CrossMark

A novel unconditionally stable explicit integration method for finite element method

Mianlun Zheng[1] · **Zhiyong Yuan**[1] · **Qianqian Tong**[1] ·
Guian Zhang[1] · **Weixu Zhu**[1]

Abstract Physics-based deformation simulation demands much time in integration process for solving motion equations. To ameliorate, in this paper we resort to structural mechanics and mathematical analysis to develop a novel unconditionally stable explicit integration method for both linear and nonlinear FEM. First we advocate an explicit integration formula with three adjustable parameters. Then we analyze the spectral radius of both linear and nonlinear dynamic transfer function's amplification matrix to obtain limitations for these three parameters to meet unconditional stability conditions. Finally, we theoretically analyze the accuracy property of the proposed method so as to optimize the computational errors. The experimental results indicate that our method is unconditionally stable for both linear and nonlinear systems and its accuracy property is superior to both common and recent explicit and implicit methods. In addition, the proposed method can efficiently solve the problem of huge computation cost in integration procedure for FEM.

Keywords Finite element method · Implicit integration · Explicit integration · Unconditionally stable

✉ Zhiyong Yuan
zhiyongyuan@whu.edu.cn

1 School of Computer, Wuhan University, Wuhan 430072, China

1 Introduction

The widespread use of physics-based simulations of deformable objects has been enabling abundant applications such as computer games, virtual reality systems, computer animation. To efficiently and accurately simulate the object's physical behaviors of deformation, many fundamental methodologies, ranging from finite element method (FEM) [1] together with their GPU acceleration [2,3] to various types of flexible mesh-free methods, have been well devised to accommodate the application-specific requirements [4]. However, in physical modeling, the partial differential equations of solid continuum mechanics are quite demanding and permit few computational shortcuts [5]. Therefore, there are a lot of methods trying to explore effective and flexible numerical approaches so as to solve the problem to reach high update frame rates.

In fact, the most computational time-consuming part of physics-based animation methods like FEM is exactly the integrating solution procedure for dynamic equations, where implicit integration method is adopted most frequently. In particular, as the number of degrees-of-freedom participating in calculation gets larger, the computational cost will increase correspondingly.

In principle, in field of structural mechanics, step-by-step time integration methods are widely used to solve the dynamic motion equations, which are usually sorted into two types: explicit and implicit [6]. For explicit integration methods, the system matrices for solving the displacements at next step are known and keep constant at the beginning and also during calculation, while the computation of implicit integration methods mainly depends on the motion responses in each time step, which would change in simulation. As a result, while dealing with the same model, explicit methods cost less time for integration than implicit methods.

Compared to explicit methods, most implicit methods possess the advantage of unconditionally stability. With regard to explicit methods, although most are conditionally stable, the computational cost for explicit methods is far less than that of implicit methods because of no factorization of any unknown system matrices. Moreover, the advantage for time-consuming promotion is more significant when the number of calculation dimensions gets larger.

In a word, the key technical challenges of presenting a new explicit integration method are highlighted as follows: (a) From the perspective of computation stability, the explicit method has to be unconditionally stable; (b) Common explicit integration methods are explicit only for linear system, but such methods also have no advantage of computational efficiency when encountering nonlinear system equipped with variable dampers. So the proposed method is supposed to be explicit and stable for both linear and nonlinear systems; (c) As far as accurate performance is concerned, the numerical property of the method to be presented should be superior to existing methods, so as to assure the modeling precision for physical animation method FEM.

In this paper, we resort to structural mechanics and mathematical analysis areas to solve the challenges above and develop a novel explicit integration method mainly for FEM in order to handle both efficient and stable deformation simulation. And the salient contributions of this paper include:

1. We present a novel explicit integration method which is unconditionally stable for both linear and nonlinear systems, compared to most existing explicit integration methods.
2. From the perspective of numerical accuracy property, there are 3 adjustable variable parameters in our integration formulas which can be determined according to the current calculation conditions, so as to optimize the calculation errors. More significantly, the accuracy property of our method is superior to both common and recent explicit and implicit methods.
3. We redescribe FEM using the proposed integration method, where the calculation procedure for dynamic equations at each time step is much more intuitive and convenient compared to current integration methods.
4. The last but not the least, the performed simulation experiments of FEM indicate that the proposed explicit integration method can more efficiently solve the problem of huge computational cost in integration procedure, compared to implicit methods.

Superiorities above enable us to assure both efficient and accurate simulations of FEM. As a side note, the derivation process of our method in this paper is also applicable for other physics-based methods with dynamic integration procedure.

2 Related work

Closely relevant to the central theme of this paper, in this section we briefly give out two previous work in following aspects: finite element method (FEM) and its integration methods.

Finite element method (FEM)
FEM has been proven to be a powerful approach to accurately simulate the physical and mechanical principles underlying deformable models. Erleben et al. [1] detailedly described the whole modeling and simulation process for FEM in their book. During the simulation process of FEM, the modeling object is represented as a 3D volumetric mesh by dividing the object into a large number of elements and the simulation is prescribed (controlled) by the displacements of the mesh vertices.

Although fully physics-based FEM is widely considered as reliable and accurate ways to simulate deformable objects, its main shortcoming is that it is computationally demanding [4], especially for its integration process. Therefore, researchers in areas of computational physics, mechanics, and applied mathematics have been developing a great deal of novel algorithms to solve the problem.

Hirota et al. [7] introduced a novel penalty method based on the concept of material depth to analytically interpolate contact forces over surfaces without raising the computational consumption. Choi et al. [8] regarded the rotational component as an infinitesimal deformation tracked by traditional linear modal analysis. By integrating small rotations, authors realized real-time simulation for large deformations. Yang et al. [9] put forward a novel framework for multi-domain subspace deformation to achieve real-time simulation, which used the Lagrange multiplier technique to impose coupling constraints at the boundary.

Model reduction has popularized itself for simplifying simulation of dynamical systems described by differential equations [10]. Barbic [5,11] presented an automatic modal derivative approach to select quality low-dimensional basis vectors and an algorithm for fast simulation of the resulting nonlinear dynamics. Yang et al. [12] proposed a complete system of precomputation pipeline as a faster alternative to the classic linear and nonlinear modal analysis. However, although model reduction plays a satisfactory role, researchers still face a fundamental problem about making well tradeoff between computation speed and simulation precision.

Integration methods
It requires a complex integration solving process for ordinary differential equations in deformable simulations [13]. Generally, the integration methods can be classified into two types: explicit and implicit, and they both possess their own advantages and disadvantages [14].

Most implicit integration methods are unconditionally stable, such as the Erleben method [1], Newmarks' constant acceleration method [15,16] and Wilson-θ method [17]. This implies their time step is limited only by the convergence or accuracy considerations. However, these methods have to solve differential equations at each time step [6], thus causing large computation cost. In contrast, the factorization of system matrix is not required for most explicit methods, which saves a lot of computational cost. But it exists an obvious problem of conditionally stability for those traditional explicit methods such as the central difference [18] and the explicit Newmark method [15]. Normally, a severe restriction on the time step has to be applied in order to receive satisfactory simulation results [19].

For improvements of implicit integration methods, Kang et al. [20] presented an approximate implicit method for creating animation of flexible objects. The method mainly focused on the mass-spring model, and it took $O(n)$ time for integration when the number of springs is $O(n)$. In addition, Oh et al. [21] proposed a new implicit integration method for stable cloth simulation, which decreased the damping artifacts rather than introduced excessive damping forces. But these two methods are not suitable for simulations with a large number of mass points due to generation of undesirable results.

