Simple Numerical Methods for Mechanical Systems Subject to Strong forces and Constraints

Claude Lacoursière
Deptartment of Computing Science
Room C240, MIT Huset
email: claude@hpc2n.umu.se

December 1, 2011

1 Introduction

When simulating mechanical systems made of point masses or rigid bodies interacting with forces, it is often useful to introduce *geometric constraints* of the form $g(q) = 0$ where q are the generalized coordinates and g is a smooth function that can be differentiated at least once.

In addition, if a numerical simulation is used within an interactive application that includes real-time components such as 3D graphics, it is often best to use a fixed integration step. In that case, one needs to pay particular attention to filter out high frequencies which cause instability.

The following lectures notes provide a simple and intuitive understanding of numerical instabilities of simple linear mechanical system. Simple and popular low order integration methods are analyzed carefully. A simple example is provided to illustrate the connection between strong forces and geometric constraints, and this is used to warn against blind use of strong penalty forces.

A discrete-time framework is then provided for the analysis of simple stepping procedures that can handle either constraints or very strong penalty forces without instabilities and without destroying the physics of the problem.

2 Stability Analysis of the numerical simple harmonic oscillator

The physical system considered here is a one dimensional point mass with coordinate $x(t)$, velocity $v = \dot{x}$, scalar mass m, and subject to the force

$$f = -m\omega^2 x, \tag{1}$$

where ω is the natural frequency of the oscillator. This force can also be derived from the potential function

$$U = \frac{m\omega^2}{2}x^2, \text{with} \quad f = -\frac{\partial U}{\partial x^T}. \tag{2}$$

The reason for the transpose in the last expression is so that f is a column vector. By convention, the gradient $\partial U(x)/\partial x$ is a row vector.

The equations of motion for this problem are easily derived to read

$$m\ddot{x} = -m\omega^2 x, \text{ and after cancelling mass factors, } \ddot{x} = -\omega^2. \tag{3}$$

The solution to this problem is easily verified to be

$$x(t) = x(0)\sin(\omega t) + \frac{v(0)}{\omega}\cos(\omega t). \tag{4}$$

This is a superposition of sine waves of constant amplitude. In fact, one can check that the energy of this system is constant, i.e.,

$$E = \frac{m}{2}\dot{x}^2 + \frac{m\omega^2}{2}x^2 = const, \tag{5}$$

which means that x and $\dot{x}$ move on an ellipse in the $(x, \dot{x})$ plane.

If time is discretized with $t_k = kh$ where $h > 0$ is the fixed time step and k is an integer, the discrete sampled solution reads

$$x_k = x(0)\sin(\tau k) + \frac{v(0)}{\omega}\cos(\tau k), \tag{6}$$

where $\tau = h\omega$ is the natural discrete time unit. Qualitatively, this solution is bounded and sinusoidal. A good discretization will have the same qualities.

Naturally, the period of the oscillator is $T = 2\pi/\omega$.

3 Linear recurrence of second order

Since the force is linear, and since any approximation of $\ddot{x}$ will be linear in the discrete time samples x_k, the discrete time equations of motion will be second order recurrence relations of the form:

$$x_{k+1} = -ax_k - bx_{k-1}, \tag{7}$$

where x_k is the position at the discrete time $t = kh$, where $h > 0$ is the time step, and k is the integer discrete time. For higher order methods, the recurrence would be of higher order, of course.

Such recurrences are easily analyzed by postulating that $x_k = \lambda^k$. After substitution and cancellation of a common factor of λ^{k-1}, we get the indicial equation

$$\lambda^2 + a\lambda + b = 0, \tag{8}$$

which has solutions

$$\lambda_\pm = -\frac{a}{2} \pm \sqrt{\frac{a^2}{4} - b}. \tag{9}$$

Since we are looking for oscillatory solutions, we assume that the discriminant (the term under the square root in (9)) is negative,

$$\begin{aligned} &\frac{a^2}{4} - b \leq 0, \quad \text{, or in other words ,} \\ &\frac{a^2}{4b} < 1 \text{ when } \quad b > 0. \end{aligned} \tag{10}$$

The sign of the inequality $a^2/4b < 1$ would be reversed if we had $b < 0$. When $b < 0$, the discriminant is necessarily positive as is obvious from the first line in (10).

