On Solving Differential Kinematic Equations for Constrained Mechanical Systems

A. W. Beshaw

Department of Mathematics
Bahr Dar University
Ethiopia, Bahr Dar

This paper proposes a method for constructing the kinematic equations of the mechanical system, which imposed geometric constraints. The method is based on the consideration of kinematic constraints as particular integrals of the required system of differential equations. Runge-Kutta method is used for the numerical solution of nonlinear differential equations. The developed methods allow us to estimate the range of variation of the parameters during the numerical solution which determine conditions for stabilization with respect to constraint equations. The numerical results illustrate the dependence on the stabilization of the numerical solution is not only due to the asymptotic stability with respect to the constraint equations, but also through the use of difference schemes of higher order accuracy. To estimate the accuracy of performance of the constraint equations additional parameters are introduced that describe the change in purpose-built perturbation equations. It is shown that unstable solution, with respect to constraint equations, obtained by the Euler method can be stable by using Runge-Kutta method.

Key words and phrases: kinematic constraints, pseudo-inverse, approximate solution, stability, Euler's method, Runge-Kutta methods.

1. Introduction

Let the kinematic state of a mechanical system be determined by the position coordinates $q_1, q_2, \ldots, q_n$ and dependent velocities $\dot{q}_1, \dot{q}_2, \cdots, \dot{q}_n$ described by the system of ordinary differential equations [1]:

$$\dot{q} = v(q, t), \ \ \dot{q} = \frac{dq}{dt}, \ q \in R^n, \tag{1}$$

where v is a vector valued function and we shall assume that the function v together with its first partial derivatives are continuous in some domain $D \subset R^n \times R$.

The functions $q(t)$ satisfy the equations (1) and take prescribed values q^0 at the initial time t_0:

$$q(t_0) = q^0. \tag{2}$$

And assume that the system is subjected to m holonomic constraints:

$$f(q, t) = 0, \quad \text{where,} \quad f \in R^m, \quad m \leqslant n \tag{3}$$

then it is usually assumed that the initial values of the coordinates also exactly satisfy the constraint equations

$$f(q^0, t_0) = 0. \tag{4}$$

As stated in [2] relations (1) should be constructed according to the constraint equations (3), which under relations (4), form a set of particular integrals of the system of differential equations (1).

Moreover, on all solutions $q = q(t), t \geqslant t_0$, of system (1) satisfying condition (3), the vector $v(q, t)$ of the right-hand side should be chosen so as to satisfy the conditions

$$f_q v + f_t = 0, \tag{5}$$

where $f_q = (f_{ij})$, $f_t = (f_{it})$, $f_{ij} = \dfrac{\partial f_i}{\partial q_j}$, $f_{it} = \dfrac{\partial f_i}{\partial t}$, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$

which form a system of m linear equations with n unknowns $v_1, v_2, \ldots, v_n$ forming the vector v.

2.   Solution of systems of differential equations

The equation (5); the system of m ordinary differential equations with n unknowns, can be rewritten as

$$Mv = b, \tag{6}$$

where $M = f_q$ is an $m \times n$ matrix and $b = -f_t$ is a column matrix. As we can see from the dimension of matrix M, we cannot apply inverse (the usual method) to solve the system in (6).

The structure of the general solution of equation (6) is described by the following theorem [3]:

Theorem 1. *The set of all solutions of the linear systems* (6), *in which the matrix M has rank m is determined by the relation*

$$v = c[MC] + M^+ b, \tag{7}$$

where c is an arbitrary scalar quantity, $[MC] = [M_1 \ldots M_m C_{m+1} \ldots C_{n-1}]$ is vector product, $M_i = (M_{ij})$, and arbitrary $C_\tau = (C_{\tau j})$, $\tau = m + 1, \ldots, n - 1$, $j = 1, \ldots, n$, $M^+ = M^T (MM^T)^{-1}$.

