Dynamic simulation of a parallel robot: Coulomb friction and stick–slip in robot joints

Nidal Farhat, Vicente Mata, Álvaro Page and Miguel Díaz-Rodríguez

Robotica / Volume 28 / Issue 01 / January 2010, pp 35 - 45
DOI: 10.1017/S0263574709005530, Published online: 07 April 2009

Robotica (2010) volume 28, pp. 35–45. © Cambridge University Press 2009
doi:10.1017/S0263574709005530

Dynamic simulation of a parallel robot: Coulomb friction and stick–slip in robot joints

Nidal Farhat†*, Vicente Mata†, Álvaro Page‡ and Miguel Díaz-Rodríguez§

†*Departamento de Ingeniería Mecánica y de Materiales, Universidad Politécnica de Valencia, Spain*
‡*Departamento de Física Aplicada, Universidad Politécnica de Valencia, Spain*
§*Departamento de Tecnología y Diseño, Facultad de Ingeniería, Universidad de Los Andes, Venezuela*

(Received in Final Form: March 06, 2009. First published online: April 7, 2009)

SUMMARY
Dynamic simulation in robotic systems can be considered as a useful tool not only for the design of both mechanical and control systems, but also for planning the tasks of robotic systems. Usually, the dynamic model suffers from discontinuities in some parts of it, such as the use of Coulomb friction model and the contact problem. These discontinuities could lead to stiff differential equations in the simulation process. In this paper, we present an algorithm that solves the discontinuity problem of the Coulomb friction model without applying any normalization. It consists of the application of an external switch that divides the integration interval into subintervals, the calculation of the friction force in the stick phase, and further improvements that enhance its stability. This algorithm can be implemented directly in the available commercial integration routines with event-detecting capability. Results are shown by a simulation process of a simple 1-DoF oscillator and a 3-DoF parallel robot prototype considering Coulomb friction in its joints. Both simulations show that the stiffness problem has been solved. This algorithm is presented in the form of a flowchart that can be extended to solve other types of discontinuity.

KEYWORDS: Dynamic simulation; Friction; Stiff ODE's; Parallel robots; Parameter identification.

1. Introduction

Friction is present in the joints of most of the robotic systems. In some cases, friction forces can reach until 20% of the maximum force.[1] Hence, any realistic simulation should include the friction phenomena correctly modeled. Despite the fact that friction phenomena is nonlinear and several models have been proposed for its modeling, such as the Dahl model,[2] the bristle model,[3] and the LuGre model,[4] the most widely implemented models in the direct dynamic parameter identification processes are the linear ones that consider Coulomb and viscous frictions (see refs. [5–8] among others). For a survey on the different friction models and their application see, for example, refs. [9, 10]. It will be important to mention that the identification of dynamic parameters is nowadays a widely accepted procedure in order to build the equations of motion for realistic simulations of complex multibody mechanical systems. In this paper, the dynamic simulation of the parallel robot is based on inertial and friction parameters experimentally identified.

However, Coulomb friction can be considered as one of those typical elements that introduce discontinuities into the equation of motion of multibody systems. Inefficient behavior or even a failure of the integration process might be produced when trying to solve the discontinuous systems without localizing the points of discontinuity explicitly.[11]

In order to solve this type of discontinuity problem several approaches had been proposed. First contributions proposed the application of a smoothing function, also known as normalization method that tries to connect the positive and negative limits of the Coulomb friction by smooth curves for a small velocity region (see, for example, refs. [12, 13]). Sextro[12] used such normalization where in the integration process switching between the different states of the friction model was made internally depending on the value of the velocity, i.e. at each integration step. As stated by Sextro, high values of the slope of this curve lead to a stiff set of differential equations. The proposed solution was the use of special integration routines, or stiff integrators, that cope with these kinds of problems. However, these integrators need additional calculations to enhance the instability of the integration process.[11] In order to make use of the conventional integration routines, Leine *et al.*[14] proposed the Shooting method to switch between the different states of the Karnopp friction model[15] internally without halting the integration process. This approach is well suited for friction with periodic cycles and can be considered as an improved version of the Karnopp model where the numerical instability has been solved. Hensen *et al.*[16] made a comparison between this approach and the LuGre dynamic friction model with respect to the shooting method. The shooting method was also used earlier by Van de Vrande *et al.*[17] in combination with the smoothing function.

It is important to mention that in all the previously described approaches—in the simulation process—exact sticking never occurs and the stick–slip is not described exactly.[12,13] Finally, Quinn[13] proposed a new normalization of the Coulomb friction model where a small interval (ε) around zero velocity was assumed in which the friction force

* Corresponding author. E-mail: nifar@doctor.upv.es

was calculated as a function of the net applied forces acting on the joint and the size of (ε). As (ε) tends to zero his model converges pointwise to the classical definition of the Coulomb friction model. As stated by the author, some of the drawbacks of this model are that in the stick phase the velocity does not come to rest in a finite time and it may lead to a stiff system of differential equations near the equilibrium. Moreover, in the normalization methods, the exact friction force is never calculated in the stick phase as exact sticking never occurs and these methods represent an approximation of the friction model in the discontinuity region. Among those that related the friction force to the net applied forces are refs. [14, 18].

From these papers the importance of the problem that accompanies the application of the static friction models in the direct dynamic problem can be revealed. Note that all of them deal with the case in which the adjacent bodies remain in contact. In the other case, where separation can occur, the problem is normally converted to a Linear Complementarity Problem. For more information about this kind of problem see, for example, ref. [19].

In this paper, we introduce a switch algorithm in which the Coulomb friction model is applied directly without any normalization. In the stick phase, the friction force was considered equal to the net forces and acting in opposite direction. The problem of stiff differential equations has been solved, and all the instability sources have been investigated and then eliminated. The proposed algorithm is verified firstly using a common simple example and then applied in the dynamic simulation of a 3-DoF RPS parallel manipulator submitted to actual control actions over a simple sinusoidal trajectory and another Fourier series one. Results show that this algorithm is stable and solves the discontinuity problem of the Coulomb friction without the need for special integrating routines specialized in stiffness problems.

2. Equations of Motion

The starting point for the dynamic simulation consists of obtaining the explicit dynamic model of the manipulator and integrating it with respect to time. The dynamic model of the robot can be obtained basing on many dynamic principles. For example, Mata *et al.*[20] proved that, starting from the Gibbs function, a computationally efficient recursive algorithm can be obtained. For an unconstrained open-chain mechanical system it has the form

$$\mathbf{D}(\vec{q}) \cdot \ddot{\vec{q}} + \vec{C}(\vec{q}, \dot{\vec{q}}) + \vec{F}_{\mathrm{f}}(\dot{\vec{q}}) = \vec{F}(t), \tag{1}$$

where $(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}})$ are the vectors of independent generalized positions, velocities, and accelerations, respectively, $\mathbf{D}$ is the inertia matrix of the mechanical system, $\vec{C}$ is the bias vector that regroups centrifugal and Coriolis accelerations, $\vec{F}_{\mathrm{f}}$ is the vector of friction forces, and finally $\vec{F}$ is the vector of the corresponding generalized forces that are time t dependent.

Starting from the Gauss principle of Least Action, the equations of motion for a closed-chain mechanical system can be found by considering it as a constrained multi-open chain system. As shown by Udwadia and Kalaba,[21] they have the form

$$\begin{bmatrix} 0 & \mathbf{D}_{ii} + \mathbf{X}^T\mathbf{D}_{ee}\mathbf{X} - \mathbf{D}_{ei}\mathbf{X} - \mathbf{X}^T\mathbf{D}_{ie} \\ \mathbf{A}_e & \mathbf{A}_i \end{bmatrix} \begin{bmatrix} \ddot{\vec{q}}_e \\ \ddot{\vec{q}}_i \end{bmatrix} = \begin{bmatrix} \left(\mathbf{X}^T\mathbf{D}_{ee} - \mathbf{D}_{ie}\right)\mathbf{A}_e^{-1}\vec{b} + \vec{F}_i - \vec{C}_i - \vec{F}_{f_i} \\ -\mathbf{X}^T\left(\vec{F}_e - \vec{C}_e - \vec{F}_{\mathrm{f}_e}\right) \\ \vec{b} \end{bmatrix}, \tag{2}$$

where the subscripts i and j stand for the independent/dependent generalized coordinates, respectively, $\mathbf{X} = \mathbf{A}_e^{-1} \cdot \mathbf{A}_i$, and

$$\begin{bmatrix} \mathbf{A}_e & \mathbf{A}_i \end{bmatrix} \begin{bmatrix} \ddot{\vec{q}}_e \\ \ddot{\vec{q}}_i \end{bmatrix} = \vec{b} \tag{3}$$

are the constraint equations at the acceleration level. For more information about how to obtain the equations of motion of closed-chain mechanisms see, for example, refs. [22, 23].