The roots can then be expressed as the complex conjugates,

$$\lambda_\pm = r \exp(\pm i\phi), \tag{11}$$

where i is the imaginary unit, and from this, we can express r and $\cos\phi$ in terms of the original coefficients

$$r^2 = r\exp(-i\phi) r \exp(i\phi) = \lambda_+ \lambda_- = \frac{a^2}{4} - (\frac{a^2}{4} - b) = b, \tag{12}$$

and

$$\cos\phi = \frac{\lambda_+ + \lambda_-}{2r} = -\frac{a}{2\sqrt{b}}, \tag{13}$$

where use was made of the known formulae

$$\frac{\exp(i\phi) + \exp(-i\phi)}{2} = \cos\phi, \quad \text{and} \quad \frac{\exp(i\phi) - \exp(-i\phi)}{2i} = \sin\phi, \tag{14}$$

for $\phi \in \mathbb{R}$, and, of course, we can solve (13) for ϕ provided $|a/\sqrt{b}| < 2$.

A general solution to the recurrence is then a linear combination of sinusoidal oscillations:

$$x_k = r^k \left[\alpha \sin(k\phi) + \beta \cos(k\phi)\right], \tag{15}$$

and the coefficients are to be determined by the initial conditions x_0, x_1.

Qualitatively, the solution escapes to infinity when $r > 1$, or decays to 0 when $r < 1$. The solution is physically correct if $r = 1$ as this produces bounded trajectories.

4 Stability analysis of the Verlet stepper

The stepping formula for the Verlet stepper is defined as

$$m\frac{x_{k+1} - 2x_k + x_{k-1}}{h^2} = f_k = -m\omega^2 x_k. \tag{16}$$

After rearrangements and the cancellation of the mass factors, this becomes

$$x_{k+1} = 2x_k - x_{k-1} - \tau^2 x_k, \tag{17}$$

and collecting the x_k terms, the result is

$$x_{k+1} = 2(1 - \tau^2/2)x_k - x_{k-1}. \tag{18}$$

The coefficients of the recurrence relations (7) are then

$$\begin{aligned} a = -2(1 - \tau^2/2), \quad &\text{and } b = 1, \text{ so that} \\ \frac{a^2}{4b} &= (1 - \tau^2/2)^2. \end{aligned} \tag{19}$$

Since $b = 1$, the solution is bounded if the discriminant is negative, which implies

$$\begin{aligned} -1 &\leq 1 - \tau^2/2 \leq 1 \\ -2 &\leq -\tau^2/2 \leq 0 \\ 0 &\leq \tau^2/2 \leq 2 \\ 0 &\leq \tau^2 \leq 4 \\ 0 &\leq \tau = h\omega \leq 2. \end{aligned} \tag{20}$$

Therefore, the Verlet integrator can integrate the simple harmonic oscillator as long as the natural discrete time $\tau = h\omega$ is less than two, which means that we need a time step $h < 2/\omega = T/\pi$. As long as we have approximately three samples per period, the integrator is doing fine.

The analytic solution is then

$$\begin{aligned} x_k &= \alpha \sin(k\phi) + \beta \cos(k\phi), \\ \cos(\phi) &= -\frac{a}{2\sqrt{b}} = 1 - \tau^2/2. \end{aligned} \tag{21}$$

Comparing with the exact solution where $\cos\tau = 1 - \tau^2/2 + \ldots$, this is not far off.