Note that $M^+ = M^T (MM^T)^{-1}$ is pseudo-inverse (Moore–Penrose inverse) of matrix M. Given an m by n matrix M, the n by m matrix M^+ is a unique matrix that satisfies the following four relations [4]:

$$
\begin{aligned}
&\text{1. } MM^+ M = M; \\
&\text{2. } M^+ MM^+ = M^+; \\
&\text{3. } (MM^+)^T = MM^+; \text{ and,} \\
&\text{4. } (M^+ M)^T = M^+ M.
\end{aligned}
\tag{8}
$$

Since $M^+ M$ and MM^+ are both symmetric matrices,

$$M^T (MM^T)^{-1} = M^T (M^+)^T M^+ = (M^+ M)^T M^+ = M^+ MM^+ = M^+.$$

In our case replace f_q instead of M in (7), accordingly $f_q = M$, $M^+ = (f_q)^+$ and

$$v = c[f_q C] + (f_q)^+ (-f_t). \tag{9}$$

In order to compensate the deviation of (5) during numerical integration it is necessary to use a correcting term and restate equation (5) as [3]:

$$f_q v + f_t = F, \tag{10}$$

where $F = F(f, q, t)$, $F(0, q, t) = 0$.

Definition 1. The function $f(q, t)$ is a particular integral of (1) if the equality (10) and $F(f, q, t)$ vanishes when the condition $f(q, t) = 0$.

It is usually assumed that the equality (4) is satisfied exactly. However, if the initial conditions $q(t_0) = q^0$ do not satisfy the constraints (3), depending on how you

will be determined by the function $F(f; q, t)$ in equation (10), the solution will either approach the manifold (3), or away from it.

Having a general solution of equation (10), we can obtain the desired set of systems of differential equations that determine the kinematic relations between the coordinates of constrained mechanical systems and their derivatives [3]

$$\dot{q} = c[f_q C] + (f_q)^+ (F - f_t). \tag{11}$$

In the expression (11), $[f_q C] = [f_{1q} f_{2q} \ldots f_{mq} C_{m+1} \ldots C_{n-1}]$ is vector product $f_q = (f_{ij})$, $f_{ij} = \dfrac{\partial f_i}{\partial q_j}$ $(i = 1, \ldots, m, \ j = 1, \ldots, n)$ is row vector matrix and $C_{m+1}(q)$, $C_{m+2}(q), \ldots, C_{n-1}(q)$ are arbitrary vectors, $(f_q)^+ = (f_q)^T (f_q f_q)^T)^{-1}$. Equation (3) represents a collection of particular integrals of the system (11).

3.   Numerical Solutions

The physical phenomena of the differential equations (11) are highly nonlinear and usually solved by numerical methods (numerical approximations). Let us assume that the initial value $q(t_0) = q^0$ be given and suppose F can be expressed as a linear product of f; $F = Kf$, where K is $m \times m$ matrix. Then we may use one of different numerical methods to solve (11). Among these methods we use the Euler's method, second order, third order and fourth order Runge-Kutta methods.

3.1.   Euler Method

For the system of differential equation in (11) we construct the equation:

$$q^{i+1} = q^i + \tau \dot{q}^i, \tag{12}$$

where $q^i = q(t_i)$, $t_{i+1} = t_i + \tau$ and $\dot{q}^i = c[f_q C]_{(q^i, t_i)} + (f_q)^+ (Kf - f_t)_{(q^i, t_i)}$, $i = 0, 1, 2, \ldots$.

In relation to (12) we state the following statement [5]:

(i) $\left\| f^0 \right\| \leqslant \epsilon,$

(ii) $\tau \leqslant \tau_1$, and for all $q = q^i$, $t = t_i$, $i = 0, 1, 2, \ldots$ fulfilled the inequality,

(iii) $\|I + \tau K\| \leqslant a \leqslant 1,$ \hfill (13)

(iv) $\left\| f^{(2)} \right\| \leqslant \dfrac{2(1-a)\epsilon}{\tau_1^2},$

where $f^{(2)} = v^T f_{q^T q} v + 2 f_{qt} v + f_{tt}$, then the solution of the difference equation (12) will satisfy the condition $\left\| f(q^i, t_i) \right\| \leqslant \epsilon$, for any $i = 1, 2, \cdots$, where a, ϵ, τ_1 are constants.

