DYNAMIC MODELLING OF HUMANOID ROBOTS USING SPATIAL ALGEBRA

M. González-Fierro, D. Hernández García, P. Pierro and C. Balaguer
mgpalaci@ing.uc3m.es,dhgarcia@ing.uc3m.es,ppierro@ing.uc3m.es,balaguer@ing.uc3m.es
Robotics Lab
Department of Systems and Automation Engineering at Universidad Carlos III de Madrid
Av. de la Universidad 30
28911, Leganés, Madrid, Spain

Abstract

Dynamics is a fundamental part for a humanoid robot, not only for locomotion, but also for any kind of movement. It relates the forces and accelerations that appear in the system. It is used in trajectory generation, stability control, mechanical design and simulation.
In this work, a complete dynamic model of the HOAP-3 robot has been obtained, using the spatial notation of inverse and forward dynamics algorithms. This provides a more compact formulation for velocity, acceleration and forces and simplify the equations.
Different proofs have been achieved using this approach. We have generated a set of movement trajectories for the humanoid robot HOAP-3 and then we have applied spatial dynamic formulation to them. First, we have designed a dance trajectory imitating a human dancer, involving only the upper part of the robot body, afterwards, we have created a locomotion trajectory using the cart table, which involves the complete robot body. Finally, results have been validated and discussed.

Keywords: Humanoid robots, dynamics, spatial algebra, postural planning

1 INTRODUCTION

Over the last years, research in humanoid robots has evolved in many directions. Advanced humanoids have been developed with a high degree of interaction abilities and mobility. Some examples are HRP-2 of AIST [1], ASIMO of Honda [2] or RH-1 of Universidad Carlos III de Madrid [3]. But if these robots have to interact with humans and share his environment, it is necessary to rise its safety, stability, mobility, manipulability and interaction ability.

The study of forces and torques produced in the robot is of main importance if we want to achieve these objectives. The problem of the humanoid robot dynamics is an interest topic since many decades, and it plays an important role in the robot movement control.

Dynamics is defined by the equation of motion, which relates joint torques with joint accelerations. Through forward dynamics, it is possible to obtain the joint accelerations knowing the joint torques and it is mainly used in simulation. Inverse dynamics performs the opposite task, it allows to obtain the joint torques knowing the joint accelerations. It is commonly used in trajectory generation, stability control and mechanical design of the robot.

There have been many works which make use of the complete dynamic model of the robot. In [4] the net force produced by all robot link is calculated to perform biped walking. In [5], whole body dynamics is taken into account to generate motion in simulated human figures. Finally, in [6], a complete dynamic model of the RH-1 humanoid robot is obtained using euclidean groups.

This work deals with the need of obtaining a complete dynamic model of the humanoid HOAP-3. Starting from this model, which allows to know all the torques and forces produced in every joint and link of the robot, it is possible to develop control algorithms and test different trajectories. Furthermore, the complete dynamic model constitute a useful tool to validate these algorithms.

To obtain the dynamics of the robot, we have used the spatial notation of $6D$ vector of the dynamic algorithms, developed in, which differs from the traditional $3D$ notation in a simpler way to formulate the algorithms and in a conciser way to express velocity, acceleration, inertia, and forces.

The document is structured as follows, the next section provides a explanation of the Lagrangian and Newton-Euler approaches of dynamics. In section III we discuss the spatial notation of 6D vectors and address the most used algorithms for inverse and forward dynamics. Section IV presents the generated trajectories for testing the algorithms. Section V describes the dynamic model approach. Section VI presents our results and finally, we discuss our conclusions in section VII.

2 FRAMEWORK OF DYNAMICS

The dynamic model of a robot is used to know the relationship between the robot motion and the forces involved in this motion [7]. Obtaining the dynamic model of the robot is essential to achieve different purposes as movement simulation, design of structure and actuators or movement control.

