Int J Interact Des Manuf
DOI 10.1007/s12008-011-0132-x

ORIGINAL PAPER

Biomechanically-based motion control for a digital human

Giuseppe Di Gironimo · Luigi Pelliccia · Bruno Siciliano · Andrea Tarallo

Received: 18 October 2011 / Accepted: 14 November 2011

Abstract An algorithm for both human-like motion generation and joint torques computation for digital humans is addressed in this paper. This goal has been achieved using techniques derived from robotics. In particular, the so-called augmented Jacobian has been used to solve the inverse kinematics problem with a single closed loop inverse kinematics algorithm. Furthermore, a position control for the center of mass of the kinematic chain, and for its projection on the support plane (center of pressure), has been implemented to achieve easy posture control. Thus, the inverse kinematics can be solved taking into account the static balance of the digital human. Moreover, the proposed algorithm allows simulating quite complex tasks, which involve the motion of the whole body, by means of only few task-related control points arbitrary located on the whole kinematic chain. The resulting movements are quite natural even for complex tasks as can be seen on the simulation experiments reported which show the effectiveness of the proposed approach. Finally, the joint torques can be computed thanks to the kinetostatics duality: the results are in accordance with biomechanical analyses.

Keywords Digital humans · Human-like motion generation · Multiple-point kinematic control · Posture control · Kinetostatic duality · Low-back biomechanical analysis

G. Di Gironimo (✉) · A. Tarallo
IDEA Lab, DiME, University of Naples Federico II,
Piazzale Tecchio 80, 80125 Naples, Italy
e-mail: giuseppe.digironimo@unina.it

L. Pelliccia · B. Siciliano
PRISMA Lab, DIS, University of Naples Federico II,
Via Claudio 21, 80125 Naples, Italy
e-mail: bruno.siciliano@unina.it

1 Introduction

Over the years, the so-called *ergonomics* and *usability* have became key product quality parameters, while an increasing attention has been devoted to the so-called *human-centred design* approach, even from the early stages of the design process [1–4].

In particular, a great deal of research has been done on modelling "digital humans" that could faithfully reproduce, in a digital environment, the movements and the behaviour of the human beings [5,6]. This has given birth to the so-called *digital human modelling* (DHM) which led to the development of many software tools [7,8].

These tools are used for instance to study human–product and human–process interaction as well as to perform ergonomic and biomechanical analyses and even manual processes simulations.

The *digital humans* essentially are kinematic chains consisting of several segments (*links*) connected by joints. The lengths of the segments, as well as their mass distribution, are derived from anthropometric databases, (e.g. [9,10]) which can be queried with respect to different percentiles in the population.

With reference to the "animation" of the digital human, commercially available software tools generally follow a "key-frame" approach that can be very time consuming, because of the difficulty to achieve, in this way, a faithful "behaviour" for the virtual humanoid.

In fact, although they do implement effective inverse kinematics (IK) algorithms, they can generally manage only a limited number of segments at time, that move independently of each other, without any care for balancing or other biomechanics issues. As a result, the accuracy of those simulations is strongly dependent on the biomechanics skills of the operator and his experience.

Published online: 01 December 2011

The humanoid model proposed in this paper is simpler than commercially available ones. It has just 39 degrees of freedom (DoF), but it can achieve good results in terms of faithfulness of the simulation, while it is very easy to control.

The proposed IK algorithm, despite the simplicity of the kinematic structure of the humanoid, allows simulating quite complex tasks, given some physical constraints and some optimality criteria.

2 Kinematic model

The approach presented in this paper is based on the idea of controlling a highly redundant kinematic chain, that is the digital human, by means of a small set of control points on its structure [12–14]. This goal has been achieved using some modelling techniques typical of serial robots [15]. Indeed, once determined an inertial base frame, the Denavit-Hartenberg (DH) method can be used to find the position and orientation the reference frame of any other point on the kinematic chain. In the field of robotics, generally the *end-effector* is the only control point. But, changing the value of some DH parameters in a proper sequence results in obtaining another kinematic equation of another manipulator whose end-effector is located before the real one: that is equivalent to move the control point along the structure. The symbolic form determined for DH representation allows identifying an arbitrary point as a *virtual end-effector* (VEE) related to a smaller manipulator. It is understood that an arbitrary number of such control points can be made in this way.

Referring to the humanoid, different kinematic chains will be considered.

During the execution of certain tasks, the posture taken by humans largely depends on balancing, as well as mechanical issues that are only partially related to the considered task itself. This implies the need for an additional posture control. For this purpose, the goal has been to develop an inverse kinematics algorithm with a limited number of task-related control points, without the need of specifying the DoFs of the chain for posture control. The position of the centre of mass (CoM) of the humanoid is calculated at run-time and it is always kept consistent with the balancing of the mechanical structure. In fact, the time-varying CoM is considered just as an additional control point. It is understood that the control points on the humanoid can be selected depending on the task and the context.

