Combining Eqs. (22) and (24), we obtain

$$\begin{bmatrix} \beta_1 & -\beta_0 & -\beta_3 & \beta_2 \\ \beta_2 & \beta_3 & -\beta_0 & -\beta_1 \\ \beta_3 & -\beta_2 & \beta_1 & -\beta_0 \\ \beta_0 & \beta_1 & \beta_2 & \beta_3 \end{bmatrix} \delta\gamma(t) = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ C_a - C \end{Bmatrix} \tag{25}$$

Since the coefficient matrix is orthogonal, the solution to the above equations can be written as

$$\delta\gamma(t) = (C_a - C)\beta(t) = \alpha_a \beta(t) \tag{26}$$

where

$$\alpha_a = C_a - C$$

Hence we have the family of admissible attitude costates given by

$$\gamma_a(t) = \gamma(t) + \alpha_a \beta(t) \tag{27}$$

It is seen from Eqs. (27) and (20) that $\gamma(t)$ is never zero.

Of particular interest is the solution with $\gamma_m(t)$ such that $\|\gamma_m^T \gamma_m\| = B_m^2$ is a minimum.

In order to perform a minimization, we form the inner product

$$\gamma_a^T(t)\gamma_a(t) = \gamma^T(t)\gamma(t) + 2\alpha_a \gamma^T \beta(t) + \alpha_a^2$$
$$= B^2 + 2\alpha_a C + \alpha_a^2 \tag{28}$$

Since $\alpha_a$ is the parameter in Eq. (28), it is easy to see by differentiation that $\alpha_m$ corresponding to the minimum norm solution is given by

$$\alpha_m = -C \tag{29}$$

and, correspondingly, the inner product $\gamma_m^T \gamma_m$ is given by

$$B_m^2 = \gamma_m^T(t)\gamma_m(t) = B^2 - 2C^2 + C^2 = B^2 - C^2 \tag{30}$$

For the minimum norm solution

$$B_m^2 = B_m^2 - C_m^2$$

or

$$C_m = \beta^T \gamma_m = 0 \tag{31}$$

The algorithm of Junkins and and Turner[1] can be simplified if the constraint equation (31) is used instead of rigorous minimization of $\gamma^T \gamma$.

## Conclusions

It has been shown that imposing an orthogonality constraint between the Euler parameter vector and the corresponding costate vector is equivalent to minimizing the norm of the Euler parameter costate vector. The normalized Euler parameter costates are shown to be Euler parameters, describing the body orientation with respect to a different inertial frame.

The results presented are purely kinematic. They hold for optimal maneuvers for which motion of the reference body axis, with respect to an inertial frame, is decribed by Euler parameters, including certain cases of flexible bodies, multiple rigid bodies, etc. The results neither depend on the form of Euler's equations nor on the performance index, as long as the Euler parameters do not appear in them.

## References

[1] Junkins, J.L. and Turner, J.D., "Optimal Continuous Torque Attitude Maneuers," *Journal of Guidance and Control*, Vol. 3, May-June 1980, pp. 210-217.

[2] Morton, H.S. Jr., Junkins, J.L., and Blanton, J.N., "Analytical Solutions for Euler Parameters," *Celestial Mechanics*, Vol. 10, No. 3, 1974, pp. 287-301.

[3] Whittaker, E.T., *A Treatise on the Analytical Dynamics of Particles and Rigid Bodies*, Dover, New York, 1944, pp. 10-11.

[4] Kirk, D.E., *Optimal Control Theory, An Introduction*, Prentice Hall, Englewood Cliffs, N.J., 1970.

# An Exponentially Fitted Adams Method of Numerical Integration

Robert A. LaFarge*
*Sandia National Laboratories, Albuquerque, New Mexico*
and
Linda R. Petzold†
*Sandia National Laboratories, Livermore, California*

## Nomenclature

$a_0, \ldots, a_k$ = constants of the approximating polynomial (in time) in conventional Adams methoda

$C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7$, = constants of approximating function in EFAM

DPE = drift probable error, equal to 0.6745 of the standard deviation in drift

$K_1, K_2, K_3, K_4$ = complex constants representing the initial magnitude and orientation of the nutation, precession, trim, and yaw of repose vectors in the complex angle-of-atack plane for a nonrolling coordinate system

$\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3, \mathcal{K}_4$ = complex constants representing the initial magnitude and orientation of the nutation, precession, trim angle, and yaw of repose vectors in the complex angle-of-attack plane for a body-fixed coordinate system

Presented as Paper 82-1342 at the AIAA 9th Atmospheric Flight Mechanics Conference, San Diego, Calif., Aug. 9-11, 1982; submitted Aug. 19, 1982; revision received July 11, 1983. 