Nowadays, there are two approaches to deal with this problem [8]. The first approach assumes that the complete model of the robot is known, including all the masses and inertias of each link. Due to the high cost to compute all this, trajectories are usually planned off-line [9]. In this paper we foolow this approach.

The second method uses a limited knowledge of the system dynamics, representing the robot as an simple or multiple inverted pendulum [10]. It relies on feedback control to correct the errors produced in the model simplification.

2.1 Equation of motion

The movement of the robot is defined by its equation of motion.

$$D(\mathbf{q})\ddot{\mathbf{q}} + H(\mathbf{q}, \dot{\mathbf{q}}) + C(\mathbf{q}) = \tau \tag{1}$$

where $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are the joint position, velocity and acceleration vectors and τ are the joint torques. D is the inertia matrix, C is the coriolis and centripetal matrix and C is the gravity matrix. To be more precise, this matrices not only depend of q and $\dot{q}$, but also of the model, so it would be more correct to write these matrices as $D(model, \mathbf{q})$, $H(model, \mathbf{q}, \dot{\mathbf{q}})$ and $C(model, \mathbf{q})$ where *model* refers to the rigid body system including the number of bodies and joints, the kind of joint, masses, inertias and the way they are connected.

2.2 Lagrangian and Newton-Euler approach

Lagrangian approach solves the dynamics of a rigid body system tacking into account the energy of the system. This formulation allows to derive the equation of motion in a systematic way, independently of the reference frame. It uses the Lagrangian L, which is the difference between the kinetic and potential energy. The equation of motion using the Lagrange theory is enunciated as:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} = \tau \tag{2}$$

The algorithms based on this approach are slower than those based on Newton-Euler Formulation. They have the advantage that they only need to compute the kinetic and potential energy, so they reduce the number of equations to derive and are less prone to errors.

Newton-Euler formulation is based on the balance of all the forces acting on the robot links. This implies that the equations can be expressed in a recursive way, which produces a big advantage, the algorithms based on this formulation are faster than non recursive ones. Newton-Euler method are described by two equations, the first one is related to the translational movement of the center of mass.

$$f_i - f_{i+1} = m_i \ddot{r}_{CM} - m_i g \tag{3}$$

where f is the force passing through the link, $\ddot{r}_{CM}$ is the center of mass acceleration, m is the link mass and g is the gravity acceleration. The second equation is based on the rotative movement of the link.

$$T_i - T_{i+1} = I_i \alpha_i + \omega_i \times (I_i \omega_i) \tag{4}$$

where T is the torque produced by the link, I is the inertia tensor of the link, α is the angular acceleration and ω the angular velocity.

3 SPATIAL FORMULATION OF DYNAMIC ALGORITHMS

Building the dynamic model of a high degree of freedom robot can be tedious. If we are working with a humanoid robot, the problem is more difficult due to the numerous joints and the closed kinematic chains. Spatial formulation of dynamics provides a compact and easy to implement notation. This formulation make use of $6D$ vector and tensors to describe velocity, acceleration, inertia and force. Using these components, a set of dynamic algorithms can be developed.

3.1 Spatial equation of motion

The equation of motion of a rigid body system (see Fig.1) is defined using the spatial notation as:

$$\mathbf{f} = \frac{d}{dt}(\mathbf{Iv}) = \mathbf{Ia} + \mathbf{v} \times \mathbf{Iv} \tag{5}$$

with

$$\mathbf{f} = \begin{pmatrix} n \\ f \end{pmatrix} \in F^6 \tag{6}$$

$$\mathbf{v} = \begin{pmatrix} \omega \\ v \end{pmatrix} \in M^6 \tag{7}$$

$$\mathbf{a} = \begin{pmatrix} \dot{\omega} \\ \ddot{c} - v \times \omega \end{pmatrix} \in M^6 \tag{8}$$

$$\mathbf{I} = \begin{pmatrix} Ic & 0 \\ 0 & m \end{pmatrix} \in M^{6\times 6} \tag{9}$$

where $\mathbf{f} \in F^6$ is the net spatial force applied in the body, which is compose by $3D$ vectors force f and torque n, $\mathbf{v}$ is the spatial velocity, composed by the linear and angular velocity of the body center of mass, $\mathbf{a}$ is the spatial acceleration and $\mathbf{I}$ is the spatial inertial, composed by the inertia tensor I_c and the mass m.