2.1 Hierarchical model of the digital human

Firstly, in order to take advantage of the systematic approaches typical of serial robots, the humanoid has been modelled as the combination of four serial kinematic chains,

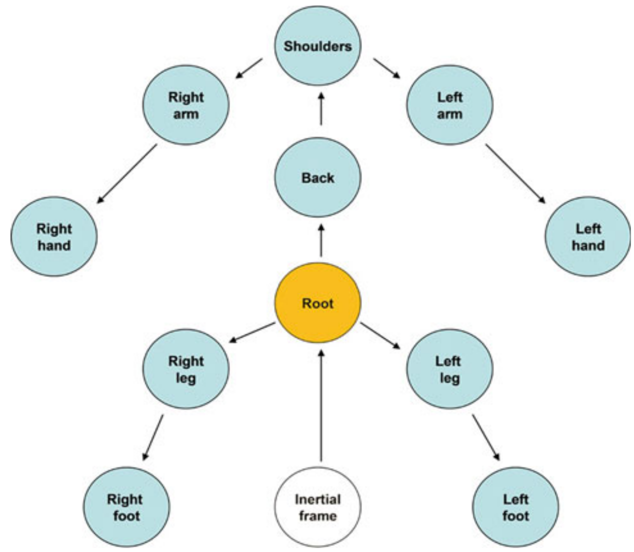


Fig. 1 Hierarchical model of the virtual human

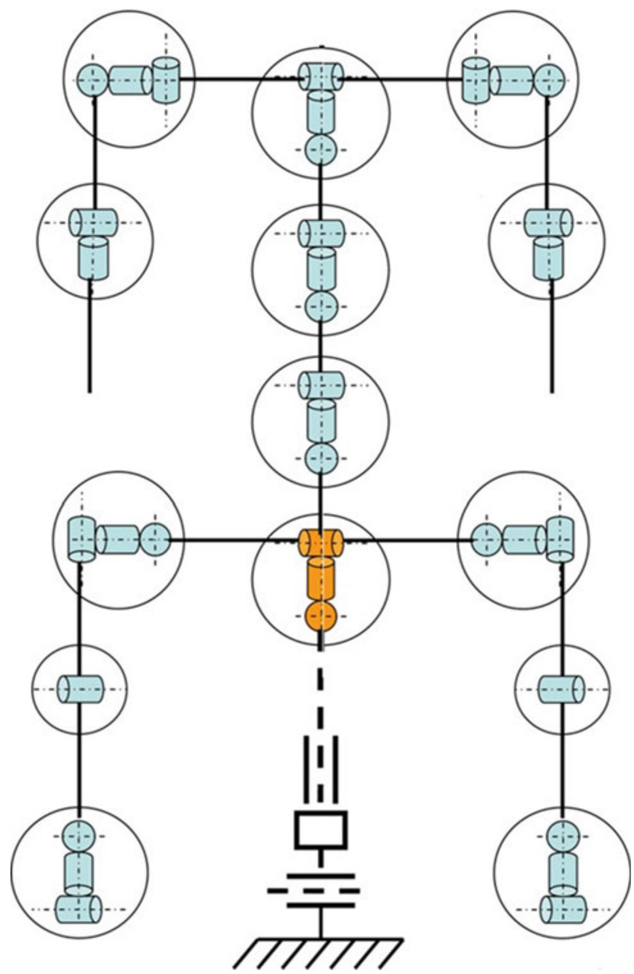


Fig. 2 DH model of the virtual human

which share the same starting point, called *root*. The resulting hierarchical structure is shown in Fig. 1.

Starting from this graph, it is possible to build up the DH model of the whole kinematic chain (Fig. 2) which consists of 33 rotational joints.

In particular, we can define a number of direct kinematic equations, with respect to the *root* reference frame. One or more control points on the provided chains can be selected, by considering the proper set of DH parameters that specify such points. As we can see, the position and orientation of the *root* node with respect to reference frame is specified by introducing 6 *virtual joints* (see Sect. 2.2). Thus, the considered kinematic structure has 39 DoFs in all.

This kind of modelling has the advantage of simplicity, but generally it may cause a physical consistency problem, since some *links* (as well as all the virtual links) are shared among different kinematic chains.

For instance the "back" of the virtual humanoid is shared between its right and left arms. This issue and its solution is discussed in Sect. 2.5.

2.2 Virtual joints

Now, we must describe the position and orientation of the multi-legged kinematic chain with respect to an inertial frame. For industrial robots identifying such a frame is intuitive, because they have a fixed base. A humanoid robot, instead is bound to the ground by a one-way constraint, that is the current support plane, for instance one foot.

But, since this reference frame periodically changes during the walk, we apparently cannot identify a fixed base starting from which the DH method can be applied (Fig. 3).