*Member of Technical Staff, Aeroballistics Division. Member AIAA.

†Member of Technical Staff, Applied Mathematics Division. Member AIAA.

$p$ = roll rate
$q$ = pitch rate
RPE = range probable error, equal to 0.6745 of the standard deviation in range
$\bar{\alpha},\bar{\beta}$ = nonrolling angles of attack and sideslip, respectively
$\delta_\alpha$ = principal axis misalignment angle
$\lambda_1,\lambda_2$ = damping rates of the nonrolling nutation and precession vectors, respectively
$\bar{\xi}$ = complex nonrolling total angle of attack
$\xi$ = complex body-fixed total angle of attack
$\omega_1,\omega_2$ = frequencies of the nonrolling nutation and precession vectors, respectively
$\omega$ = frequency of the solution assumed by EFAM

## Introduction

EXPONENTIALLY fitted Adams methods can be used in numerical integrators to reduce the computational times required to solve systems of differential equations dominated by high frequencies. These methods take advantage of knowledge of the dominance of high frequencies by incorporating them into the approximating function. EFAM (exponentially fitted Adams methods), a numerical integrator using these methods, was written into two trajectory codes: SANDSHELL,[1] a six degree-of-freedom (6-DOF) computer program for artillery shells, and VIM (vibrating internal member), an 8-DOF extension of SANDSHELL. Solutions to trajectories involving principal axis misalignment effects (SANDSHELL) or nonrigid body effects (VIM) were computed. The solutions are dominated by frequencies at least as high as the roll rate. The roll rate can be as high as 270 rev/s in a typical run. Compared to ODE,[2] the state-of-the-art library integrator previously used, the computational times are reduced by as much as 30%.

To help understand how EFAM works on these shell trajectories, we will briefly discuss the dominance of the roll rate in the solutions. We will also show how a conventional Adams method integrator is constructed and now EFAM differs from it. Finally, we will present results demonstrating EFAM's shorter computation times and discuss how EFAM can be used with other problems.

## Numerical Dominance of Roll Rate

In order to demonstrate how EFAM speeds up the computations, we must show why this is such an expensive computational problem. The nature of the problem, i.e., either mass asymmetries or a vibrating internal member, dictates that a point mass model cannot be used. In fact, both mass asymmetries and vibrating internal members are body-related problems. This means that the roll rate $p$ is part of the solution. The quasisteady solution extended by Hodapp[3] further illustrates this point.

Hodapp[3] extended the classical quasisteady solution of the pitching yawing motion to include a slightly asymmetric shell. We can use his solution to gain insight into the behavior of the problem. His solution for a nonrolling coordinate system is given by

$$\bar{\xi}=\bar{\beta}+i\bar{\alpha}=K_1e^{(\lambda_1+i\omega_1)t}+K_2e^{(\lambda_2+i\omega_2)t}+K_3e^{ipt}+K_4$$

where $K_1$ and $K_2$ describe the initial magnitude and orientation of the transient nutation and precession vectors, $K_3$ represents the trim angle, and $K_4$ represents the yaw of repose. The nutation frequency $\omega_1$ is typically one-tenth the roll rate, while the precession frequency $\omega_2$ is typically one-hundredth the roll rate.

Since SANDSHELL uses a body-fixed system and since EFAM was designed as a replacement for the previous integrator, we need to study the trajectory solutions in a body-fixed system. Hodapp's solution can be converted to a body-fixed system by multiplying the right-hand side by $e^{-ipt}$. This results in

$$\xi=\beta+i\alpha=\mathcal{K}_1e^{[\lambda_1+i(\omega_1-p)]t}+\mathcal{K}_2e^{[\lambda_2+i(\omega_2-p)]t}+\mathcal{K}_3+\mathcal{K}_4e^{-ipt}$$

As can be seen from the above equation, the numerical model tracks the position a projectile takes as it spins about its axis at frequencies of $p$, $0.9p$, and $0.99p$. Thousands of revolutions are completed during a typical trajectory. Therefore, the numerical model gives rise to a system of differential equations where some components are oscillating at a very rapid rate while others (such as altitude) are changing relatively slowly. Whatever integrator is used must solve the differential equations for thousands of these revolutions.