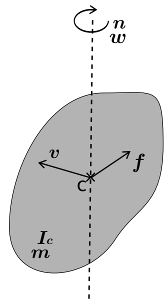

Figura 1: Forces and velocities acting on a rigid body

3.2 Spatial inverse and forward dynamics

Inverse dynamics deals with the problem of obtaining the torques applied in every joint starting from the acceleration of the rigid body system. The generic formula can be expressed as:

$$\tau = ID(model, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$$

The most used algorithm to calculate inverse dynamic is the Recursive Newton Euler Algorithm (RNEA) [11], whose spatial formulation can be found in [12]. This algorithm has a complexity of $O(n)$, where n is the number of degrees of freedom.

RNEA has two phases. First, it calculates recursively the velocity and acceleration of every joint, and then, using (5), it calculates the force transmitted in every joint. In a second stage, it computes the joint forces starting at the terminal links and working towards the base.

Forward dynamics consists on determining the accelerations that appears in the actuated joints as a function of the torques applied. The general formulation can be expressed as:

$$\ddot{\mathbf{q}} = FD(model, \mathbf{q}, \dot{\mathbf{q}}, \tau)$$

One of the most cited algorithm for forward dynamics is the Composite Rigid Body algorithm (CRBA), first developed in [13]. This algorithm computes the inertia matrix of a set of composite rigid bodies and then solve for every joint acceleration. This matrix can be computed efficiently by applying successively inverse dynamics with joint velocity and acceleration set to zero and depends on the connectivity of the kinematic chain.

Another approach for solving the forward dynamic is the Articulated Body Algorithm (ABA), developed in [14]. It is based on the propagation of the equations of an articulated body. Forward dynamic problem presents two set of unknowns, joint accelerations and joint forces. ABA calculates the coefficients of this equation locally, taking into account one joint at every step. It calculates the acceleration that appears in a joint formed by two bodies, one is the parent body, the second one is an articulated body formed by all the other links of the kinematic chain. It computes this equation recursively, until it finds a local solution (usually at the terminal link) and propagates backwards to obtain a global solution.

Both CRBA and ABA are algorithms to compute forward dynamics. Generally speaking, CRBA is faster than ABA, but ABA is more precise [15]. Also, they have a spatial formulation than can be found in [12].

4 TRAJECTORY GENERATION

In this paper we study two group of trajectories that have been generated for the humanoid HOAP-3. The first one is a dance performance imitation of a professional dancer. The second one is a locomotion trajectory generated using the cart-table method.

4.1 Dance performance trajectory

There are many works regarding dance performance imitation in humanoid robots. In [16] a method to scale human upper body motion capture data to a humanoid robot is proposed. Other example of upper body motion imitation can be found in [17], where the dance performance speed is taken into accounet to control the robot.

Another approach is to accomplish whole body robot movement through imitating a human. Usually, the problem is divided in two. The upper part of the body, which can be imitated without taking into account the stability of the robot and the lower part, where the ZMP criteria is used. This study [18] is an example of this.

We have obtain a set of trajectories for the upper body of the humanoid robot, imitating the dance performance of a professional dancer. The complete dance performance can be seen in this video[1].

To simplify the dance adaptation to the robot, we have constructed the set of motion primitives, which have been performed by the dancer and im-

[1] http://www.youtube.com/watch?v=mu5psxG7bwA

itated by the robot. The complete dance routine consist of 12 different motion primitives, which combines arms and legs movement. In this paper we only study the motion of the upper body.