Moreover, the presence of multiple end-effectors (two hands and two feet) implies the need to describe the position and orientation of many frames, differently from industrial robots, in which the kinematic chain has only one end-effector.

This problem was overcome using the *virtual joints* approach [16]. Namely, when describing the position and orientation of a robot arm, it was conceived connected to the ground plane through a virtual manipulator consisting of three prismatic and three revolute joints, which characterize its position and orientation. The attaching point has been called *root* (Fig. 4). It is worth noticing that, obviously, the actuation torques (as well as the static torques) would be always zero for the virtual joints because "they just do not exist".

With this approach, the hands and the feet (and any other control point) simply are end-effectors that can be controlled with velocity references.

In other words, the posture of the virtual humanoid is completely specified by the following parameters vector:

$$\mathbf{q} = \left[\, \mathbf{q}_r^T \; q_1 \; q_2 \; \ldots \; q_n \,\right]^T \tag{2.1}$$

where $\mathbf{q}_r = [\, \mathbf{p}_r^{0^T} \; \boldsymbol{\varphi}_r^{0^T} \,]^T$ identifies the joints in the *root*.

Moreover, virtual joints technique makes unnecessary the management of closed kinematic chains (e.g. during the phase of double support), since the latter condition becomes merely equivalent, from a kinematic point of view, to impose a null velocity reference to the feet.

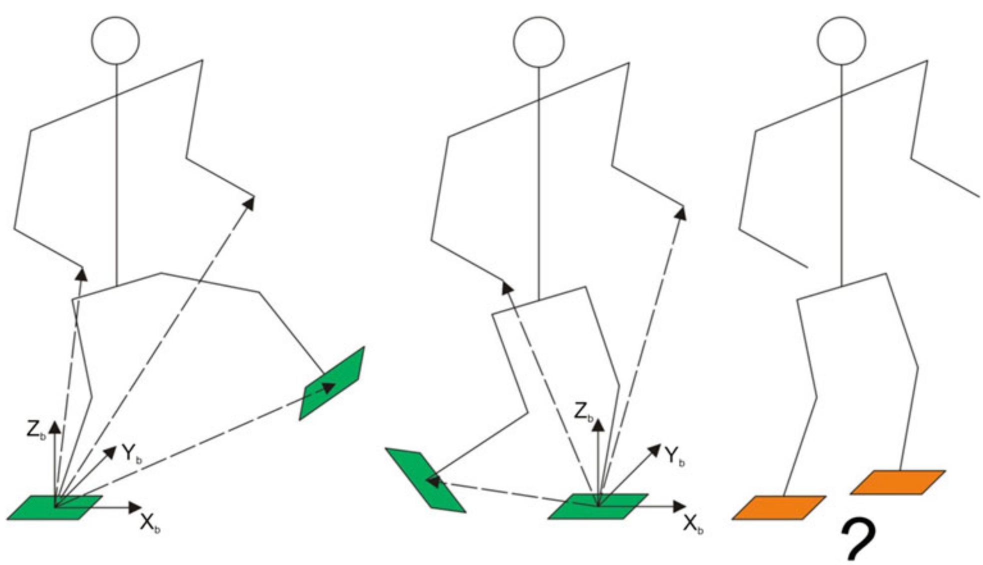


Fig. 3 Which reference frame?

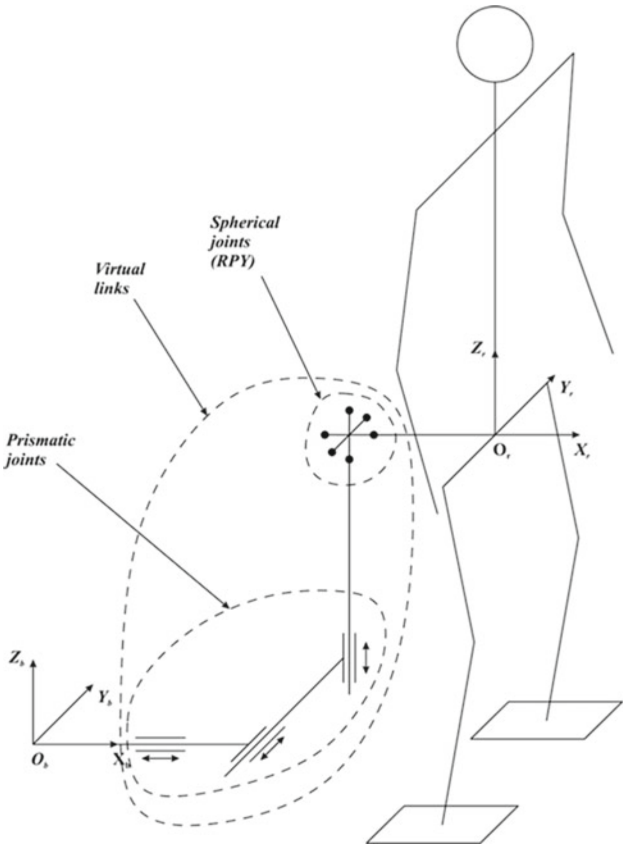


Fig. 4 Virtual joints

2.3 Augmented Jacobian

Each chain has its own kinematic function; therefore a Jacobian matrix can be computed for a generic control point of the structure. Generally, considering n control points we can define the following set of equations:

$$\begin{aligned} \mathbf{v_1} &= \mathbf{J_1}\dot{\mathbf{q}} \\ &\vdots \\ \mathbf{v_i} &= \mathbf{J_i}\dot{\mathbf{q}} \\ &\vdots \\ \mathbf{v_n} &= \mathbf{J_n}\dot{\mathbf{q}} \end{aligned} \tag{2.2}$$

where the generic element $\mathbf{J_i}$ is the Jacobian matrix related to a specific control point i. It is understood that, if the generic joint variable $\mathbf{q_j}$ does not affect $\mathbf{v_i}$, it is $(\mathbf{J_i})_{kj} = 0$ (where k is the generic row of $\mathbf{J_i}$). This set of equations can be summarized as follow:

$$\mathbf{v} = \mathbf{J}_{\mathrm{AU}}\dot{\mathbf{q}} \tag{2.3}$$

where $\mathbf{J}_{\mathrm{AU}}$ is the so called augmented Jacobian. On one hand, this approach allows us to solve the IK problem with only one CLIK algorithm [17], i.e.:

$$\dot{\mathbf{q}} = \mathbf{W}^{-1}\mathbf{J}_{\mathrm{AU}}^{\mathrm{T}}\left(\mathbf{J}_{\mathrm{AU}}\mathbf{W}^{-1}\mathbf{J}_{\mathrm{AU}}^{\mathrm{T}}\right)^{-1}\mathbf{ve} \tag{2.4}$$

In particular, if $\mathbf{W}$ is the identity matrix, then:

$$\dot{\mathbf{q}} = \mathbf{J}_{\mathrm{AU}}^{\dagger}\mathbf{v} \tag{2.5}$$

where

$$\mathbf{J}_{\mathrm{AU}}^{\dagger} = \mathbf{J}_{\mathrm{AU}}^{\mathrm{T}}\left(\mathbf{J}_{\mathrm{AU}}\mathbf{J}_{\mathrm{AU}}^{\mathrm{T}}\right) \tag{2.6}$$

is the so called *right pseudo-inverse* of $\mathbf{J}_{\mathrm{AU}}$.

On the other hand, the trajectories defined for the control points will be all considered as tasks with the same priority (primary tasks), unlike other solution methods do, such as null-space based approaches [18,19].

In particular, in order to define the structure of $\mathrm{J_{AU}}$, the joint velocities vector must be properly sorted. Since the humanoid structure is composed by four kinematic chains, we can write four different vectors of unknowns:

$$\begin{aligned} \dot{\mathbf{q}}_1 &= \left[\, \dot{\mathbf{q}}_{\mathrm{r}}^{\mathrm{T}}\ \dot{\mathbf{q}}_{\mathrm{rl}}^{\mathrm{T}} \,\right]^{\mathrm{T}} && \text{right leg} \\ \dot{\mathbf{q}}_2 &= \left[\, \dot{\mathbf{q}}_{\mathrm{r}}^{\mathrm{T}}\ \dot{\mathbf{q}}_{\mathrm{ll}}^{\mathrm{T}} \,\right]^{\mathrm{T}} && \text{left leg} \\ \dot{\mathbf{q}}_3 &= \left[\, \dot{\mathbf{q}}_{\mathrm{r}}^{\mathrm{T}}\ \dot{\mathbf{q}}_{\mathrm{b}}^{\mathrm{T}}\ \dot{\mathbf{q}}_{\mathrm{ra}}^{\mathrm{T}} \,\right]^{\mathrm{T}} && \text{right arm} \\ \dot{\mathbf{q}}_4 &= \left[\, \dot{\mathbf{q}}_{\mathrm{r}}^{\mathrm{T}}\ \dot{\mathbf{q}}_{\mathrm{b}}^{\mathrm{T}}\ \dot{\mathbf{q}}_{\mathrm{la}}^{\mathrm{T}} \,\right]^{\mathrm{T}} && \text{left arm} \end{aligned} \tag{2.7}$$

where $\dot{\mathbf{q}}_{\mathbf{r}}$ is the vector of the virtual joint velocities that are shared among four kinematic chains. These vectors can be summarized in only one vector of unknowns

$$\mathbf{q} = \left[\, \mathbf{q}_{\mathrm{r}}^{\mathrm{T}}\ \mathbf{q}_{\mathrm{rl}}^{\mathrm{T}}\ \mathbf{q}_{\mathrm{ll}}^{\mathrm{T}}\ \mathbf{q}_{\mathrm{b}}^{\mathrm{T}}\ \mathbf{q}_{\mathrm{ra}}^{\mathrm{T}}\ \mathbf{q}_{\mathrm{la}}^{\mathrm{T}} \,\right]^{\mathrm{T}} = \left[\, \mathrm{q}_1\ \mathrm{q}_2\ \ldots\ \mathrm{q}_{39} \,\right]^{\mathrm{T}} \tag{2.8}$$