## Adams Methods

A conventional general-purpose code for integrating differential equations, such as ODE, which is a variable-step and variable-order Adams code, can be very inefficient in this situation. To understand why, we must better understand how these types of methods work.

Adams methods are designed to solve a system of differential equations given by

$$\bar{y}'=\bar{f}(\bar{y},t) \qquad \bar{y}(t_0)=\bar{y}_0$$

A common technique for solving such a system is an Adams predictor/corrector scheme. An explicit Adams-Bashforth formula, given by

$$y_n^{(0)}=y_{n-1}+h\sum_{i=1}^{k}\beta_{k_i}f_{n-1}$$

$$f_n=f(y_n,t)$$

is used as the predictor formula. Here $y_n$ is the numerical solution at time $t_n$ and $h=t_n-t_{n-1}$ is the mesh spacing.

The coefficients $\{\beta_{k_i},i=1,2,\ldots,k\}$ of this method are chosen so that the method is *exact* on a set of basis functions $\{1,t\ldots,t^k\}$. This means that whenever the above formula is applied to a differential equation whose solution is a linear combination of the basis functions, i.e., $a_0+a_1t\ldots+a_kt^k$, the method produces the exact solution at the points $t_n=t_0+nh$, $n=1,2\ldots$ .

After the predictor computes an initial guess, a corrector method is used. In this case, we can use a $k^{th}$ order Adams-Moulton method given by

$$y_n=y_{n-1}+h\sum_{i=0}^{k-1}\beta^*k_i\,f_{n-i}$$

This is an implicit method since $f_n$ appears on the right-hand side of the equation. We can solve the above equation for $y_n$ by using functional iteration with $y_n^{(0)}$ as the initial guess; i.e., the next guess of $y_n$ is given by

$$y_n^{(m)}=y_{n-1}+h\sum_{i=1}^{k-1}\beta^*k_if_{n-i}+h\beta_{k_0}f(y_n^{(m-1)},t_n)$$

**Table 1 Comparison of EFAM and ODE Using SANDSHELL[a]**

| Method | $V_0$ | $q_0$ | $R$ | $D$ | $t_f$ | $t_{comp}$ |
|---|---|---|---|---|---|---|
| EFAM | 2710.0 | 0 | 23,974 | 637 | 81.734 | 880 |
| | | 1 | 23,966 | 638 | 81.702 | |
| EFAM | 1060.6 | 0 | 28,171 | 464 | 43.604 | 218 |
| | | 1 | 27,998 | 467 | 43.486 | |
| ODE | 2710.0 | 0 | 23,979 | 636 | 81.760 | 1297 |
| | | 1 | 23,972 | 637 | 81.727 | |
| ODE | 1060.6 | 0 | 28,173 | 464 | 43.608 | 307 |
| | | 1 | 28,000 | 467 | 43.491 | |

[a] $V_0$ = velocity, ft/s; $R$ = range computed by the code, ft; $D$ = drift computed by the code, ft; $t_f$ = final time to impact, s; $t_{comp}$ = computation time (seconds on CDC 7600) (this is the combined time for computing the trajectory with $q_0 = 0$ and then with $q_0 = 1$); $q_0$ = initial pitch rate, rad/s.