3.2.   Second Order Runge-Kutta Method

In Euler method there was no attempt made to include information from the point t_{i+1} at which the solution is being obtained. Heun (Second order Runge-Kutta) method uses Euler method but takes the result $q^i + \tau \dot{q}^i$ to calculate the gradient there and then, whereas Euler method just uses $\dot{q}^i$ for the gradient over the interval, Heun method uses the mean of that value and the approximately calculated value at t_{i+1}.

$$q^{i+1} = q^i + \frac{\tau}{2} [v(q^i, t_i) + v(q^i + \tau, q^i + \tau v(q^i, t_i))], \tag{14}$$

where $\dot{q}^i = v(q^i, t_i)$.

For our system (11):

$$v(q^i, t_i) = c[f_q C]_{(q^i, t_i)} + (f_q)^+ (Kf - f_t)_{(q^i, t_i)}. \tag{15}$$

3.3.   Third Order Runge-Kutta Method

The solution of (11) can also be solved using Runge-Kutta method in more efficient order:

$$q^{i+1} = q^i + \tau(c_1 k_1 + c_2 k_2 + c_3 k_3), \tag{16}$$

where the multipliers $k_j (j = 1, 2, 3)$ and $c_j (j = 1, 2, 3)$ are obtained as in [6]:

$$q^{i+1} = q^i + \frac{\tau}{6}(k_1 + 4k_2 + k_3), \tag{17}$$

and

$$k_1 = v(q^i, t_i); \quad k_2 = v\left(q^i + \frac{\tau}{2}k_1, t_i + \frac{\tau}{2}\right); \quad k_3 = v(q^i - \tau k_1 + 2\tau k_2, t_i + \tau).$$

Now to solve (11) in this method, expand each expressions of k_1, k_2, k_3 up to order three:

$$v(q^i, t_i) = c[f_q C]_{(q^i, t_i)} + (f_q)^+ (Kf - f_t)_{(q^i, t_i)};$$

$$k_2 = v\left(q^i + \frac{\tau}{2}v, t_i + \frac{\tau}{2}\right);$$

$$k_3 = v\left(q^i - 2\tau v(q^i, t_i) + 2\tau v\left(q^i + \frac{\tau}{2}v(q^i, t_i), t_i + \frac{\tau}{2}\right), t_i + \tau\right).$$

The second and third order Runge-Kutta methods are available in MATLAB as built in functions called ODE solvers *ode*23. It is also possible to construct user defined functions and solve ODE problems with efficiency as desired. The approximate solution will be close to the actual solution if n is taken sufficiently large enough.

3.4.   Fourth Order Runge-Kutta Method

The fourth order Runge-Kutta method is the popular version of the members, for which reference can be made to any book on numerical methods. They require the ability to solve the initial value problem for a system of ordinary differential equations at a number of points intermediate between t_i and t_{i+1} [7].

If the function v and the data q^0, t_0 are given, then

$$q^{i+1} = q^i + \frac{\tau}{6}(k_1 + 2k_2 + 2k_3 + k_4), \tag{18}$$

where

$$k_1 = v(q^i, t_i) = c[f_q C]_{(q^i, t_i)} + (f_q)^+ (Kf - f_t)_{(q^i, t_i)};$$

$$k_2 = v\left(t_i + \frac{\tau}{2}, q^i + \frac{\tau}{2}v(q^i, t_i)\right);$$

$$k_3 = v\left(t_i + \frac{\tau}{2}, q^i + \frac{\tau}{2}v\left(t_i + \frac{\tau}{2}, q^i + \frac{\tau}{2}v(t_i, q^i)\right)\right);$$

$$k_4 = v\left(t_i + \tau, q^i + \tau v\left(t_i + \frac{\tau}{2}, q^i + \frac{\tau}{2}v\left(t_i + \frac{\tau}{2}, q^i + \frac{\tau}{2}v(q^i, t_i)\right)\right)\right).$$