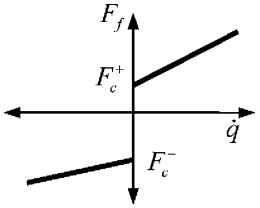

Fig. 1. Coulomb and viscous friction model.

3. Discontinuity Problem of the Friction Model

Discontinuous friction models, such as the Coulomb friction model, are commonly used to model friction in many mechanical systems, including robot joints. Figure 1 shows a friction model where Coulomb and viscous frictions are considered. It can be expressed by the following equation:

$$F_{\mathrm{f}} = \begin{cases} F_{\mathrm{c}}^+ + F_{\mathrm{v}}^+ \cdot \dot{q} & \text{if } \dot{q} > 0 \\ -F_{\mathrm{c}}^- + F_{\mathrm{v}}^- \cdot \dot{q} & \text{if } \dot{q} < 0 \end{cases}, \tag{4}$$

where, F_{c} and F_{v} are the Coulomb and viscous friction coefficients, respectively, and the (+) and (−) superscripts are to indicate the moving sense. As can be observed, this friction model does not determine the friction force at zero velocity. It can take any value between F_{c}^- and F_{c}^+.

In order to be able to solve the problem of applying such a discontinuous friction model it is necessary to investigate its behavior in the integration process. Consider a joint of the robot that is passing zero-velocity state. The dynamic equilibrium for this joint will be one of the two cases represented in Fig. 2: the net force* (F_{net}) acting on the joint of the link—excluding the friction force—lies either between the positive and the negative Coulomb friction coefficients or outside them. For both cases assume that the net force is positive constant.

*Hereafter, the net force refers to all the forces acting on the joint excluding the friction force.

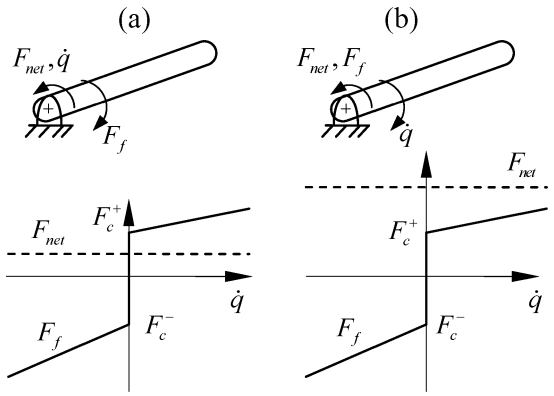

Fig. 2. (a) The net force lies between the Coulomb friction limits and (b) higher than the positive friction limit.

3.1. Net force lies between Coulomb friction coefficients: Fig. 2(a)

Consider the first case where the net force has a value that lies between the Coulomb friction limits while the joint moves at a positive velocity, i.e. $F_c^+ > F_{\text{net}} > F_c^-$ and $\dot{q} > 0$. Then the acceleration is negative $\ddot{q} < 0$. This makes the velocity decrease until it reaches zero. At zero velocity the friction has two possibilities; it may take the value of the positive Coulomb friction limit or the negative one. Consider the positive one (there will be no difference if it takes the negative one), then still $\ddot{q} < 0$ since $F_{\text{net}} - F_{\text{f}} < 0$. The integrator tries to take a further step; the velocity continues decreasing because the acceleration is negative but when the velocity changes its sign to the negative one, the friction switches to the negative velocity side $F_{\text{f}} < 0$, thus $F_{\text{net}} - F_{\text{f}} > 0$ resulting in positive acceleration. This makes the velocity change its sign once again from negative to positive, returning to the previous point with the same previous conditions: $F_{\text{c}}^+ > F_{\text{net}}$, $\dot{q} > 0$, and $\ddot{q} < 0$. The integrator will try to minimize the integration step but without any improvement (see the flowchart presented in Fig. 3). Hence it will be hard for the integrator to integrate this point, leading to a type of the so-called "*stiff differential equations*." Actually—holding the previous conditions—when the velocity of the link reaches zero the friction force exactly equals the net force in a negative sense, leading to zero acceleration, and so the link stops and does not move until the net force increases/decreases outside the Coulomb friction limits.

3.2. Net force lies outside the Coulomb friction coefficients: Fig. 2(b)

The second case can be considered as a special case of the first one (Fig. 2(b)). Let the previous link move in a negative

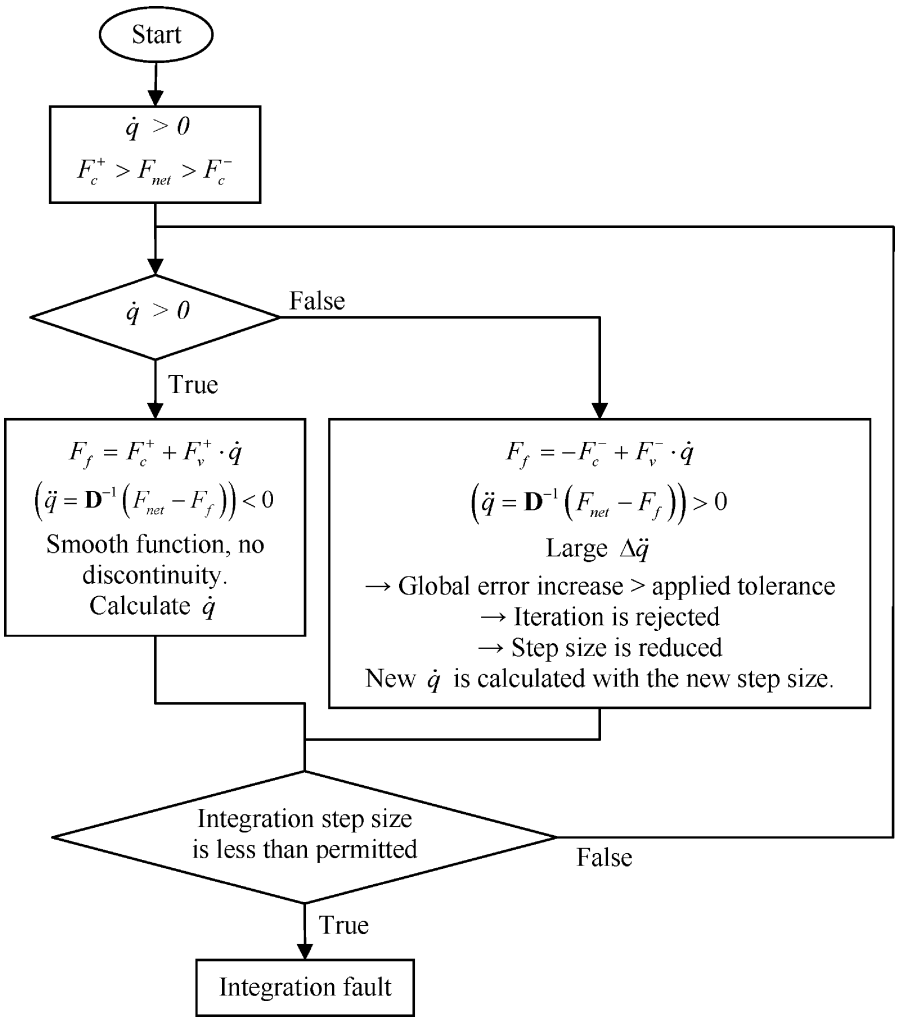

Fig. 3. The behavior of conventional nonstiff integrators in the presence of the discontinuous Coulomb friction model.

direction and let the net force has a positive value that is larger than the positive Coulomb friction coefficient, i.e. $\dot{q} < 0$ and $F_{\text{net}} > F_{\text{c}}^{+}$, then $F_{\text{f}} < 0$, $F_{\text{net}} - F_{\text{f}} > 0$ and $\ddot{q} > 0$. This makes the velocity decrease down to zero value. Then at this point of zero velocity the friction changes its sign from positive to negative, leading to a sudden increase in the $F_{\text{net}} - F_{\text{f}}$ value. As a result, the acceleration will not be a continuous function of time. This also forces the integrator to make many evaluations and to reduce the size of the integration step at this point to conform with the applied integration tolerances.

