JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 30, No. 6, November–December 2007

Stabilizing the Explicit Euler Integration of Stiff and Undamped Linear Systems

Pini Gurfil* and Itzik Klein†
Technion—Israel Institute of Technology, 32000 Haifa, Israel

DOI: 10.2514/1.29148

Euler's integration methods are frequently used for numerical integration as well as for real-time implementation of linear systems. However, when the integrated system is either undamped or stiff, Euler's explicit integration becomes unstable, regardless of the forcing input. In this work, it is shown that this instability can be avoided by a judicious selection of the state variables. Instead of the generalized coordinates and velocities, it is proposed to define state variables based on the method of variation of parameters. It is proven that the variation-of-parameters-based state variables yield a bounded numerical error for undamped and stiff systems, provided that the forcing inputs are bounded. The analysis is performed for both deterministic and stochastic inputs. In the stochastic case, the numerically calculated covariance matrix entries diverge when using the generalized coordinates and velocities, but remain bounded when implementing the variation-of-parameters-based approach. The newly developed formalism is illustrated by a number of examples of practical interest, showing that the variation-of-parameters-based approach is also more computationally efficient than the standard approach.

Nomenclature

A	=	system matrix
B	=	input matrix
$\boldsymbol{c}$	=	variation-of-parameters constants
C	=	damping matrix
$\boldsymbol{e}$	=	local integration error
$\boldsymbol{f}$	=	forcing term vector
h	=	time step
K	=	stiffness matrix
M	=	mass (inertia) matrix
Q	=	covariance matrix
$\boldsymbol{q}$	=	generalized coordinates
$\dot{\boldsymbol{q}}$	=	generalized velocities
$\boldsymbol{q}_h$	=	homogenous solution
q_{ij}	=	jth fundamental solution of the ith DOF
V_s	=	fundamental solutions matrix
$\boldsymbol{v}$	=	white noise
w	=	Wronskian determinant
$\boldsymbol{x}$	=	state variables vector
$Z\{\cdot\}$	=	unilateral Z transform
ζ	=	damping coefficient
ω_n	=	natural frequency

Subscripts

$(\cdot)_d$	=	discrete variable/matrix
$(\cdot)_e$	=	explicit Euler integration
$(\cdot)_{\text{vop}}$	=	variation of parameters

I. Introduction

THERE are many important problems in engineering and science that require solving ordinary differential equations (ODEs). The study of differential equations dates back to the dawn of calculus in the 17th century [1]. Over the course of years, numerous analytical techniques for solving differential equations have been developed; however, there are many practical problems on which these analytical methods cannot be applied. It was Euler who first attempted to overcome this situation by solving initial value problems (IVPs) using discretization [2]. Euler's idea was to propagate the solution of an IVP forward by a sequence of small time steps. In each step, the rate of change of the solution is treated as constant and is found from a formula for the derivative evaluated at the beginning of the step. This method is considered to be *explicit*, in the sense that the algorithm for determining an approximation at the end of the step consists of a definite sequence of derivative calculations.

The *implicit* Euler method [3] is usually used for solving stiff problems. In a stiff system of ODEs, some solution components are slowly varying, whereas other components are rapidly decaying, thus forcing a severe stability restriction on the step size used to compute the numerical approximation.

Although the explicit Euler method is of limited accuracy, it is frequently used for numerical integration of linear ODEs emerging in diverse fields such as control and estimation theory [4], dynamics [5], aeroelasticity [6], and computer animation [7]. Furthermore, this method is used for real-time implementation of control algorithms, due to the fact that it introduces minimum delay into the computational cycle. However, despite its simplicity and utility, the explicit Euler method rapidly diverges when applied either to undamped or stiff linear systems. Although this divergence can be avoided by using the implicit Euler method, the latter approach is of limited use in undamped systems, due to the fact that the implicit scheme introduces artificial damping, thus compromising accuracy [3].

Mitigating the deficiency of the explicit Euler method without resorting to the implicit formalism is the main goal of the current paper. In particular, we shall ask the following question: Can the instability of Euler's explicit integration method be cured by a nonorthodox selection of the state variables?

Given a second-order vector ODE, the common approach is to use the generalized coordinates and velocities as state variables to reduce the ODEs into a set of integrable first-order ODEs. This set of variables will be referred to as the *standard* state variables. In this work, we show that the choice of the state variables has a strong impact on the numerical stability of Euler's explicit method. Moreover, we prove that a different selection of state variables can completely eliminate the divergence of Euler's method. To that end, we propose an alternative selection of state variables based on the variation of parameters (VOP).

Received 4 December 2006; revision received 24 April 2007; accepted for publication 24 April 2007.

*Senior Lecturer, Faculty of Aerospace Engineering; pgurfil@technion.ac.il. Associate Fellow AIAA

†Graduate Student, Faculty of Aerospace Engineering; iklein@tx.technion.ac.il. Student Member AIAA.

The VOP method was invented by Euler [8] for solving highly nonlinear problems emerging in celestial mechanics. Lagrange [9] then perfected this method for deriving his system of equations describing the evolution of orbital elements. According to the VOP rationale, the integration constants of the homogeneous solution of a given system of ODEs are endowed with a time variation due to the presence of an external force. The VOP method thus involves a transformation from the standard state variables of the original problem into new state variables, defined as the time-varying constants. Although in the current work, VOP is used to numerically solve inhomogeneous linear differential equations, this method can be applied for solving any nonlinear ODE.

One of the main contributions of this work is proving that replacing the standard state variables by VOP-based state variables stabilizes the explicit Euler method in the sense of bounded-input/bounded-output (BIBO). In the stochastic case, it is proven that the state covariance matrix entries will always be bounded when using VOP-based state variables in undamped and stiff systems, whereas they are bound to diverge with the standard state variables. The improved behavior of the VOP-based state-space model does not compromise the computational efficiency. In fact, we show that the number of floating-point operation counts (FLOPS) is reduced when using the VOP-based state variables (for a given integration error). This renders the VOP-based approach attractive for real-time implementation.

II. Explicit Euler Integration Method

In this section, we briefly outline the numerical integration of linear systems of ODEs by the explicit Euler integration method. Although the Euler integration method can be applied both to linear and nonlinear ODEs, we address linear ODEs only, due to the ubiquity of these ODEs in various engineering and scientific fields and due to the fact that Euler's integration is rarely applied to nonlinear systems.

Problems involving high-order ODEs can often be reduced into a system of first-order ODEs by introducing new state variables; it is this selection of state variables with which the current paper is concerned. Assume, for example, that a given high-order system possesses the following linear time-varying (LTV) first-order state-space representation:

$$\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t) + B(t)\boldsymbol{f}(t), \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{1}$$

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T$, $\dot{\boldsymbol{x}} = \mathrm{d}\boldsymbol{x}/\mathrm{d}t$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $\boldsymbol{f} \in \mathbb{R}^m$, and $\boldsymbol{x}_0$ is the initial condition vector at $t = t_0$, which is required to numerically integrate Eq. (1) using the explicit Euler integration method. This operation transforms Eq. (1) into the discrete LTV form:

$$\boldsymbol{x}(k+1) = A_d(k)\boldsymbol{x}(k) + B_d(k)\boldsymbol{f}(k), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{2}$$

To find the system and input matrices A_d and B_d, respectively, of Eq. (2), recall that the explicit Euler formula is given by [10]

$$\boldsymbol{x}(k+1) = \boldsymbol{x}(k) + h\dot{\boldsymbol{x}}(k) = \boldsymbol{x}(k) + h[A(k)\boldsymbol{x}(k) + B(k)\boldsymbol{f}(k)] \tag{3}$$

where h is the constant time step, k is the step index, and $t_k = t_0 + kh$.

