution path no. 2, for scaled/projected variable m_1 (left) and its physical value (right).

together, and path jumping can easily occur. One method to get around this problem is to detect the singularity, and at the next time step re-solve for all of the points using the original continuation method with an updated scaling. Essentially, start over at the new time step. However, because two of the solutions are very close to each other, path crossing can still easily happen as $\delta \to 1$.

Another more efficient, albeit less elegant, solution, is to first detect for path jumping; are there two distinct solutions after the singularity? If so, then we need not do anything. If there is path jumping and both solutions are the same, then we must determine if the solution went from real to imaginary, or vice versa. If the solution is complex after the singularity, then the solution to our problem is easy. We can easily recover the second solution, because it is just the complex conjugate of the first solution. If the solution is real on the other side of the singularity, it is less straightforward to find the second solution. Since Newton's method finds the closest solution to the initial guess, all one has to do is seed Newton's method appropriately. We know the solution must lie near the singularity, so if we seed Newton's method with random small deviations from the singular point, chances are we will recover the solution after only a few attempts. If all else fails, one could move away from the singularity a few time steps, re-solve for every solution point, and work backwards to the singularity.

B. Continuing the Current Example

Expanding on the current example, the spacecraft will start in the previous configuration, and we will assume that the vehicles are already initially rotating in their plane with an angular velocity of one revolution every 30 min. We will also assume that the plane formed by the spacecraft also rotates about the x-axis at a rate of one revolution every 4 h, such as might be expected from a formation acting as an interferometer. The spacecraft rotate in their physical plane to fill the ultraviolet plane (spatial wavelengths), while the rotation about the x-axis can be thought of as a repointing maneuver. The free dipole in this example will be chosen such that it always lies in the plane of the vehicles and is normal to the line that connects it and the center of mass of the formation. Initially, the free dipole has a magnitude of $40,000$ A-m^2, but we will vary that later on in the example.

Conveniently, the solution for the previous example is the initial solution for this example. Thus, we already have the initial solutions to input into the solver.

Figure 7 (left) shows the real solutions to m_1 (the x component of the magnetic dipole on vehicle no. 2), as found initially in the analysis above. As we can see, all four of the solution paths behave very well, meaning that there appear to be no singularities or path jumping. While some of the paths in this plot do cross, it is the n-dimensional paths that must cross for a true path crossing to take place, and so they must cross in all of the variables. We can tell that no paths cross, because there are no singularity points where the real solutions appear or disappear. We can also look at the matrix condition of the Jacobian (Fig. 7, right) of the equations and see that all are well-behaved, meaning that there are no singular points.

Our free dipole has a magnitude of $40,000$ A-m^2, and as seen in Fig. 7, the magnitude of m_1 ranges from $50,000$–$150,000$ A-m^2. It is logical to try to balance the magnitudes by increasing the magnitude of our free dipole. Figure 8 shows the solution when the magnitude of the free dipole is increased to just $42,000$ A-m^2.

In this plot, the solution path still remains continuous, but there are points where the matrix-condition number is rapidly increasing. It should be noted that the points always increase in pairs, which is due to the fact that at a singularity the matrix condition is infinite, and two paths must cross. Therefore, the matrix-condition number for two paths must reach infinity at the same time.

As we continue to increase the free dipole to $50,000$ A-m^2 (Fig. 9), we see that the paths are no longer continuous, with gaps in the solutions. Singular points occur where the paths meet, and when the paths disappear the solutions become complex. It is also interesting to note that there are always at least two real solutions to the problem.