Here q^{i+1} is the Runge-Kutta approximation of $q(t_{i+1})$, and the next value (q^{i+1}) is determined by the present value (q^i) plus the weighted average of corresponding (2,3,4) increments, where each increment is the product of the size of the interval,

τ, and an estimated slope specified by the function G on the right-hand side of the differential equation [8]:

- k_1 is the increment based on the slope at the beginning of the interval, using $\dot{q}$;
- k_2 is the increment based on the slope at the mid-point of the interval, using $\dot{q} + \frac{\tau}{2}k_1$;
- k_3 is the increment based on the slope at the mid-point of the interval, but now using $\dot{q} + \frac{\tau}{2}k_2$;
- k_4 is the increment based on the slope at the end of the interval, using $\dot{q} + \tau k_3$.

Note that the $4^{th}, 5^{th}$ order Runge-Kutta methods are combined together and termed as $'ode45'$, which is available in MATLAB. It is also possible to build user defined functions.

Example 1. Let us consider systems of ordinary differential equations of two variables:

$$\begin{cases} \dot{x} = v_1(x,y), \\ \dot{y} = v_2(x,y), \end{cases} \tag{19}$$

and satisfy a constraint equation:

$$f(x,y) \equiv 1/2(9x^2 + y^2 - 9) = 0. \tag{20}$$

Differentiating (20) with respect to time and assigning the result to a function F we get

$$9x\dot{x} + y\dot{y} = F, \quad F = F(f,x,y). \tag{21}$$

We assume this function F as a linear combination of the given constraint function f; $F = kf$.

The solution for this equation (21) can be found using Theorem 1, it gives

$$\begin{cases} \dot{x} = cy + \dfrac{9x}{81x^2 + y^2}(kf), \\ \dot{y} = -9cx + \dfrac{y}{81x^2 + y^2}(kf). \end{cases} \tag{22}$$

Substitute f in relation (22) we obtain

$$\begin{cases} \dot{x} = cy + \dfrac{9kx}{2(81x^2 + y^2)}(9x^2 + y^2 - 9), \\ \dot{y} = -9cx + \dfrac{ky}{2(81x^2 + y^2)}(9x^2 + y^2 - 9). \end{cases}$$

Next we solve (22) numerically and investigate the stability of the solution as k varies. For simplicity we consider k as constant and $c = 1$.

case I: Let $k > 0$, say $k = 2$, and initial values $(x^0, y^0) = (1,0)$. Therefore, relation (22) becomes

$$\begin{cases} \dot{x} = y + \dfrac{9x}{81x^2 + y^2}(9x^2 + y^2 - 9), \\ \dot{y} = -9x + \dfrac{y}{81x^2 + y^2}(9x^2 + y^2 - 9). \end{cases} \tag{23}$$

As we can see from Fig. 1, the numerical solution of (23) is unstable, neither it is solved in Euler method nor in Runge-Kutta methods.

case II: let $k = 0$, the system reduces to

$$\begin{cases} \dot{x} = y, \\ \dot{y} = -9x. \end{cases} \tag{24}$$
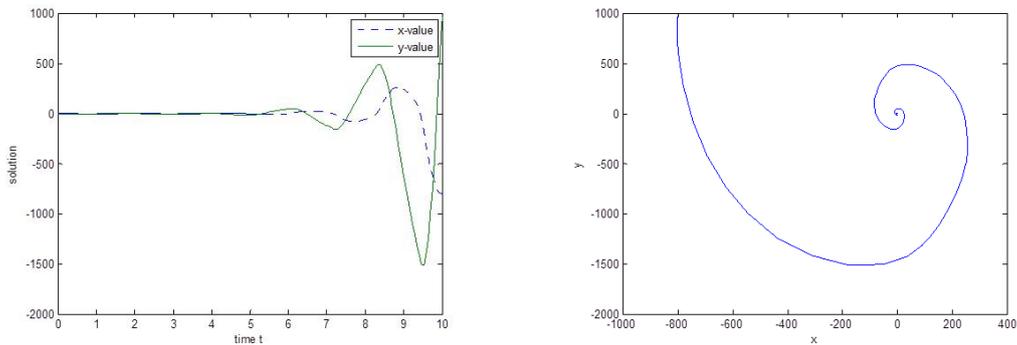
a Position versus time $k = 2$; unstable solution

b Phase portrait graph for $k = 2$; unstable solution

Figure 1. The numerical solution of (23)

