

# Trajectory tracking for robot manipulators using differential flatness

## Seguimiento de trayectorias en manipuladores robóticos usando Differential Flatness

Elkin Veslin Diaz<sup>1</sup>, Jules Slama<sup>2</sup>, Max Dutra<sup>3</sup>, Omar Lengerke<sup>4</sup>, Magda Morales Tavera<sup>5</sup>

### ABSTRACT

This paper proposes applying differential flatness to robot manipulator trajectory tracking. The trajectories for each generalised coordinate are proposed as a function and the corresponding input must be found to guarantee tracking. It is shown that the position in the generalised coordinates and their derivatives are flat inputs which, together with a PD controller, could determine (with some restrictions) manipulator movement having minimal deviation throughout its trajectory in both plane movements and in space.

**Keywords:** condition-based maintenance, fault detection, neuronal networks, permanent magnet synchronous motor.

### RESUMEN

Este documento propone una aplicación con Differential Flatness para el problema de seguimiento de trayectorias en manipuladores robóticos. Para cada coordenada generalizada, se proponen sus trayectorias como una función en el tiempo donde deben encontrar las entradas correspondientes para garantizar el seguimiento. Se demuestra que la posición de cada coordenada generalizada del manipulador robótico y sus correspondientes derivadas son salidas planas que, en conjunto con un controlador PD pueden determinar, con algunas restricciones, los valores de fuerza para conseguir un movimiento en el manipulador con una mínima desviación a lo largo del trayecto, tanto en movimientos planos como en el espacio.

**Palabras Claves:** Seguimiento de trayectorias, Manipuladores robóticos, Brazo Humano, Differential Flatness, Control PD.

**Received:** May 31th 2010

**Accepted:** June 3th 2011

### Introduction and the state of the art

Robot manipulators are defined as all automatically programmable systems which can position and orientate an object following variable trajectories inside a defined workspace. A robot manipulator is formed by different segments united through joints and system's movement capability depends on each joint's degree of freedom.

Manipulator trajectory tracking consists of determining the necessary input (forces) in each system's generalised coordinates to make the model move between one point and another following a defined path. The first step consists of specifying a sequence of points in the manipulator's work space; these points will become desired positions along such pathway and are integrated using an interpolation function (typically in polynomial form). Different techniques are used for trajectory planning: (i) final and ending positions are given regarding point-to-point motion, or (ii) working with a finite point sequence needs motion through a sequence of points. Both techniques give a time function that describes the desired behaviour (Siciliano, 2009, van Nieuwstadt, 1997a).

The second step consists of implementing such function in a dynamic model of the proposed system and verifying that trajectory tracking is effectively given. A decision must be taken regarding whether the problem is going to be worked in the opera-

tional space or at each joint. The first solution could result in singularity problems and redundancy caused by the number of joints' degrees of freedom and the nonlinear effects could hamper prediction, given its generality. Meanwhile, joint position path parameterisation would seem more suitable because this study allows working on each element, considering factors like movement restriction and each joint's degrees of freedom.

Joints' parameterised trajectory is defined by a  $q(t)$  function. This function is the source of a joint desired position for the manipulator's movement as time elapses. Control techniques, such as PD control with gravity compensation, inverse dynamic control (IDC) or adaptive control, enable manipulator movement along a defined path having minimal deviation (Marguitu, 2009, Carvajal, 2007, Spong, 1992, Wang, 2009).

It has also been demonstrated that this mechanical system is differentially flat (Lévine, 2009), meaning that the system has a collection of outputs (called flat outputs) which, according to differential flatness, could describe the whole system (as well as its derivatives) (Fliess, 1994).

The differential flatness concept was introduced by Michel Fliess and his group, using differential algebra terminology (Fliess, 1994). A system is seen to be a differential camp generated by a set of variables (states and inputs). Subsequently, Martin (1997) redefined this concept within a whole geometric context with

<sup>1</sup> Mechatronics Engineering UNAB, M.Sc. Mechanical Engineering. COPPE/UFRJ, Brazil. Universidad de Boyaca., Colombia. eyveslin@uniboyaca.edu.co

<sup>2</sup> Electronique Electrotechnique Automatique, Université De Provence-Faculté De St Jerome, M.Sc. Acoustic" DEA d'Acoustique" - Université de Provence Ph.D. Acoustique et Dynamique des Vibrations, Université d'AixMarseille II. Professor COPPE/UFRJ, Brazil. Jules@mecanica.coppe.ufrj.br

<sup>3</sup> Dr-Ing. in Robotics, DUISBURG, Germany. M.Sc. in Mechanical Engineering COPPE/UFRJ. Mechanical Engineer UFF, Brazil. Professor COPPE/UFRJ, Brazil, max@mecanica.coppe.ufrj.br

<sup>4</sup> Ph.D. Mechanical Engineering COPPE UFRJ, M.Sc. Automation and Control, ITEMS-CEM México, Computational Systems Engineer, UNAB, Colombia. Researcher COPPE UFRJ, Brazil. Professor UNAB, Colombia. olengerke@unab.edu.co

<sup>5</sup> Mechatronics Engineering UNAB, M.Sc. Mechanical Engineering. COPPE UFRJ, Brazil. Ph.D. Mechanical Engineering COPPE UFRJ. mjmtavera@ufrj.br

which flat systems could be described in absolute equivalence terms.