The integration error $\boldsymbol{e}$ can now be defined in the following standard manner. Let $\boldsymbol{x}_a(t_0, \boldsymbol{x}_0, t)$ be a solution of Eq. (1), and suppose that $\boldsymbol{x}(k)$ is the numerical approximation thereof obtained via the explicit method; then

$$\boldsymbol{e}(k) = \boldsymbol{x}_a(t_0, \boldsymbol{x}_0, t = t_k) - \boldsymbol{x}(k) \tag{4}$$

Equation (4) may also be written componentwise, so that the integration error for x_i becomes

$$e_{x_i}(k) = x_{ia}(t_0, \boldsymbol{x}_0, t = t_k) - x_i(k) \tag{5}$$

A closed-form expression for the local integration error $\boldsymbol{e}$ of the explicit Euler formula can be obtained as follows: Define the discrete timescale (wherein the time points are not necessarily equally spaced) $t_k \le t_i^\star \le t_{k+1}$, where $k, i = 0, 1, \cdots$. Then the local error is given by [10]

$$\boldsymbol{e}(t_i^\star) = \frac{h^2}{2}\ddot{\boldsymbol{x}}(t_i^\star) \tag{6}$$

An equivalent expression is

$$\boldsymbol{e}(t_k) = \frac{h^2}{2}\ddot{\boldsymbol{x}}(t_k) + \mathcal{O}(h^3) \tag{7}$$

The third-order terms $\mathcal{O}(h^3)$ will be neglected throughout this study.

In the subsequent discussion, we shall study the stability of Euler's explicit numerical integration of Newtonian n-degree-of-freedom (DOF) linear time-invariant (LTI) systems of the form

$$M\ddot{\boldsymbol{q}}(t) + \tilde{C}\dot{\boldsymbol{q}}(t) + \tilde{K}\boldsymbol{q}(t) = \tilde{\boldsymbol{f}}(t), \qquad \boldsymbol{q}(t_0) = \boldsymbol{q}_0, \qquad \dot{\boldsymbol{q}}(t_0) = \dot{\boldsymbol{q}}_0 \tag{8}$$

where $\boldsymbol{q} = [q_1, q_2, \ldots, q_n]^T \in \mathbb{R}^n$ denotes the *generalized coordinates*; $\dot{\boldsymbol{q}} = \mathrm{d}\boldsymbol{q}/\mathrm{d}t \in \mathbb{R}^n$ are the *generalized velocities*; $M, \tilde{C}, \tilde{K} \in \mathbb{R}^{n\times n}$ are the constant mass (inertia), damping, and stiffness matrices, respectively; $\tilde{\boldsymbol{f}}$: $\mathbb{R}_{\ge 0} \to \mathbb{R}^n$ is a forcing function; and $\boldsymbol{q}_0$ and $\dot{\boldsymbol{q}}_0$ are the initial condition vectors at $t = t_0$. If M is nonsingular, Eq. (8) can be rewritten into

$$\ddot{\boldsymbol{q}}(t) + C\dot{\boldsymbol{q}}(t) + K\boldsymbol{q}(t) = \boldsymbol{f}(t) \tag{9}$$

where

$$C = M^{-1}\tilde{C}, \qquad K = M^{-1}\tilde{K}, \qquad \boldsymbol{f} = M^{-1}\tilde{\boldsymbol{f}} = [f_1, f_2, \cdots, f_n]^T \tag{10}$$

The literature offers a few sufficient conditions [11] guaranteeing the stability of the homogenous solution of system (9). In particular, the following Lemma is a well-known result [12,13]:

Lemma 1: System (9) is stable if $K > 0$ and $C \ge 0$, namely, if K is positive definite and C is positive semidefinite.

We are now ready to discuss how the selection of state variables affects the stability of Euler's explicit integration method.

III. Generalized Coordinates and Velocities as State Variables

Equation (9) can be reduced to a system of first-order ODEs by introducing the standard state variables $\boldsymbol{x}_1 = \boldsymbol{q}$ and $\boldsymbol{x}_2 = \dot{\boldsymbol{q}}$, the generalized coordinates and velocities, respectively, yielding

$$\begin{bmatrix}\dot{\boldsymbol{x}}_1(t)\\ \dot{\boldsymbol{x}}_2(t)\end{bmatrix} = \begin{bmatrix}0 & I_{n\times n}\\ -K & -C\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_1(t)\\ \boldsymbol{x}_2(t)\end{bmatrix} + \begin{bmatrix}0_{n\times n}\\ I_{n\times n}\end{bmatrix}\boldsymbol{f}(t) = A_s\boldsymbol{x}(t) + B_s\boldsymbol{f}(t) \tag{11}$$

In this case, both the system matrix $A = A_s$ and the input matrix $B = B_s$ appearing in the state-space model (1) are *constant*, and hence the resulting discrete time model will be time-invariant. To see this, we apply the explicit Euler formula (3) to Eq. (11), yielding

$$\begin{bmatrix}\boldsymbol{x}_1(k+1)\\ \boldsymbol{x}_2(k+1)\end{bmatrix} = \begin{bmatrix}I_{n\times n} & hI_{n\times n}\\ -hK & I_{n\times n} - hC\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_1(k)\\ \boldsymbol{x}_2(k)\end{bmatrix} + \begin{bmatrix}0_{n\times n}\\ hI_{n\times n}\end{bmatrix}\boldsymbol{f}(k) = A_{de}\boldsymbol{x}(k) + B_{de}\boldsymbol{f}(k) \tag{12}$$

where $A_d = A_{de}$ and $B_d = B_{de}$ are the system and input matrices, respectively, of the discrete state-space representation in Euler's explicit formalism.

When applying the Euler explicit integration method to a state-space model with the standard state variables, the original continuous LTI system (9) is transformed into a discrete linear time-invariant form, for which the stability can be studied using some well-established stability analysis tools [14]. The most straightforward

way to analyze stability is to use the unilateral Z transform, defined as

$$Z\{x(k)\} = X(z) = \sum_{k=0}^{\infty} x(k) z^{-k} \tag{13}$$

and to use its shift property,

$$Z\{x(k+1)\} = zX(z) - zx_0 \tag{14}$$

It was shown by Dahlquist [15] that the set

$$S = \{z \in \mathbb{C}; |R(z)| \leq 1\} \tag{15}$$

constitutes the stability domain of the integration method, where $R(z)$ is the *stability function*. For the explicit Euler method, it can be shown [10] that $R(z) = 1 + z$, yielding a stability domain in which the eigenvalues of the system matrix lie inside a unit circle centered at $z = -1$. To illustrate the instability of the explicit Euler method in either stiff or undamped systems, consider the second-order 1-DOF ODE:

$$\ddot{q}(t) + 2\zeta\omega_n \dot{q}(t) + \omega_n^2 q(t) = f(t), \qquad \dot{q}(0) = \dot{q}_0, \qquad q(0) = q_0 \tag{16}$$

where $\omega_n \neq 0$ is the natural frequency and $0 \leq \zeta \leq 1$ is the damping coefficient. Applying the explicit Euler method (12) yields

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & h \\ -h\omega_n^2 & 1 - 2h\zeta\omega_n \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ h \end{bmatrix} f(k) \tag{17}$$

Next, we perform the Z transform given in Eqs. (13) and (14):

$$\begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} z - \begin{bmatrix} x_{1_0} \\ x_{2_0} \end{bmatrix} z = \begin{bmatrix} 1 & h \\ -h\omega_n^2 & 1 - 2h\zeta\omega_n \end{bmatrix} \begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} + \begin{bmatrix} 0 \\ h \end{bmatrix} F(z) \tag{18}$$

After rearranging, we obtain

$$\begin{bmatrix} z-1 & -h \\ h\omega_n^2 & z - 1 + 2h\zeta\omega_n \end{bmatrix} \begin{bmatrix} X_1(z) \\ X_2(z) \end{bmatrix} = \begin{bmatrix} x_{1_0} \\ x_{2_0} \end{bmatrix} z + \begin{bmatrix} 0 \\ h \end{bmatrix} F(z) \tag{19}$$

The system's eigenvalues are then

$$\lambda_{1,2} = -1 + h\zeta\omega_n \pm \sqrt{h^2\omega_n^2(\zeta^2 - 1)} \tag{20}$$

Dahlquist's test (15) yields two stability conditions: The first, $h < 0$, is infeasible because the time step must be a positive scalar. The second condition is given by

$$h < \frac{2\zeta}{\omega_n} \tag{21}$$

If $\zeta/\omega_n \to 0$, an *infinitely small* time step is required for stability. This implies that the explicit Euler method, implemented using the standard state variables, will *always* be unstable for undamped ($\zeta = 0$) as well as stiff ($\zeta \ll \omega_n$) systems.