4. Switch Algorithm

4.1. Switch between different phases

To solve the problem of discontinues differential equations, Eich Soellner and Führer[11] proposed the use of an external switch which can be described as follows: testing the sign of the switch function at each integration step, the discontinuity is checked. If there is a change of sign—a root of the switch function—then integration is stopped and the sign of the switch function is changed, otherwise integration is continued. It will be important to emphasize that the integrator must be admitted to accomplish some integration steps without changing the phase to the other side of the discontinuity even though the discontinuity has been reached, otherwise the same problem will be faced due to discontinuity. Leine *et al.*,[14] among others, objected to use such a switch as halting the integration process is undesirable from the numerical point of view, and special integration routines to detect the switch point "event" are needed. Herein, as the discontinuity of the equation of motion occurs at velocity change of sign points, the switch function is the velocity value.

Recently, some integration routines that are capable of detecting the point/points of discontinuity have become available. This will detract from the second objection. In addition, simulating the actual sticking is more important than the computational cost of halting the integration process. Nevertheless, the results presented in this paper show that the application of the proposed switch algorithm is computationally more efficient than the normalization method, to a great extent. Moreover, there will be no need to apply a force in the stick threshold to eliminate the small velocity as the event of the external switch is zero velocity.

The application of the external switch in parallel with the calculation of the actual friction force when sticking has not been addressed in the past. It avoids the stiffness problem and simulates the stick–slip phenomenon. The external switch solves the problem mentioned in the previous section, second case. On the other hand, when accompanied with a friction force calculation in the stick phase, the problem that appeared in the first case is solved. In this paper, this idea is implemented with some important improvements that will be introduced in a later subsection, resulting in the proposed switch algorithm. It is important to indicate that most of the investigations that were mentioned in Section 1 consider simple simulated examples—in most cases of 2-DoF—in which the net force when sticking could be obtained as direct relations. Here, the case is generalized and systematic relations are extracted for their calculation for any multibody mechanical system. In addition, the switch algorithm is verified using actual data.

4.2. Friction calculation in the stick phase

In this section, an efficient way to calculate the friction force as a function of the net external force in zero-velocity region is presented. In the first place, for simplicity, stick–slip is assumed to be present in only one joint, while the others are still moving and they are away from the stick–slip region. After that, this will be generalized to span all of them.

In order to simulate the stick–slip phenomenon, the friction model must be introduced to the equation of motion in such a way that when the velocity of a joint reaches zero value it sticks if the net force lies between the two Coulomb friction limits. I.e. denoting the joint that passes the zero-velocity value by the subscript i, if

$$F_{\text{c}_i}^{+} > \left[F(t)_i - \left(\mathbf{D}(\vec{q})\ddot{\vec{q}} + \vec{C}(\vec{q}, \dot{\vec{q}})\right)_i\right] > F_{\text{c}_i}^{-} \quad \text{at} \quad \dot{q}_i = 0, \tag{5}$$

then the acceleration of this joint is zero. This can be done by searching the value of the friction that automatically leads to the zero acceleration of this joint, which can be found solving the following linear equation:

$$F_{\text{f}}(\dot{q})_i = \left[F(t)_i - \left(\mathbf{D}(\vec{q})\ddot{\vec{q}} + \vec{C}(\vec{q}, \dot{\vec{q}})\right)_i\right] \quad \text{at} \quad \dot{q}_i = 0, \quad \text{subject to } \ddot{q}_i = 0. \tag{6}$$

This equation cannot be solved simply as accelerations of the other joints are unknown, but it can be rewritten in the following form:

$$\begin{bmatrix} \ddot{q}_1 \\ \vdots \\ \ddot{q}_i^* = 0 \\ \vdots \\ \ddot{q}_n \end{bmatrix} = \mathbf{D}^{-1} \left[\begin{bmatrix} F_1 \\ \vdots \\ F_i^* \\ \vdots \\ F_n \end{bmatrix} - \begin{bmatrix} C_1 \\ \vdots \\ C_i^* \\ \vdots \\ C_n \end{bmatrix} - \begin{bmatrix} F_{f_1} \\ \vdots \\ F_{\text{f}_i}^* \\ \vdots \\ F_{\text{f}_n} \end{bmatrix} \right] \tag{7}$$

where $F_{\text{f}_i}^*$ is the desired friction value that eliminates the acceleration $\ddot{q}_i^*$ at this point. The remaining values of the friction are calculated by the friction model implemented, as a function of the corresponding joint velocity. Here we are interested only in the ith row of the right-hand side of the previous equation that is the product of the ith row of the inverse of the system inertia matrix by the adjacent vector. Note that the inverse of the inertia matrix is also needed to obtain the accelerations at this integration step.

When there are more than one joint passing the stick–slip region, the previous expression (expression (7)) produces a set of linear equations with the same number of unknown friction forces, $F_{\text{f}_i}^*$, $F_{\text{f}_j}^*$, . . ., etc., which can be solved easily. After solving these equations, if one of these frictions is outside the Coulomb friction limits, then it takes the Coulomb friction coefficient opposing motion tendency and the set of the previous linear equations are re-solved for remaining unknown friction values. This procedure is repeated until no

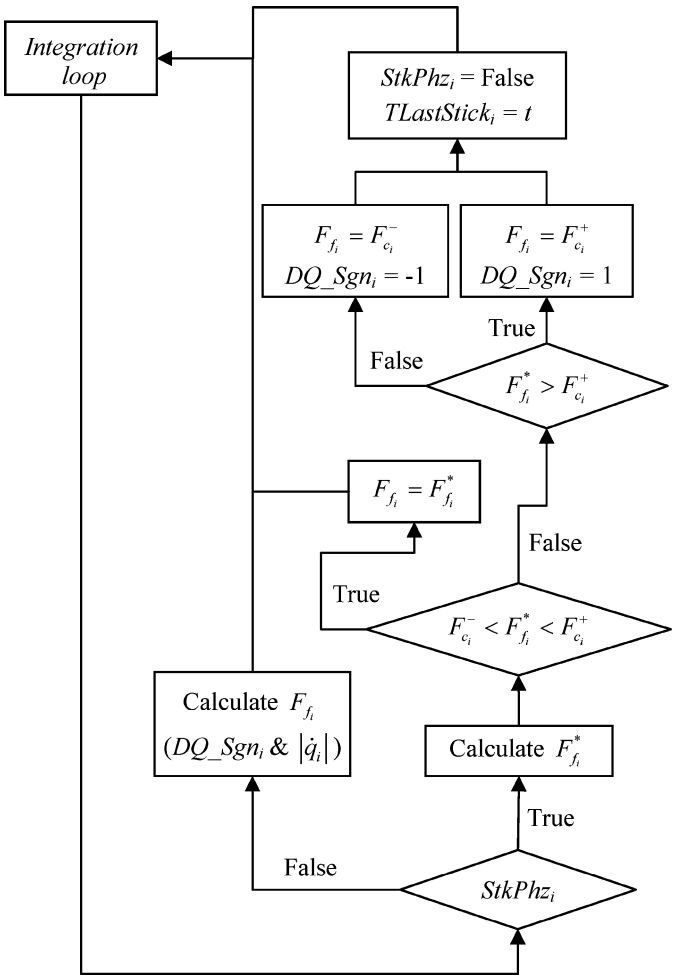

Fig. 4. Friction calculation in and out of the stick phase in the integration process. The term ($StkPhz_i$) indicates that joint i is passing the stick phase. $TLastStick_j$ and DQ_Sgn_j are terms that will be introduced later on.

friction value calculated solving the set of linear equations lies outside the Coulomb friction limits.