5 Stability analysis of the explicit Euler stepper

The explicit Euler stepper is defined from the following system of ODEs

$$\begin{aligned} \dot{x} &= v \\ m\dot{v} &= f, \end{aligned} \tag{22}$$

and these are simply a reformulation of $m\ddot{x} = f$. The system (22) is then discretized as

$$\begin{aligned} x_{k+1} &= x_k + hv_k \\ v_{k+1} &= v_k - h\omega^2 x_k. \end{aligned} \tag{23}$$

This only uses information from he past to step forward. Eliminating the velocity variables leads to

$$\frac{x_{k+2} - x_{k+1}}{h} = \frac{x_{k+1} - x_k}{h} - h\omega^2 x_k \tag{24}$$

which is rearranged to

$$x_{k+2} = 2x_{k+1} - (1 + \tau^2)x_k, \tag{25}$$

and after shifting the index k by one, the coefficients of the recurrence relation are

$$\begin{aligned} a &= -2, \\ b &= 1 + \tau^2 > 1, \\ \frac{a^2}{4b} &= \frac{1}{1 + \tau^2} < 1. \end{aligned} \tag{26}$$

So, this solution is fundamentally unstable since $b > 1$ but nevertheless, it is oscillatory. The solution grows as $(1 + \tau^2)^{k/2}$ which is not good at all. Some damping can help. In fact, you can repeat the analysis and see what happens with the introduction of a damping force of the form $-m\gamma\dot{v}$. It turns out that damping does not help unless the parameter γ is carefully chosen in a tight range that depends on h and ω. Since there is no stability here, let's move on.

6 Stability analysis of the implicit Euler stepper

Starting from the system (22), the implicit Euler stepper reads

$$\begin{aligned} x_{k+1} &= x_k + hv_{k+1} \\ v_{k+1} &= v_k + h\omega^2 x_{k+1}. \end{aligned} \tag{27}$$

Notice that the unknown x_{k+1} appears both on the left and the right hand side of the equations which means that we now have to solve a system of equations. Because this is the formula you would get by reversing time in the explicit Euler discretization (23), it is often called the "backward Euler formula". In general, there is a nonlinear system to solve but here, it is a linear system with one variable so we don't have to work too hard.

After eliminating v_k as was done in § 5, the result is

$$x_{k+1} = 2x_k - x_{k-1} - \tau^2 x_{k+1}, \tag{28}$$

and this is renormalized to produce

$$x_{k+1} = \frac{2}{1 + \tau^2} x_k - \frac{1}{1 + \tau^2} x_{k-1}. \tag{29}$$

The coefficients of the recurrence are now

$$\begin{aligned} a &= -\frac{2}{1+\tau^2}, \\ b &= \frac{1}{1+\tau^2} < 1, \\ \frac{a^2}{4b} &= \frac{1}{1+\tau^2} < 1. \end{aligned} \tag{30}$$

This solution decays to 0 as $1/(1+\tau^2)^{k/2}$ and is always oscillatory. The frequency of oscillation is now

$$\cos(\phi) = -\frac{a}{2\sqrt{b}} = 1/\sqrt{1+\tau^2}, \tag{31}$$

and as τ becomes large, we have $\cos\phi \to 0$, which is not the correct physics.

The first order implicit Euler stepper is very stable but overly dissipative which is not necessarily good, unless you do want to damp the dynamics of your system. To be used with caution.

7 Stability analysis of the implicit midpoint stepper

The implicit midpoint rule is due to Gauss and it is particularly useful in simulating physical conservative systems. Starting from (22), we express all the terms on the right hand side as averages as follows

$$\begin{aligned} x_{k+1} &= x_k + h\frac{v_k + v_{k+1}}{2} \\ v_{k+1} &= v_k + h\omega^2\frac{x_{k+1} + x_k}{2}. \end{aligned} \tag{32}$$

To eliminate v_k out of this, we need to add the shifted velocity equation

$$v_k = v_{k-1} + h\omega^2\frac{x_k + x_{k-1}}{2}, \tag{33}$$

and after adding that to the second line of (32), we get

$$\frac{v_{k+1} + v_k}{2} = \frac{v_k + v_{k-1}}{2} + \frac{h\omega^2}{2}\left[\frac{x_{k+1} + x_k}{2} + \frac{x_k + x_{k-1}}{2}\right]. \tag{34}$$

Substituting for x using the first line of (32), we get,

$$x_{k+1} - x_k = x_k - x_{k-1} - \frac{\tau^2}{4}(x_{k+1} + 2x_k + x_{k-1}). \tag{35}$$

This is a lot like the Verlet formula, replacing f_k with the three point weighted average:

$$< f_k > = \frac{1}{4}[f_{k+1} + 2f_k + f_{k-1}]. \tag{36}$$

It is this averaging that makes things better as we now show.