According to Fliess (1994), a system is defined as being flat if a set of outputs (having the same number of inputs), called *flat outputs*, can determine all system states and inputs without the need for integration. That means that a system of states  $x \in n$  and inputs  $u \in m$  is flat if a set of outputs  $y \in m$  can be found in the following form:

$$y = h(x, u, \dot{u}, \dots, u^{(r)}) \quad (1)$$

so that:

$$x = f(y, \dot{y}, \dots, u^{(q)}) \quad (2)$$

$$u = g(y, \dot{y}, \dots, u^{(q)}) \quad (3)$$

As seen in (2) and (3), all flat system states and inputs are defined regarding flat output and a finite number of its derivatives. This property has been called diffeomorphism by Fliess; such association is helpful in situations where explicit trajectory generation is required (Martin, 1997), i.e. a system has desired behaviour which is achieved by using a defined path in the output space. Equations (2) and (3) determine appropriate input for developing such path. Several studies have illustrated the use of differential flatness, including mobile robots (Murray, 1995a, Deligiannis, 2006, Defoort, 2006), mobile robot with  $n$  trailers (Siciliano, 2009), Chua's system (Aguilar-Ibañez, 2004), magnetic synchronous motor (Achir, 2005), flexible beam (Barczyk, 2008) and robot manipulator (Murray, 1995b). All these systems can work with these theories because there is a relationship between flat input and output; a system's flatness thus depends on its configuration, thereby making it an intrinsic property of the model itself.

This paper highlights this property for trajectory tracking; the desired path has been formulated in each generalised coordinate  $q$ , using a time function. Parameterisation, combined with differential flatness concepts and PD control, has determined the input set allowing manipulator movement control regarding such trajectory.

## Problem resolution

### Differential flatness in Lagrangian models

Robot manipulator modelling can be addressed via two approaches: Lagrangian and Newton-Euler analysis. Both differ in procedure, the former using an energetic conservation concept whilst the latter analyses forces and equilibrium based on Newton's second law. Both ideas provide the same answer: a differential equation system describing manipulator dynamics.

A model obtained by Lagrangian theory would be defined as:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + K(q) = F \quad (4)$$

It has been stated a Lagrangian system is fully actuated when it has the same number of actuators and generalised coordinates. A fully actuated model is a differential flat system (van Nieuwstadt, 1997b, Rathinam, 1997) since all states and non-conservative forces  $F$ , could be determined from flat output  $q$ .

The advantage of such structure is the small number of flat output derivatives necessary to transform the system into a flat concept.

Even though Van Nieuwstadt(1997b) and Rathinam (1997) have detailed such flat property in Lagrangian systems, Lévine (2009) has demonstrated it. Starting from defining a robot manipulator having  $n$  degrees of freedom and  $n$  actuators, its dynamic model would be:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = Q(\dot{q}, q)u \quad (5)$$

where  $q$  are generalised coordinates and  $u$  system input represented by actuators. Assuming that system input and generalised coordinates have the same dimension and that  $M(q)$  represents the inertial forces' matrix, then  $C(q, \dot{q})\dot{q}$  represents the centrifugal and centripetal forces' matrix and  $Q(q, \dot{q})$  characterises the actuators (e.g. arm location or the presence of motored units). If  $x_1 = q, x_2 = \dot{q}$ , equation (5) becomes a set of EDOs:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= M(x_1)^{-1}(C(x_1, x_2) - Q(x_1, x_2)u) \end{aligned} \quad (6)$$

If the  $Q$  dimension is  $n$ , all elements in(6) could be deleted, so the answer becomes reduced to the trivial form  $0=0$ . Thus a set of inputs  $u$  could be:

$$u = Q^{-1}(x_1, \dot{x}_1)(M(x_1)\ddot{x}_1 + C(x_1, \dot{x}_1)) \quad (7)$$

where  $x_1$  represents the flat output vector (each of the manipulator's generalised coordinates); a system's input representation depending on flat output may thus be obtained. The (7) formula is also known as computed torque or IDC, this being an inverse dynamics technique for determining the necessary amount of momentum to develop any kind of movement in a robot manipulator.

It should be membered that not all system outputs are flat, so these must be differentiated from normal ones by the letter  $Z$  (Van Nieuwstadt, 1997b). Equation (7) can thus be represented as:

$$u = Q^{-1}(z, \dot{z})(M(z)\ddot{z} + C(z, \dot{z})) \quad (8)$$

Diffeomorphism is thus demonstrated.

### Flat output and trajectories

Tracking trajectories do not present complications in a flat system; as input and its states are defined in terms of flat output then paths can be created for defining its output behaviour and its respective derivatives, determining the inputs needed by the system to give a response approaching the one desired by the path. These paths could be polynomial or  $C^\infty$  type functions, such as trigonometrics or exponentials (Rotella and Zambettakis, 2008). Applying flat theories leads to transforming a differential equation system (7) into polynomial order  $n$ , thereby simplifying

problem solution by making it into a completely algebraic solution.

**Implementation in a human arm-type manipulator**

The behaviour of a human arm-type robotic manipulator should be studied to implement this theory (such system emulating human arm performance) as this forms the basis for exoskeleton and upper limb prosthesis research. Akinematic model has been studied (Veslin, 2009) and is described in Figure 1; it is a multi-body system formed by 2 segments.  $L_1$  is the arm and is formed by joint ( $P_0$ ) that is analogous to a human shoulder. It has three degree of freedom, allowing the system's free movement around the shoulder.  $L_2$  is the forearm ( $P_1$ ) and only has one degree of freedom, rotation around the Y-axis, a movement known as flexion. Hand movements have not been considered but parameters like mass and length will be added to the forearm, turning the forearm-arm system into a single body one.