In the integration process, the joint that passes the stick–slip region remains in this region until the net force without friction lies outside the Coulomb friction limits. Then the net force can overcome the Coulomb friction and makes the joint move in its direction. After that, the friction is calculated by the normal friction model as a function of the corresponding velocity. These concepts are introduced in the integration as shown by the flowchart presented in Fig. 4.

4.3. Observations

The previous two subsections solve the majority of the discontinuity problem of the Coulomb friction model. Some sources of instability still need to be resolved before applying the algorithm in the integration process. They can be summarized in the following subsections.

4.3.1. Incorrect change of sign. From the definition of the aforementioned switch function one can face the problem that not all the events found correspond to a true change of sign of the switch function. For example, a positive velocity can decrease to zero, event of the switch function, and after that the conditions, net force and friction force, make the joint moves in the same previous direction (the velocity does not change its sign).

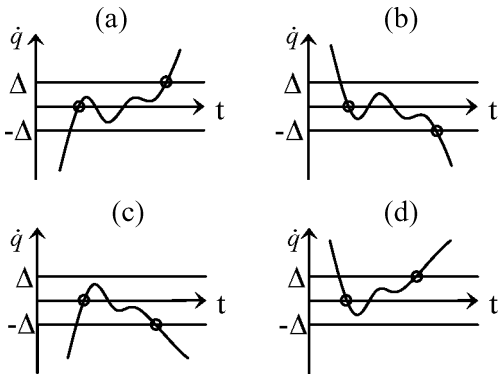

Fig. 5. The different cases where an event is detected by the switch function (represented by the small circles).

$$F_{\mathrm{f}} = \begin{cases} F_{\mathrm{c}}^{+} + F_{\mathrm{v}}^{+} \cdot \dot{q} & \text{if} \quad \mathrm{DQ\_Sgn} > 0 \\ -F_{\mathrm{c}}^{-} + F_{\mathrm{v}}^{-} \cdot \dot{q} & \text{if} \quad \mathrm{DQ\_Sgn} < 0 \end{cases}. \tag{8}$$

To solve this problem, a sign variable that represents the actual sign of the velocity is introduced. Initially, it has the same sign as the velocity. When an event is found its sign does not change if stick occurs. If the net force is higher than the Coulomb friction coefficients, then it takes the sign that opposes the net force. Friction force outside the stick phase is calculated using the normal friction model, but this time as a function of the sign of this variable (see expression (8) and the flowchart presented in Fig. 4). Another advantage of applying this variable is that it permits the integrator to accomplish some integration steps without changing the phase to the other side of discontinuity, as indicated previously, to locate the discontinuity point freely.

4.3.2. Machine precision (near zero switching points). Due to the limited machine precision an incorrect velocity change of sign can be detected. The velocity in the stick region should have exact zero value. Nevertheless, any zero-velocity value crossing, because of the machine precession, accumulated errors, or the applied integrating tolerances, could be considered as an event even though the current phase is stick. As an alternative, root detection can be stopped in the stick phase. Unfortunately, not all event finding integration routines permit this action.

This problem can be solved by applying velocity margins about the zero value where the switch function is modified in such a manner that it detects an event in two consecutive cases: when velocity changes its sign crossing zero value and when the velocity goes outside these margins. None of them can be detected unless the other had been detected previously, i.e. if joint velocity goes out of these margins then the argument of the switch function is the zero velocity. After detecting this event the argument is changed to these margins. All the possible sequences are shown in Fig. 5.

Note that only in the first case—the zero-velocity argument—sticking occurs and the net force should be calculated using Eq. (7). The corresponding switch function

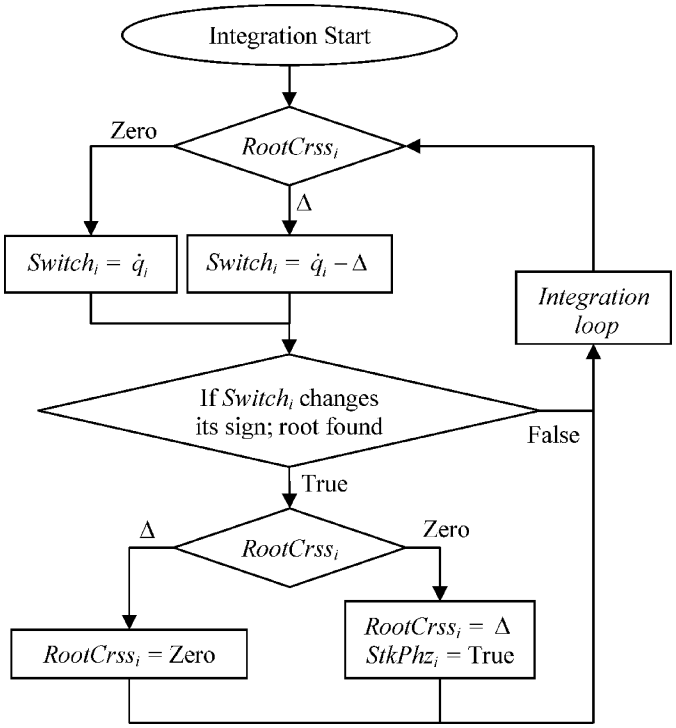

Fig. 6. An algorithm to solve the problem of near zero switching points.

can be expressed by the following equation:

$$\begin{cases} \text{Switch}_i = \dot{q}_i & \text{RootCrss} = 0 \\ \text{Switch}_i = |\dot{q}_i| - \Delta & \text{RootCrss} = \Delta \end{cases} \tag{9}$$

where, in the integration process, the integrator indicates a change of sign of the switch function (Switch_i) if the velocity changes its sign in the case of ($\text{RootCrss} = 0$) or crosses the margins in the other case ($\text{RootCrss} = \Delta$). The flowchart presented in Fig. 6 shows the method of implementation of the previous equation in the integration process.

4.3.3. Rejected integration iterations. Suppose that the joint is in the stick phase and the integrator performs an iteration in which the net force is higher/lower than the positive/negative Coulomb friction coefficient, respectively. As a result, the switch algorithm will change the state from stick to slip. But what happens if this iteration does not satisfy the applied tolerances? It will be rejected by the integrator. As a result, it will go back and try to find another iteration that satisfies tolerances. Unfortunately, the switch algorithm will not be able to calculate the net force as the phase has already been changed from stick to slip in the previous rejected iteration.

This problem is solved by applying another variable, denoted by (TLastStick_i) in the flowchart shown in Fig. 7. It conserves the time of the iteration (t) at which the switch algorithm changes the state from stick to slip. Since time in the integration process is increasing unless the iteration has been rejected, this situation can be detected by comparing (TLastStick_i) with the current iteration time. If it is lower, nothing is changed and the integrator continues its task outside the stick phase. Otherwise, the phase is returned to stick. I.e. the phase will not be changed from stick to slip unless the iteration performed has been accepted by the integrator. The value of (TLastStick_i) is assigned once $F_{f_i}^*$ is out of the Coulomb friction force limits (see Fig. 4).

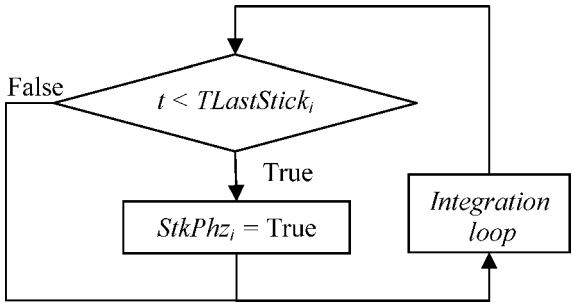

Fig. 7. An algorithm to solve the problem of rejected iterations.