Collecting terms and normalizing, this becomes

$$x_{k+1} = 2\frac{1-\tau^2/4}{1+\tau^2/4}x_k - x_{k-1}. \tag{37}$$

The coefficients of the recurrence relation are

$$\begin{aligned} a &= -2\frac{1-\tau^2/4}{1+\tau^2/4}, \\ b &= 1, \\ \frac{a^2}{4b} &= \frac{(1-\tau^2/4)^2}{(1+\tau^2/4)^2} < 1. \end{aligned} \tag{38}$$

So here, irrespective of the value of τ, we have oscillatory and uniformly bounded dynamics. The cosine of the phase here is then

$$\cos\phi = -\frac{a}{2\sqrt{b}} = \frac{1-\tau^2/4}{1+\tau^2/2}, \tag{39}$$

which has the correct limit both as $\tau \to 0$, and as $\tau \to \infty$. In the latter case, this gives $\cos\phi \to -1$, as $\tau \to \infty$.

This stepper also preserves energy exactly. Mind you, a trajectory x_k for very large τ will be essentially random noise with amplitude 2 and mean 0, i.e., $x_k \in [-1, 1]$, $< x_k >= 0$, where $< \cdot >$ is the expectation value.

8 Strong potentials and constraints

For macroscopic systems, we do not always have a force law but instead, we have some geometric condition such as $g(q) = 0$, where q are the generalized coordinates and g is a continuous function with at least one good derivative. For instance, a pendulum satisfies the constraint $\|q\| - l_0 = 0$, where l_0 is the rest length.

If the constraint $g(q) = 0$ is met at all times, then, $\dot{g}(q) = G\dot{q} = 0$, where $G = \partial g/\partial q$ is the Jacobian matrix of $g(q)$. In 2D at least, the Jacobian is easily demonstrated to be normal to the curve $g(q) = 0$ and the same holds in n dimensions: the matrix G at the point q_0, determines the tangent plane to $g(q_0) = 0$ with the equation $G(q_0) \cdot (q - q_0) = 0$. So, basically, if there are forces f_e acting on the system, the effect of the constraint is prevent any force acting in the direction normal to the constraint surface. In other words, a quantity $G^T\lambda$ has to be subtracted from f_e to make sure that the velocity stays tangent to the surface.

At least in terms of analytic differential equations, this statement amounts to writing the following system of equations

$$\begin{aligned} M\ddot{q} &= f_e + G^T\lambda \\ g(q) &= 0. \end{aligned} \tag{40}$$

So, there is one component of λ for each constraint component, $g_i(q)$. At least formally, we have enough equations to solve for $\ddot{q}$ and λ given q and $\dot{q}$. However, there is no term involving $\dot{\lambda}$ and that is what makes the system (40) difficult to solve.

Of course, there is no perfectly rigid kinematic constraint in the whole universe. There are exact symmetries and conservation laws but even then, at the quantum mechanical level, these are not entirely exact either. Therefore, the interesting thing to study is what happens *near* the limit.

To understand that, consider a small number $\epsilon > 0$ and the potential

$$U(q) = \frac{1}{2\epsilon}\|g(q)\|^2, \tag{41}$$

where $\|x\|^2$ is the standard Euclidean norm. Such a potential produces the force

$$f_\epsilon = -\frac{\partial U}{\partial q^T} = -\frac{1}{\epsilon}G^T g(q), \tag{42}$$

so we almost have the right equations if we now write $\lambda_\epsilon = -(1/\epsilon)g(q)$ and thus

$$\begin{aligned} M\ddot{q} &= f_e + G^T\lambda \\ \epsilon\lambda + g(q) &= 0. \end{aligned} \tag{43}$$

Formally, nothing has changed since λ can be eliminated from this system of equations. The problem is to understand what happens near $\epsilon \to 0$. The full theory for that is beyond the scope of these notes though but the following section contains an illustrative example.