Arm movement was studied regarding the sagittal plane where the shoulder and forearm rotate to produce flexion (rotations around the Y axis) and then movement using all degrees of freedom (four). Figure 1 describes the elements exerting influence, such as the position of the segment's centre of mass (defined by  $C_1$  and  $C_2$ ), system orientation on the axis and rotations.

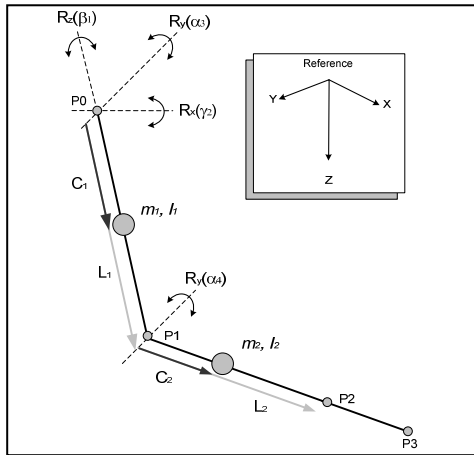


Figure 1. Human arm structure on which the dynamic study was based

**Plane movements**

Plane movement means movements developed by the system in one direction without the intervention of other shoulder rotations. It means both articulations move around the same axis. This system was modelled (9,10) taking into account that rotation axis angular and kinetic velocities did not influence the model, thereby being considered null. The following was obtained for the first segment:

$$\begin{aligned}
 & -m_2 g s_{14} C_2 + m_2 C_2^2 \ddot{q}_4 + m_2 (\ddot{q}_1) L_1^2 - m_1 g c_1 C_1 \\
 & + 2m_2 C_2 (\dot{q}_1) L_1 c_4 - m_2 g s_1 L_1 \\
 & + m_1 (\ddot{q}_1) C_1^2 + \frac{1}{2} m_1 (\ddot{q}_1) L_1^2 \\
 & + m_2 C_2 (\ddot{q}_4) L_1 c_4 \\
 & - 2m_2 C_2 (\dot{q}_1) (\dot{q}_4) L_1 s_4 + \frac{1}{2} m_2 L_2^2 \ddot{q}_1 \\
 & + \frac{1}{12} m_2 L_2^2 \ddot{q}_4 - m_2 C_2 (\dot{q}_4)^2 L_1 s_4 = U_1
 \end{aligned} \tag{9}$$

and the following for the second segment:

$$\begin{aligned}
 & -m_2 g s_{14} C_2 + m_2 C_2^2 \ddot{q}_1 + m_2 C_2^2 \ddot{q}_4 + m_2 C_2 (\ddot{q}_1) L_1 c_4 \\
 & + m_2 C_2 (\dot{q}_1)^2 L_1 s_4 + 1/12 m_2 L_2^2 \ddot{q}_1 + 1/12 m_2 L_2^2 \ddot{q}_4 = U_2
 \end{aligned} \tag{10}$$

where  $m_1$  and  $m_2$ , were the mass of each articulation and  $s_n$  and  $c_n$  were  $sen(q_n)$  and  $cos(q_n)$ , respectively, for reducing the equation's complexity. The inertia tensor was not considered, as dynamics studies have shown that it becomes reduced to a precise force located in the centre of mass, thereby having no strong influence on simulation model behaviour (Veslin, 2010). Both equations represented a nonlinear system, and according to (8), a differential flat system having flat outputs  $q_1$  and  $q_4$ , corresponding to the model's generalised coordinates. Henceforth, these outputs will be denoted as  $z_1(t)$  and  $z_4(t)$ , to emphasise that output was used to apply the flat system concept. This led to:

$$\begin{aligned}
 & -m_2 g s_{14} C_2 + m_2 C_2^2 \ddot{z}_1 + m_2 C_2^2 \ddot{z}_4 + m_2 (\ddot{z}_1) L_1^2 - m_1 g c_1 C_1 + \\
 & 2m_2 C_2 (\ddot{z}_1) L_1 c_4 - m_2 g s_1 L_1 + m_1 (\ddot{z}_1) C_1^2 + \frac{1}{12} m_1 (\ddot{z}_1) L_1^2 + \\
 & m_2 C_2 (\ddot{z}_4) L_1 c_4 - 2m_2 C_2 (\dot{z}_1) (\dot{z}_4) L_1 s_4 + \frac{1}{12} m_2 L_2^2 \ddot{z}_1 + \\
 & \frac{1}{12} m_2 L_2^2 \ddot{z}_4 - m_2 C_2 (\dot{z}_4)^2 L_1 s_4 = U_1
 \end{aligned} \tag{11}$$

$$\begin{aligned}
 & -m_2 g s_{14} C_2 + m_2 C_2^2 \ddot{z}_1 + m_2 C_2^2 \ddot{z}_4 + m_2 C_2 (\ddot{z}_1) L_1 c_4 \\
 & + m_2 C_2 (\dot{z}_1)^2 L_1 s_4 + 1/12 m_2 L_2^2 \ddot{z}_1 + 1/12 m_2 L_2^2 \ddot{z}_4 = U_2
 \end{aligned} \tag{12}$$