4.4. Switch algorithm flowchart

All the flowcharts introduced in this section can be joined together in a single flowchart as presented in Fig. 8. Note that the value of (Δ) must be greater than the applied integration tolerances in order to prevent any unexpected margins crossing or numerical instability (this size was also proposed by Leine *et al.*[24] for the Karnopp model).

5. Results

In this section, validation of the proposed switch algorithm is presented using three examples. The first one is a benchmark to compare the behavior of the proposed algorithm with the normalization techniques. The second one is a simple sinusoidal trajectory to be able to make a comparison before and after applying the algorithm in actual simulation tasks. And finally, the last example is introduced to make evident its stability and capability in the simulation of an actual robot over a trajectory with many velocity changes of sign. The actual simulation tasks are over a 3-DoF RPS parallel manipulator using real data. The implemented integrator is the (D02QFF) integration routine that uses a variable-order variable-step Adams method with root-finding capability provided by the NAG library.[25]

5.1. Simple 1-DoF oscillator

In this subsection, the advantages of the proposed switch algorithm are shown by applying it to a simple 1-DoF oscillator (Fig. 9). The unforced case, $F = 0$, will be considered. The results obtained can be compared with those obtained by Quinn[13] in his normalization of the Coulomb friction model. The equivalent equation that represents the mechanical system can be written in the following form:

$$\ddot{x} + x = F_{\text{f}}, \tag{10}$$

where $\ddot{x}$ is the second time derivative of the position x and F_{f} is the friction force that is modeled using a symmetric Coulomb friction model with $F_{\text{C}} = 1\text{N}$. Now, starting from the same initial conditions as in the previously mentioned work, $(x(0), \dot{x}(0)) = (2.75, 0)$, and integrating to $t = 10$ s using the previously mentioned NAG routine, the response of the switch algorithm is shown as the time history of the

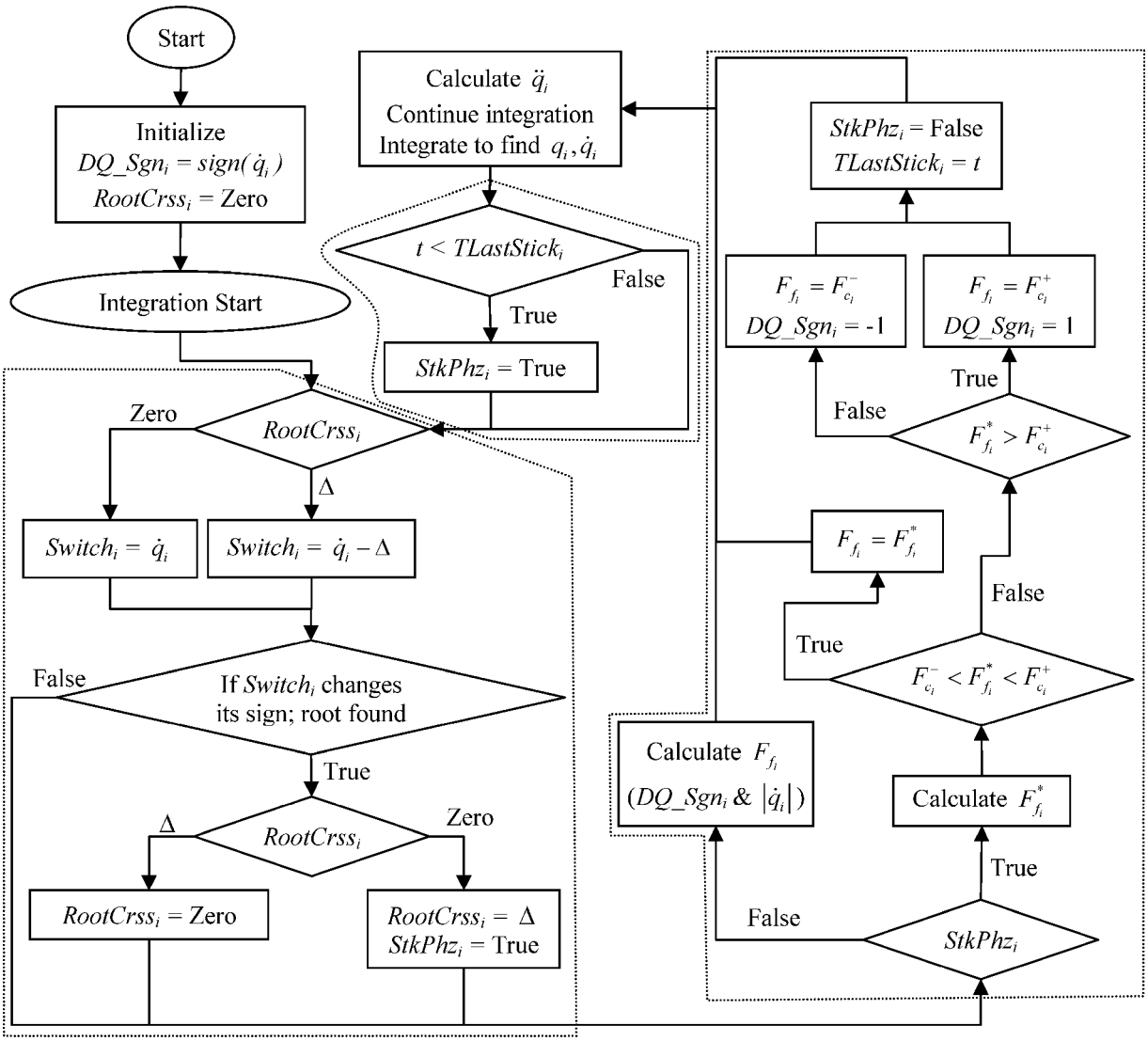

Fig. 8. Switch algorithm flowchart.

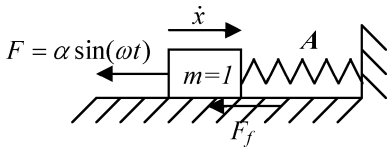

Fig. 9. A simple 1-DoF oscillator.

integration process presented in Fig. 10. This figure presents all the used iterations by the integration routine, as internal calls to the function to be integrated, whether they were accepted or rejected in the process.

As can be observed from the previous figures, the integrator accomplished the first part of integration (slip portion) as normal and located the zero-velocity point easily. Observe that it had been permitted to accomplish the opposite velocity sign iterations without changing the sign of the friction force (the marked velocity point). Once zero velocity had been found, the corresponding friction force was calculated as proposed in the flowchart and then compared with the Coulomb friction limits. Since it was within these limits, the corresponding calculated acceleration was zero (to the machine precession) and similarly the following velocities had maintained the zero value (to the machine precession and the applied integration tolerances). Another important observation is the number of iterations accomplished in the stick phase; in fact, it was a simple task for the integrator away from oscillation and instability problems, simpler even than the slip phase! Note that when the system reached the stick phase $x = -0.74999996$, the only difference between this value and the true one $x = -0.750000$ calculated by Quinn[13] is because of the applied integration tolerances (no normalization error is being introduced).