Furthermore, researchers also proposed several unconditionally stable explicit methods to eliminate the stability drawback. Chang et al. [22–24] put forward a series of explicit algorithms with excellent stability as well as Ding et al. [25]. Nevertheless, if provided with a nonlinear system whose damping or stiffness parameters does not keep constant, the Chang family methods are required to solve nonlinear equations due to non-diagonal damping or stiffness matrix. Consequently, for a nonlinear system, these explicit methods lose advantages of computation efficiency. To overcome such problem, many methods are proposed to explicitly solve the nonlinear dynamics equations, such as Chang et al. [26], Chen et al. [27], and Yao et al. [6]. Methods of Chang [26] and Chen [27] offer an explicit estimate of velocity and displacement as well. Although they are unconditionally stable for any linear elastic or stiffness softening system, they are still conditionally stable for an instantaneous stiffness hardening system. Yao et al. [6] developed a new series of explicit integration methods based on the discrete control theory whose accuracy and stability properties are very well for both linear and nonlinear systems. However, the calculation stage at each time step for FEM of the methods is quite complex.

For other areas of deformation simulation like position-based dynamics, researchers have developed a set of approaches for integration process. Bouaziz et al. presented a new method for implicit time integration of physical systems, which builds a bridge between nodal finite element methods and position-based dynamics, leading to a simple, efficient, robust and accurate solver [28]. Wang et al. demonstrated the use of the Chebyshev semi-iterative approach in projective and position-based dynamics to accelerate the convergence by at least one order of magnitude [29]. Wang et al. also proposed a gradient descent method using Jacobi preconditioning and Chebyshev acceleration for elastic body simulation approaches. Moreover, with step length adjustment, initialization and invertible model conversion techniques, the final method is simple, fast, scalable, memory-efficient and robust [30]. Liu et al. [31] interpreted projective dynamics as a quasi-Newton method that enables very efficient simulation of hyperelastic materials, and the final method is more than 10 times faster than one iteration of Newton's method.

3 Basics

In this paper, the proposed method mainly aims at finite element method (FEM), and its derivation process could be extended to other physics-based methods with dynamic integration procedure. In this section, we give a brief introduction about FEM and its two integration methods: the implicit and explicit integration method.

3.1 Finite element method

In FEM, the continuous vibration is replaced by a finite number of displacements [32]. For a multiple-degree-of-freedom (MDOF), solid, dynamic, damped FEM system, an ordinary differential equation of motion can be expressed as,

$$M\ddot{u}_i + D(\dot{u}_i) + K(u_i) = F_i \tag{1}$$

where i stands for the current time step, $u \in R^{3n}$ contains the displacements of n mesh vertices away from the rest configuration, $M \in R^{3n,3n}$ is the mass matrix, $D(\dot{u}_i) \in R^{3n}$ is the damping force, $K(u_i) \in R^{3n}$ is the internal elastic force and $F_i \in R^{3n}$ restores the external forces like gravity, pushing, dragging etc.

$K(u_i) = K \cdot u_i$ and $K \in R^{3n,3n}$ is called the tangent stiffness matrix. Similarly, $D(\dot{u}_i) = D \cdot \dot{u}_i$ and $D \in R^{3n,3n}$ is called the damping matrix. Moreover, by setting values of K and D, one can obtain either linear or nonlinear FEM system.

During simulation, Eq. 1 has to be solved for each time step to obtain the new displacement u and its calculation scale depends on the number of mesh vertices n. As a result, the computational cost increases when the value of n gets larger. If dealing with a certain model with more than 10^4 element vertices, the computing speed of traditional FEM is rather slow which leads to abnormal simulation display.

3.2 Implicit and explicit integration method

For the sake of simulation stability, the most widely used integration method for solving Eq. 1 is implicit method with unconditionally stability. A popular one in structural dynamics is the implicit Newmark integrator [5,33], which assumes the expressions of $\dot{u}_{i+1}$ and u_{i+1} as follows:

$$\dot{u}_{i+1} = \dot{u}_i + [(1-\delta)\ddot{u}_i + \delta\ddot{u}_{i+1}]\Delta t \quad (2)$$

$$u_{i+1} = u_i + \dot{u}_i\Delta t + [(-1/2-\alpha)\ddot{u}_i + \alpha\ddot{u}_{i+1}]\Delta t^2 \quad (3)$$

To assure the method to be unconditionally stable, the values of δ and α must meet the conditions of $\delta \geq 0.5$, $\alpha \leq 0.25(0.5+\delta)^2$. Here we choose $\delta = 0.5$, $\alpha = 0.25$, which is a common setting for many applications. Besides, the method is second-order accurate in terms of local error.

Basically, one implicit Newmark time step means solving the current dynamic equilibrium equations of motion, namely Eq. 1. By inserting Eqs. 2 and 3 into the motion equation and solving it, we can finally get the results. A detailed algorithm of one step of the implicit Newmark subspace integration is shown in Algorithm 1.

Algorithm 1 The implicit Newmark integration method

Input: values of u_i, $\dot{u}_i$, $\ddot{u}_i$ at time step i; external force F_{i+1} at time step $i+1$; time step size Δt; max number of iterations per step j_{max}; tolerance TOL to avoid unnecessary iteration.

Output: values of u_{i+1}, $\dot{u}_{i+1}$, $\ddot{u}_{i+1}$ at time step $i+1$.

1: $u_{i+1} \leftarrow u_i$;
2: **for** $j = 1$ to j_{max} **do**
3:     Calculate internal forces $R(u_{i+1})$
4:     Calculate stiffness matrix $K(u_{i+1})$;
5:     Calculate the damping matrix $D = M + \beta K(u_{i+1})$;
6:     Calculate the system matrix $A = \alpha_1 M + \alpha_4 C + K(u_{i+1})$;
7:     $residual \leftarrow M(\alpha_1(u_{i+1}-u_i) - \alpha_2\dot{u}_i - \alpha_3\ddot{u}_i) + C(\alpha_4(u_{i+1}-u_i) + \alpha_5\dot{u}_i) + R(q_{i+1}) - F_{i+1}$;
8:     **if** $||residual||_2 < TOL$ **then**
9:       break out of **for** loop;
10:     **end if**
11:     Solve the symmetric system: $A(\Delta u_{i+1}) = -residual$;
12:     $u_{i+1} \leftarrow u_{i+1} + \Delta u_{i+1}$;
13: **end for**
14: $\dot{u}_{i+1} \leftarrow \alpha_4(u_{i+1}-u_i) + \alpha_5\dot{u}_i$;
15: $\ddot{u}_{i+1} \leftarrow \alpha_1(u_{i+1}-u_i) - \alpha_2\dot{u}_i - \alpha_3\ddot{u}_i$;
16: **return** $u_{i+1}, \dot{u}_{i+1}, \ddot{u}_{i+1}$;

In Algorithm 1, parameters are set as follows: $\alpha_1 = 4/\Delta t^2, \alpha_2 = 4/\Delta t, \alpha_3 = 1, \alpha_4 = 2/\Delta t, \alpha_5 = -1$. It can be seen in Algorithm 1 that it requires recalculation (Line 6) and factorization (Line 11) of the system matrix $A = \alpha_1 M + \alpha_4 C + K(u_{i+1})$ in each iteration of one time step due to its non-constant feature. Consequently, the computational cost for implicit integration method is in direct proportion to the square of the number of degrees of freedom.