The next step consisted of applying differential flat theories to the model; the general objective was to control system behaviour regarding a desired action. This behaviour was parameterised regarding a particular trajectory. Such trajectories could be easily implemented with knowledge of a system's flat output. The model demonstrated in (9) and (10) shows that system input can be interpreted by knowing its flat output behaviour until the second derivative; this function and its derivatives were thus added to the model. For example, if the arm is to rise starting from an original position with gradual positioning variation, then a cubic function could be used to command its behaviour, (2007), i.e.:

$$z_1(t) = 0,0094t^2 - 0,0006t^3 \tag{13}$$

$$z_4(t) = 0,0157t^2 - 0,001t^3 \tag{14}$$

Equation (14) led the arm from its original position at  $0^\circ$  to a position at  $18^\circ$  in equation (13) to  $30^\circ$  (Fig. 3). Both segments had zero speed at the beginning and the end and took 10 seconds to move. These equations were derived twice and the following was obtained for the first flat output:

$$\begin{aligned}
 \dot{z}_1(t) &= 0,0188t - 0,0018t^2 \\
 \ddot{z}_1(t) &= 0,0188 - 0,0036t
 \end{aligned} \tag{15}$$

and the following for the second flat output:

$$\begin{aligned}
 \dot{z}_4(t) &= 0,0314t^2 - 0,003t^2 \\
 \ddot{z}_4(t) &= 0,0628t - 0,006t
 \end{aligned} \tag{16}$$

If these equations were inserted into (11) and (12) then this

would lead to a set of polynomial equations, transforming differential equations into an algebraic-type system. The system must be solved in a given time to determine inputs  $U_n$ .

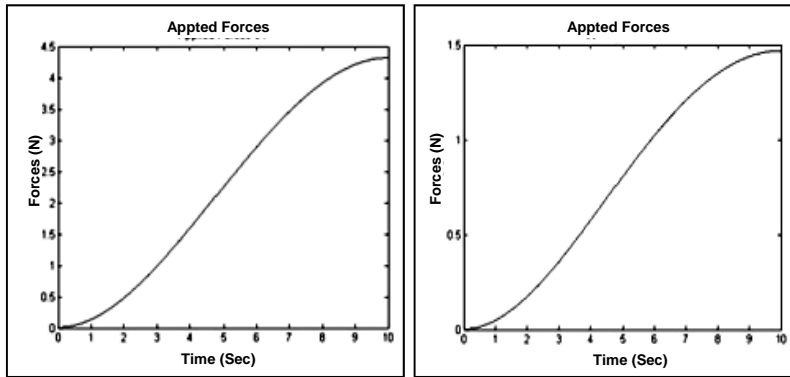


Figure 2. Set of inputs  $U_1$  and  $U_2$  used for the desired trajectory

The response provided by Figure 2 shows a set of outputs applied to a system and displace it according to the behaviour implemented in equations (13) and (14). The system was rewritten as a set of ordinary differential equations, similar to (6) and its offset response is described in Figure 3.

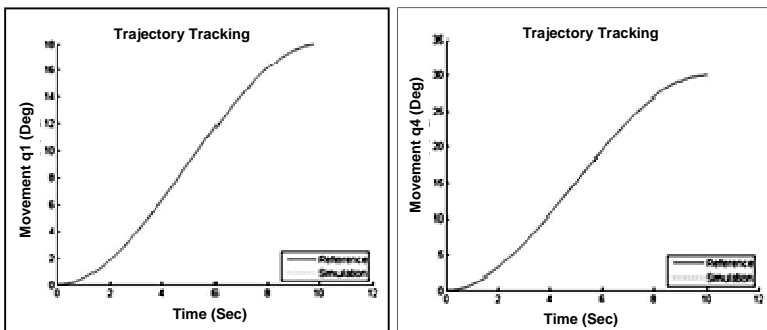


Figure 3. System response for  $U_1$  and  $U_2$  input

Figure 3 presents input behaviours specified for equation system (11) and (12) and compares it to the set of cubic equations (13) and (14), demonstrating that the system effectively displayed the desired behaviour. The trajectories' difference can be seen in Figure 4, showing an oscillatory error trend, maximum difference being  $10^{-2}$ .

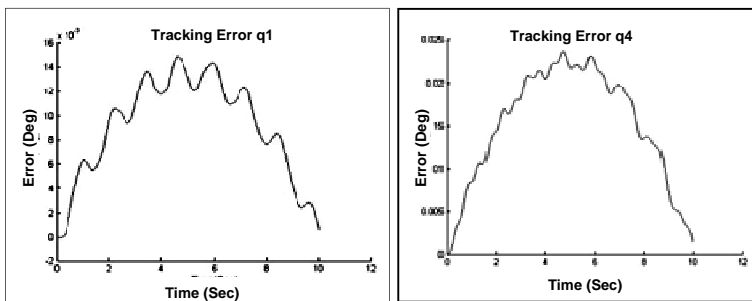


Figure 4. Tracking error for  $q_1$  and  $q_4$

While the response obtained was desirable, stability problems were encountered for position tests following the same cubic behaviour which were caused by a leak in the desired path system involving random movement (Figure 5), when trying to move both segments from  $0^\circ$  to  $90^\circ$ .