First, a tracking vision system to capture the movement of the dancer have been developed. Using a set of 3 tags placed at the shoulder, elbow and hand, the movement of the dancer arms have been tracked. To obtain the $3D$ trajectories of the tags, we have used segmentation by histogram and a Kalman filter. The noisy trajectory obtained is smoothed using a third order spline.

The trajectory performed by the dancer has to be adapted to the robot (see Fig.2). To do so, we have obtained the kinematic model of the human arm as a 4 degree of freedom manipulator, with 3 degrees of freedom (yaw, pitch,roll) in the shoulder and 1 degree of freedom in the elbow (yaw). This is the same model as the HOAP arm.

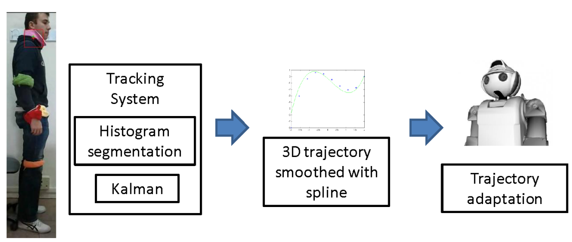

Figura 2: Vision tracking system and obtaintion of an adapted 3D trajectory for the humanoid robot.

In such way, it is possible to use Inverse Kinematics algorithms in order to get the joint angles of the robot arm. In Fig. 3 the used algorithm is presented.

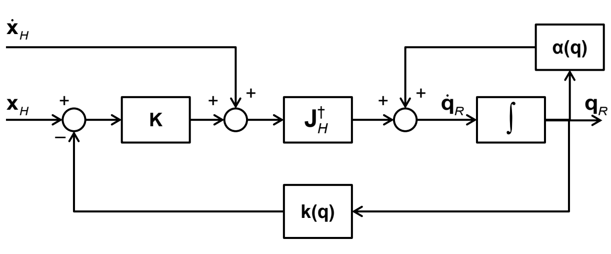

Figura 3: Inverse kinematics for the human arm

The reference position and velocities of the human arm are used as input. The human arm angle velocities can be calculated using the well-known equation

$$\dot{\mathbf{q}}_H = \mathbf{J}_H^{\dagger}\left[\dot{\mathbf{x}}_H + \mathbf{K}\left(\mathbf{x}_H - k(\mathbf{q})\right)\right] \qquad (10)$$

where the pseudo-inverse of the Jacobian Matrix is used since only the position of the arm is considered. The remaining degrees of freedom can be used in order to adapt the different range of movements of the HOAP-3 robot with respect to the human arm. So, the velocity of robot arms are calculated as:

$$\dot{\mathbf{q}}_R = \mathbf{J}_H^{\dagger}\left[\dot{\mathbf{x}}_H + \mathbf{K}\left(\mathbf{x}_H - k(\mathbf{q})\right)\right] + \alpha(\mathbf{q}_R) \qquad (11)$$

where

$$\alpha(\mathbf{q}_R) = \left[\mathbf{I} - \mathbf{J}_H^{\dagger}\mathbf{J}_H\right]\dot{\mathbf{q}}_0 \qquad (12)$$

The vector $\dot{\mathbf{q}}_0$ can be calculated in order to get a solution of joint angles being far from the HOAP-3 joints limits, while getting the same end-effector trajectory:

$$\dot{\mathbf{q}}_{0,i} = -k_l \frac{\mathbf{q}_{H,i} - \overline{\mathbf{q}}_{R,i}}{(\mathbf{q}_{R,i,M} - \mathbf{q}_{R,i,m})^2} \qquad (13)$$

with $k_l > 0$.