5.2. Actual simulation

Once the model has been verified using a simple example, it is applied in the simulation process of an actual 3-DoF RPS parallel manipulator shown in Fig. 11. As can be seen, it consists of a fixed base and a moving platform that is guided by three linear actuators. These are actuated by three DC motors controlled by an open architecture PID. The dynamic model was constructed using dynamic parameters obtained from an identification process established by Farhat *et al.*[26] Usually, in this kind of parallel robots, the friction

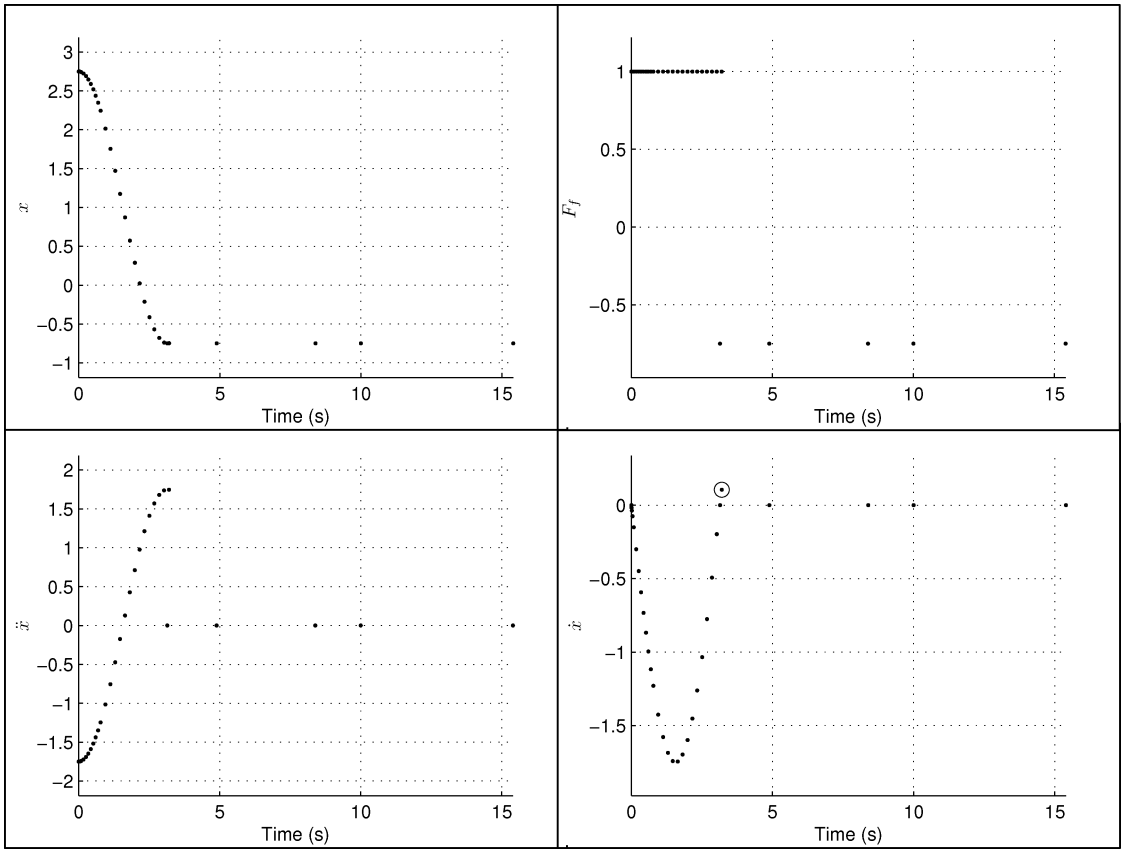

Fig. 10. The response of the switch algorithm for a simple example.

in passive joints (revolute and spherical joints) is negligible compared with the active joints (prismatic ones). This issue was experimentally verified in the previously mentioned identification process.

In the simulation process, the applied forces are those calculated by the control unit applied to the parallel manipulator to follow the corresponding trajectory. Results are shown for two types of trajectories. Firstly, they are shown in detail for a simple sinusoidal trajectory and then the robustness of the switch algorithm is demonstrated applying it over a complex Fourier series exciting trajectory.

5.3. Simple sinusoidal trajectory

The sinusoidal trajectory and the corresponding applied forces are show in Fig. 12.

Firstly, when the equations of motion were integrated without considering the proposed algorithm the integrator was not able to continue after the first velocity change of sign. Stiff differential equations were obtained. Integration was held with absolute and relative errors of 1E-06 and 1E-08, respectively. In order to make the stiffness of the process evident, figures are presented showing the time history of the independent generalized velocities, the calculated friction

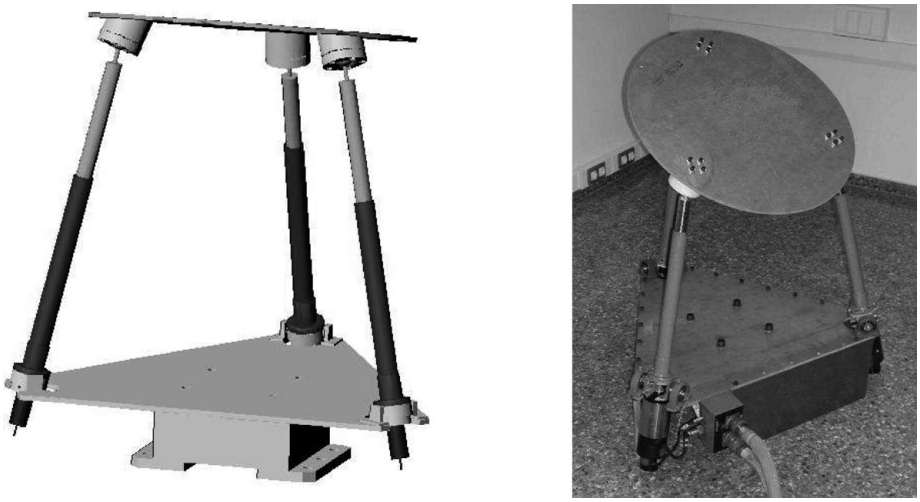

Fig. 11. The 3-DoF RPS parallel manipulator used in the simulation process. Simulated model on the left and the actual one on the right.

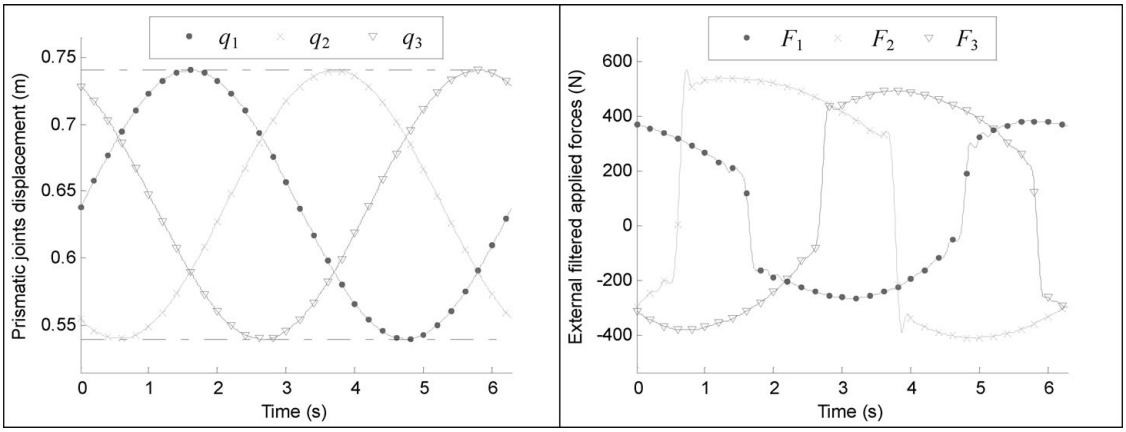

Fig. 12. A simple sinusoidal trajectory with the corresponding applied forces.

force by the corresponding friction model, the applied forces, and the resulting generalized accelerations for each integration iteration (see Fig. 13). As for Fig. 10, all the internal iterations made by the integration routine are shown in this figure, whether they were accepted or rejected in the process.

As can be seen, when the velocity of the second joint reached zero value, at $t = 0.64$ s, the corresponding friction force oscillated between the two Coulomb friction limits, as a result the calculated accelerations for all joints have a similar oscillation. Observe that all the iterations made posterior to $t = 0.64$ s were rejected as they did not satisfy the applied tolerances. The integrator was not capable to find an iteration posterior to this time, hence the integration has been interrupted. On the other hand, when the proposed algorithm had been applied, this problem was solved and integration was accomplished over the entire trajectory (see Fig. 14 for the corresponding integration history).

As can be observed from Fig. 14, the oscillation due to the discontinuity problem at $t = 0.64$ s has been removed with the application of the proposed switch algorithm. Similarly for the other zero-velocity points of the trajectory.