It was demonstrated that the presence of gravitational forces were one of the main causes of path deviation when trying to resolve the aforementioned problem; removing them led to establishing track near the desired trajectory. The resulting magnitudes in  $U_1$  and  $U_2$  did not offer a satisfactory solution since their value was really low.

This did not mean that flat system theory could not be applied to certain positions, only that such solutions did not guarantee the tracking and the necessary stability if applied in an open-loop. The focus of this moment had to be changed to take advantage of this scheme's benefits by implementing a controller that feedback information from real state and compared it to desired output. This difference produced a control action allowing the system to track the desired trajectory with minimum error.

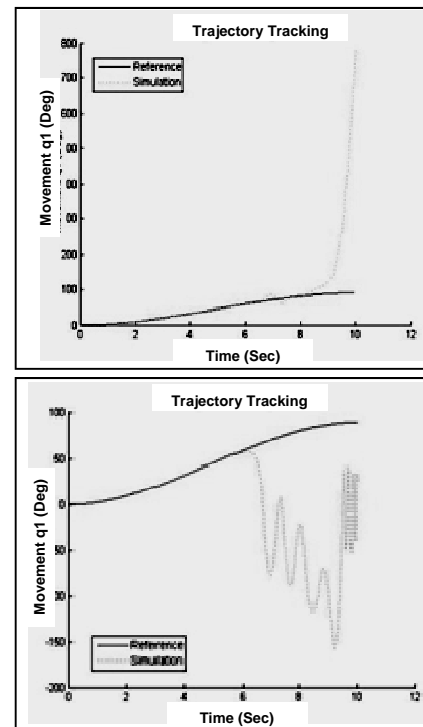


Figure 5. Trajectory deviation for  $q_1$  and  $q_4$  in higher moves

### Degrees of freedom control

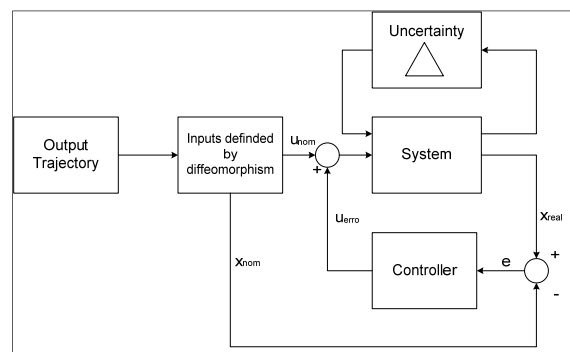


Figure 6. Two degrees of freedom control system (Van Nieuwstadt, 1997b)

A two degrees of freedom control (Fig. 6) is any closed-loop sys-

tem containing a controller and a trajectory generator. The controller modifies output regarding existing error between actual output and nominal or desired output (Van Nieuwstadt, 1997b).

This system actually had three degrees of freedom. A higher level produced output depending on desired behaviour paths, an intermediate level generated input based on these outputs and a lower level stabilised the system concerning nominal trajectory. Uncertainty was defined as variation caused by gravitational, frictional and nonlinear forces in the model.

A feedback law has been established for the controller (Franch, 2003):

$$u_n = u_n + k_p(z_n - x_n) + k_d(\dot{z}_n - \dot{x}_n) \tag{17}$$

with  $n=1,2$ ; this equation was selected to search for nominal state  $X_n$  following desired path  $Z_n$  in line with correct choice  $k_p$  and  $k_d$  constants. Proportional constant  $k_p$  produced an action that modified input from flatness system control. Derivative constant  $k_d$  removed variation (oscillations) caused by this correction. Figure 5 shows that there were jumps in position ranging from  $100^\circ$  to  $800^\circ$  in less than a second, i.e. abrupt change in proportional control action and, consequently, oscillations in the model. If each segment had its own control loop, with constant  $k_p=25$  and  $k_d=0.5$  for the arm segment and  $k_p=2.5$  and  $k_d=0.5$  for the forearm, then Figure 7 shows how these values would be implemented. The paths so obtained very nearly followed the desired paths. Figure 8 shows arm perform-

Once a controller had been defined, it was easier to control the system using other trajectories. Any position within a manipulator workspace could be parameterised by a polynomial for differential flatness system theory. For example, if the arm were to make a movement oscillating from a cosine function (18) the forearm would be lifted to a  $90^\circ$  position following the same method for the previous path's behaviour.

$$z_1(t) = \cos(t) \tag{18}$$

The set of inputs described in Figure 9 was obtained by applying function (18) and its derivatives (11) and (12).