Finally, increasing the free dipole strength to $60,000$ A-m^2 (Fig. 10), we see the solutions become continuous again, but this time there are only two real solutions. The condition number of the matrix remains quite low, indicating that these solutions appear to be relatively far from singular points. Compared with the other cases with lower free-dipole magnitudes (Figs. 7–9), the solutions of the other dipoles in Fig. 10 can also be seen to be very similar in magnitude to that of the free dipole. This may indicate that balancing the dipoles across the vehicles is more desirable for providing well-behaved solution paths. However, the condition numbers in Fig. 10

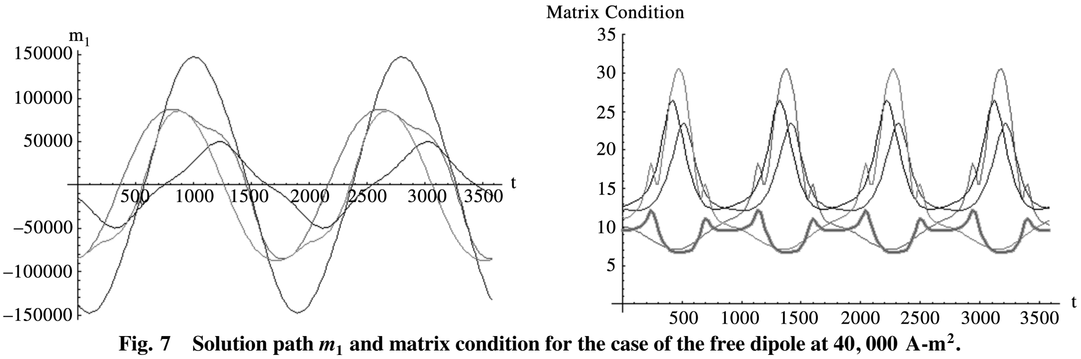

Fig. 7 Solution path m_1 and matrix condition for the case of the free dipole at $40,000$ A-m^2.

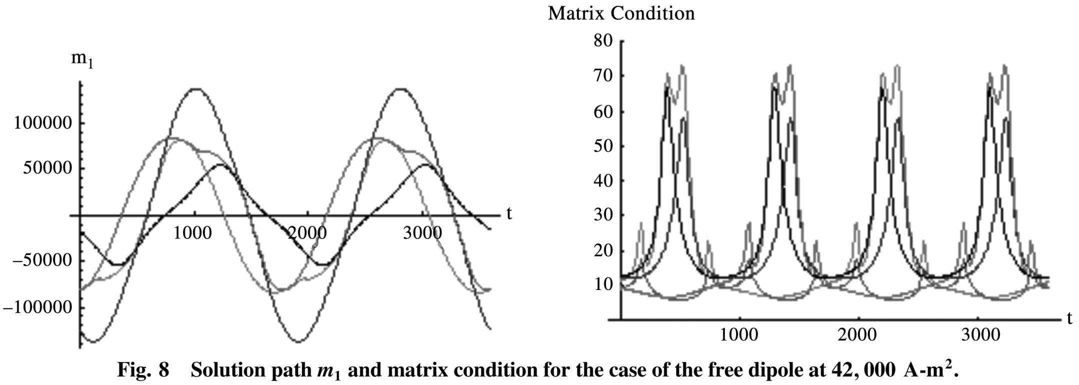

Fig. 8 Solution path m_1 and matrix condition for the case of the free dipole at $42,000$ A-m^2.

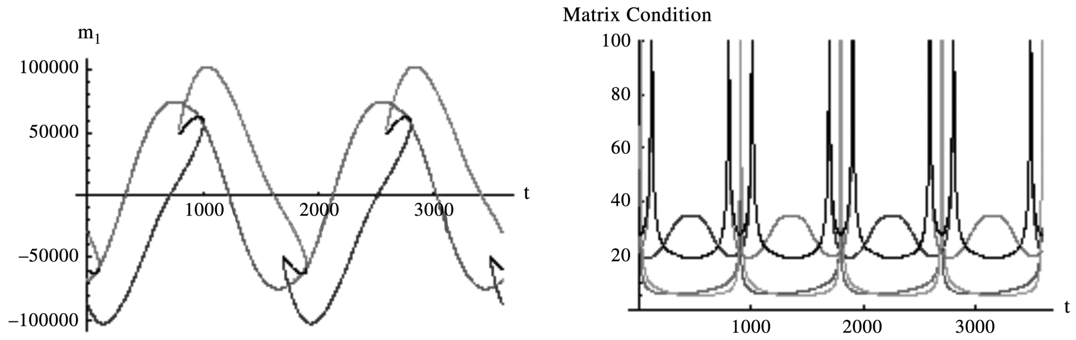

Fig. 9 Solution path m_1 and matrix condition for the case of the free dipole at $50,000$ A-m^2.