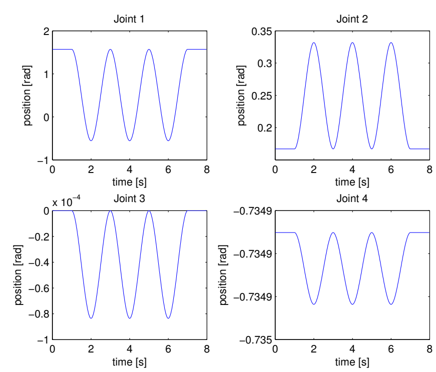

Figura 4: Articular dancing trajectory of the 4 DoF left arm

In Fig.4 the articular trajectory of the left arm is showed. In addition, in Fig.5 some snapshots of the dancing performance imitation is showed.

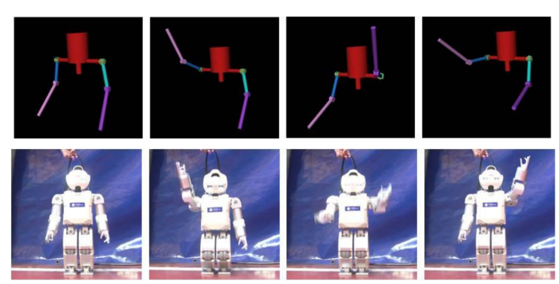

Figura 5: Upper body dance performance imitation (simulated and real)

4.2 Locomotion trajectory

The other trajectory studied is a stable locomotion trajectory. We have used the cart-table model

[19] to generate the gait. This model is based on ZMP a preview control scheme to obtain the COG trajectory from a defined ZMP trajectory. This method generates a dynamically stable gait trajectory using the 3D Linear Inverted Pendulum Model [20] to approximate the dynamics of the humanoid.

The relationship between ZMP trajectory and COG trajectory is defined by the following equations:

$$p_x = x - \frac{\ddot{x}}{g} z_c \tag{14}$$

$$p_y = y - \frac{\ddot{y}}{g} z_c \tag{15}$$

where p_x is the ZMP reference, x is the COG trajectory, $\ddot{x}$ the COG acceleration, z_c is the COG height and g is the gravity. In cart table model

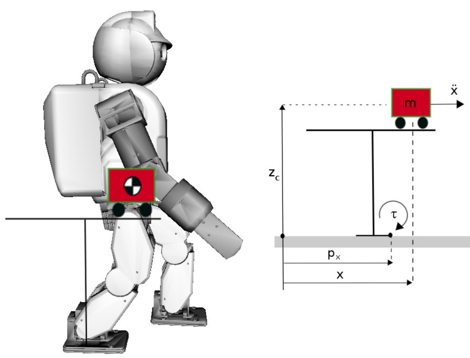

Figura 6: Cart table model in sagital plane

(Fig. 6), the cart mass corresponds to the center of mass of the robot. If the cart accelerates at a proper rate, the table can be upright for a while. At this moment, the moment around p_x is equal to zero, so the ZMP exists.

$$\tau_{ZMP} = mg(x - p_x) - m\ddot{x}z_c = 0 \tag{16}$$

The trajectories have been calculated with a step distance of 8 cm, COG height of 32 cm and a preview time of 0.75 sec. The robot walks 12 steps forward. Some snapshot of the biped locomotion are showed in Fig. 7.

5 DYNAMIC MODEL APPROACH

There are many works regarding simplified dynamic models, which make use of the pendulum, simple or multiple, to control humanoid robots. It have been proved, that in some cases as walking, the error between computing the simple pendulum dynamics and the complete model dynamics is small [19]. However, there are some cases, as for example dancing, where the complete model is necessary. In these cases, the inertia forces produced by the arms or by the floating leg are not neglectable, and may have an important influence in robot stability.

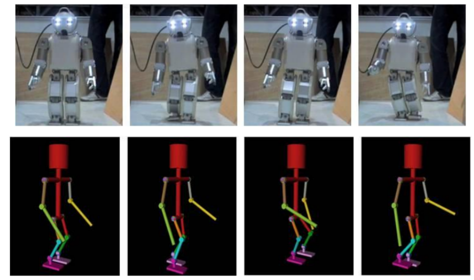

Figura 7: Biped locomotion trajectory (real and simulated)

In order to apply the spatial dynamic algorithms previously presented, we have developed two dynamic models to study the dance and locomotion trajectories. The model used to obtain the dance performance dynamics is a kinematic tree, with the base frame in the robot chest. It has two branches and 8 degrees of freedom, 4 for every arm.