**Table 2 Comparison of EFAM and ODE Using VIM[a]**

| $m_2/m$ | $X_{20}/d$ | $p_c/p_0$ | $q_0$ | $R$ | $D$ | $t_f$ | $t_{comp}$ |
|---|---|---|---|---|---|---|---|
| EFAM | | | | | | | |
| 0.075 | 0.0 | 1.5 | 0.0 | 23,979.8 | 625.2 | 81.746 | |
| 0.075 | 0.0 | 1.5 | 1.0 | 23,972.5 | 626.3 | 81.714 | 2117 |
| 0.075 | 2.0 | 1.5 | 0.0 | 23,977.1 | 667.0 | 81.806 | |
| 0.075 | 2.0 | 1.5 | 1.0 | 23,987.5 | 667.9 | 81.813 | 2124 |
| 0.025 | 0.0 | 1.5 | 0.0 | 23,980.1 | 625.4 | 81.755 | |
| 0.025 | 0.0 | 1.5 | 1.0 | 23,972.9 | 626.5 | 81.723 | 2000 |
| ODE | | | | | | | |
| 0.075 | 0.0 | 1.5 | 0.0 | 23,978.0 | 637.5 | 81.742 | |
| 0.075 | 0.0 | 1.5 | 1.0 | 23,970.7 | 637.9 | 81.710 | 2502 |
| 0.075 | 2.0 | 1.5 | 0.0 | 23,975.6 | 678.5 | 81.803 | |
| 0.075 | 2.0 | 1.5 | 1.0 | 23,985.6 | 680.3 | 81.809 | 2608 |
| 0.025 | 0.0 | 1.5 | 1.0 | 23,978.6 | 637.1 | 81.751 | |
| 0.025 | 0.0 | 1.5 | 1.0 | 23,971.93 | 638.4 | 81.718 | 2475 |

[a] $m_2$ = mass of vibrating member, slugs; $m$ = total mass of system, slugs; $X_{20}$ = difference in c.g. locations between the two bodies, ft; $d$ = diameter of shell, ft; $p_c = P_{crit}$; $p_0$ = initial roll rate, rad/s; $R$ = range computed by the code, ft; $D$ = drift computed by the code, ft; $t_f$ = total flight time, s; $t_{comp}$ = computation time, s; initial velocity = 2710.0 ft/s for all problems.

Again, the coefficients $(\beta^*_{k_i} i=0,1...k-1)$ are chosen so that the above formula is exact on the basis set $\{1,t,...t^k\}$.

Codes based on Adams' methods have been very successful in solving a wide range of nonstiff problems, and they are especially competitive when function evaluations are at least moderately expensive. In fact, when ODE was first incorporated into SANDSHELL, it reduced the computation time by two-thirds. However, if a solution is oscillating, ODE must take a time step that is much smaller than the period of oscillation because it approximates the solution by polynomials that are accurate over only very short time intervals where the solution is changing very rapidly. Thus, the method must take many time steps, sometimes millions, for a full trajectory. EFAM was designed to reduce the number of times steps and thereby reduce the computation time.

## Construction on EFAM

The exponentially fitted Adams methods in EFAM are designed to take advantage of the knowledge that some components of the solution are oscillating with frequencies equal, or nearly equal, to the roll rate. Instead of using only the basis functions $\{1,t,t^2...,t^k\}$, these methods use trignometric functions such as sin $(\omega t)$, cos $(\omega t)$, etc. In fact, the basis functions of EFAM are $\{1,t,t^2,t^3 \sin(\omega t), \cos(\omega t)\}$. This means that EFFAM approximates the solution by

$$F(t) = C_0 + C_1 t + C_2 t^2 + C_3 t^3 + C_4 \sin(\omega t) + C_5 \cos(\omega t) + C_6 t \sin(\omega t) + C_7 t \cos(\omega t)$$

*where* $\omega$ is set to the roll rate.

The basis functions $\{1,t,t^2,t^3\}$ are chosen to model the nonoscillatory components. The basis functions $[\sin \omega t]$ and $[\cos \omega t]$ are chosen because they work well if the frequency of the oscillation is known accurately. The basis functions $[t\sin\omega t]$ and $[t\cos\omega t]$ are chosen because they are less sensitive to our exact knowledge of frequency. We chose not to add more basis functions for two reasons: 1) each additional basis function requires an additional past value of $y$ to be carried which can be inefficient in terms of storage and overhead; and 2) as more basis functions are added, the method becomes less and less stable. We arrived at our choice of basis functions after careful experimentation.

Although these types of methods have been investigated before, EFAM is the first such method tailored to an artillery shell. Skelboe[4] describes an automatic code that implements these methods, and EFAM uses some of his idea. However, there are some important differences between his codes and EFAM. EFAM uses a fixed order (seventh) and selects the step size automatically to control the local truncation error of the method. We have found that the step sizes are nearly constant over large segments of the problem. This is not unexpected since the physical problem is nearly constant over several oscillations.