With this choice, the Augmented Jacobian takes the following form:

$$\mathbf{J}_{\mathrm{AU}} = \begin{bmatrix} \mathbf{J_r} & \mathbf{J_{rl}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J_r} & \mathbf{0} & \mathbf{J_{ll}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J_r} & \mathbf{0} & \mathbf{0} & \mathbf{J_b} & \mathbf{0} & \mathbf{0} \\ \mathbf{J_r} & \mathbf{0} & \mathbf{0} & \mathbf{J_b} & \mathbf{J_{ra}} & \mathbf{0} \\ \mathbf{J_r} & \mathbf{0} & \mathbf{0} & \mathbf{J_b} & \mathbf{0} & \mathbf{J_{la}} \end{bmatrix} \tag{2.9}$$

The matrix $\mathbf{J}_{\mathrm{AU}}$, with the proposed humanoid model, has 39 columns, while the number of its rows depends on the number of considered control points.

2.4 Center of mass Jacobian

Unlike industrial manipulators and, more generally, non-legged robots, bipedal robots must explicitly address balance control while performing any task. If this does not happen, obviously, the robot would lean over and fall. Moreover, humanoid robots are inherently hyper-redundant, having a much higher number of joints than traditional industrial robots. Consequently, there are many postures that achieve the same position for its body terminals, corresponding to control points. Also, taking into account the balancing issues allows the humanoid to attain more natural postures, similar to those of human beings.

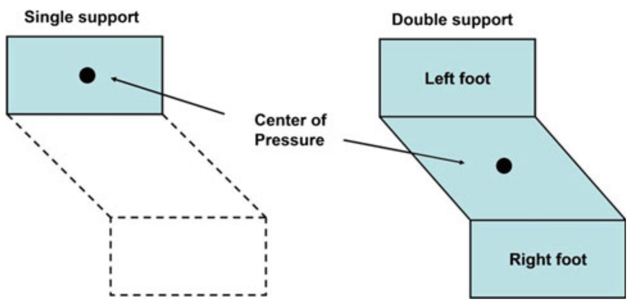


Fig. 5 Center of pressure and support polygon

For this, the multiple VEEs approach [13], has been implemented also with respect to the CoM of the digital humanoid, which becomes a further control point for the kinematic chain.

In particular, the trajectory of the CoM can be defined in such a way that the center of pressure (CoP) belongs to the stability polygon formed by the feet (Fig. 5).

It is worth noticing that the constraint about the CoP will be treated as a primary task, as well as the other tasks. The basic idea is to obtain a differential relationship like

$$\mathbf{v}_G = \mathbf{J}_G \dot{\mathbf{q}} \tag{2.10}$$

where $\mathbf{J}_G$ is a $3 \times n$ matrix, called center-of-mass Jacobian. Then, Eq. (2.10) will be inserted in the equations set (2.2) as a further control point. For this purpose, we can calculate the position of the CoM for a kinematic chain composed of n links as:

$$\mathbf{p}_G = \frac{\sum_{i=1}^{n} m_i \mathbf{p}_{G_i}}{\sum_{i=1}^{n} m_i}. \tag{2.11}$$

Deriving the Eq. (2.11) with respect of time we obtain the CoM's velocity

$$\mathbf{v}_G = \frac{\sum_{i=1}^{n} m_i \mathbf{v}_{G_i}}{\sum_{i=1}^{n} m_i}. \tag{2.12}$$

Since the center of mass of each link can be considered as a *virtual end-effector* it is always possible to write the differential relationship

$$\mathbf{v}_{G_i} = \mathbf{J}_{G_i} \dot{\mathbf{q}} \tag{2.13}$$

where

$$\mathbf{J}_{G_i} = \begin{bmatrix} \gamma_{x,1} \dots \gamma_{z,1} & 0 \dots 0 \\ \gamma_{y,1} \dots \gamma_{z,1} & 0 \dots 0 \\ \gamma_{z,1} \dots \gamma_{z,1} & 0 \dots 0 \end{bmatrix}. \tag{2.14}$$