By repeating the preceding analysis for the n-DOF case, we get

$$\begin{bmatrix} (z-1)I_{n\times n} & -hI_{n\times n} \\ hK & (z-1)I_{n\times n} + hC \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_1(z) \\ \boldsymbol{X}_2(z) \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_{1_0} \\ \boldsymbol{x}_{2_0} \end{bmatrix} z + \begin{bmatrix} 0_{n\times n} \\ hI_{n\times n} \end{bmatrix} \boldsymbol{F}(z) \tag{22}$$

At this point, observe that A_{de} can be written as

$$A_{de} = \begin{bmatrix} I_{n\times n} & 0 \\ 0 & I_{n\times n} \end{bmatrix} + h \begin{bmatrix} 0_{n\times n} & I_{n\times n} \\ -K & -C \end{bmatrix} \tag{23}$$

with $h \ll 1$. This order-of-magnitude difference permits the use of the approximation [16]

$$|A_{de}| = 1 - h\,\mathrm{trace}(C) + \mathcal{O}(h^2) \tag{24}$$

so that when the damping matrix entries are close to zero, $\{c_{ij}\} \approx 0$, $|A_{de}| \approx 1$. This implies that either all eigenvalues are equal to 1, yielding the stability limit of the system, or the magnitude of at least one eigenvalue is larger than 1, resulting in instability. Therefore, undamped systems *cannot* be numerically integrated using the explicit Euler method when the selected state variables are the generalized coordinates and velocities. In fact, it can be shown that there exists a finite time step for which the Euler method (either the explicit or the implicit) implemented with standard state variables will be stable if K and C are positive definite [7].

Consequently, the explicit Euler method using standard state variables will diverge if the strict positive definiteness constraint on C is violated. Our goal here is to show that this situation can be alleviated by a different selection of state variables. Specifically, we shall allow C to be *positive semidefinite*. This situation arises in systems such as flexible space structures, in which some of the DOFs are weakly damped or undamped. In these systems, the matrix K is usually positive definite, and hence the sufficient conditions guaranteeing stability of the analytical homogenous solution, stated in Lemma 1, are satisfied.

IV. Choosing State Variables Based on Variation of Parameters

We shall now show that in cases wherein Euler's explicit numerical scheme (implemented using the generalized position and velocity state variables) diverges, the numerical integration implemented using VOP-based state variables yields a bounded integration error that can be made arbitrarily small.

A. Variations-of-Parameters Formalism

To illustrate the main idea of the VOP method, we begin with the single-DOF case. We shall then show that this method can be straightforwardly implemented on multiple DOFs as well. To that end, consider the one-dimensional, second-order forced ODE given in Eq. (16). The corresponding homogeneous equation is

$$\ddot{q}(t) + 2\zeta\omega_n \dot{q}(t) + \omega_n^2 q(t) = 0 \tag{25}$$

Assuming an underdamped case ($\zeta < 1$), the homogeneous solution is given by

$$q_h(t) = c_1 q_1(t) + c_2 q_2(t) \tag{26}$$

where $q_1(t)$ and $q_2(t)$ are the fundamental solutions

$$q_1(t) = e^{-\zeta\omega_n t}\cos(\omega_d t) \tag{27a}$$

$$q_2(t) = e^{-\zeta\omega_n t}\sin(\omega_d t) \tag{27b}$$

and $\omega_d = \omega_n\sqrt{1-\zeta^2}$. The main idea of the VOP method is to replace the constants c_1 and c_2 in Eq. (26) with the time-dependent functions $c_1(t)$ and $c_2(t)$, respectively, to solve the inhomogeneous Eq. (16); therefore, Eq. (26) becomes

$$q(t) = c_1(t)q_1(t) + c_2(t)q_2(t) \tag{28}$$

Equation (28) constitutes a general solution candidate to Eq. (16); that is, adding time dependence to the coefficients of the homogenous solution may render it a general solution. Differentiating Eq. (28) yields

$$\dot{q}(t) = \dot{q}_1(t)c_1(t) + \dot{q}_2(t)c_2(t) + \dot{c}_1(t)q_1(t) + \dot{c}_2(t)q_2(t) \tag{29}$$

It is common to use the *Lagrange constraint* [17] (the implications of using a different constraint are discussed elsewhere [18])

$$\dot{c}_1(t)q_1(t) + \dot{c}_2(t)q_2(t) = 0 \tag{30}$$

to facilitate the mathematical derivation and to solve for the excess

freedom introduced by the mapping $(q, \dot{q}) \mapsto (c_1, c_2, \dot{c}_1, \dot{c}_2)$. To continue, we substitute Eq. (30) into Eq. (29), yielding

$$\dot{q}(t) = \dot{q}_1(t)c_1(t) + \dot{q}_2(t)c_2(t) \tag{31}$$

and

$$\ddot{q}(t) = \dot{q}_1(t)\dot{c}_1(t) + \dot{q}_2(t)\dot{c}_2(t) + \ddot{q}_1(t)c_1(t) + \ddot{q}_2(t)c_2(t) \tag{32}$$

Substituting Eqs. (28), (31), and (32) into Eq. (16) and rearranging entails

$$\begin{aligned} c_1(t)[\ddot{q}_1(t) + 2\zeta\omega_n\dot{q}_1(t) + \omega_n^2 q_1(t)] + c_2(t)[\ddot{q}_2(t) + 2\zeta\omega_n\dot{q}_2(t) \\ + \omega_n^2 q_2(t)] + \dot{q}_1(t)\dot{c}_1(t) + \dot{q}_2(t)\dot{c}_2(t) = f(t) \end{aligned} \tag{33}$$

Each of the expressions in square brackets in Eq. (33) equals zero, because both q_1 and q_2 are solutions of the homogeneous Eq. (25). Therefore, Eq. (33) reduces to

$$\dot{q}_1(t)\dot{c}_1(t) + \dot{q}_2(t)\dot{c}_2(t) = f(t) \tag{34}$$

Equations (30) and (34) form a system of two linear algebraic equations for the derivatives $\dot{c}_1(t)$ and $\dot{c}_2(t)$. By solving this system, we obtain

$$\dot{c}_1(t) = \frac{-q_2(t)f(t)}{w[q_1(t), q_2(t)]} \tag{35a}$$

$$\dot{c}_2(t) = \frac{q_1(t)f(t)}{w[q_1(t), q_2(t)]} \tag{35b}$$

where $q_1(t)$ and $q_2(t)$ are given in Eq. (27) and $w[q_1(t), q_2(t)]$ is the Wronskian determinant:

$$w[q_1(t), q_2(t)] = \begin{vmatrix} q_1(t) & q_2(t) \\ \dot{q}_1(t) & \dot{q}_2(t) \end{vmatrix} = \text{const} \neq 0 \quad \forall\ t \tag{36}$$

Division by $w[q_1(t), q_2(t)]$ is allowed because $q_1(t)$ and $q_2(t)$ constitute a fundamental set of solutions and therefore their Wronskian determinant is always nonzero. Thus, we have transformed the second-order ODE (16) into the two first-order ODEs (35). The initial conditions for system (35) are found from Eqs. (28) and (31):

$$q(t_0) = c_1(t_0)q_1(t_0) + c_2(t_0)q_2(t_0) \tag{37a}$$

$$\dot{q}(t_0) = \dot{q}_1(t_0)c_1(t_0) + \dot{q}_2(t_0)c_2(t_0) \tag{37b}$$

Solving Eq. (37) for $c_1(t_0)$ and $c_2(t_0)$ yields

$$c_1(t_0) = \frac{q_0\dot{q}_2(t_0) - q_2(t_0)\dot{q}_0}{w[q_1(t_0), q_2(t_0)]} \qquad c_2(t_0) = \frac{q_1(t_0)\dot{q}_0 - q_0\dot{q}_1(t_0)}{w[q_1(t_0), q_2(t_0)]} \tag{38}$$

Integrating system (35) with these initial conditions will yield the solution $q(t)$, given by Eq. (28). Equation (28) is the solution of the inhomogeneous system (16), comprising the time-dependent functions $c_1(t)$ and $c_2(t)$.