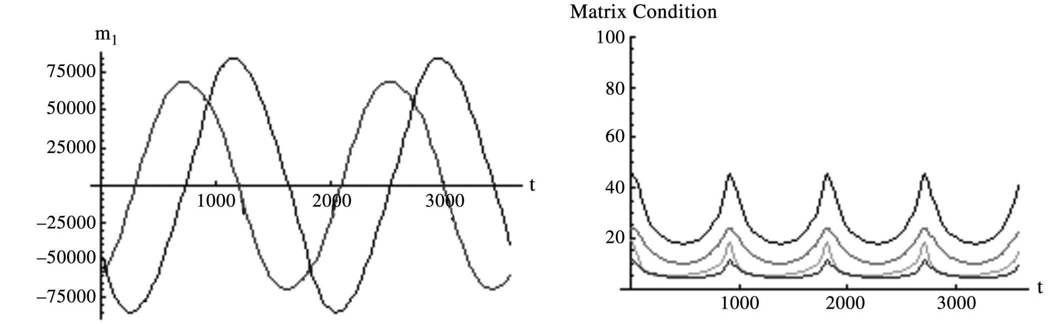

Fig. 10 Solution path m_1 and matrix condition for the case of the free dipole at $60,000$ A-m^2.

are not substantially lower than in Fig. 7, so even if it is found that balancing dipole strengths is sufficient to find well-behaved solutions (which cannot be claimed at this point), it does not appear to be necessary.

Of the examples shown, the case where the magnitude of the free dipole is set to $42,000$ A-m^2 (Fig. 8) appears to be the most problematic, in terms of matrix conditioning during solution. This is seen to correspond to transitions between real and complex solutions. In Figs. 7 and 8, all four solutions are seen to be real, and in Fig. 10 two of the solutions apparently remain complex over the entire path. Only in Fig. 9 does a transition occur. Because of the nonlinear nature of the problem, it is difficult, and perhaps not possible, to predict if and when such transitions will occur or to know how they can be avoided.

If a solution to a particular maneuver is found using an optimal-control approach, the solution can be analyzed using the preceding method to investigate its optimality. Given the paths that result from the optimization, and choosing one of the vehicles as being the free dipole, all of the other dipole solutions that would produce the given trajectory can be found and compared with the one that was chosen. This can be done by choosing each of the vehicles in succession to be the free dipole. In the current example, there would be up to three alternative solutions, each resulting in a different value of the cost function. While a lower cost function will not necessarily correspond to a global minimum, it at least represents an improvement.

The solutions generated by this method, when applied to an undisturbed formation, provide the feedforward control that will cause the formation to follow the desired trajectory. In this case, the result is that the three spacecraft perform a 30 min rotation around the normal to their instantaneous plane of motion, while simultaneously rotating the normal of this plane around the x-axis over a 4 h period. The highly nonlinear force equations are naturally unstable to perturbations, and as such will need an additional feedback-control law to maintain stability over the course of the maneuver. The continuation method discussed will allow for new solutions to be rapidly found, in the vicinity of the previous solutions, that will be self-consistent with the desired trajectory as the feedback controller requests new forces to be applied.