## Results from SANDSHELL

Table 1 compares the solution and computation times obtained by using EFAM and ODE in SANDSHELL. The trajectories were calculated for a 155 mm shell. They represent the trajectories we typically run. An initial velocity of 2710 ft/s ($V_0=2710$) represents the highest roll rate we usually run. The problem has been modeled without an initial disturbance ($q_0=0$) and with an initial disturbance ($q_0=1$). The main variables of interest are the range $R$ and drift ($D$). The results for range and drift are essentially the same no matter which integration is used, but the computation time is significantly reduced by using the exponentially fitted methods in EFAM. These results are representative of the results for other problems of this type. In this problem, the code was asked to be locally accurate to a tolerance of $10^{-6}$, where the error is measured as a combination of relative and absolute error. ODE was under a similar constraint.

## Results from VIM

As mentioned earlier, VIM is an 8-DOF extension of SANDSHELL used to model a shell with one internal member free to vibrate. The problem in VIM is not only complicated by the additional degrees of freedom but also by the additional frequencies $p+p_{\text{crit}}$ and $p-p_{\text{crit}}$, where $p_{\text{crit}}$ is the natural frequency of the vibrating member.[5] We cannot fit the method to all the different frequencies ($p$, $p+p_{\text{crit}}$, and $p-p_{\text{crit}}$) because of stability problems, so we have chosen to fit it to the highest frequency ($p+p_{\text{crit}}$) because it is the frequency that most severely limits the step size. So, instead of setting $\omega=p$ as in SANDSHELL, we set $\omega=p+p_{\text{crit}}$ in our approximating function. The relative efficiency of EFAM over ODE is smaller for this problem because there are so many high frequencies, i.e., $p$, $p+p_{\text{crit}}$, $p-p_{\text{crit}}$. Solutions to 8-DOF trajectories have been computed using EFAM and ODE. The results are given in Table 2. Again, these represent a 155 mm projectile with a muzzle velocity of 2710 ft/s. The improvement in the efficiency of EFAM dropped to about 20%.

## Conclusion

EFAM, a numerical integrator that uses exponentially fitted Adams methods, was written to reduce the computational time required to compute the six degree-of-freedom trajectories of spinning artillery projectiles. EFAM takes advantage of our knowledge of the numerical dominance of a high frequency in the solution by incorporating periodic terms in its approximating function. Specifically, these terms are $\sin(\omega t)$, $\cos(\omega t)$, $t\sin(\omega t)$, and $t\cos(\omega t)$. These terms allow the approximating function to model the solution better than the state-of-the-art library routine previously used. The improved approximating function allows a bigger time step for a given error tolerance and thus result in less computational time. The reduction in time can be as much as 30% in SANDSHELL (a six degree-of-freedom computer program for artillery shells) or about 20% in VIM (an eight degree-of-freedom extension of SANDSHELL). The reduced efficiency in VIM can be attributed to the fact that there are three dominant frequencies in VIM ($p$, $p+p_{\text{crit}}$, and $p-p_{\text{crit}}$) rather than only one. However, the savings using EFAM in VIM can be greater since the computation times for a trajectory are greater in VIM than in SANDSHELL.

The use of exponentially fitted Adams methods need not be restricted to artillery projectile problems. EFAM can be used on any vibration problem, particularly one that has a single, high frequency. EFAM can also be used on some orbital problems. If the needed solution has only oscillatory motion, the basis functions in EFAM can be changed to just periodic terms.

## References

[1] LaFarge, R. A., "SANDSHELL: A 6-DOF Trajectory Code for Spinning Artillery Shells," Sandia National Laboratories, Albuquerque, N. Mex., Rept. SAND80-2717, Sept. 1981.

[2] Shampine, L. F. and Gordon, M. F., "Solving Ordinary Differential Equations with ODE, STEP, and INTRP," Sandia National Laboratories, Albuquerque, N. Mex., Rept. SLA-73-1060, Jan. 1974.

[3] Hodapp, A.F., Jr., *Effects of Mass Asymmetry on Ballistic Match of Projectiles,* Sandia National Laboratories, Albuquerque, N. M., SAND 77-0324, July 1977.

[4] Skelboe, S., "Numerical Methods for the Analysis of Mildly Nonlinear Electrical Circuits," Institute for Numerical Analysis, Technical University of Denmark, Rept. 75-08, Oct. 1975.

[5] Hodapp, A. E. Jr. and LaFarge, R. A., "A Ballastic Similitude Design Criterion for Artillery Projectiles," AIAA Paper 82-1344, Aug. 1982.