This system has exact solution: $u(t) = (c_1 \cos 3t, c_2 \sin 3t)$ and take $x^0 = 1, y^0 = 0$ we get $u(t) = (\cos 3t, -3 \sin 3t)$.

case III: Let $k < 0$, say $k = -2$, the system (22) becomes

$$\begin{cases} \dot{x} = y - \dfrac{9x}{81x^2 + y^2}(9x^2 + y^2 - 9), \\ \dot{y} = -9x - \dfrac{y}{81x^2 + y^2}(9x^2 + y^2 - 9). \end{cases} \tag{25}$$

The numerical solution of (25) is stable in any of the methods discussed above, see Fig. 2. But the stability of the solution depends on not only the value of the parameter k but also the method we use to solve it. For instance, taking $k = -400$ the solution is unstable when we use Euler method but it is stable in Runge-Kutta methods as shown in Fig. 3.
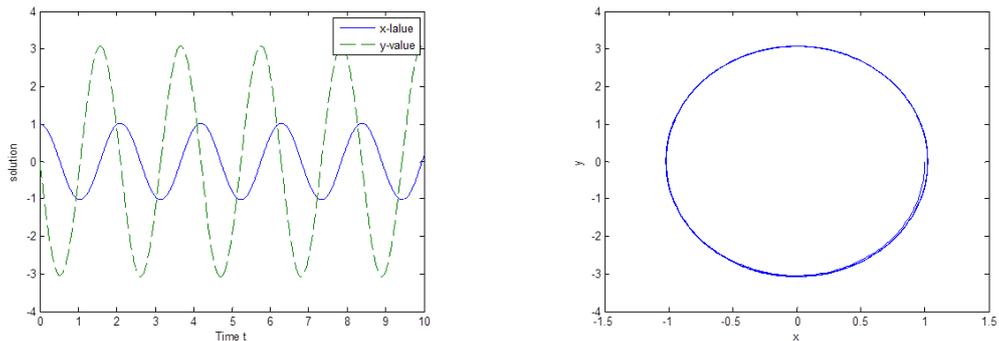
a Position-time graph for $k = -2$; stable solution

b Phase portrait graph for $k = -2$; stable solution

Figure 2. The numerical solution of (25)

From the above cases the solutions are
− Unstable for all values of k in case 1 ($k > 0$);
− Stable for case 2 ($k = 0$) or it gives more sense if we say $F = 0$;
− Stable for case 3 ($k < 0$) for certain domain of k values.
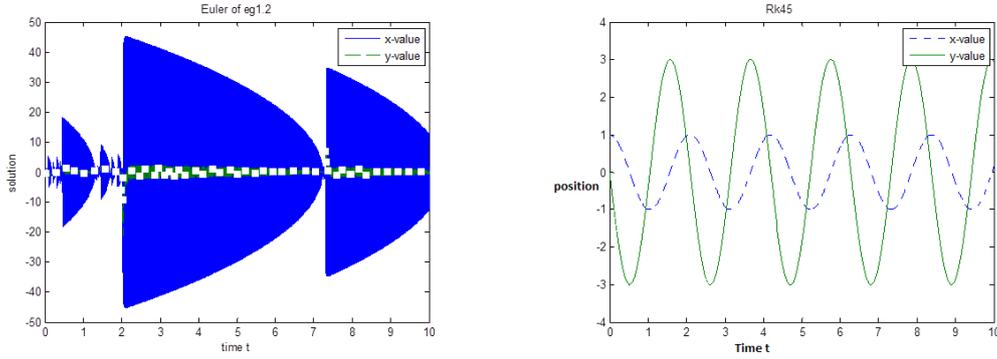
a Position versus time graph using Euler method for $k = -400$; unstable solution

b Position versus time graph using Runge-Kutta methods for $k = -400$; stable solution