VI. Conclusions

In this paper, a method for determining the desired magnetic-dipole strengths for each spacecraft in an electromagnetically controlled spacecraft cluster was presented. Determining the dipole strengths was accomplished by first realizing that the governing equations comprise a system of bilinear polynomials, which allowed the use of algorithms specifically developed to solve such equations. This also allowed insight to be gained, concerning the solution paths and difficulties encountered when solving for the dipole trajectories. For instance, the example problem could have up to four solutions, depending on the chosen magnitude of the free dipole, but in some cases a particular solution could disappear before the maneuver is completed. Choosing a free-dipole magnitude that matched those of the other vehicles appeared to be beneficial; however, the chosen example also had a highly symmetric nature, and in general this approach may not necessarily be best. Newton's method is well-suited for finding particular solutions to the governing equations, but the continuation method is a powerful tool when all of the solutions are necessary. After finding the solutions at one point in time, the continuation method was extended to solve for all points in time.

References

[1] Kong, E. M. C., Kwon, D. W., Schweighart, S. A., Elias, L. M., Sedwick, R. J., and Miller, D. W., "Electromagnetic Formation Flight for Multi-Satellite Arrays," *Journal of Spacecraft and Rockets*, Vol. 41, No. 4, 2004, pp. 659–666.
doi:10.2514/1.2172

[2] Elias, L. M., Kong, E. M., and Miller, D. W., "An Investigation of Electromagnetic Control for Formation Flight Applications," *Proceedings of the SPIE*, Vol.4849, Aug. 2002, pp. 166-180.

[3] Hashimoto, T., Sakai, S., Ninomiya, K., Maeda, K., and Saitoh, T., "Formation Flight Control Using Super-Conducting Magnets," *International Symposium on Formation Flying Missions and Technologies*, European Space Agency and the European Space Research and Technology Center, Toulouse, France, 2002.

[4] Kwon, D. W., Neave, M., Lee, S., Ramirez, J., and Sedwick, R. J., "Development and Test of a High Temperature Superconducting Testbed for Electromagnetic Formation Flight," *Journal of Spacecraft and Rockets* (submitted for publication).

[5] Kwon, D. W., and Sedwick, R. J., "Cryogenic Heat Pipe for Cooling High Temperature Superconductors," *Cryogenics* (to be published).

[6] Kwon, D. W., and Sedwick, R. J., "Cryogenic Heat Pipe for Cooling High Temperature Superconducting Coils," *Journal of Thermophysics and Heat Transfer*, Vol. 23, No. 4, 2009, pp. 732–740.

doi:10.2514/1.43728.

[7] Ahsun, U., “Dynamics and Control of Electromagnetic Satellite Formations in Low Earth Orbits,” AIAA Guidance, Navigation, and Control Conference, AIAA Paper 2006-6590, 2006.

[8] Ahsun, U., Miller, D. W., and Ahmed, S., “A Hybrid Systems Approach to Closed-Loop Navigation of Electromagnetically Actuated Satellite Formations Using Potential Functions,” *Proceedings of the 17th World Congress, IFAC*, International Federation of Automatic Control, Seoul, Korea, 2008, pp. 6810–6814.

[9] Ahsun, U., Rodgers, L., and Miller, D. W., “Comparison Between Structurally Connected, Propellant Formation Flying and Electromagnetic Formation Flying Spacecraft Configurations for Gen-X Mission,” *Proceedings of SPIE*, Vol. 5899, 2005, pp. 209-220.

[10] Kwon, D., Miller, D. W., and Sedwick, R. J., “Electromagnetic Formation Flight for Sparse Aperture Arrays,” *Second International Symposium on Formation Flight Missions and Technologies*, ESA and the European Space Research and Technology Center, Noordwijk, The Netherlands, 2004.

[11] Wang, S., and Schaub, H., “Spacecraft Collision Avoidance Using Coulomb Forces with Separation Distance Feedback,” *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 3, 2008, pp. 740–750.
doi:10.2514/1.29634

[12] Morgan, A., *Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems*, Prentice–Hall, Englewood Cliffs, NJ, 1987.