Indeed, if the vector $\dot{\mathbf{q}}$ has been properly sorted, $\mathbf{v}_{Gi}$ can be affected at most by the first *i* links of the chain. Now, Eq. (2.12) can be written, placing $m = \sum_{i=1}^{n} m_i$, as

$$\mathbf{v}_G = \left[\frac{1}{m} \sum_{i=1}^{n} m_i \mathbf{J}_{G_i} \right] \dot{\mathbf{q}}. \tag{2.15}$$

By comparing Eqs. (2.15) and (2.10), we can finally assume

$$\mathbf{J}_G = \frac{1}{m} \sum_{i=1}^{n} m_i \mathbf{J}_{G_i}. \tag{2.16}$$

Given $\mathbf{J}_G$, the CoM becomes a further control point for the kinematic chain. Thus, we can insert the kinematic relation (2.10) in the equations set (2.2).

As a result, we will have an augmented Jacobian matrix with two more rows that are related to the components of $\mathbf{v}_G$ projected on the current support plane (CoP's velocity). As mentioned above, the implemented inversion algorithm assures that a constraint on CoM velocity becomes a high-priority task to be achieved, without using null-space projection.

Finally, it is worth emphasizing that the expression of $\mathbf{J}_G$ suggests also the possibility to use the *kinetostatic duality* (KSD) [20] to compute the aliquot of the joint torques needed to balance the gravity that affects the humanoid's mechanical structure.

2.5 Conflicting tasks

As mentioned above, some tracts of the humanoid structure are shared among apparently different kinematics chains. For instance, the right and left arms of the humanoid share a common tract, namely the back. But, if actually left and right arms were modelled as independent chains, they could perform different or even conflicting tasks.

For this, inverse kinematics algorithms for multi-legged robots generally provide two different solutions for the left and right arm. In particular, for example, for the back it will be

$$\dot{\mathbf{q}}_{bl} \neq \dot{\mathbf{q}}_{br} \tag{2.17}$$

where $\dot{\mathbf{q}}_{bl}$ and $\dot{\mathbf{q}}_{br}$ are different solutions obtained considering the back belonging respectively to right and to left arm. Several strategies have been adopted to avoid this problem. More in general, optimality criteria can be adopted for solving redundancy.

The CLIK algorithm based on the Augmented Jacobian cleverly resolves also this issue. Indeed, the vector of solution $\dot{\mathbf{q}}$ has been sorted in such a way that its elements appear just once, thus the inversion algorithm provides only one solution that is consistent with all the physical constraints.

On the other side, the main problem related to the application of Augmented Jacobian method is the matrix inversion, due to its dimensions ($\mathbf{J}_{AU}$ has 39 columns) and consequently to the detection of its singularities.

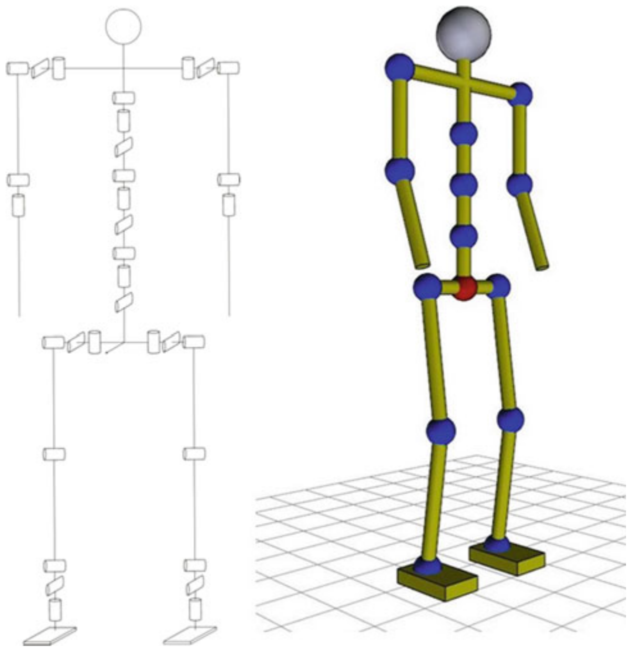


Fig. 6 Kinematic structure of the humanoid and the corresponding VRML model

3 Geometric model

A geometric model for the digital humanoid has been built up with a typical hierarchical approach. The result has been a VRML model that can be used for the simulations. Figure 6 shows the Kinematic structure of the humanoid and the corresponding VRML model.

After that, the kinematic model of the virtual humanoid and its CLIK algorithm has been implemented with MathWorks MATLAB-Simulink software. Since a weighted pseudo-inverse has been adopted to compute the inverse kinematics, a proper choice of weights and of some optimization criteria have granted quite natural and fluid movements for the virtual humanoid. As a result, despite the ease of planning the movements of the digital humanoid, we can simulate quite complex tasks, by planning the trajectory for only a limited number of control points.