Figure 3. The numerical solution of (25)

To determine the possible values of k so that the solutions obtained are stable we proceed as follows. To approximate equation (22) numerically we guess the possible values of the parameter k

$$z^{i+1} = z^i + \tau \dot{z}^i, \quad z^i = z(t_i), \quad t_{i+1} = t_i + \tau, \quad z = (x, y),$$

$$\dot{x} = cy + \frac{9kx}{2(81x^2 + y^2)}(9x^2 + y^2 - 9), \quad \dot{y} = -9cx + \frac{ky}{2(81x^2 + y^2)}(9x^2 + y^2 - 9).$$

Referring these with the statements of (22) [3]:

$$f^{(2)} = \frac{\partial^2 f}{\partial x^2}\dot{x}^2 + \frac{\partial^2 f}{\partial y^2}\dot{y}^2 = 9\dot{x}^2 + \dot{y}^2.$$

From the last condition of (13), we have $\left|9\dot{x}^2 + \dot{y}^2\right| \leqslant \dfrac{2(1-a)\epsilon}{\tau_1^2}$.

Letting $k = c$ and using (22), condition 3 can be rewritten as

$$9(9x^2 + y^2) + \frac{90\lambda |x| |y| \left|9x^2 + y^2 - 9\right|}{81x^2 + y^2} + \frac{\lambda^2(729x^2 + y^2)\left|9x^2 + y^2 - 9\right|^2}{4(81x^2 + y^2)^2} \leqslant \frac{2(1-a)\epsilon}{c^2\tau_1^2}.$$

Now we take $|x| \leqslant 1.1$, $|y| \leqslant 3.1$, $81x^2 + y^2 \geqslant x^2 + y^2 \geqslant 0.81$, and substitute these to variables in the above inequality results

$$81.9 + 37.9\lambda + 3.4\lambda^2 \leqslant \frac{2(1-a)\epsilon}{c^2\tau_1^2}.$$

To have possible range of the discriminates of the inequality must be nonnegative:

$$(37.9)^2 - 4(3.4)\left(81.9 - \frac{2(1-a)\epsilon}{c^2\tau_1^2}\right) \geqslant 0, \quad \text{this leads,} \quad \tau \leqslant \sqrt{\frac{(1-a)\epsilon}{48.3}}.$$

Now choose $c = 1$, $a = 0.99$, $\epsilon = 0.1$, $\tau \leqslant 0.0045$, say $\tau = 0.002$.

Then from the quadratic inequality we get $-359.6 \leqslant \lambda \leqslant 6$.

On the other side from condition 2 above; $|1 + \tau k| \leqslant a < 1$, we get

$$-\frac{(a+1)}{\tau} \leqslant k \leqslant \frac{a-1}{\tau}, \quad -995 \leqslant k \leqslant -0.5.$$

Hence $-359.6 \leqslant k = \lambda \leqslant -0.5$.

4. Conclusion

The paper considers a method of constructing kinematic equation of mechanical system that contains holonomic constraints. It is shown that the sign and size of the factor (parameter) in the general equation affects the stability of its solution. Experimental investigations using MATLAB are made to find the range and sign of the factor especially for Euler numerical method.

A remarkable result is obtained that stability depends not only on the size and sign of factor but also on the numerical method. In the demonstrative example, the Euler method results in unstable solution for $k = -400$ whereas Runge-Kutta method produces stable solution, see Fig. 3. This paper can be used as an initial for further research on the factor (indicated in the paper) in other numerical methods.