Table 1 Experimental models and their integration time (ms), frame rate (FPS) for implicit Newmark integration (Algorithm 1)

Model	Vertices	Facets	Elements	Time (ms)	FPS
Kidney	854	1472	433	157.0	6.3
Armadillo	2905	25,200	1499	429.0	2.3
Bunny	5553	41,330	2819	780.0	1.2
Lung	8334	86,084	4188	1298.0	0.7
Heart	9580	95,729	5607	1610.0	0.6
Brain	12,473	110,046	6605	1940.0	0.5

Furthermore, there also exists the other integration method called explicit integration which costs less time. A general explicit method is central difference, which is used to express velocity and acceleration at a certain point as,

$$\dot{u}_i = \frac{(u_{i+1} - u_{i-1})}{2\Delta t} \quad (4)$$

$$\ddot{u}_i = \frac{(u_{i+1} - 2u_i + u_{i-1})}{\Delta t^2} \quad (5)$$

Insert Eqs. 4 and 5 into Eq. 1 we can get an explicit dynamic equation for u_{i+1},

$$\left(M + \frac{\Delta t}{2}D\right)u_{i+1} = \Delta t^2(F_i - K(u_i))$$
$$+ \frac{\Delta t}{2}Du_{i-1} + M(2u_i - u_{i-1}) \quad (6)$$

It is quite obvious that for a linear system, the matrix $\left(M + \frac{\Delta t}{2}D\right)$ is constant and can be precomputed and factorized in preprocess stage. Thus the computation cost is proportional to the number of degrees of freedom. Compared with implicit method, the explicit method has the advantage of less computation consumption. Furthermore, if the number of elements for simulation becomes larger, the superiority of explicit method will be more significant.

However, the explicit method does exist two shortcomings. The first one is that for a system equipped with nonlinear dampers, the explicit algorithm has no advantage of computational efficiency. Secondly, when the time step is rather large, the computational error of the explicit scheme will accumulate over time, thus causing the dynamic system extremely unstable.

3.3 Experimental data

In this paper, we mainly adopt six simulation models with different numbers of elements to perform the experiments. The detailed information of these six models is listed in Table 1.

We have obtained integration time (ms) and frame rate (FPS) of the implicit method shown in Algorithm 1 in advance. Table 1 shows only the integration time of one
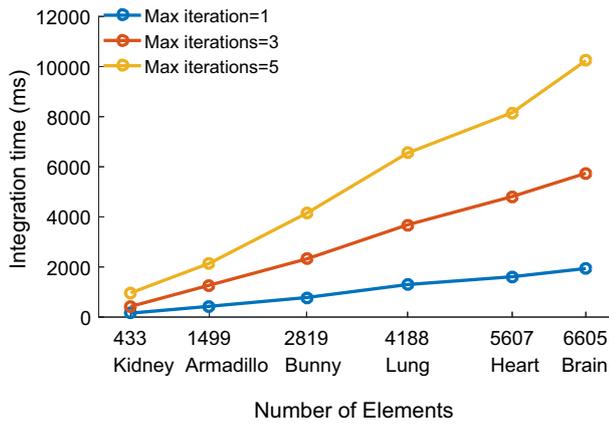
Fig. 1 The implicit Newmark integration time (Algorithm 1) for six models with different values of max iterations

iteration per time step, which means j_{max} (max number of iterations per step) in Algorithm 1 is set as 1 (also called semi-implicit method). In addition, the value of TOL is set as 10^{-6}. While considering values of frame rate, we here ignore the rendering time for models because it will be influenced by model's complexity.

As indicated in Table 1, it takes 157.0 ms per iteration for Kidney model with 433 simulation elements. Obviously such integration time will lead to unacceptable frame rate, let alone those models with more elements. If the number of elements exceeds approximately 3000, the frame rate will be <1 FPS.

Meanwhile, we set j_{max} as two other values, 3 and 5, and obtain their integration time, as shown in Fig. 1. Similarly, the integration time increases greatly with the increase in elements, especially for more iterations per integration time step.

Although the implicit integration method is considered as a pretty one for stable FEM simulation, it still needs to be improved due to its demanding computational cost. Therefore, in consideration of the strength and weaknesses of explicit integration method, this paper aims at proposing a new explicit integration method for dynamic equation solution to improve FEM's computational efficiency.

4 Development of a novel explicit method

In this section, we present a novel unconditionally stable explicit integration method for both linear and nonlinear FEM. Firstly, we present an explicit integration formula with three unknown parameters. Then by analyzing the spectral radius of transfer function's amplification matrix, the range of three parameters is determined so as to meet the unconditional stable condition. Finally, we give a detailed description for FEM with the proposed integration method.

4.1 Integration for linear system

Firstly, we adopt Taylor's expansion to derivate the expressions of velocity $\dot{u}_{i+1}$ and displacement u_{i+1}, which is similar to the derivation of central difference method as,

$$\dot{u}_{i+1} = \dot{u}_i + \ddot{u}_i \Delta t + u_i^{(3)} \frac{\Delta t^2}{2} + \cdots + u_i^{(m+1)} \frac{\Delta t^m}{m!} + o(\Delta t)^m \tag{7}$$

$$u_{i+1} = u_i + \dot{u}_i \Delta t + \ddot{u}_i \frac{\Delta t^2}{2} + \cdots + u_i^{(m)} \frac{\Delta t^m}{m!} + o(\Delta t)^m \tag{8}$$

In Eqs. 7 and 8, save the term of u_i, $\dot{u}_i$, $\ddot{u}_i$ for expressions of $\dot{u}_{i+1}$ and u_{i+1}. The rest terms do not appear in a dynamic equations, so we omit them and use a coefficient α to adjust the formula instead, in order to compensate the errors.

$$\begin{cases} \dot{u}_{i+1} = \dot{u}_i + \alpha \ddot{u}_i \Delta t \\ u_{i+1} = u_i + \dot{u}_i \Delta t + \alpha \ddot{u}_i \Delta t^2 \end{cases} \tag{9}$$

α is an integration parameter to be determined, and Δt is the time step size. Next, we substitute Eq. 9 into Eq. 1 so as to obtain explicitly integration motion equation. Besides, for more convenient and simply representation in the following analysis, we here use $1/\alpha$ to replace α in all formulas followed.

$$\alpha M u_{i+1} = \Delta t^2 (F_i - K(u_i)) + D \Delta t (u_{i-1} - u_i) + \alpha M (2u_i - u_{i-1}) \tag{10}$$

Transform Eq. 10 into a discrete transfer function of recursive matrix form as,

$$\begin{bmatrix} u_{i+1} \\ u_i \end{bmatrix} = U \begin{bmatrix} u_i \\ u_{i-1} \end{bmatrix} + F \tag{11}$$

where U is the amplification matrix and F is related to the external force which are defined as,

$$U = \begin{bmatrix} \frac{-K^2 \Delta t^2 - D \Delta t + 2\alpha M}{\alpha M} & \frac{D \Delta t - \alpha M}{\alpha M} \\ 1 & 0 \end{bmatrix} \tag{12}$$

$$F = \frac{1}{\alpha M} \Delta t^2 F_i \tag{13}$$

According to the discrete control theory [34,35], the stability of a transfer function relies on its amplification matrix's spectral radius. If the value of spectral radius (the maximum absolute eigenvalues) is always less than 1 for any time step size, the system is unconditionally stable. For simplicity, α is derived from a single-degree-of-freedom (SDOF) system in the following. Rewrite the expression of matrix U into a linear SDOF system format and insert $k = m\omega^2$, $d = 2\xi\omega m$, $\omega \Delta t = \Omega = \Delta t / T$ into it (here m is the mass, ω is the natural vibration frequency, ξ is the viscous damping ratio and T is the natural vibration period), we can finally obtain,

$$U = \begin{bmatrix} \frac{-\Omega^2 - 2\Omega\xi + 2\alpha}{\alpha} & \frac{2\Omega\xi - \alpha}{\alpha} \\ 1 & 0 \end{bmatrix} \qquad (14)$$