4 Biomechanical model

The formulation of the IK problem for a virtual humans in terms of a single CLIK algorithm suggests also the possibility of computing the torque at the joints of the humanoid by means of KSD [20]. The main feature of the proposed model is the possibility of implement multiple-point kinematic control for a humanoid using a single CLIK algorithm [11,15] (something like the serial robots), through the $\mathbf{J}_{AU}$ matrix. Thus, the torque joints related to a force applied to a generic point of the structure will be:

$$\boldsymbol{\tau}_\mathbf{i} = \mathbf{J}_\mathbf{i}^\mathbf{T}\mathbf{f}_\mathbf{i} \tag{4.1}$$

where $\mathbf{f}_\mathbf{i}$ is the generalized vector of forces applied at generic point of the structure, $\mathbf{J}_\mathbf{i}$ is its Jacobian matrix and $\boldsymbol{\tau}_\mathbf{i}$ the resulting torque joints vector.

We considered as reference low back biomechanical model the one illustrated in [21] by Chaffin and shown in Fig. 7a. Figure 7b shows the low back model of the developed digital human.

4.1 Validation of the biomechanical model

Taking into account the dynamic biomechanical model of lifting described in [21] (see Fig. 7) the moment at the hip can become quite large, especially when a load is lifted far from the body. In particular [22] proposed that the load moment about the lumbosacral disc L5/S1 should be used for setting the limit for lifting and carrying loads of various sizes in order to avoid muscle fatigue in the erector spinal muscle group.

From a biomechanical point of view the fact that large moments are created at the lumbar spine, when heavy loads are lifted, raises the question of the nature of the internal forces that must be present to stabilize the spine while incurring such load moments.

A simple static model of the lumbar spine during lifting was proposed by Morris et al. [23]. They assumed that two kind of internal forces are involved in order to contrast the external load moment. One is produced by the extensor erector spinal muscles which exert their action approximately at 5 cm posterior to the centre of rotation in the spinal discs; the second, due to the abdominal pressure acting on the diaphragm [24].

This model showed also that while the load is lifted a large compression forces raises in the spinal column that acted to compress the disc. The existence of this compression was also later confirmed experimentally by [25].

A simple static analysis, which does not consider the abdominal pressure, can be made first identifying the relevant forces involved, shown in Fig. 8, where $\mathbf{P}_\mathrm{L}$ is the load due to the lifted weight, $\mathbf{P}_\mathrm{B}$ is the weight of the whole humanoid, $\mathbf{P}_\mathrm{BW}$ the weight of the body part located above the L5/S1 level, $\mathbf{R}$ is the reaction of the plan, $\mathbf{F}_\mathrm{M}$ is the force produced by the extensor erector spinal muscles, $\mathbf{F}_\mathrm{C}$ and $\mathbf{F}_\mathrm{S}$ are, respectively, the compression and the shear force acting on the lumbosacral discs L5/S1. In particular $\mathbf{P}_\mathrm{L}$, $\mathbf{P}_\mathrm{B}$, $\mathbf{P}_\mathrm{BW}$ and R are external forces and $\mathbf{F}_\mathrm{M}$, $\mathbf{F}_\mathrm{C}$ and $\mathbf{F}_\mathrm{S}$ are internal forces.

The analysis of the forces acting on the L5/S1 discs begins with the computation of the second member of Eq. (4.2) by using the KSD,