9 A constraint realization example

Consider a 2D particle with unit mass subject to downward unit gravity producing the force $f_g = -(0,1)^T$. The particle is forced to live on the plane $g(q) = y = 0$, so the equations of motion of the constrained problem read

$$\begin{aligned} \ddot{x} &= 0 \\ \ddot{y} &= -1 + G^T\lambda \text{ (note that } G = 1) \\ g(q) = y &= 0. \end{aligned} \tag{44}$$

This is easily solved to yield the solution

$$\begin{aligned} x(t) &= v_0 t + x_0 \\ y(t) &= 0 \\ \lambda(t) &= 1. \end{aligned} \tag{45}$$

However, now, starting with $y(0) = 0$ and $\dot{y}(0) = 0$, subject the particle to the strong force $-\frac{1}{\epsilon}y$ and the equations are now

$$\begin{aligned} \ddot{x} &= 0 \\ \ddot{y} &= -1 - \frac{1}{\epsilon}y. \end{aligned} \tag{46}$$

Before continuing, observe here that the energy in this conservative system is just $E = (1/2)(\dot{x}^2 + \dot{y}^2) + y + (1/\epsilon)y^2$. Since this is a constant, we have

$$\frac{1}{\epsilon}y^2 \leq E, \tag{47}$$

and this means in fact that

$$\sqrt{\|g(q)\|^2} = O(\sqrt{\epsilon}). \tag{48}$$

This holds for general constraints and it does show that the constraint violation does decrease in a uniformly bounded way. You can already see that the limit as $\epsilon \to 0$ is not a slam dunk though since $\lambda_\epsilon = -(1/\epsilon)g(q)$, so the best estimate from the energy argument is the useless bound $\lambda_\epsilon = O(1/\sqrt{\epsilon})$.

But now, the solution to (46), given the initial conditions, are found to be

$$\begin{aligned} x &= v_0 t + x_0 \\ y &= \epsilon(-1 + \sin(t/\sqrt{\epsilon})), \\ \lambda_\epsilon &= -\frac{1}{\epsilon}g(q) = -\frac{1}{\epsilon}y = (-1 + \cos(t/\sqrt{\epsilon})). \end{aligned} \tag{49}$$

Now, as $\epsilon \to 0$, $y(t) \to 0$ uniformly, without any problem. However, λ_ϵ does not converge because it contains a highly oscillatory term. In fact, $\dot{y}$ does also converge nicely.

The general case here, and this is the Rubin and Ungar theorem[5], is that the trajectory and the velocities converge uniformly as $\epsilon \to 0$ provided we start on the constraint surface and in a direction tangent to it. However the strong forces do not converge in general except in the weak sense, i.e.,

$$\int_{t_0}^{t_1} \mathrm{d}s \lambda_\epsilon(s) a(s) \to \int_{t_0}^{t_1} \mathrm{d}s \lambda(s) a(s), \tag{50}$$

for any smooth function $a(s)$ and any interval $[t_0, t_1], t_1 > t_0$, and where $\lambda(s)$ is the constraint force as computed by solving the constrained system (40). One way to understand weak limits is to say that the time average of a quantity converges, but not the samples themselves.

Of course, adding constraint damping so that

$$\epsilon\lambda + g(q) + \gamma\dot{g}(q) = 0, \tag{51}$$

does also guarantee strong and uniform convergence. There is no room here to discuss that fact further even though that will be part of the stepping equations.

This simple example is a very strong warning that penalty forces should not be used without doing a lot of analysis first, or without damping. The fast oscillatory terms which develop as $\epsilon \to 0$ can be trying for any integrator, with the possible exception of implicit Euler. The implicit midpoint rule would generate random noise from that which is not a pleasant thing to see. These terms have to be damped adequately, in a numerically stable way, is one is to simulate a very nearly constrained system.

10 Linearly Implicit Euler for strong forces

The graphics literature "rediscovered" linearly implicit integrators with a famous SIGGRAPH 1998 papers about cloth which shall remain nameless here. In there, the wonderful stability virtues of the implicit Euler stepper were described as a panacea (note: panacea is a universal remedy that cures everything). But as we now show, it is a panacea only if death is regarded as the healthy state.