5.4. Fourier series trajectory

In this example, the robustness of the switch algorithm is presented. The trajectory is used in the identification process of the dynamic parameters of the parallel robot mentioned previously. So it is an exciting trajectory that contains many points of velocity changes of sign (see Fig. 15).

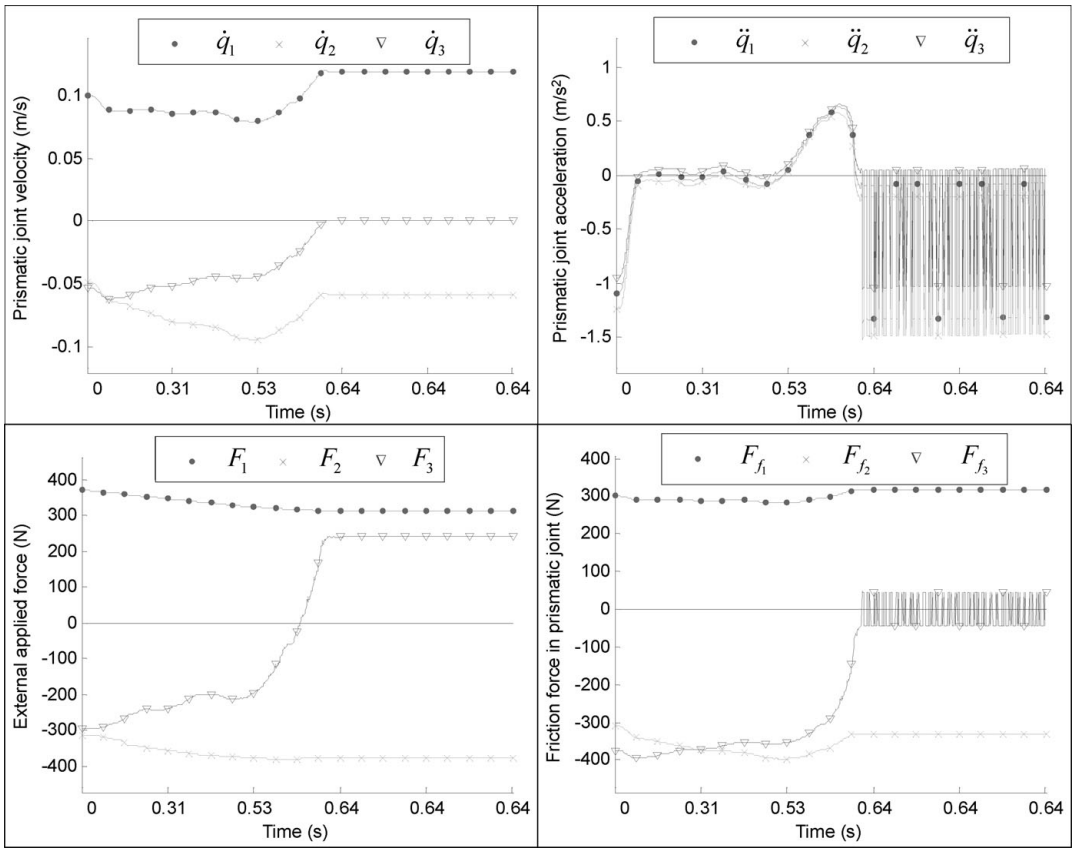

Fig. 13. Stiff differential equations behavior.

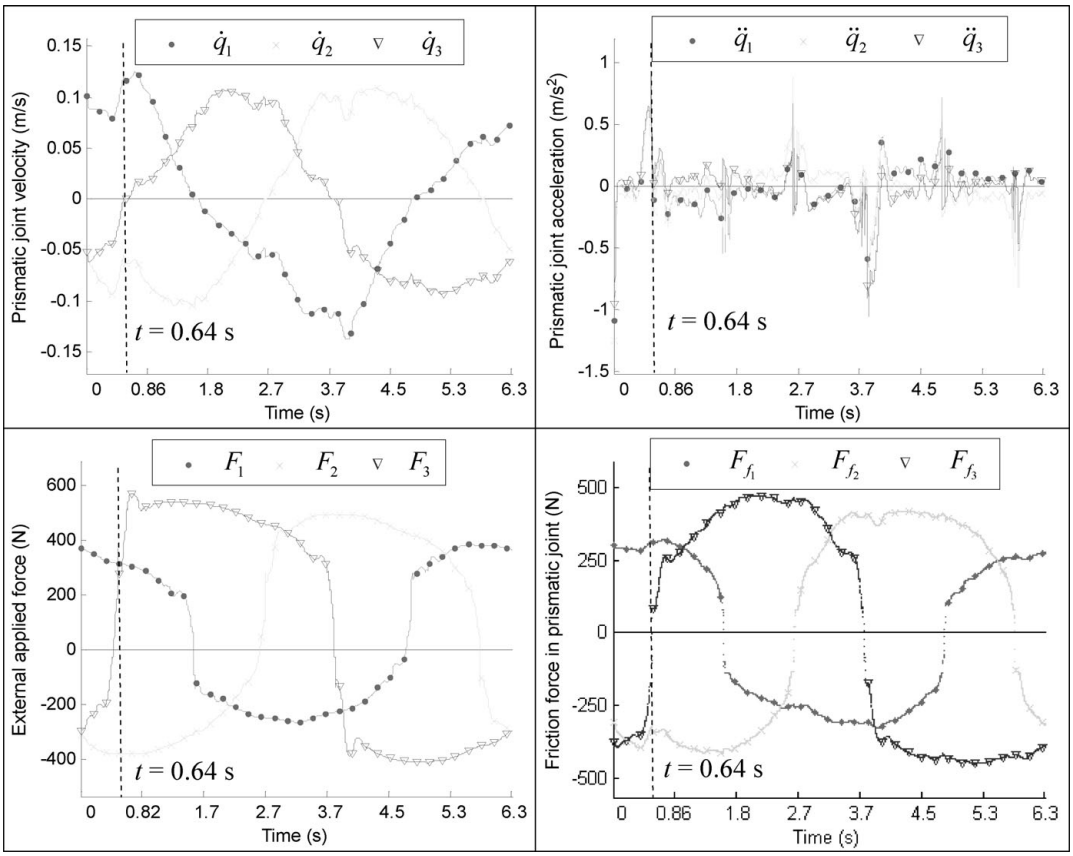

Fig. 14. Integration considering the proposed switch algorithm.