The dynamic model used in the locomotion trajectory is an open branched kinematic tree, with the base frame located in the supporting leg. It has 23 degrees of freedom, corresponding to all the actuated joints of the HOAP-3 robot. It starts in the feet, with the supporting leg kinematic tree until the waist, where it branches in 3 kinematic trees, corresponding to the other leg an the two arms.

We have used a dynamic model of the robot taking into account only the single support phase, where only one leg is in contact with the floor at the same time. The reason to do this is because the single support phase is more critical that the double support phase. In single support, the strongest instabilities appear and the torques are bigger. It also allows an easier formulation of the dynamic algorithms, avoiding kinematic closed loops and contact forces.

To compute the robot dynamics in the locomotion trajectory, we have used a complete dynamic model for the right support phase. In the left support phase, we have used a model symmetrically equal to the first one, but with the fixed base in the left foot.

Robot links lengths, masses and inertia tensors have been provided by the manufacturer.

6 RESULTS

We have computed the inverse and forward dynamics for the dance (Fig.5) and locomotion trajectories (Fig. 7), usign the spatial formulation of the dynamic algorithms.

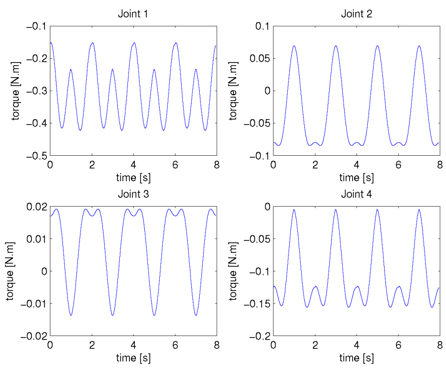

Figura 8: Joint torques for the right arm in the dance trajectory obtained usign RNEA.

We have obtained the arm joint torques for the first trajectory using the RNEA algorithm of inverse dynamic. In Fig. 8 the torques of only one arm is showed. The complete dance performance last about 3 minutes, we only use a small part of it to test our algorithms.

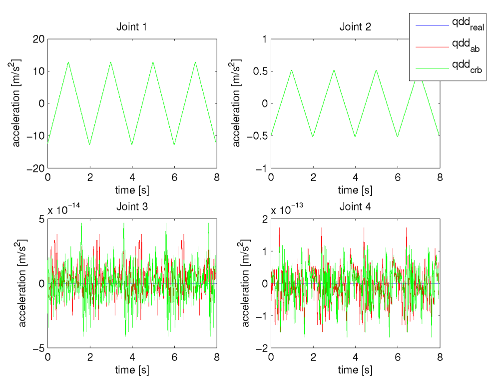

Figura 9: Joint accelerations for the right arm in the dance trajectory. Desired acceleration is showd in blue, acceleration obtained by CRBA is showed in green and acceleration obtained by ABA is showed in red.

Using the previous torques as an input, we have calculated the forward dynamics using the CRBA and ABA algorithms (Fig. 9), where the acceleration obtained by CRBA is printed in green, ABA in red and the real acceleration is printed in blue.

The real acceleration is the desired acceleration, which means that it is the second derivative of the original tracked motion. It can be seen that the dynamic model is correct, as the error between the solutions of the CRBA and ABA algorithm and the real acceleration is very small.

In Fig. 10 are showed the joint torques produced in the floating and supporting leg when HOAP-3 humanoid is walking forward. The motion used here consist on a left step forward, being the right leg the supporting leg and being the left leg the floating leg. As it can be seen, the joints supporting all the humanoid weight are the ones with the higher torque value, as well, the torques in the floating leg are lower than the others.