In order to analyze the spectral radius of matrix U, we use $A = \frac{-\Omega^2 - 2\Omega\xi + 2\alpha}{\alpha}$ and $B = \frac{2\Omega\xi - \alpha}{\alpha}$ to simplify the matrix U as,

$$U = \begin{bmatrix} A & B \\ 1 & 0 \end{bmatrix} \qquad (15)$$

The characteristic equation for amplification matrix U can be derived by solving the equation of $|U - \lambda I| = 0$ and it is found to be,

$$f(\lambda) = \lambda^2 - A\lambda - B = 0 \qquad (16)$$

where λ is the eigenvalue and I is a unit matrix.

According to Eq. 16, the two characteristic roots can be expressed as,

$$\lambda_{1,2} = \frac{A \pm \sqrt{A^2 + 4B}}{2} \qquad (17)$$

Taking unconditional stability into consideration, $|\lambda_1|$ and $|\lambda_2|$ have to be less than 1 for any value of Ω. While observing the formula of $|\lambda_{1,2}|$, A and B, we will definitely find that if $A = \frac{-\Omega^2 - 2\Omega\xi + 2\alpha}{\alpha}$ and $B = \frac{2\Omega\xi - \alpha}{\alpha}$ tend toward infinitude when Ω is infinite, the unconditional stability will not be satisfied. Hence, here comes to a conclusion that the expression of α must contains a term of Ω^2 in order to avoid $|\lambda_{1,2}|$ tending toward more than 1 with the increasement of Ω.

Since then, the expression of α is assumed as,

$$\alpha = p\Omega^2 + q\Omega\xi + r \qquad (18)$$

where p, q and r are all uncertain parameters to be determined. Next, we will determine the parameters of p, q and r to satisfy the conditions of unconditional stability for the proposed method.

1. If $A^2 + 4B < 0$ in Eq. 17, $\lambda_{1,2}$ are complex numbers. The modulus of $\lambda_{1,2}$ is

$$|\lambda_{1,2}| = \sqrt{\left(\frac{A}{2}\right)^2 + \left(\frac{\sqrt{A^2 + 4B}}{2}\right)^2} = \sqrt{-B} \qquad (19)$$

In this case, the unconditional stability condition can be derived from $A^2 + 4B < 0$ and $\sqrt{-B} \le 1$ as,

$$-1 \le B < -\frac{A^2}{4} \qquad (20)$$

2. If $A^2 + 4B \ge 0$, then $\lambda_{1,2}$ are real numbers. Because Eq. 16 represents a parabolic curve opening upward, the unconditional stability condition of $|\lambda_{1,2}| < 1$ leads to the following inequalities:

$$f(1) \ge 0; \, f(-1) \ge 0; \, -1 \le \frac{A}{2} \le 1 \qquad (21)$$

From the inequalities in Eq. 21 and $A^2 + 4B \ge 0$, the unconditional stability condition can be derived as,

$$B \ge -\frac{A^2}{4}; \, B \le 1 - A; \, B \le 1 + A; \, -2 \le A \le 2 \quad (22)$$

Combining Eqs. 20, 22 and $\alpha = p\Omega^2 + q\Omega\xi + r$ results in the following unconditional stability condition of the proposed explicit integration method for a linear system:

$$p \ge \frac{1}{4}; \, q \ge 1; \, r \ge 0 \qquad (23)$$

Up to now, we have analyzed the expression and unconditional stability condition of the proposed explicit integration method via SDOF system. Here we recommend literature [36–38] for details of how results of α could be extended to MDOF system. If dealing with the MDOF system, $\boldsymbol{\alpha}$ should be rewritten as the following equation and the limitation for p, q and r is the same as the SDOF system.

$$\boldsymbol{\alpha} = \frac{p\boldsymbol{K}\Delta t^2 + \frac{q}{2}\boldsymbol{D}\Delta t + r\boldsymbol{M}}{\boldsymbol{M}} \qquad (24)$$

4.2 Integration for nonlinear system

For a SDOF structure system with nonlinear stiffness or nonlinear damping, α is assumed to be invariant in the entire integration procedure and is determined from the initial damping ratio d_0 and the stiffness value k_0 [6] as,

$$\alpha = p\Omega_0^2 + q\Omega_0\xi + r \, (\alpha > 0) \qquad (25)$$

Hence, the amplification matrix U can be changed to,

$$U = \begin{bmatrix} \frac{-\Omega_i^2 - 2\Omega_i\xi + 2\alpha}{\alpha} & \frac{2\Omega_i\xi - \alpha}{\alpha} \\ 1 & 0 \end{bmatrix} \qquad (26)$$

And with the same analysis step in Sect. 4.1, the unconditionally stable condition for nonlinear SDOF system can be derived as,

$$p \ge \frac{\omega_i^2}{4\omega_0^2}; \, q \ge \frac{\omega_i}{\omega_0}; \, r \ge 0 \qquad (27)$$

Back to the nonlinear MDOF system, $\boldsymbol{\alpha}$ is determined as,

$$\alpha = \frac{pK_0\Delta t^2 + \frac{q}{2}D_0\Delta t + rM}{M} \qquad (28)$$

4.3 Description of FEM with the proposed method

Initialization

1. Initialize nodal displacements: $u_i = u_{i-1} = 0$.
2. Initialize the chosen time step size.
3. Apply load for the first time step: forces $F_i \leftarrow F_0$.
4. Obtain the damping matrix D_i and stiffness matrix K_i (D_i and K_i keep invariable for linear system).

Precomputation stage

1. Load mesh and boundary conditions.
2. Select perfect values of p, q and r according to model's physical property and time step size. Compute and diagonalize the matrix $\alpha M = pK_0\Delta t^2 + \frac{q}{2}D_0\Delta t + rM$ (if the system is linear, then $K_0 = K_i$, $D_0 = D_i$).

Time stepping

1. Take element nodal displacements u_i and u_{i-1} from previous time step.
2. Perform the proposed explicit integration method:
   (1) Evaluate internal force $K_i(u_i)$ and damping force $D_i(\dot{u}_i)$.
   (2) Obtain the external force F_i.
   (3) Obtain the current displacement u_{i+1} by solving the equation: $\alpha M u_{i+1} = \Delta t^2 (F_i - K_i(u_i)) + D_i\Delta t (u_{i-1} - u_i) + \alpha M (2u_i - u_{i-1})$.
3. Update nodal displacement u_i and u_{i-1}.

Since the matrix αM is invariant during the whole computation, there is no need for solution of coupled equation in the algorithm . As a consequence, the computation cost is saved a lot by adopting the proposed explicit integration method, which is testified in detail in the next section.

5 Experiments and results

In this section, we have done two experiments to testify our method for linear system (since the analysis for nonlinear system is similar, thus here we do not show it). The first one is to compare the numerical properties of the proposed method with other existing unconditionally stable integration method. Then we conducted experiments to utilize both the implicit and proposed methods into FEM to compare their computational cost and deformation effects.