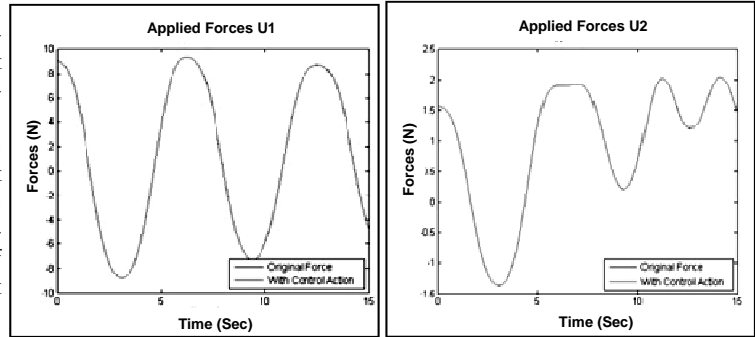
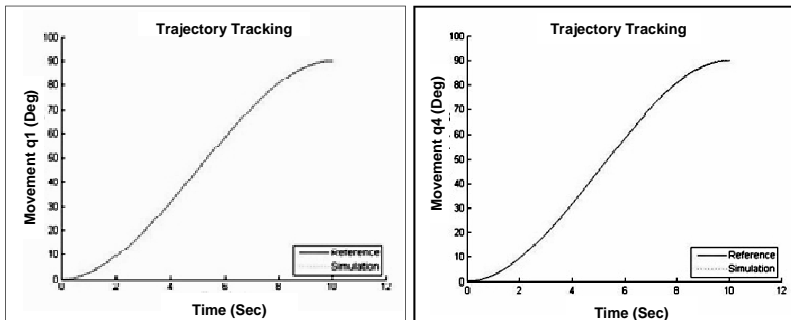


Figure 9. Force trajectories obtained by diffeomorphism

Figure 10 and Figure 11 show the input trajectories implemented in the system. The model's behaviour affected the magnitude of the forces applied, proving the oscillatory form shown in Figure 9. The controller was responsible for ensuring that the system moved through the desired trajectory and stability in movement without influencing the result generated by diffeomorphism.



ance according to input.

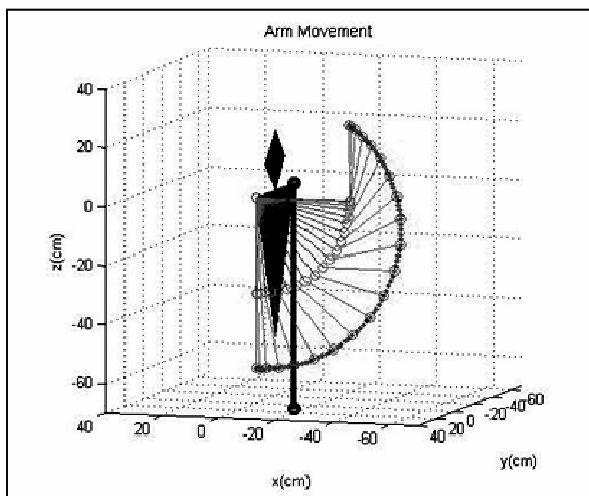


Figure 7. Trajectory tracking using a controller

Figure 8. Model behaviour using input defined by diffeomorphism

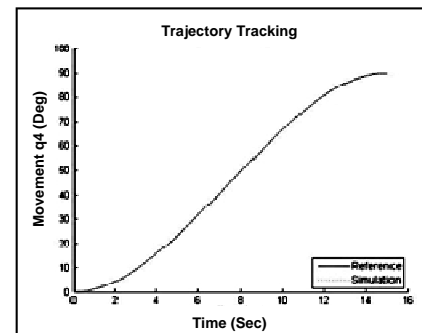
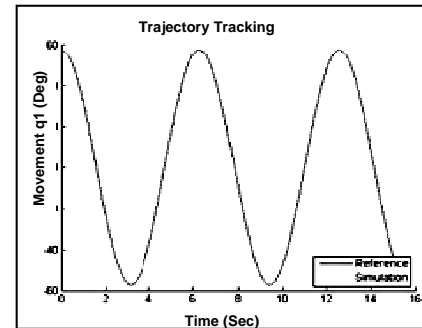


Figure 10. System behaviour using the desired trajectories

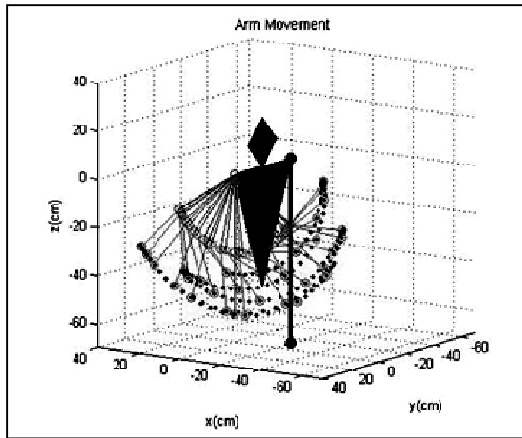


Figure 11. Model behaviour using diffeomorphism-defined input

**Space control**

The concept was also applicable to systems having more degrees of freedom and, although equations of motion are more complex, paths and behaviour could be tracked. Figure 12 presents the same system with four degrees of freedom (generalised coordinates  $q_1, q_2, q_3$  and  $q_4$ ). The equations of motion took rotations on the Z-plane and shoulder X into account and determined equations of motion.

The implemented trajectories were cube shaped using the controller mentioned in the previous paragraph to ensure follow-up, thus reducing path system error.

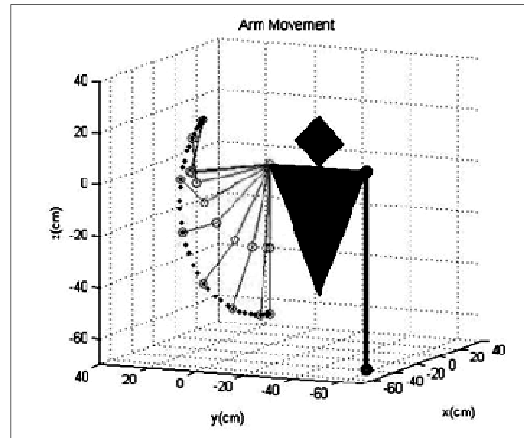


Figure 13. Simulation model behaviour

The ability to follow-up trajectories using differential flatness systems for robotics manipulators depends on the quality of input values. It was found that small changes in these settings offered better performance, proving that input resolution is an important part of the results. Models were analysed in 0.01 second increments in previously carried out tests. However, testing with greater increases did not provide appropriate behaviour (see Figure 14). This hampered determining the controller's constant values or, even worse, prevented it. This restriction on using short times made it difficult to calculate input due to high consumption of computing resources; a strategy for resolving this situation was thus reconfigured.