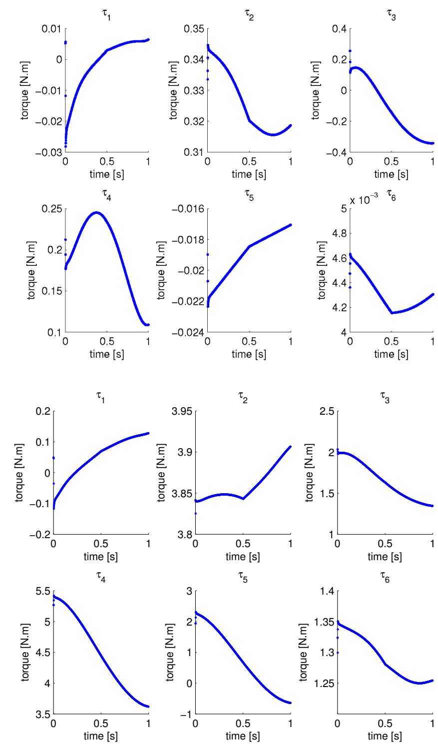

Figura 10: Joint torques of the floating leg (up) and supporting leg (down) in the locomotion trajectory.

The dynamic algorithms have been programed in MATLAB, usign a variation a the dynamic library created by Featherstone. All these algorithms are pretty fast and have been computed in a Daul Core 5200+ with 2 Gb of RAM. The computing time for a 23 degree of freedom model using RNEA has been $3,31 \cdot 10^{-3}s$, CRBA has been $3,76 \cdot 10^{-3}s$ and ABA $5,13 \cdot 10^{-3}s$.

They are good results tacking into account that the usual sampling time of the HOAP-3 robot is $3ms$.

7 CONCLUSIONS

From our point of view, the complete dynamic model of a humanoid robot provides some important advantages. We can obtain a accurate knowledge of the forces and torques which appears in every link. We also can predict more precisely the behavior of the robot, under complex trajectories or uncertain environments. Concentrated models as inverted pendulum simplified the equations with the inconvenient of loosing precision. In our approach, we put together the spatial notation of dynamics, which make use of a quite easy formulation and the precision of knowing the complete dynamic model.

In this paper we have applied the spatial formulation of dynamic algorithms to a set of different trajectories. The first one is a dance performance, initially executed by a professional dancer. Using a tracking vision system, we have obtained the $3D$ movement of the upper body, which have been adapted to the HOAP-3 humanoid robot, taking into account its joint limits.

The second trajectory is a stable walking gait obtained by the cart-table method. In this case, we have applied the dynamic algorithms to the single support phase, which is the less stable phase and where the higher torques are produced. We have studied the joint torques which appears in both the supporting and floating leg.

We have also created two dynamic models of the robot in order to test the algorithms and trajectories. We have made an upper body model to deal with the dance trajectory and a complete body model to study the locomotion trajectory.

Using these models, we have been able to compute the inverse dynamics, using the spatial version of the RNEA, and the forward dynamics, using the spatial versions of CRBA and ABA, proving that the the algorithms are consistent.

Finally, we have achieved a complete dynamic model of the humanoid HOAP-3. With this model, we can test different control algorithms which includes all joint masses and inertias and also, we can know the forces acting in every link.

In future works, we will try to add constraint forces to the model and will try to decrease the computing time to obtain the dynamics in real time.

8 Acknowledgements

The research leading to these results has received funding from the COMANDER project CCG10-UC3M/DPI-5350 funded by Comunidad de Madrid and UC3M (University Carlos III of Madrid), and ARCADIA project DPI2010-21047-C02-01 funded by CICYT project grant on behalf of Spanish Ministry of Economy and Competitiveness.