Consider the strong potential $U = \frac{1}{2\epsilon}\|g(q)\|^2$ generating the force $f_\epsilon = -\frac{1}{\epsilon}G^T g(q)$. The complete stepping equations for the implicit Euler method are then

$$\begin{aligned} Mv_{k+1} &= Mv_k + hf_e + \frac{h}{\epsilon}[G^T g(q)]_{k+1} \\ q_{k+1} &= q_k + hv_{k+1}. \end{aligned} \tag{52}$$

The constraint force in this case should be evaluated at $q_{k+1} = q_k + hv_{k+1}$. However, since we don't know v_{k+1} or q_{k+1} yet, we make an approximation. As explained before (though regrettably, without proof here for lack of space) $\|g(q)\| = O(\sqrt{\epsilon})$ for a conservative system, if ϵ is small enough. This means that the changes in the Jacobian cannot be so large since we never go far from the constraint surface. Therefore, a reasonable and simple approximation to the implicit force term reads

$$\begin{aligned} [G^T g(q)]_{k+1} &\approx G_k^T g_{k+1} = G_k^T(g_k + hG_k v_{k+1}) \\ &= G_k^T g_k + hG_k^T G_k v_{k+1}. \end{aligned} \tag{53}$$

Inserting that in the stepping equation produces the result

$$\begin{aligned} \left[M + \frac{h^2}{\epsilon}G_k^T G_k\right] v_{k+1} &= Mv_k + hf_e + \frac{h}{\epsilon}G_k^T g_k \\ x_{k+1} &= x_k + hv_{k+1}. \end{aligned} \tag{54}$$

This now looks like treating the strong potentials implicitly is like adding inertia to the system. However the pseudo mass matrix–the factor $M+(h^2/\epsilon)G^T G$ that multiplies v_{k+1} in (54)–is ill conditioned, even though in the present form, it is symmetric and positive definite. Incidentally, that is not the case if Jacobian derivative terms are included in the approximation (53).

The problem here is that we want to make $\epsilon \to 0$ which means that the terms added to the mass matrix can be enormous in comparison to M. When this is done, as far as the numerical computation goes, the equations look like

$$G_k^T G_k v_{k+1} = -\frac{1}{h}G^T g_k, \tag{55}$$

and, basically, that's just the steepest descent [3] method to find a zero of $g(q)$. Implicit Euler kills the rest of the dynamics accidentally, but it does damp everything good. Next time you look at a cloth animation, notice how everything looks like rubber: this is why. And by the way, if using implicit Euler, don't bother adding damping: it is already included in the stepping equations for free, and in copious amounts. Save the headache!

Had we used the midpoint rule, the stepping equations would be slightly different but not so much. That will not be investigated further here but it has appeared in the literature [4]. When using the implicit midpoint rule though, it is necessary to introduce some damping since otherwise, you will have a lot of white noise. The problem here is that the limit as $\epsilon \to 0$ is still not well behaved and for that reason, the implicit midpoint rule is not investigated further.

11 Linearly Implicit Verlet for strong forces

The question now is how to combine the good properties of the simple Verlet stepper with those of an implicit method for the strong potentials. Let's start by writing down what we know.

The strong forces should be written as $G_k^T \lambda$ so that we don't have to worry with the derivatives of the Jacobian.

The equation for λ should be an implicit relationship so it does not generate explicit terms of the form $-(1/\epsilon)g_k$ in the equation for v_{k+1}.

Whenever $g(q)$ appears in the equation, it should be replaced by some sort of local average, including values at the next time step to be implicit. That would hopefully recover the good properties of the implicit midpoint rule where they are needed. That means that

$$g(q) \approx \frac{1}{4}(g_{k+1} + 2g_k + g_{k-1}), \tag{56}$$

for instance. In addition, there should be some damping on $\dot{g}$ so we can write

$$\begin{aligned} Mv_{k+1} &= Mv_k + hf_e + hG^T\lambda \\ \epsilon\lambda + \frac{1}{4}(g_{k+1} + 2g_k + g_{k-1}) + \gamma G_k v_{k+1} &= 0. \end{aligned} \tag{57}$$