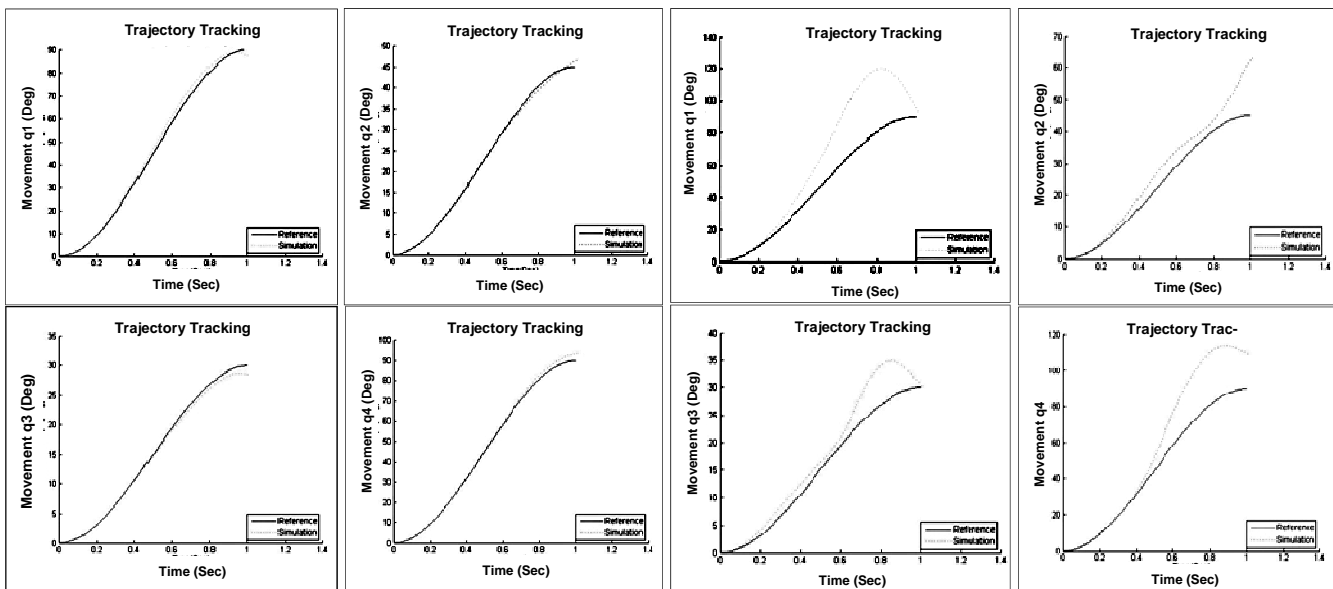


Figure 12. Behaviour of the system for each trajectory. The dashed line describes actual movement while the continuous line describes the desired trajectory.

Implementation presented difficulties. Input equations regarding four outputs were extensive and prevented an acceptable result being achieved for extended times. This error forced the use of a controller. The open-loop system presented difficulties during follow-up. The controller made modifications imperceptible regarding input but enough so that there was a stable trajectory.

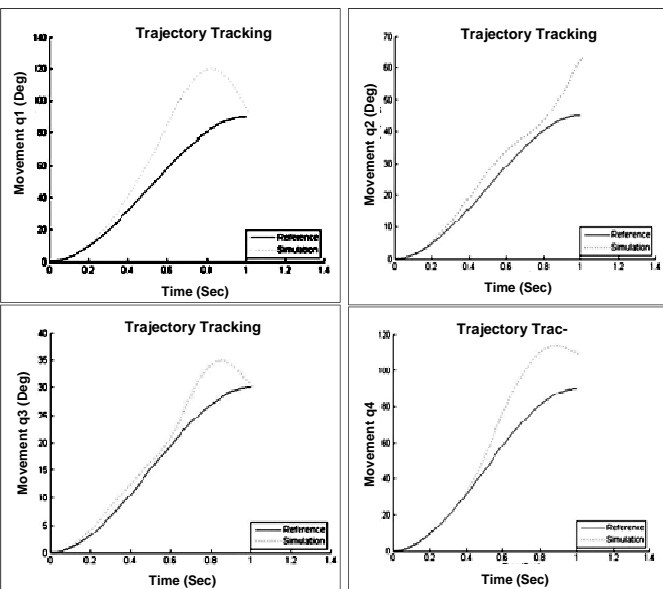


Figure 14. Behaviour for a model having longer time period (showing difficulty in following-up desired output)

**Conclusions**

This article has examined the application of differential flatness system theories to robotic manipulators' trajectories. The model's generalised coordinates concerned system flatness output; as flatness output can establish a relationship with output and obtain an input set implemented in the manipulator, this led to trajectory planning.