B. Variation of Parameters for *n* Degrees of Freedom

Consider the n-DOF system given by Eq. (9). The solution of the corresponding homogenous equation is

$$\boldsymbol{q}_h(t) = c_1\boldsymbol{q}_1(t) + \cdots + c_{2n}\boldsymbol{q}_{2n}(t) \tag{39}$$

where $c_1, \ldots, c_{2n}$ are constants and $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{2n}$ are the $2n$ linearly independent solutions of the n-DOF homogeneous differential equation. Following the VOP method, we replace the constants $c_1, \ldots, c_{2n}$ in Eq. (39) by the time-dependent functions $c_1(t), \ldots, c_{2n}(t)$, respectively, to solve the forced system; therefore, Eq. (39) becomes

$$\boldsymbol{q}(t) = c_1(t)\boldsymbol{q}_1(t) + \cdots + c_{2n}(t)\boldsymbol{q}_{2n}(t) \tag{40}$$

Let $\boldsymbol{q}_i = [q_{1i}, q_{2i}, \ldots, q_{ni}]^T$ represent a solution of the homogenous equation (i.e., q_{11} refers to the first component of the first solution $\boldsymbol{q}_1$). Consider the matrix V_s, for which the columns are the vectors $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{2n}$:

$$V_s(t) = \begin{bmatrix} q_{1,1}(t) & \cdots & q_{1,2n}(t) \\ \vdots & \ddots & \vdots \\ q_{n,1}(t) & \cdots & q_{n,2n}(t) \end{bmatrix}_{[n\times 2n]} \tag{41}$$

Consequently, the mapping $(\boldsymbol{q}, \dot{\boldsymbol{q}}) \mapsto (c_1, c_2, \ldots, c_{2n}, \dot{c}_1, \dot{c}_2, \ldots, \dot{c}_{2n})$ results in

$$\dot{\boldsymbol{c}}(t)_{[2n\times 1]} = W^{-1}(t)\boldsymbol{\Psi}(t) = B_{\text{vop}}(t)\boldsymbol{\Psi}(t) \tag{42}$$

where $\boldsymbol{c} = [c_1, c_2, \ldots, c_{2n}]^T$ and

$$W(t) = \begin{bmatrix} \dot{V}_s(t) \\ V_s(t) \end{bmatrix}_{[2n\times 2n]}, \qquad \boldsymbol{\Psi}(t) = \begin{bmatrix} \boldsymbol{f}(t) \\ \boldsymbol{0} \end{bmatrix}_{[2n\times 1]} \tag{43}$$

The corresponding initial conditions are found via Eq. (39) and its derivative and the original initial conditions given in Eq. (9). Note that in the VOP formalism, the system matrix $A = 0$ and the input matrix is *time-varying*: $B = B_{\text{vop}}(t)$. Thus, the VOP procedure has transformed the LTI model (11) into a *driftless* (i.e, the linear velocity vector field does not depend on the state vector) LTV model (42). This has been achieved by the following transformation, relating the generalized coordinates and velocities (the standard states) to the VOP state variables:

$$\begin{bmatrix} \boldsymbol{x}_1(t) \\ \boldsymbol{x}_2(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}(t) \\ \dot{\boldsymbol{q}}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_1(t) & \cdots & \boldsymbol{q}_{2n}(t) \\ \vdots & & \vdots \\ \dot{\boldsymbol{q}}_1(t) & \cdots & \dot{\boldsymbol{q}}_{2n}(t) \end{bmatrix} \begin{bmatrix} c_1(t) \\ \vdots \\ c_{2n}(t) \end{bmatrix} \tag{44}$$

Therefore, with $\boldsymbol{x} = \boldsymbol{c}$, the resulting discrete state-space model will assume the form

$$\boldsymbol{x}(k+1) = I\boldsymbol{x}(k) + hB_{\text{vop}}(k)\boldsymbol{f}(k) = I\boldsymbol{x}(k) + \boldsymbol{u}(k) \tag{45}$$

so that $A_d = I$ and $B_d = hB_{\text{vop}}(k)$. The input to the discrete model (45) is $\boldsymbol{u} = [u_1, u_2, \ldots, u_{2n}]^T$.

C. Stability Conditions

The VOP procedure transformed the ODE into a *quadrature* problem. Thus, regardless of the type of Euler method (explicit/implicit) used, the corresponding transition matrix is, by definition, a unit matrix [cf. Eq. (45)]. To determine stability, we shall examine the local integration error and study its boundedness using the notion of *input–output stability*.

When numerically integrating Eqs. (35), we obtain $c_1(k)$ and $c_2(k)$, the numerical error of which is given by Eq. (7). However, to derive the integration error of the solution $q(k)$, both errors must be superimposed according to Eq. (28). In other words, if e_{c_1} and e_{c_2} are the local integration errors of c_1 and c_2, respectively, then following Eq. (7),

$$e_{c_1}(t_k) = \tfrac{1}{2}h^2\ddot{c}_1(t_k), \qquad e_{c_2} = \tfrac{1}{2}h^2\ddot{c}_2(t_k) \tag{46}$$

Thus, for the one-dimensional case, the integration error of the VOP method e_{vop} becomes

$$\begin{aligned} e_{\text{vop}}(t_k) &= q_1(t_k)e_{c_1}(t_k) + q_2(t_k)e_{c_2}(t_k) \\ &= \frac{h^2}{2}[\ddot{c}_1(t_k)q_1(t_k) + \ddot{c}_2(t_k)q_2(t_k)] \end{aligned} \tag{47}$$

Differentiating Eq. (35) and substituting into Eq. (47) yields

$$e_{\text{vop}}(t_k) = \frac{h^2}{2}\left[\frac{1}{w}(-\dot{q}_2 f - q_2\dot{f})q_1 + \frac{1}{w}(\dot{q}_1 f + q_1\dot{f})q_2\right]$$
$$= \frac{h^2}{2}\left[-\frac{1}{w}(\dot{q}_2 q_1 - \dot{q}_1 q_2)f\right] = -\frac{h^2}{2}f(t_k) \tag{48}$$

As can be plainly seen, the VOP integration error is proportional to the forcing term. What is the implication of this result?

If the stability conditions stated by Lemma 1 are satisfied, it is guaranteed that the fundamental solutions are bounded. Thus, if $\|B_{\text{vop}}\|_i$ is the matrix norm induced by some vector $\boldsymbol{s} \neq 0$, then

$$\max_{t_k}\|B_{\text{vop}}(t_k)\|_i = \max_{t_k}\max_{s}\frac{\|B_{\text{vop}}(t_k)\boldsymbol{s}\|}{\|\boldsymbol{s}\|} < \infty \tag{49}$$

Because

$$\|\boldsymbol{u}(t_k)\|_\infty = h\|B_{\text{vop}}(t_k)\boldsymbol{f}(t_k)\|_\infty \leq h\|B_{\text{vop}}(t_k)\|_{i,\infty}\|\boldsymbol{f}(t_k)\|_\infty \tag{50}$$

if all the components of $\boldsymbol{f}$ are bounded, $\|f_j(t_k)\|_\infty < \infty \quad \forall\ j$, then the numerical error will be bounded, and system (45) is then stable in the BIBO sense.

Stated more formally, let L^∞ denote the normed space of measurable, bounded, discrete signals with $\|s(t_k)\|_\infty \triangleq \sup_{t_k}|s(t_k)| < \infty$, where $k \in \mathbb{N}$, and assume that the fundamental solutions of system (9) are bounded (according to Lemma 1, this happens if $K > 0$ and $C \geq 0$). Then $e_{\text{vop}_j} \in L^\infty$ if, and only if, $f_j \in L^\infty$, implying that Eq. (45) is BIBO stable. This holds even for *undamped* systems, whereas for the standard state variables, if C is singular or close to singularity ($\zeta = 0$ in the single-DOF case), then the integration error is unbounded. This result can now be formalized by the following Theorem.

Theorem 1: Let e_{vop_j} be the integration error for the jth DOF, $j \in [1, \ldots, n]$. Then $e_{\text{vop}_j} \in L^\infty$ if, and only if, $f_j \in L^\infty$.

Proof: We shall prove Theorem 1 constructively, by showing that

$$e_{\text{vop}_j}(t_k) = \frac{h^2}{2}\sum_{m=1}^{2n}\ddot{c}_m q_{jm} = -\frac{h^2}{2}f_j(t_k) \tag{51}$$

To that end, rewrite Eq. (42) into

$$\dot{\boldsymbol{c}}(t)_{[2n\times 1]} = \frac{1}{|W|}\text{adj}(W)\boldsymbol{\Psi} \tag{52}$$

where $\text{adj}(W)$ is the adjugate matrix of W, defined as

$$\text{adj}\,(W) \triangleq \begin{bmatrix} \Delta_{11} & \Delta_{21} & \cdots & \Delta_{n1} \\ \Delta_{12} & \Delta_{22} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \Delta_{12n} & \cdots & \cdots & \Delta_{2n2n} \end{bmatrix} \tag{53}$$

where $\Delta_{im} = (-1)^{i+m}M_{im}$ is the cofactor of the imth minor of W, M_{im}, defined as the determinant of the $(2n-1)\times(2n-1)$ matrix resulting from deleting row i and column m of W.