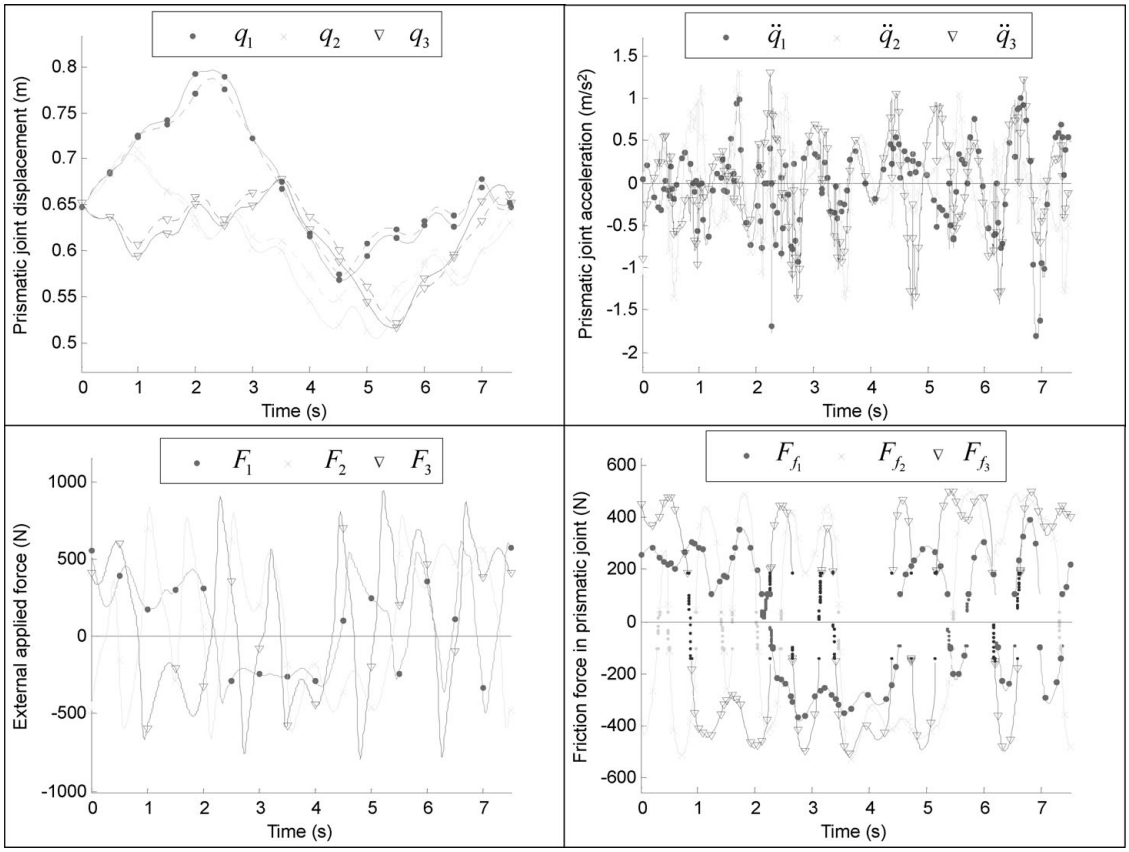

Fig. 15. Fourier series trajectory integration results. In the first plot continuous and dashed lines represent the actual trajectory followed by the robot and the integrated one, respectively.

As can be observed from Fig. 15, the problem of discontinuity that accompanies the direct application of the Coulomb friction model has been totally eliminated; the correct value of the friction force has been calculated in the stick phase (points not interconnected by lines in the friction figure), hence there is no oscillation in the corresponding value of the acceleration (acceleration figure) ensuring the stability of the integration process. Therefore, integration could be accomplished using a conventional nonstiff integration routine.

6. Conclusion

In this paper, we have presented an algorithm that solves the stiff differential equations problem that accompanies the discontinuity of the Coulomb friction model. With this algorithm the exact stick–slip phenomenon has been simulated without any normalization of the friction model. Fundamentally, it is based on two concepts: the application of the switch function that divides the integration interval into subintervals and the calculation of the friction force in the stick phase. Thorough investigation of integrator behavior in the simulation process considering the Coulomb friction model led to further improvements that enhanced the stability and robustness of the simulation process and totally eliminated the stiffness problem, enabling the use of nonstiff integration routines.

This algorithm has been applied in the simulation process of a 1-DoF oscillator, unforced model, to show its advantages over the normalization method. No integration problem has been faced in the stick phase. On the contrary, it was an even simpler task than the slip phase. Using real data, applied forces calculated by the control unit, and identified dynamical parameters, it has been applied to the simulation of the movement of an actual 3-DoF parallel manipulator over simple and complex trajectories. The results of the first trajectory were presented in detail, with and without the application of the switch algorithm, to show its behavior in the stick–slip phase in the simulation process. Meanwhile, the complex one shows its robustness and stability in accomplishing the real complex simulation tasks. The algorithm has been provided in a flowchart for ease of future implementation. It could be improved to span other sources of discontinuity in the equations of motion such as the contact problem.

Acknowledgment

This research has been partially funded by the Ministerio Español de Ciencia y Tecnología (project # DPI2005-08732-C02-01).

References

1. J. C. Piedbœuf, J. de Carufel and R. Hurteau, "Friction and stick–slip in robots: Simulation and experimentation," *Multibody Syst. Dyn.* **4**, 341–354 (2000).
2. P. Dahl, *A Solid Friction Model* (The Aerospace Corporation, El Segundo, CA, 1968).
3. D. A. Haessig and B. Friedland, "On the modelling and simulation of friction," *J. Dyn. Syst. Meas. Control Trans. ASME* **113**(3), 354–362 (1991).
4. C. Canudas-de-Wit, H. Olsson, K. J. Aström and P. Lischinsky, "A new model for control of systems with friction," *IEEE Trans. Automat. Control* **40**(3), 419–425 (1995).
5. K. Kozlowski, *Modelling and Identification in Robotics* (Springer-Verlag, London, 1998).
6. W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots* (Taylor & Francis, London, 2002).
7. M. Grotjahn, B. Heimann and H. Abdellatif, "Identification of friction and rigid-body dynamics of parallel kinematic structures for model-based control," *Multibody Syst. Dyn.* **11**(3), 273–294 (2004).
8. V. Mata, F. Benimeli, N. Farhat and A. Valera, "Dynamic parameter identification in industrial robots considering physical feasibility," *J. Adv. Rob.* **19**(1), 101–120 (2005).
9. B. Armstrong-Helouvry, P. Dupont and C. C. De Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction," *Automatica* **30**(7), 1083–1138 (1994).
10. H. Olsson, K. J. Åström, C. Canudas-de-Wit, M. Gäfvert and P. Lischinsky, "Friction models and friction compensation," *Eur. J. Control* **4**(3), 176–195 (1998).
11. E. Eich Soellner and C. Führer, *Numerical Methods in Multibody Dynamics* (B. G. Teubner Stuttgart, Germany, 1998).
12. W. Sextro, *Dynamical Contact Problems with Friction: Models, Methods, Experiments, and Applications* (Springer-Verlag, Berlin, 2002).
13. D. D. Quinn, "A new regularization of Coulomb friction," *J. Vibration Acoust.* **126**(3), 391–397 (2004).
14. R. I. Leine, D. H. Van Campen and B. L. Van De Vrande, "Bifurcations in nonlinear discontinuous systems," *Nonlinear Dyn.* **23**(2), 105–164 (2000).
15. D. Karnopp, "Computer simulation of slip–stick friction in mechanical dynamic systems," *J. Dyn. Syst. Meas. Control* **107**(1), 100–103 (1985).
16. R. H. A. Hensen, M. J. G. van de Molengraft and M. Steinbuch, "Friction induced hunting limit cycles: A comparison between the LuGre and switch friction model," *Automatica* **39**(12), 2131–2137 (2003).
17. B. L. Van de Vrande, D. H. van Campen, B. d. Kraker and Eindhoven, "Some Aspects of the Analysis of Stick–Slip Vibrations with an Application to Drillstrings," *Proceedings of ASME Design Engineering Technical Conference, 16th Biennial Conference on Mechanical Vibration and Noise*, Sacramento (1997).
18. F. Altpeter, *Friction Modeling, Identification and Compensation*, vol. 166 (Mechanical Engineering Department, Federal Polytechnic of Lausanne, Valais, 1999).
19. F. Pfeiffer and C. Glocker, *Multibody Dynamics with Unilateral Contacts* (John Wiley & Sons, New York, 1996).
20. V. Mata, S. Provenzano, F. Valero and J. I. Cuadrado, "Serial-robot dynamics algorithms for moderately large numbers of joints," *Mech. Mach. Theory* **37**(8), 739–755 (2002).
21. F. Udwadia and R. Kalaba, "The explicit Gibbs–Appell equation and generalized inverse forms," *Q. Appl. Math.* **56**(2), 277–288 (1998).
22. F. M. L. Amirouche, *Computational Methods in Multibody Dynamics* (Prentice-Hall, New Jersey, 1992).
23. J. García de Jalón and E. Bayo, *Kinematic and Dynamic Simulation of Multibody Systems: The Real-Time Challenge* (Springer-Verlag, New York, 1994).
24. R. I. Leine, D. H. V. Campen and A. D. Kraker, "Stick–slip vibrations induced by alternate friction models," *Nonlinear Dyn.* **16**(1), 41–54 (1998).
25. NAG, *The NAG Fortran Library Manual, Mark 20* (The Numerical Algorithms Group Ltd., Oxford, UK, 2001).
26. N. Farhat, V. Mata, Á. Page and F. Valero, "Identification of dynamic parameters of a 3-DOF RPS parallel manipulator," *Mech. Mach. Theory* **43**(1), 1–17 (2008).