The authors gratefully acknowledge the help of Eva de Frutos, who performs the dance routine and help us with the robot. We also want to thank the Rh-1 and Rh-2 humanoid robot team, our students and the Robotics Lab members for their support during our research.

References

[1] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi, “Humanoid robot hrp-2.” Proceedings of IEEE International Conference on Robotics and Automation, 2004, pp. 1083–1090.

[2] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, “The intelligent asimo: System overview and integration.” IEEE/RSJ international conference on intelligent robots and systems, 2002, pp. 2478–2483.

[3] M. Arbulú, D. Kaynov, L. M. Cabas, and C. Balaguer, “The rh-1 full-size humanoid robot: design, walking pattern generation and control,” *Journal of Applied Bionics and Biomechanics.*, vol. 6, no. 3, pp. 301–344, 2009.

[4] J. Yamaguchi, E. Soga, S. Inoue, and A. Takanishi, “Development of a bipedal humanoid robot-control method of whole body cooperative dynamic biped walking,” in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 1. IEEE, 1999, pp. 368–374.

[5] K. Yamane and Y. Nakamura, “Dynamics filter-concept and implementation of online motion generator for human figures,” in *Robotics and Automation, 2000. Proceedings. ICRA’00. IEEE International Conference on*, vol. 1. IEEE, 2000, pp. 688–694.

[6] *Aiming for multibody dynamics on stable humanoid motion with Special Euclideans groups.* IEEE/RSJ international conference on intelligent robots and systems, 2010.

[7] A. Barrientos, L. F. Peñín, C. Balaguer, and R. Aracil, *Fundamentos de Robótica*, 2nd ed. Mc Graw-Hill, 2007.

[8] D. Katic and M. Vukobratovic, “Survey of intelligent control techniques for humanoid robots,” *Jounal of Intelligent and Robotics Systems*, no. 37, pp. 117–141, 2003.

[9] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of the honda humanoid robot." Proceedings of IEEE International Conference on Robotics and Automation, 1998, pp. 1321–1326.

[10] M. Arbulú and C. Balaguer, "Real-time gait planning for rh-1 humanoid robot using local axis gait algorithm," *International Journal of Humanoid Robotics*, vol. 6, no. 1, pp. 71–91, 2009.

[11] J. Luh, M. Walker, and R. Paul, "On-line computational scheme for mechanical manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 102, p. 69, 1980.

[12] R. Featherstone, *Rigid Body Dynamics Algorithms*. Springer, 2008.

[13] M. Walker and D. Orin, "Efficient dynamic computer simulation of robotic mechanisms," *Journal of Dynamic Systems, Measurement, and Control*, vol. 104, p. 205, 1982.

[14] R. Featherstone, "The calculation of robot dynamics using articulated-body inertias," *The International Journal of Robotics Research*, vol. 2, no. 1, p. 13, 1983.

[15] U. M. Asher, D. Pai, and B. P. Cloutier, "Forward dynamics, elimination methods, and formulation stiffness in robot simulation," *International Journal of Robotic Research*, vol. 6, no. 6, pp. 749–758, 1997.

[16] N. Pollard, J. Hodgins, M. Riley, and C. Atkeson, "Adapting human motion for the control of a humanoid robot," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 2. IEEE, 2002, pp. 1390–1397.

[17] T. Shiratori, S. Kudoh, S. Nakaoka, and K. Ikeuchi, "Temporal scaling of upper body motion for sound feedback system of a dancing humanoid robot," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 3251–3257.

[18] S. Nakaoka, A. Nakazawa, F. Kanehiro, K. Kaneko, M. Morisawa, and K. Ikeuchi, "Task model of lower body motion for a biped humanoid robot to imitate human dances," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*. IEEE, 2005, pp. 3157–3162.

[19] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 2. Ieee, 2003, pp. 1620–1626.

[20] F. Kanehiro, K. Kaneko, K. Yokoi, H. Hirukawa, and S. Kajita, "The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, 2001, pp. 239–246.