Writing Eq. (52) for the lth equation ($1 \leq l \leq 2n$) gives

$$\dot{c}_{\,l} = \frac{1}{|W|}(\Delta_{1l}f_1 + \Delta_{2l}f_2 + \cdots + \Delta_{nl}f_n) \tag{54}$$

The jth integration error of the jth DOF is then

$$e_j = \frac{h^2}{2}(\ddot{c}_1 q_{j1} + \cdots + \ddot{c}_l q_{jl} + \cdots + \ddot{c}_{2n}q_{j2n}) \tag{55}$$

Substituting Eq. (54) into Eq. (55) and rearranging entails

$$e_j = \frac{h^2}{2|W|}\left(\sum_{m=1}^{n}\dot{f}_m\sum_{i=1}^{2n}\Delta_{mi}q_{ji} + \sum_{m=1}^{n}f_m\sum_{i=1}^{2n}\dot{\Delta}_{mi}q_{ji}\right) \tag{56}$$

Notice that

$$\sum_{i=1}^{2n}\Delta_{mi}q_{ji}$$

is the determinant of a matrix containing the line

$$\sum_{i=1}^{2n}q_{ji}$$

twice; this determinant thus equals zero. This fact nullifies the first term in Eq. (56) $\forall\ m$, $1 \leq m \leq n$. Each of the additional terms in Eq. (56) has the form

$$\dot{q}_{\,j1}g_1 + q_{j1}\dot{g}_1 + \dot{q}_{j2}g_2 + q_{j2}\dot{g}_2 + \cdots + \dot{q}_{j,2n}g_{2n} + q_{j,2n}\dot{g}_{2n} \tag{57}$$

where the g terms are the expressions for the corresponding cofactors of the jth DOF. These terms can be rewritten into the determinants

$$G_1 = \begin{vmatrix} q_{j1} & q_{j2} & \cdots & q_{j,2n} \\ \vdots & \ddots & \ddots & \vdots \\ q_{j1} & q_{j2} & \cdots & q_{j,2n} \\ \vdots & \ddots & \ddots & \vdots \\ \dot{q}_{2n,1} & \dot{q}_{2n,2} & \cdots & \dot{q}_{2n,2n} \end{vmatrix}$$
$$G_2 = \begin{vmatrix} q_{j1} & q_{j2} & \cdots & q_{j,2n} \\ \vdots & \ddots & \ddots & \vdots \\ \dot{q}_{j1} & \dot{q}_{j2} & \cdots & \dot{q}_{j,2n} \\ \vdots & \ddots & \ddots & \vdots \\ \dot{q}_{j1} & \dot{q}_{j2} & \cdots & \dot{q}_{j,2n} \\ \vdots & \ddots & \ddots & \vdots \\ \dot{q}_{2n,1} & \dot{q}_{2n,2} & \cdots & \dot{q}_{2n,2n} \end{vmatrix} \tag{58}$$

which are valid for $1 \leq m \leq n$ and $m \neq j$. As can be seen, both G_1 and G_2 are zero, because the matrices are linearly dependant. For the case $m = j$, G_1 has the same form and is thus zero, and $G_2 = -|W|$. Substituting these expressions into Eq. (56) yields

$$e_j = -\frac{h^2}{2}f_j \tag{59}$$

which completes the proof of Theorem 1. □

To summarize, Theorem 1 shows that if a linear, stable, second-order n-DOF system (represented by the VOP state variables) is subjected to an L^∞-bounded forcing term, the integration error will also be L^∞-bounded. The magnitude of this integration error (i.e., the accuracy) will depend on the time step. This is an important improvement over the standard state variables, which yield an unbounded integration error for undamped, as well as for stiff, systems.

Although the instability of the explicit method can be removed by using the implicit Euler integration, the latter approach is unsuitable for integrating undamped systems: The implicit scheme using the generalized coordinates and velocities introduces artificial numerical damping, thus distorting the phase space of the original problem. The VOP-based variables, however, maintain the topology of the original problem. This fact is illustrated in the Appendix.

V. Stochastic Inputs Case

In addition to deterministic systems, the VOP-based selection of state variables can also be proven useful when modeling the response of large-scale linear differential systems to white noise. To show this, consider the system

$$\ddot{\boldsymbol{q}}(t) + C\dot{\boldsymbol{q}}(t) + K\boldsymbol{q}(t) = \boldsymbol{v}(t) \tag{60}$$

Here, $\boldsymbol{v}(t)$ is a zero-mean white noise with covariance $V = E[\boldsymbol{v}(t)\boldsymbol{v}^T(t)]$. Using the generalized coordinates and velocities as state variables yields

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + B\boldsymbol{v}(t), \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \tag{61}$$

where $\boldsymbol{x}(t) = [\,\boldsymbol{q}^T(t) \quad \dot{\boldsymbol{q}}^T(t)\,]^T$, $A = A_s$, $B = B_s$, A_s and B_s are given in Eq. (11), $\boldsymbol{x}_0$ is a random vector independent of $\boldsymbol{v}(t)$ with mean $\boldsymbol{m}_0$, and covariance $Q_0 = E[(\boldsymbol{x}_0 - \boldsymbol{m}_0)(\boldsymbol{x}_0 - \boldsymbol{m}_0)^T]$. The covariance matrix $Q(t) = E\{[\boldsymbol{x}(t) - \boldsymbol{m}][\boldsymbol{x}^T(t) - \boldsymbol{m}]\}$ satisfies the matrix differential equation [19]:

$$\dot{Q}(t) = AQ(t) + Q(t)A^T + BVB^T, \qquad Q(t_0) = Q_0 \tag{62}$$

For a single-DOF ODE driven by white noise, the covariance matrix has the following form:

$$Q = \begin{bmatrix} E(q^2) & E(q\dot{q}) \\ E(\dot{q}q) & E(\dot{q}^2) \end{bmatrix} \tag{63}$$

When using the VOP-based state variables, Eq. (60) becomes

$$\dot{\boldsymbol{c}}(t) = B_{\text{vop}}(t)\boldsymbol{\Omega}(t), \qquad \boldsymbol{c}(t_0) = \boldsymbol{c}_0 \tag{64}$$

where

$$\boldsymbol{\Omega}(t) = \begin{bmatrix} \boldsymbol{v}(t) \\ \boldsymbol{0} \end{bmatrix}_{[2n\times 1]}, \qquad B_{\text{vop}}(t) = \begin{bmatrix} \dot{V}_s(t) \\ V_s(t) \end{bmatrix}_{[2n\times 2n]} \tag{65}$$

Substituting Eq. (64) into Eq. (62) yields the covariance matrix of c_1 and c_2:

$$Q_{\text{vop}}(t) = \begin{bmatrix} E[c_1^2(t)] & E[c_1(t)c_2(t)] \\ E[c_2(t)c_1(t)] & E[c_2^2(t)] \end{bmatrix} \tag{66}$$

To obtain the covariance matrix of the solution, additional calculations are required. Recall that the VOP solution has the form $q(t) = c_1(t)q_1(t) + c_2(t)q_2(t)$; thus,

$$\begin{aligned} E[q^2(t)] = E\{[c_1(t)q_1(t) + c_2(t)q_2(t)]^2\} = q_1^2(t)E[c_1^2(t)] \\ + 2q_1(t)q_2(t)E[c_1(t)c_2(t)] + q_2^2(t)E[c_2^2(t)] \end{aligned} \tag{67}$$

In the same manner,

$$\begin{aligned} E[\dot{q}q] = E[q\dot{q}] = q_1\dot{q}_1E[c_1^2] + q_1\dot{q}_2E[c_1c_2] + q_2\dot{q}_1E[c_1c_2] \\ + q_2\dot{q}_2E[c_2^2] \end{aligned} \tag{68}$$

$$E[\dot{q}^2] = \dot{q}_1^2E[c_1^2] + 2\dot{q}_1\dot{q}_2E[c_1c_2] + \dot{q}_2^2E[c_2^2] \tag{69}$$

Because the expectation operators of c_1 and c_2 are known from Eq. (66), we can calculate the covariance matrix of solution (63) by using Eqs. (67–69). The covariance matrix of the VOP representation is given by

$$Q_{\text{vop}} = \begin{bmatrix} E[c_1^2] & E[c_1c_2] & E[c_1c_3] & \cdots E[c_1c_{2n}] \\ E[c_2c_1] & E[c_2^2] & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ E[c_{2n}c_1] & E[c_{2n}c_2] & \cdots & E[c_{2n}^2] \end{bmatrix} \tag{70}$$

IV. Illustrative Examples

In this section, we shall illustrate the merit of the VOP-based selection of state variables using several practical examples. We begin with the most simple example, a harmonic oscillator with a deterministic periodic input, continue with a discussion of computational complexity, present a multiple-DOF example, and conclude by considering a harmonic oscillator subjected to a white-noise input.