References

1. R. G. Muharlyamova, N. V. Abramov, J. K. Kirgiyaev, Equations of Dynamical Systems with Program Constraints, URGU, 2013, in Russian.
2. R. G. Mukharlyamov, On the Construction of Differential Equations of Motion of Constrained Mechanical Systems, Differential Equations, Differential Equations 39 (3) (2003) 343–353.
3. R. G. Mukharlyamov, Equations of Motion of Mechanical Systems, PFUR, 2001, in Russian.
4. F. E. Edwadia, Recent Advances in Multi-body Dynamics and Nonlinear Control, ABCM 28 (3) (2006) 311–315.
5. R. G. Mukharlyamov, On the Equations of Kinematics and Dynamics of Constrained Mechanical Systems, Multibody System Dynamics (6) (2001) 17–28.
6. R. G. Mukharlyamov, A. W. Beshaw, Solving Differential Equations of Motion for Constrained Mechanical Systems, Bulletin of PFUR. Series "Mathematics. Information Sciences. Physics" (3) (2013) 81–91.
7. J. D. Fenton, Numerical Methods, Karlsplatz (3) (2010) 19–22.
8. L. Xie, A Criterion for Hurwitz Polynomials and its Applications, Modern Education and Computer Science (1) (2011) 38–44.

УДК 531.3
Численное решение уравнений кинематики механических систем

А. В. Бешау

Кафедра математики
Бахрдарский университет
Эфиопия, Бахрдар

Работа посвящена решению задачи стабилизации связей при численном решении дифференциальных уравнений, описывающих кинематические соотношения в механической системе. В статье предлагается метод построения системы дифференциальных уравнений, соответствующих кинематическим соотношениям в механической системе, на которую наложены геометрические связи. Предлагаемый метод основан на преставлении

кинематических связей в качестве частных интегралов соответствующей системы дифференциальных уравнений. Для определения численного решения нелинейных дифференциальных уравнений используется метод Рунге-Кутта. Разработанный метод позволяет в процессе численного решения дифференциальных уравнений оценить границы изменения параметров управляющих воздействий, которые соответствуют условиям стабилизации решения по отношению к заданным уравнениям связей. Результаты вычислений показывают, что стабилизация численного решения зависит не только от асимптотической устойчивости по отношению к уравнениям связей, но также от точности используемой той или иной разностной схемы. Для оценки точности выполнения уравнений связей вследствие стабилизации связей вводятся дополнительные параметры, изменение которых определяется специально построенными дифференциальными уравнениями возмущений связей. Показано, что численное решение, полученное методом Эйлера, которое оказывается неустойчивым, может оказаться устойчивым при использовании метода Рунге-Кутта.

Ключевые слова: кинематические ограничения, приближенное решение, псевдообратный, устойчивость, метод Эйлера, методы Рунге–Кутты.

Литература

1. *Мухарлямов Р. Г., Абрамов Н. В., Киргираев Ж. К.* Уравнение днамики системы с программными связями. — Юргу, 2013.
2. *Mukharlyamov R. G.* On the Construction of Differential Equations of Motion of Constrained Mechanical Systems, Differential Equations // Differential Equations. — 2003. — Vol. 39, No 3. — Pp. 343–353.
3. *Мухарлямов Р. Г.* Уравнение движения механических систем. — РУДН, 2001.
4. *Edwadia F. E.* Recent Advances in Multi-body Dynamics and Nonlinear Control // ABCM. — 2006. — Vol. 28, No 3. — Pp. 311–315.
5. *Mukharlyamov R. G.* On the Equations of Kinematics and Dynamics of Constrained Mechanical Systems // Multibody System Dynamics. — 2001. — No 6. — Pp. 17–28.
6. *Мухарлямов Р. Г., Бешау А. В.* Решение дифференциальных уравнений движения для механических систем со связями // Вестник рУДН. Серия «Математика. Информатика. Физика». — 2013. — № 3. — С. 81–91.
7. *Fenton J. D.* Numerical Methods // Karlsplatz. — 2010. — No 3. — Pp. 19–22.
8. *Xie L.* A Criterion for Hurwitz Polynomials and its Applications // Modern Education and Computer Science. — 2011. — No 1. — Pp. 38–44.