The main configuration of our experimental platform is as follows: Intel® Xeon® CPU E3-1230 V2 3.30 GHz quad-core processor, 8.0 GB memory.

5.1 Numerical properties

5.1.1 Stability

To verify the stability property, we operated experiments for several existing integration methods and the proposed method with different values of p, q and r.

First of all, we analyze the stable property for the general central difference explicit method [19] and the general implicit method [1] by spectral radius.

Next, we perform experiments for several integration methods of Chang [24], Gui [6] and Ding [25], whose detailed equations are shown in these papers. It is worth mentioning that these three methods are all unconditionally stable explicit, but they do have differences from our proposed method. These three methods are explicit for terms of $\ddot{u}_i$, while our method is explicit for terms of u_i. The solution strategy for Chang [24], Gui [6] and Ding [25] is as follows: firstly compute velocity $\dot{u}_i$ and displacement u_i; then substitute $\dot{u}_i$ and u_i into their integration algorithm to obtain the acceleration term $\ddot{u}_i$ which can be used to calculate displacement u_{i+1}. Obviously, the calculation stage for these three methods is more complex than ours. Besides, both Chang [24] and Gui [6] take nonlinear system into consideration but Ding [25] does not. After we have expanded equations in Ding [25], it is found that its integration formula is the same as that in Gui [6] for a linear system. For simplicity, here we only choose the most accurate integration formula in Chang [24], Gui [6] and Ding [25] to compare their numerical properties with ours.

At last, we choose some different values of p, q and r for our proposed method to measure its stability property. What is more, it has to be emphasized that when $p = \frac{1}{4}$, $q = 1$, $r = 1$, the results happen to overlap results of methods in Gui [6] and Ding [25].

Figure 2 shows the stability property analysis for all the methods above.

As seen, all the methods satisfy the stability condition except the central difference method. In addition, when Ω is inside a range of $(0, 2.0)$, the central difference method always keeps stable no matter how ξ changes.

5.1.2 Accuracy

Herein, the relative period elongation (PE) and the amplitude decay (AD) are introduced to further analyze the computational error of the proposed method [36]. PE is usually employed to measure period distortion for a step-by-step integration method, and AD is a measure of numerical dissipation. These two properties are theoretically analyzed to measure the solution error for motion equation solution. In this section, we will compare the deformation effects to visu-
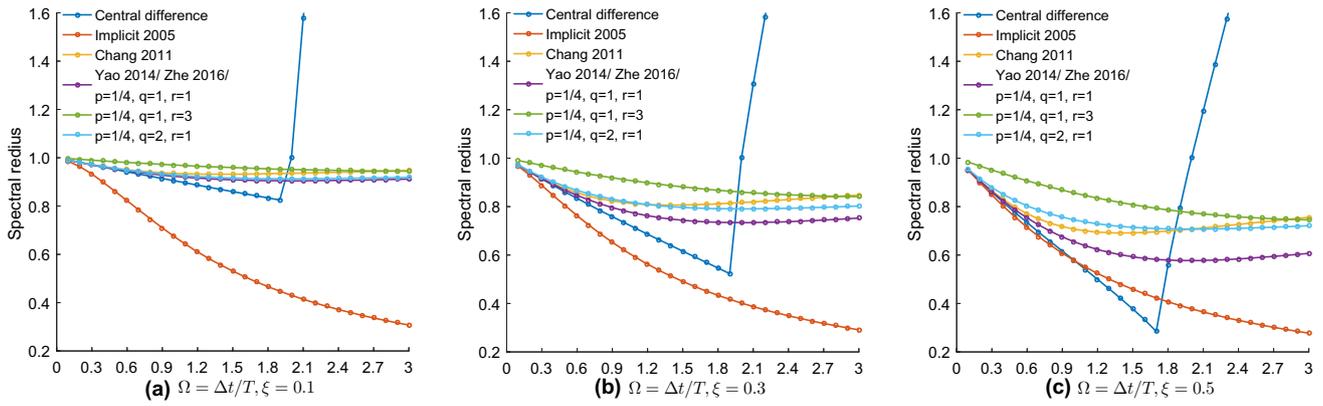
Fig. 2 Stability property: results of spectral radius for several integration methods and proposed explicit integration method
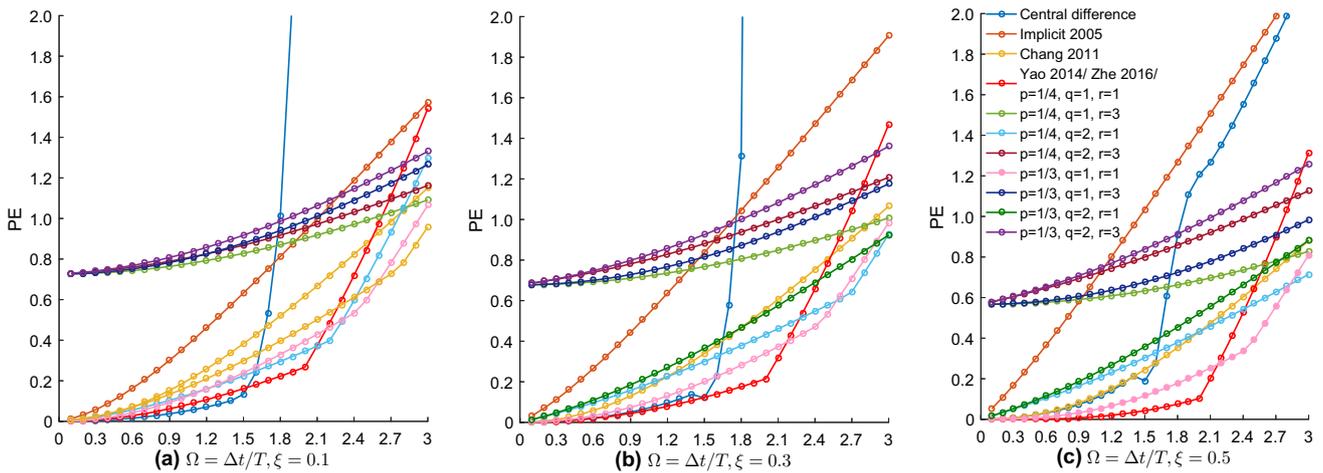
Fig. 3 Accuracy property: results of PE for several integration methods and proposed explicit integration method

ally show the difference between the implicit method and the proposed method.

If the equation for an integration algorithm is known, the eigenvalues for its corresponding amplification matrix can be expressed in an exponential form similar to Eq. 17 as,

$$\lambda_{1,2} = \varepsilon \pm \sigma i \tag{29}$$

where $i = \sqrt{-1}$, and AD and PE are defined as,

$$PE = \frac{\Omega - \bar{\Omega}}{\bar{\Omega}} \tag{30}$$

$$AD = \ln\left(\sigma^2 + \varepsilon^2\right) / (2\bar{\Omega}) \tag{31}$$

$$\bar{\Omega} = \frac{\arctan\left(|\sigma/\varepsilon|\right)}{\sqrt{1 - \xi^2}} \tag{32}$$

We compute the accuracy property of PE and AD for methods mentioned in Sect. 5.1.1 which are shown in the following two figures.

The period elongation result shown in Fig. 3 indicates the following several points.