A. Harmonic Oscillator

Consider the driven harmonic oscillator:

$$\ddot{q} + \omega_n^2 q = A_0\cos(\omega t) \tag{71}$$

where A_0 is the amplitude and ω is the frequency of the driving force. Let $\dot{q}_0 = q_0 = 0.1$, $\omega_n = A_0 = 1$, $t_0 = 0$, and $\omega = 2$.

An Euler integration using generalized coordinates and velocities as state variables requires the state-space representation:

$$\dot{x}_1 = x_2 \tag{72a}$$

$$\dot{x}_2 = \cos(2t) - x_1 \tag{72b}$$

Applying the explicit Euler method (3) on Eq. (72) gives

$$A_{de} = \begin{bmatrix} 1 & h \\ -h & 1 \end{bmatrix}, \qquad B_{de} = \begin{bmatrix} 0 \\ h \end{bmatrix} \tag{73}$$

The VOP method, Eq. (42), yields the state-space representation:

$$\dot{c}_1 = \cos t\cos(2t), \qquad \dot{c}_2 = -\sin t\cos(2t) \tag{74}$$

for which

$$B_{d_{\text{vop}}} = h\begin{bmatrix} \cos kh & \sin kh \\ -\sin kh & \cos kh \end{bmatrix} \tag{75}$$

Because the damping coefficient of Eq. (71) is zero, then regardless of the time step, the explicit Euler integration of the generalized coordinates and velocities will diverge. As opposed to that, the integration error with the variational state variables will be bounded regardless of the time step. However, the value of the time step will define the accuracy of the numerical solution. For the current example, we chose the time step $h = 0.01$.

The results of the comparison are plotted in Fig. 1. As expected, the numerical error of the standard state variables diverges when using the explicit Euler method, whereas the numerical error for the same system with the VOP state variables remains bounded.

B. Computational Complexity

Another feature of the VOP approach is its reduced computational complexity compared with the standard implementation. The VOP approach is less computationally expensive than the standard implementation when a certain threshold of the maximum integration error is required.

We shall illustrate this observation by a simple example. Consider a damped harmonic oscillator with $\zeta = 0.1$, $\omega_n = 1$, and $\sin(t)$ as the deterministic periodic forcing term. The largest allowable time step for the state implementation to be stable is obtained from Eq. (21) and is equal to 0.2. To achieve better accuracy, let $h = 0.005$.

To examine the computational complexity, we estimate the required floating-point operation counts (FLOPS). The state implementation requires about 30 FLOPS per time step, whereas the VOP approach requires about 300 FLOPS per time step. Thus, if we take the same time step h with a simulation time of T_{sim}, the VOP approach will require $10T_{\text{sim}}/h$ more FLOPS. However, the VOP approach gives a much lower integration error.

In our example, $T_{\text{sim}} = 150$ and $h = 0.005$, resulting in 3×10^5 more FLOPS in the VOP approach. Yet, the integration error is lowered by about six orders of magnitude. Now, suppose that we have a threshold for the numerical integration error that is equal to the numerical error we had received for the state implementation. A natural question would be how much we can increase the time step for the VOP implementation to reach the integration threshold. We found that one can use a time step of $h = 1.2$ (240 times larger) and still be under the required threshold. Thus, for the same simulation time, far fewer time steps are needed for the VOP implementation, yielding much less FLOPS. In the state implementation we have 3×10^4 time steps giving 9×10^5 FLOPS, whereas in the VOP implementation we have 125 time steps, which corresponds to 21,625 FLOPS, that is, about 40 times less than the state implementation for the same numerical integration error.

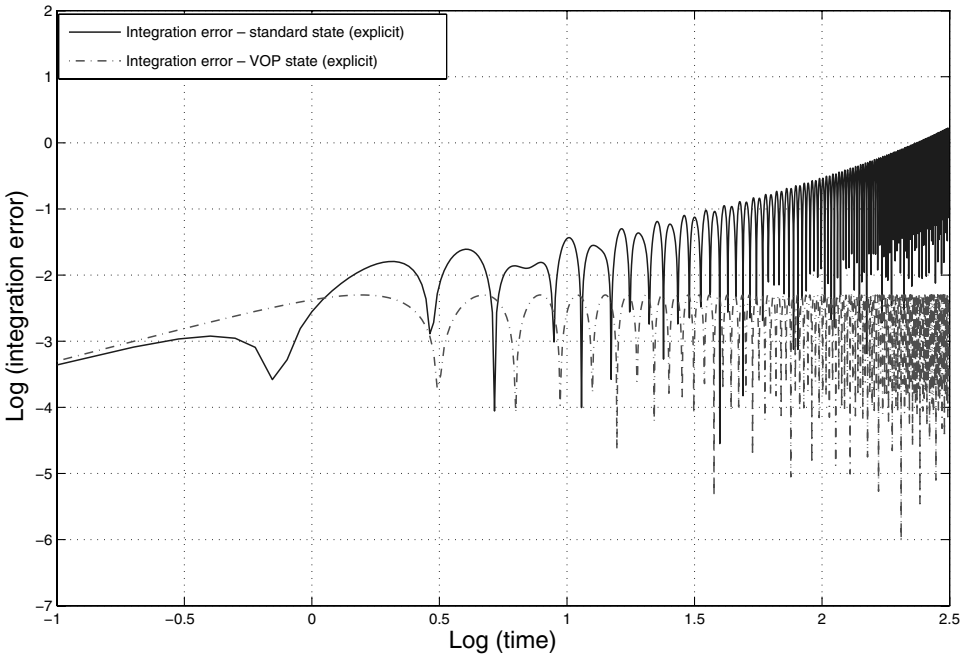

Fig. 1 A comparison of the numerical integration error of the standard state variables and the VOP-based state variables shows that the VOP-based variables give a bounded error whereas the standard states cause the error to rapidly diverge.

C. Three-DOF Spring-Mass System

One of the methods for determining the vibration modes of a flexible structure is to use finite elements. The first step of finite element modeling of an elastic structure is to divide it into several lumped-mass elements. By evaluating the physical properties of the individual finite elements and combining them appropriately, one can find the equations of motion of the complete structure. The stiffness matrix of a single finite element can be found by defining the interpolation function and using the principle of displacement [20]. The stiffness matrix of the complete structure can then be determined by adding the element stiffness coefficients.

To illustrate how the VOP-based approach improves numerical integration of finite element schemes, consider the 3-DOF spring-mass system depicted by Fig. 2.