The study showed that the system could execute any parameterised trajectory regarding all the system's generalised coordinates for certain conditions. A PD controller had to be included which influenced the system's behaviour, thereby ensuring necessary control for following a route with minimum deviation.

However, for complex systems the extension of motion equations generated an increase in the consumption of computational resources, preventing an implementation for a long time. Nevertheless, the analysis showed that the system had the necessary input and using the controller enabled following up the desired trajectory. Future work should be aimed at finding a new solution to apply differential flatness to complex robotics manipulators, such as the application of another controller.

## Acknowledgments

This work was partially financed by CNPq and CAPPES; additional support was provided by COPPE/UFRJ.

## References

- Achir, A. Sueur, C., Dauphin-Tanguy, G., Bond graph and flatness-based control of a salient permanent magnetic synchronous motor., *Proceedings IMenche*, Vol. 219, Part I: J. Systems and Control Engineering, 2005, pp. 461-476.
- Aguilar-Ibañez, C., Suárez-Castañón, M., Sira-Ramírez H., Control of the Chua's System based on a Differential Flatness Approach., *International Journal of Bifurcation and Chaos*, Vol. 14, No. 3, 2004, pp. 1059-1069.
- Barczyk, M., Lynch, A. F., Flatness-based estimated state feedback control for a rotating flexible beam: experimental results., *IET Control Theory and Applications*, Vol. 2, No.4, 2008, pp. 288-302
- Carvajal, J. H., *Modelamiento y diseño de Robots Industriales*, Ediciones Unisalle, Bogotá-Colombia, 2007.
- Defoort, M., Floquet, T., Kokosy, A., Perruqueth, W., Integral sliding mode control for trajectory tracking of an unicycle type mobile robot., *Integrated Computer-Aided Engineering*, Vol. 13, 2006, pp.277-288.
- Deligiannis V., Davrazos, G., Manesis, S., Arampatzis, T., Flatness Conservation in the N-Trailer System Equipped with a Sliding Kingpin Mechanism., *Journal of Intelligent and Robotic System*, Vol. 46 No. 2, 2006, pp. 151-162.
- Fliess M., Lévine, J., Martin, P., Rouchon, P., Flatness and Defect of Non Linear Systems: Introductory Theory and Examples., CAS internal report A-284, January ,1994.
- Franch, J., Agrawal, S.K., Design of differentially Flat Planar Space Robots : A Stepforward in their Planning and Control, *Proceedings of the IEEE/RSJ International Conference on Intelligente Robots and Systems*, 2003, pp. 3053-3058.
- Lévine, J., *Analysis and Control of Non Linear Systems: A Flatness Based Approach.*, Springer-Verlag Berlin Heilderberg, 2009, pp.131-145.
- Marguitu, D. B., *Mechanics and Robot Analysis with MATLAB.*, Springer-Verlag London Limited, 2009, pp. 183-204.
- Martin, P., Murray, R., Rouchon, P., Flat Systems., Mini-course European Control Conference, Brussels, July 1997.
- Murray, R., Rathinam, M., Sluis, W., *Differential Flatness of Mechanical Control Systems: A Catalog of Prototype Systems*, ASME International Mechanical Engineering Congress and Expo., San Francisco, California, USA. November 12-17, 1995a, pp. 1-9.
- Murray, R., *Non Linear Control of Mechanical Systems A Lagrangian Perspective.*, In IFAC Symposium on Nonlinear Control Systems Design (NOLCOS), 1995b.
- Rathinam, M., *Differentially flat nonlinear Control Systems*.Ph.D dissertation., University of California at Berkeley, Berkeley, California, USA, 1997.
- Rotella F., Zambettakis, I., *Comande des Systèmes par Platitude* Available: <http://www.techniquesingenieur.fr/book/s7450/comande-des-systemes-par-platitude.html>, Site Access: August 2008, pp. 5-6.
- Siciliano, B.,Sciavicco, L., Villani, L.,Oriolo,G., *Robotics, Modeling, Planing and Control.*, Springer-Verlag London Limited, 2009, pp. 161-189.
- VanNieuwstadt, M.J., *Trajectory Generation for Nonlinear Control Systems.*, Technical Report CS 96-011 del Department of Mechanical Engineering, California Institute of Technology, Pasadena, California, 1997a, pp. 54-57.
- Spong, M.W., *On the Robust Control of Robot Manipulators.*, *Transactions on Automatic Control of Robot Manipulators*, Vol. 37 No. 11, 1992, pp. 1782-1786.
- VanNieuwstadt, M.J., *Trajectory Generation of Non Linear Control Systems.*, Ph. D. dissertation, University of California at Berkeley, Berkeley, California, USA, 1997b.
- Veslin E., Slama, J., Dutra, M., Lengerke, O., *Análisis Cinemático de un Exoesqueleto de Partes Superiores de 7 GDL.*, *Proceedings of II Congreso Internacional de Ingeniería Mecatrónica*, Bucaramanga, Colombia, Sept. 30, Oct.1-2, 2009.
- Veslin E., *Implementação dos Sistemas diferencialmente planos para o controle de um manipulador robótico tipo braço humano.*, M.Sc. Dissertation, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, Brazil, 2010.
- Wang H., Xie, Y., *Adaptative Inverse Dynamics Control of Robot with Uncertain Kinematics and Dynamics.*, *Automatica* 45, 2009, pp. 2114-2119.