(a) Among all the methods, central difference explicit integration method shows the least period distortion at a small value of Ω and ξ. But if Ω increases, its PE values dramatically becomes larger and exceeds all the other methods;

(b) Although the implicit method is prior to most explicit methods because of its unconditionally stability, its PE value and growth rate is pretty large;

(c) Great selection for values of p, q and r for our method should be based on the value of Ω. For example, if Ω is at a range of rather small value, a selection of $p = 1/4, q = 1, r = 1$ is the best. When Ω is pretty large, another selection like $p = 1/4, q = 1, r = 3$ offers the least period distortion;

(d) The PE property of Chang [24] method is at a middle state and that of Gui [6] and Ding [25] happen to coincide to the proposed method with $p = 1/4, q = 1, r = 1$. However, our method can switch its PE value by initializing different parameters p, q and r and its calculation process is much more convenient;
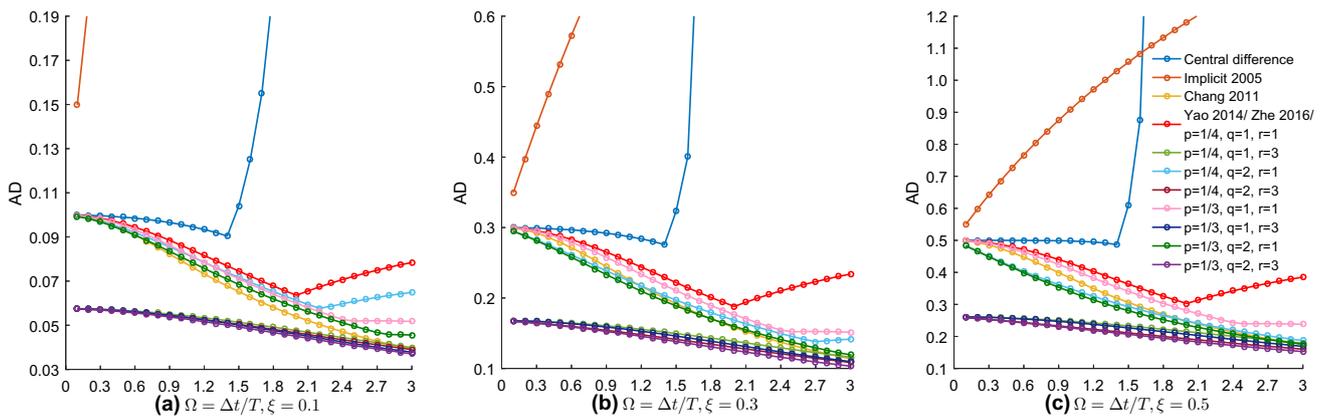
Fig. 4 Accuracy property: results of AD for several integration methods and proposed explicit integration method

(e) For the proposed method, PE value of a larger ξ is smaller than that of a smaller ξ. Therefore damping force can restrain period distortion in some way.

Similarly, here are some conclusions for the AD analysis in Fig. 4.

(a) The AD values of central difference explicit integration method and implicit integration method indicate the extremely large amplitude decay;
(b) AD falls down when values of p, q, r increase. In the details of such variation, changes of r bring the biggest influence to growth or reducing rate;
(c) In contrast to Fig. 3, the tendency of AD values first shows negative growth and then returns to positive growth with the increasement of Ω;
(d) The AD property of Chang [24] method is also at a middle state, and that of Gui [6] and Ding [25] is larger;
(e) Opposite to PE results, damping force cannot restrain amplitude decay.

If only concerned about the numerical properties, the experiments in this section have proved that the proposed explicit integration method has the following advantages.

1. Compared to most existing explicit integration method, our method is unconditionally stable for both linear and nonlinear systems.
2. The computation procedure for FEM's dynamic equations at each time step of our method is much more intuitive and convenient than other explicit methods presented recently.
3. There are 3 adjustable parameters in our method which can be determined according to the current calculation conditions, thus optimizing computational errors.

Table 2 Integration time and frame rate of FEM using implicit and proposed method for six models

Model	Method	Time	FPS
Kidney	Implicit	157.0	6.3
	Proposed	36.0	27.7
Armadillo	Implicit	429.0	2.3
	Proposed	137.0	7.2
Bunny	Implicit	780.0	1.2
	Proposed	272.0	3.6
Lung	Implicit	1298.0	0.7
	Proposed	409.0	2.4
Heart	Implicit	1610.0	0.6
	Proposed	541.0	1.8
Brain	Implicit	1940.0	0.5
	Proposed	701.0	1.4

5.2 FEM analysis

5.2.1 Efficiency analysis

To apply the proposed method to FEM, we have employed six different models to perform our experiments, which are mentioned before as shown in Table 1.

We compare the integration time and frame rate for FEM by utilizing implicit method and proposed method in this section. As mentioned before, while considering values of frame rate, we here ignore the rendering time because it will be influenced by model's complexity. The integration time and frame rate of the two methods for six models are shown in Table 2 and Fig. 5.

It is quite clear that the integration time for the proposed method decreases a lot compared to the implicit method. More importantly, the computational cost advantage is more significant if the number of dimensions is larger. Correspon-
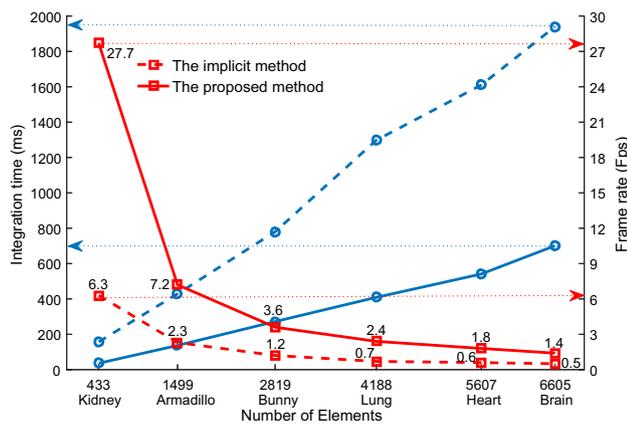
Fig. 5 Integration time and frame rate of FEM using implicit and proposed method for six models
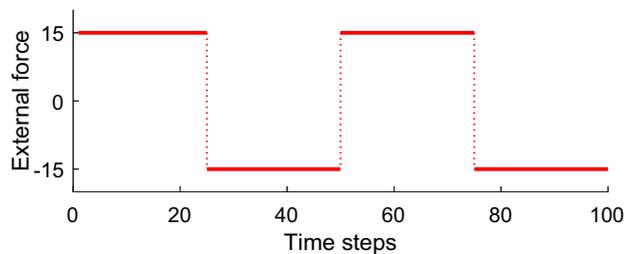
Fig. 6 External force applied in deformation simulation

Fig. 7 Displacement at X-axis of the implicit and proposed method for Kidney model

dently, it elevates the frame rate of six models in Table 2 for the proposed method compared to the implicit method.

Therefore, herein we can draw a conclusion from the experimental results above that the proposed explicit integration method can efficiently decrease the computational cost in integration procedure compared to implicit methods. However, during FEM simulation process, such advantage for frame rate is significant for models with medium scales. With regard to too larger-scale model, such advantage is, however, unconspicuous resulting from the incremental consuming time for integration procedure.

5.2.2 Deformation effects

In this subsection we simply present the deformation effects of the Kidney model applied with the implicit integration method and the proposed method. Firstly we choose a certain point on the Kidney model. Then we exert external force on the point and the direction of the force is to the left which is also at the X-axis during calculation. The value of external force for the whole 100 time steps is indicated in Fig. 6, which is expressed as f in the following. In addition, we set $p = 1/4, q = 1, r = 1$ for α in our proposed integration method.