The equations of motion can be obtained using a Lagrangian formalism by deriving the kinetic and potential energies [21]. After rearranging, we have

$$\begin{aligned} m_1\ddot{x}_1 + (k_1+k_2)x_1 - k_2x_2 &= F_0\sin(\omega t)\\ m_2\ddot{x}_2 + (k_2+k_3)x_2 - k_3x_3 - k_2x_1 &= 0\\ m_3\ddot{x}_3 + (k_3+k_4)x_3 - k_3x_2 &= 0 \end{aligned} \tag{76}$$

Let $m_1 = m_2 = m_3 = 1$ kg, $k_1 = k_4 = 1/4$ N/m, $k_2 = k_3 =$ 1 N/m, $\omega = 1$ rad/s, and $F_0 = 1$ N. Substituting these values into Eq. (76) and rewriting the equations in matrix form yields

$$\begin{bmatrix}\ddot{x}_1\\ \ddot{x}_2\\ \ddot{x}_3\end{bmatrix} + \begin{bmatrix}5/4 & -1 & 0\\ -1 & 2 & -1\\ 0 & -1 & 5/4\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} = \begin{bmatrix}\sin t\\ 0\\ 0\end{bmatrix} \tag{77}$$

Let $\boldsymbol{x} = [x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3]^T$ be the standard state variable vector, then

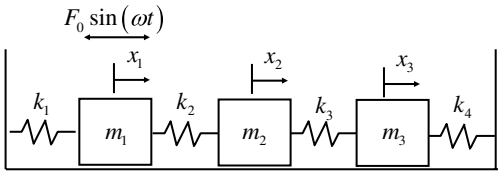

Fig. 2 A general 3-DOF spring-mass system.

$$A_{de} = \begin{bmatrix}0_{n\times n} & I_{n\times n}\\ -K & 0_{n\times n}\end{bmatrix}, \quad B_{de} = \begin{bmatrix}0_{n\times n}\\ I_{n\times n}\end{bmatrix}, \quad \boldsymbol{f} = \begin{bmatrix}\sin t\\ 0\\ 0\end{bmatrix}$$
$$K = \begin{bmatrix}5/4 & -1 & 0\\ -1 & 2 & -1\\ 0 & -1 & 5/4\end{bmatrix} \tag{78}$$

The corresponding VOP equations are

$$\begin{bmatrix}\dot{c}_1\\ \dot{c}_2\\ \dot{c}_3\\ \dot{c}_4\\ \dot{c}_5\\ \dot{c}_6\end{bmatrix} = \begin{bmatrix}-0.447\cos(1.118t)\\ 0.447\sin(1.118t)\\ 0.106\cos(1.757t)\\ -0.106\sin(1.757t)\\ 0.780\cos(0.402t)\\ -0.780\sin(0.402t)\end{bmatrix}\sin t$$

We note that this model, discretized by simply writing $t = t_0 + kh$, does not contain any artificial damping and is thus superior to the implicit formalism, as explained in the Appendix.

The numerical solution of the system is obtained by integrating both models using the explicit Euler method with a time step of $h = 0.01$. The results are depicted in Fig. 3. This figure shows the integration errors of x_1, x_2, and x_3. The integration errors using the standard state variables diverge, because the system contains no damping. The VOP-based integration, however, yields bounded errors throughout the integration time interval. We thus see that the VOP-based approach enables using explicit Euler integration even in undamped systems, as predicted by the theory.

D. Response of Linear Differential Systems to White Noise

For our final example, we consider the following ODE:

$$\ddot{q} + 10q = v(t) \tag{79}$$

where $v(t)$ is a disturbing normalized input modeled as a zero-mean white noise. Using the standard state variables gives the system

$$\dot{x}_1 = x_2, \qquad \dot{x}_2 = v(t) - 10x_1 \tag{80}$$

The VOP method, Eq. (42), yields

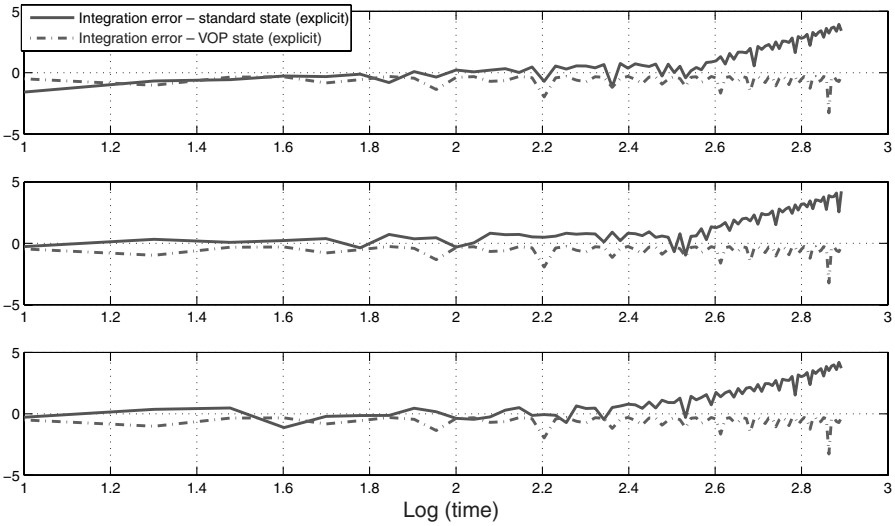

Fig. 3 Integration errors of a 3-DOF spring-mass model; standard state variables vs VOP variables; VOP-based representation yields small and bounded integration errors.

$$\dot{c}_1 = \tfrac{1}{10}\sqrt{10}\cos(\sqrt{10}t)v(t), \qquad \dot{c}_2 = -\tfrac{1}{10}\sqrt{10}\sin(\sqrt{10}t)v(t) \tag{81}$$

The system and input matrices, respectively, for the standard state variables are

$$A = \begin{bmatrix} 0 & 1 \\ -10 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{82}$$

The input matrix for the VOP-based representation is

$$B_{\text{vop}} = \frac{1}{\sqrt{10}} \begin{bmatrix} \cos(\sqrt{10}t) & \sin(\sqrt{10}t) \\ -\sin(\sqrt{10}t) & \cos(\sqrt{10}t) \end{bmatrix} \tag{83}$$

To obtain the true reference solution of the covariance, we solved Eq. (62) using a high-order Runge–Kutta integrator. This reference solution is regarded as the analytical solution for the purpose of integration error analysis. The numerical solution using explicit Euler integration is performed for both representations using the time step $h = 0.01$. Figure 4 depicts the integration errors of the 2×2 covariance matrix entries $e(Q_{11})$, $e(Q_{12})$, $e(Q_{21})$, and $e(Q_{22})$, obtained using explicit Euler integration for the standard states and the VOP states.

From the results depicted in Fig. 4, it is evident that the covariance matrix obtained using the VOP representation practically converges to the analytical solution, whereas the covariance of the standard state variables diverges.

VII. Conclusions

It was shown that the stability of the explicit Euler integration method depends on the selection of state variables used to model the original continuous differential equation. In weakly damped and/or stiff linear systems that are numerically integrated using the explicit Euler formalism, choosing the generalized coordinates and velocities as state variables requires an infinitesimally small time step, meaning that the numerical integration will diverge. However, when representing the system using the variational state variables, this divergence is avoided, and the resulting numerical integration error is bounded for bounded forcing inputs and a finite step size.

The new insight into the connection between selecting state variables and its effect on stability of the explicit Euler method could be used to simulate the effects of external disturbances on large-scale

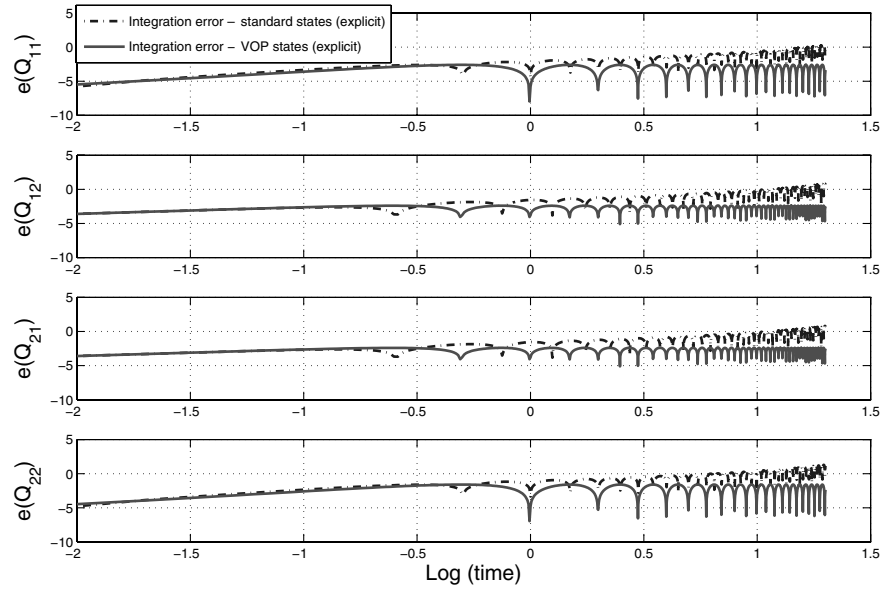

Fig. 4 A numerical integration of the covariance matrix entries using Euler's explicit method shows that the VOP-based state representation yields small and bounded errors, whereas the standard state variables cause numerical divergence of the covariance matrix components.

multiple-degree-of-freedom systems, subjected to both stochastic and deterministic inputs. A model of the system, comprising the fundamental solutions of the unforced linear model, must first be written. The effect of the arbitrary forcing inputs can then be studied using the variation of the integration constants. This variation-of-parameters (VOP) formalism is particularly convenient because the transformation from generalized velocities and coordinates to variational parameters yields a driftless linear time-varying system.