Now, we can expand the average term

$$\begin{aligned} g_{k+1} &= g(q_k + hv_{k+1}) = g_k + hG_k v_{k+1}, \\ g_{k-1} &= g(q_k - hv_k) = g_k - hG_k v_k, \\ \frac{1}{4}(g_{k+1} + 2g_k + g_{k-1}) &= g_k + \frac{h}{4}G_k v_{k+1} - \frac{h}{4}G_k v_k. \end{aligned} \tag{58}$$

After writing $h\lambda = \tilde{\lambda}$ and collecting all similar terms, we get

$$\begin{aligned} Mv_{k+1} - G_k^T\tilde{\lambda} &= Mv_k + hf_e \\ G_k v_{k+1} + \frac{4\epsilon}{h^2(1+4\gamma/h)}\tilde{\lambda} &= -\frac{4}{h(1+4\gamma/h)}g_k + \frac{1}{(1+4\gamma/h)}G_k v_k. \end{aligned} \tag{59}$$

And this is now the spook stepper that has been presented earlier in the course.

In matrix formulation, the system (59) reads

$$\begin{bmatrix} M & -G_k^T \\ G_k & \frac{4\epsilon}{h^2(1+4\frac{\gamma}{h})} \end{bmatrix} \begin{bmatrix} v_{k+1} \\ \tilde{\lambda} \end{bmatrix} = \begin{bmatrix} Mv_k + hf_e \\ -\frac{4}{h(1+4\frac{\gamma}{h})} g_k + \frac{1}{h(1+4\frac{\gamma}{h})} G_k v_k \end{bmatrix}, \tag{60}$$

where it is understood that the block $4\epsilon/(h^2(1+4\gamma/h))$ is a diagonal matrix of appropriate dimensions.

The fundamental difference with the linearly implicit Euler is that the stepping matrix

$$H_\epsilon = \begin{bmatrix} M & -G_k^T \\ G_k & \frac{4\epsilon}{h^2(1+4\frac{\gamma}{h})} \end{bmatrix} \tag{61}$$

has a very nice limit as $\epsilon \to 0$, provided the Jacobian G_k has full row rank. Matrix H_ϵ is positive definite but not symmetric for $\epsilon > 0$ and can be easily factored using a good sparse matrix code such as SuperLU [2], or UMFPACK [1].

If you eliminate v_{k+1} from the system (60), you get

$$\begin{aligned} \left[G_k M^{-1} G_k^T + \frac{4\epsilon}{h^2(1+4\frac{\gamma}{h})} \right] \tilde{\lambda} &= -\frac{4}{h(1+4\frac{\gamma}{h})} g_k \\ &\quad + \left(\frac{1}{(1+4\frac{\gamma}{h})} - 1 \right) G_k v_k + h G_k M^{-1} f_e, \end{aligned} \tag{62}$$

and now, the matrix that needs to be factored is the Schur complement

$$S_\epsilon = \left[G_k M^{-1} G_k^T + \frac{4\epsilon}{h^2(1+4\frac{\gamma}{h})} \right], \tag{63}$$

which is symmetric, positive definite, and well conditioned if $\epsilon > 0$ is large enough. On that matrix, you can use a sparse Cholesky method as found in the SparseSuite collection for instance [1].

If you want to use Gauss-Seidel iterations [3], you will have to solve the linear system (63) since H_ϵ is not symmetric.

References

[1] T. A. Davis, *Algorithm 832: UMFPACK — an unsymmetric-pattern multifrontal method*, ACM Transactions on Mathematical Software, 30 (2004), pp. 196–199.

[2] J. W. Demmel, S. C. Eisenstat, J. R. Gilbert, X. S. Li, and J. W. H. Liu, *A supernodal approach to sparse partial pivoting*, SIAM J. on Mat. Anal.and Appl., 20 (1999), pp. 720–755. superlu.

[3] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, vol. 16 of SIAM Frontiers, SIAM Publ., Philadelphia, 1995.

[4] S. Oh, J. Ahn, and K. Wo, *Low damped cloth simulation*, Visual Comput., 22 (2006), pp. 70–79.

[5] H. Rubin and P. Ungar, *Motion under a strong constraining force*, Communications on Pure and Applied Mathematics, X (1957), pp. 65–87.