During simulation, we record the 100-time step displacement values for the force loaded point at X-axis of the implicit

method and proposed method, respectively. As shown in Fig. 7, We exhibit the displacement value for the implicit method and proposed (explicit) method. However, as a result of numerical errors described in Sect. 5.1.1, the solutions of the two methods with f are much different. Loaded with f, the deformation amplitude of the implicit method is obvious. On the contrary, it seems no deformation change for the proposed method under external force f.

We additionally record the proposed method's deformation displacement values under external force of $10f, 100f, 200f$, as shown in Fig. 7. Apparently, the deformation trends of the two methods are the same, while the deformation degrees are different. Compared to the implicit with f, the proposed method with 100 has the most similar displacement values.

Actually, in dynamic deformation simulation, the two methods are both visually realistic although they have different solutions. The accuracy superiority for our method demonstrated in Sect. 5.1.1 owns its advantages for those applications with accuracy requirement.

The deformation effects are shown in Fig. 8, where five figures of 100-time step simulation are selected, respectively, for the implicit method with f and proposed method with $100f$ and $200f$. The deformation effects on the force bearing point are much similar, while on the other parts of the Kidney model they have differences. On the side aspect, it demonstrates that during calculation, the solution values of two methods exist difference, as indicated in Sect. 5.1.2.

6 Conclusion and future work

In this paper, we resort to structural mechanics and mathematical analysis to present a novel unconditionally stable explicit integration method for both linear and nonlinear
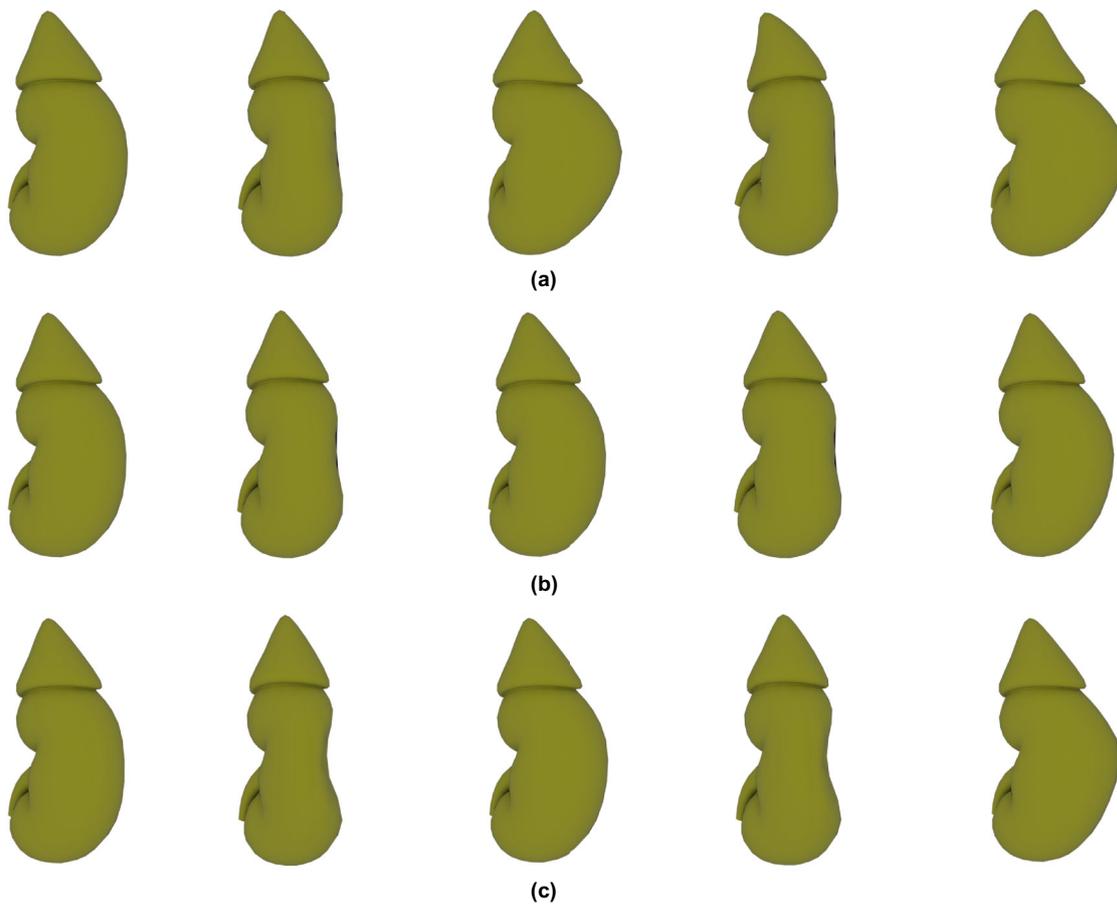
Fig. 8 Deformation effects of the implicit and proposed method for Kidney model. **a** Deformation effect of the implicit method with f, **b** deformation effect of the proposed method with $100 f$, **c** deformation effect of the proposed method with $200 f$

FEM to achieve more efficient integration calculation. First we advocate an explicit integration formula with three parameters. Then by mathematically analyzing the spectral radius of the dynamic transfer function's amplification matrix, we obtain limitations for these three parameters' values in order to meet the unconditional stability conditions. Besides, the 3 parameters can be determined according to the current calculation conditions, so as to optimize the calculation errors. Finally, the accuracy property of the proposed method is analyzed theoretically and verified numerically.

The experimental results indicate that our method is unconditionally stable for both linear and nonlinear systems compared to most existing explicit integration methods. More significantly, the accuracy property of our method is superior to not only the common explicit and implicit methods used in FEM but also methods put forward recently. Then FEM is redescribed using the proposed integration methods, and it is found that its computation procedure for dynamic equations at each time step is much more intuitive and convenient compared to current explicit and implicit methods. According to real implementation of our method

for FEM, the method can efficiently solve the problem of huge computational cost in integration procedure because of its none-factorization of unknown matrices. Superiorities above enable researchers to realize both efficient and accurate simulations of FEM. As a side note, the derivation process of our method is also applicable for other physics-based methods with dynamic integration procedure.

Currently, our method has observably improved the computational time-consuming in integration procedure compared to implicit methods. Nonetheless, there exist some shortcomings in our method. For example, generally an explicit integration method is bad at handling collisions in graphics field. Therefore, we will concentrate on introducing and realizing collision detection functions as well as self-collision caused by elastic deformation in our experiments to make our explicit method more applicable and flexible. In addition, in this paper we compare our method with the implicit Newmark method (frequently used in finite element Method). However, there are many state-of-art methods focusing on integration optimization in other deformation simulation areas like position-based dynamics, as mentioned

in related work. We should further improve our method to extend its applicability for more simulation techniques and compare it with those state-of-art methods in future work.