In addition to numerical simulations, the newly developed method can be used for real-time implementation of controllers. The explicit Euler method introduces minimum delay into the computational cycle due to the fact that it is a one-step forward method. Although standard state variables do not allow real-time implementation when the original continuous model is undamped or stiff, the VOP-based approach extends the real-time implementability of the Euler method to new realms, providing stability and reduced computational complexity.

Appendix: Euler's Implicit Method Introduces Artificial Numerical Damping

The implicit Euler integration scheme is given by

$$\boldsymbol{x}(k+1) = \boldsymbol{x}(k) + h\dot{\boldsymbol{x}}(k+1) = \boldsymbol{x}(k) + h[A(k+1)\boldsymbol{x}(k+1) + B(k+1)\boldsymbol{f}(k+1)] \tag{A1}$$

Applying Eq. (A1) on Eq. (11) yields the discrete state-space representation:

$$\begin{bmatrix} \boldsymbol{x}_1(k+1) \\ \boldsymbol{x}_2(k+1) \end{bmatrix} = \begin{bmatrix} [I_{n\times n} + h^2(I_{n\times n} + hC)^{-1}K]^{-1} & hP \\ -hPK & P \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1(k) \\ \boldsymbol{x}_2(k) \end{bmatrix} + \begin{bmatrix} h^2P \\ hP \end{bmatrix} \boldsymbol{f}(k+1) = A_{di}\boldsymbol{x}(k) + B_{di}\boldsymbol{f}(k+1) \tag{A2}$$

where $P = (h^2K + hC + I_{n\times n})^{-1}$ and $A_d = A_{di}$ and $B = B_{di}$ are the system and input matrices, respectively, of the discrete state-space representation in Euler's implicit formalism.

We will now show that the implicit Euler method introduces artificial damping when used for integrating undamped systems. To that end, consider the 1-DOF unforced system

$$\ddot{q} + \omega_n^2 q = 0, \qquad q(t_0) = q_0, \qquad \dot{q}(t_0) = \dot{q}_0 \tag{A3}$$

for which $C = 0$ and $K = \omega_n^2$. Using the standard state variables $x_1 = q$ and $x_2 = \dot{q}$ transforms Eq. (A2) into

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \frac{1}{1 + h^2\omega_n^2} \begin{bmatrix} 1 & h \\ -h\omega_n^2 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \tag{A4}$$

The magnitudes of the (complex) eigenvalues of the system matrix A_{di} in this case satisfy

$$|\lambda_1| = |\lambda_2| = \frac{1}{\sqrt{1 + h^2\omega_n^2}} < 1 \quad \forall\ h > 0 \tag{A5}$$

and thus the integrated system will exhibit exponential decay, although the original system (A3) exhibits harmonic oscillations. However, when using a VOP-based representation, the state variables c_1 and c_2 will be constant [because system (A3) is unforced; cf. Eq. (35)], and thus the analytical solution (28) will be identical to the numerical solution.

This analysis can be extended to the n-DOF case in the following manner: Assume that $K > 0$ and $C = 0$, so that all the modes of the original unforced system exhibit harmonic oscillations. The system matrix A_{di} becomes

$$A_{di} = \begin{bmatrix} P & hP \\ -hPK & P \end{bmatrix} = \begin{bmatrix} P & 0 \\ 0 & P \end{bmatrix} \begin{bmatrix} I_{n\times n} & hI_{n\times n} \\ -hK & I_{n\times n} \end{bmatrix} \tag{A6}$$

thus,

$$|A_{di}| = \frac{1}{|I_{n\times n} + h^2K|^2} |I_{n\times n} + h^2K| = \frac{1}{|I_{n\times n} + h^2K|} < 1 \tag{A7}$$

where the last inequality in Eq. (A7) stems from the fact that $K > 0$. Therefore, *at least* one mode of the discrete system will exhibit exponential decay, although *all* the modes of the original systems are oscillatory. In contrast, the VOP-based state variables will yield the exact solution because, in the absence of forcing inputs, $c_i = \text{const}$, where $i = 1, \ldots, n$.

Acknowledgments

This research was partially supported by the Asher Space Research Institute of the Technion—Israel Institute of Technology. The authors are in debt of gratitude to Moshe Idan, Barry Greenberg, and Daniella Raveh of Technion for providing useful insights.

References

[1] Newton, I., *Philosophiae Naturalis Principia Mathematica*, Edmond Halley, London, 1687, Chap. 2, pp. 65–117.
[2] Ball, W. W., *A Short Account of the History of Mathematics*, 4th ed., Dover, New York, 1960, Chap. 18, pp. 398–400.
[3] Butcher, J. C., *The Numerical Analysis of Ordinary Differential Equations: Runge–Kutta and General Linear Methods*, Wiley, New York, 1987, pp. 25–46, Chap. 2.
[4] Ogata, K., *Modern Control Engineering*, 4th ed., Pearson Education International, Upper Saddle River, NJ, 2002, Chap. 3, pp. 53–58.
[5] Cannon, R. H., *Dynamics of Physical Systems*, McGraw–Hill, New York, 1967, Chap. 2, pp. 32–57.
[6] Bisplinghoff, R. L., Ashley, H., and Hlfman, R. L., *Aeroelasticity*, Dover, New York, 1996, Chap. 11, pp. 695–715.
[7] Shinya, M., "Theories for Mass-Spring Simulation in Computer Graphics: Stability, Costs and Improvements," *IEICE Transactions on Information and Systems*, Vol. E88-D, No. 4, Apr. 2005, pp. 767–774.
[8] Euler, L., "Recherches sur la Question des Inegalites du Movement de Saturne et de Jupiter," *Piece qui a Remporte le Prix*, de l'Académie Royale des Sciences, Année, France, 1748.
[9] Lagrange, J. L., "Sur la Theorie des Variations des Elements des Planetes et en Particulier des Variations des Grands Axes de Leurs Orbites," *l'Institut de France*, Paris, 1808.
[10] Wanner, E. H. G., *Solving Ordinary Differential Equations II- Stiff and Differential-Algebraic Problems*, Springer, New York, 1996, pp. 15–37, Chap. 4.
[11] Shieh, L., Mehio, M. M., and Dib, H. M., "Stability of the Second-Order Matrix Polynomial," *IEEE Transactions on Automatic Control*, Vol. AC-32, No. 3, 1987, pp. 231–233.
[12] Bellman, R., *Introduction to Matrix Analysis*, McGraw–Hill, New York, 1970, pp. 10, 67.
[13] Anderson, B., and Bitmead, R. E., "Stability of Matrix Polynomials," *International Journal of Control*, Vol. 26, No. 2, 1977, pp. 235–247.
[14] Franklin, G. F., *Digital Control of Dynamic Systems*, Addison Wesley Longman, Reading, MA, 1998, Chap. 4, pp. 79–80.
[15] Dahlquist, G., "Stability of Two-Step Methods for Variable Integration Steps," *SIAM Journal on Numerical Analysis*, Vol. 20, No. 5, 1983, pp. 1071–1085.
[16] Horn, R. A., and Johnson, C., *Matrix Analysis*, Cambridge Univ. Press, Cambridge, England, U.K., 1985, pp. 7–12.
[17] Boyce, W. E., and Diprima, R. C., *Elementary Differential Equations*, 6th ed., Wiley, New York, 1996, Chap. 4, pp. 237–242.
[18] Gurfil, P., and Klein, I., "Mitigating the Integration Error in Numerical Simulations of Newtonian Systems," *International Journal for Numerical Methods in Engineering*, Vol. 68, No. 2, 2006, pp. 267–297.
[19] Kwakernaak, H., and Sivan, R., *Linear Optimal Control Systems*, Wiley Interscience, New York, 1972, Chap. 1.
[20] Wie, B., *Space Vehicle Dynamics and Control*, AIAA Education Series, AIAA, Reston, VA, 1998, Chap. 8, pp. 463–501.
[21] Meirovitch, L., *Introduction to Dynamics and Control*, Wiley, New York, 1985, pp. 42-56.