References

1. Erleben, K., Sporring, J., Henriksen, K., Dohlmann, H.: Physics-based animation. Point Based Graph. **30**(6), 340–387 (2005)
2. Taylor, Z.A., Cheng, M., Ourselin, S.: High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. IEEE Trans. Med. Imaging **27**(5), 650–663 (2008)
3. Dick, C., Georgii, J., Westermann, R.: A real-time multigrid finite hexahedra method for elasticity simulation using cuda. Simul. Model. Pract. Theory **19**(2), 801–816 (2011)
4. Yang, C., Li, S., Lan, Y., Wang, L., Hao, A., Qin, H.: Coupling time-varying modal analysis and fem for real-time cutting simulation of objects with multi-material sub-domains. Comput. Aided Geom. Des. **43**, 53–67 (2016)
5. Barbic, J.: Real-time subspace integration for St. Venant–Kirchhoff deformable models. Acm Trans. Graph. **24**(3), 982–990 (2005)
6. Gui, Y., Wang, J.T., Jin, F., Chen, C., Zhou, M.X.: Development of a family of explicit algorithms for structural dynamics with unconditional stability. Nonlinear Dyn. **77**(4), 1157–1170 (2014)
7. Hirota, G., Fisher, S., State, A.: An improved finite-element contact model for anatomical simulations. Vis. Comput. **19**(5), 291–309 (2003)
8. Choi, M.G., Ko, H.S.: Modal warping. Real-time simulation of large rotational deformation and manipulation. IEEE Trans. Vis. Comput. Graph. **11**(1), 91–101 (2005)
9. Yang, Y., Xu, W., Guo, X., Zhou, K., Guo, B.: Boundary-aware multidomain subspace deformation. IEEE Trans. Vis. Comput. Graph. **19**(10), 1633–1645 (2013)
10. Krysl, P., Lall, S., Marsden, J.E.: Dimensional model reduction in non-linear finite element dynamics of solids and structures. Int. J. Numer. Methods Eng. **51**(4), 479–504 (2001)
11. Barbic, J.: Real-time reduced large-deformation models and distributed contact for computer graphics and haptics, Ph.D. thesis. Carnegie Mellon University, Pittsburgh (2007), AAI3279452
12. Yang, Y., Li D., Xu, W., Tian, Y., Zheng, C.: Expediting precomputation for reduced deformable simulation. ACM Trans. Graph. **34**(6), 243:1–243:13 (2015)
13. Hauth, M., Etzmuss, O., Strasser, W.: Analysis of numerical methods for the simulation of deformable models. Vis. Comput. **19**(7), 581–600 (2003)
14. Hussein, B., Dan, N., Shabana, A.A.: Implicit and explicit integration in the solution of the absolute nodal coordinate differential/algebraic equations. Nonlinear Dyn. **54**(4), 283–296 (2008)
15. Newmark, N.M.: A method of computation for structural dynamics. J. Eng. Mech. Div. **85**(1), 67–94 (1959)
16. Belytschko, T.: An overview of semidiscretization and time integration procedures. In: Belytschko, T., Hughes, T.J.R. (eds.) Computational Methods for Transient Analysis, pp. 1–66. North-Holland Publ., North-Holland, Amsterdam (1983)
17. Wilson, E.L.: A computer program for the dynamic stress analysis of underground structures, SEL. In: Technical Report 68-1, University of California, Berkeley (1968)
18. Bathe, K., Wilson E.L.: Numerical Methods in Finite Element Analysis. Prentice-Hall, Englewood Cilffs, New Jersey (1976)
19. Miller, K., Joldes, G., Lance, D., Wittek, A.: Total lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. Commun. Numer. Methods Eng. **23**(2), 121–134 (2007). **(Gui2014Development)**
20. Kang, Y.M., Choi, J.H., Cho, H.G., Lee, D.H.: An efficient animation of wrinkled cloth with approximate implicit integration. Vis. Comput. **17**(3), 147–157 (2001)
21. Oh, S., Ahn, J., Wohn, K.: Low damped cloth simulation. Vis. Comput. **22**(2), 70–79 (2006)
22. Chang, S.Y.: Explicit pseudodynamic algorithm with unconditional stability. Am. Soc. Civ. Eng. **128**(9), 935–947 (2002)
23. Chang, S.Y.: An explicit method with improved stability property. Int. J. Numer. Methods Eng. **77**(8), 1100–1120 (2009)
24. Chang, S.Y., Yang, Y.S., Chi, W.: A family of explicit algorithms for general pseudodynamic testing. Earthq. Eng. Eng. Vib. **10**(1), 51–64 (2011)
25. Ding, Z., Li, L., Hu, Y., Li, X., Deng, W.: State-space based time integration method for structural systems involving multiple nonviscous damping models. Comput. Struct. **171**, 31–45 (2016)
26. Chang, S.Y.: Improved explicit method for structural dynamics. J. Eng. Mech. **133**(7), 748–760 (2007)
27. Chen, C., Ricles, J.M.: Development of direct integration algorithms for structural dynamics using discrete control theory. J. Eng. Mech. **134**(8), 676–683 (2008)
28. Bouaziz, S., Martin, S., Liu, T., Kavan, T., Pauly, M.: Projective dynamics: fusing constraint projections for fast simulation. ACM Trans. Graph. **33**(4), 154:1–154:11 (2014)
29. Wang, H.: A Chebyshev semi-iterative approach for accelerating projective and position-based dynamics. ACM Trans. Graph. **34**(6), 246:1–246:9 (2015)
30. Wang, H., Yang, Y.: Descent methods for elastic body simulation on the GPU. ACM Trans. Graph. **35**(6), 212:1–212:10 (2016)
31. Liu, T., Bouaziz, S., Kavan, L.: Towards real-time simulation of hyperelastic materials. arXiv:1604.07378
32. Pentland, A., Williams, J.: Good vibrations: modal dynamics for graphics and animation. Acm Siggraph Computer Graphics. **23**(3), 207–214 (1989)
33. Wriggers P.: Computational contact mechanics. Wiley (2002)
34. Hughes, T.R.J.: The finite element method. In: Linear static and dynamic finite element analysis. Prentice-Hall, Englewood Cilffs, New Jersey (2000)
35. Rezaiee-Pajand, M., Sarafrazi, S.R., Hashemian, M.: Improving stability domains of the implicit higher order accuracy method. Int. J. Numer. Methods Eng. **88**(9), 880–896 (2011)
36. Bathe, K.J., Wilson, E.L.: Stability and accuracy analysis of direct integration methods. Earthq. Eng. Struct. Dyn. **1**(3), 283–291 (1972)
37. Tamma, K.K., Zhou, X., Sha, D.: A theory of development and design of generalized integration operators for computational structural dynamics. Int. J. Numer. Methods Eng. **50**(7), 1619–1664 (2001)
38. Tamma, K.K., Sha, D., Zhou, X.: Time discretized operators. part 1: towards the theoretical design of a new generation of a generalized family of unconditionally stable implicit and explicit representations of arbitrary order for computational dynamics. Comput. Methods Appl. Mech. Eng. **192**(3), 291–329 (2003)

Mianlun Zheng is a master student at the School of Computer, Wuhan University. Her research interests include computer graphics and virtual reality.

Guian Zhang is a Ph.D. student at the School of Computer, Wuhan University. His research interests include computer vision and image processing.

Zhiyong Yuan received his B.Sc and M.Sc degrees from the Department of Computer of Wuhan Technical University of Surveying and Mapping in 1986 and 1994, respectively, and received his Ph.D. from the Department of Control Science and Engineering of Huazhong University of Science and Technology in 2008, China. He is currently a professor at the School of Computer, Wuhan University, China. His research interests include virtual reality, human-computer interaction, embedded system, internet of things technology, machine learning and pattern recognition.

Weixu Zhu is a master student at the School of Computer, Wuhan University. His research interests include human-computer interaction and artificial intelligence.

Qianqian Tong is a Ph.D. student at the School of Computer, Wuhan University. Her research interests include human-computer interaction, virtual reality and embedded system.