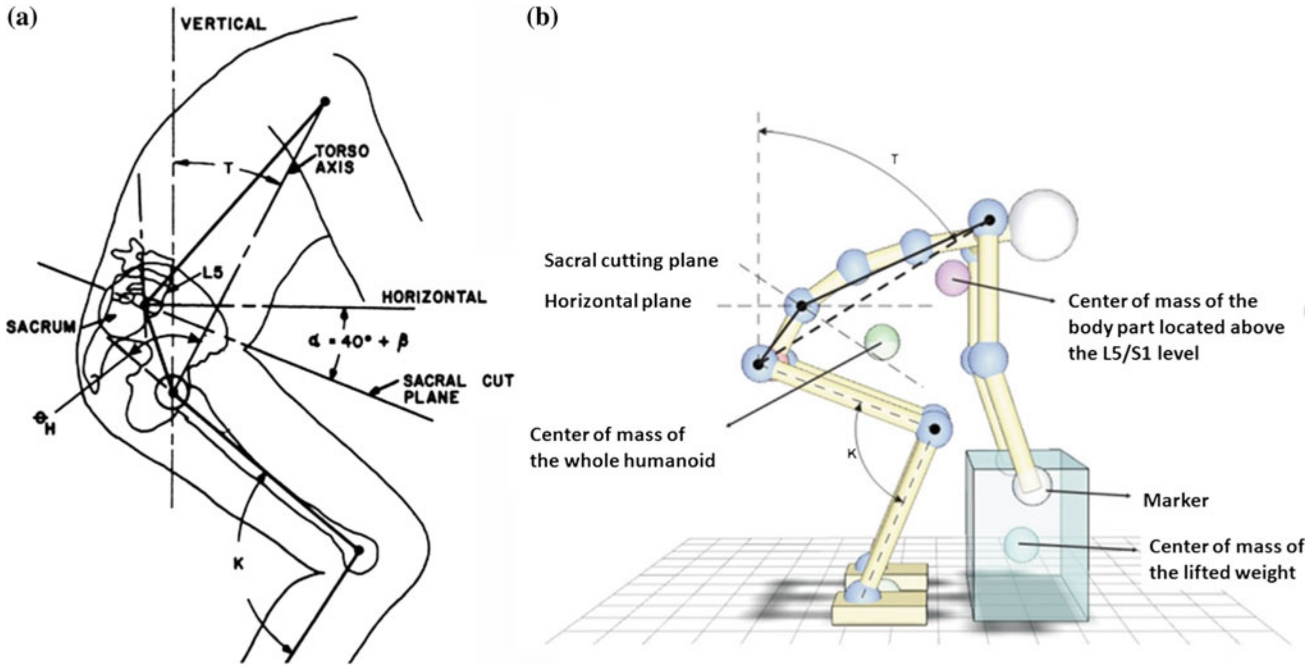
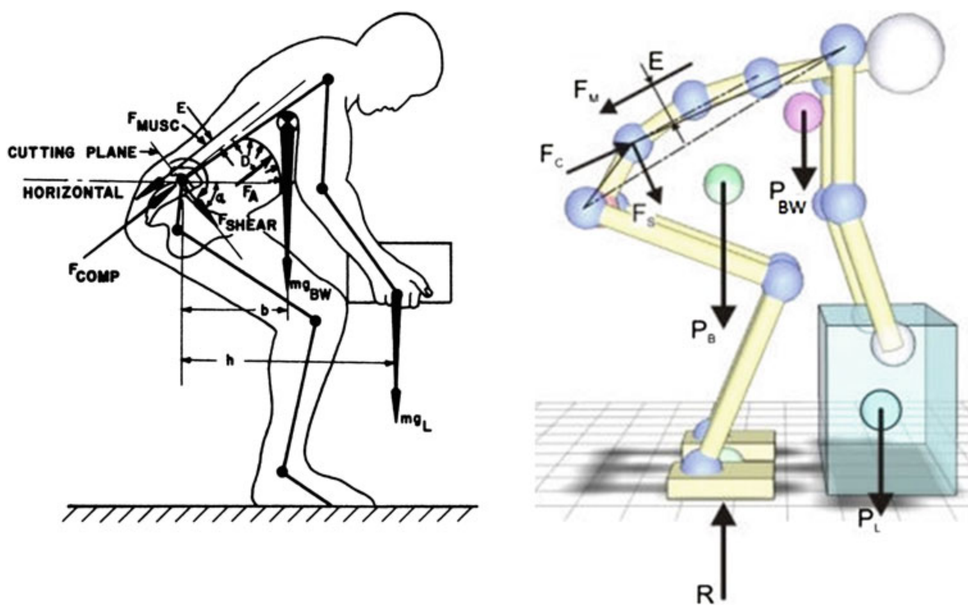


Fig. 7 **a** Low-back model as adapted by Chaffin [21], **b** low back model of the digital human

Fig. 8 Simple cantilever low-back model of lifting as adapted by Chaffin for static coplanar lifting analyses [21] and virtual simulation with the digital humanoid



$$\mathbf{M}_{\mathrm{L5/S1}} = \sum_{\mathrm{i}} \mathbf{M}\mathrm{i}. \tag{4.2}$$

Known the average moment arm "E" between the erector spinal muscles and the lumbosacral discs L5/S1 it is possible to compute the muscular effort F_M as:

$$\mathbf{F}_{\mathrm{M}} \cdot \mathrm{E} = \sum_{\mathrm{i}} \mathbf{M}\mathrm{i}. \tag{4.3}$$

For the compression and shear stress it is enough to project $\mathbf{P_{BW}}$ along the directions perpendicular and parallel to the sacral cutting plane. The shear and compression forces can be then calculated through static equilibrium equations.

$$\begin{aligned} &P_{BW}b + P_L h + F_M E = 0\\ &P_{BW}cosa + P_L cosa + F_M + F_C = 0\\ &P_{BW}sina + P_L sina - F_S = 0. \end{aligned} \tag{4.4}$$

With reference to the lifting task showed in Fig. 7 the efforts F_M, F_C and F_S related to lifting a load have been computed and, then, compared with the *Low-Back Biomechanical Model* results proposed in [21]. The results proposed in

Table 1 Lifting analysis: simulation results compared to the data from the published literature

		Literature [21]	Simulation
F_M	Force produced by the extensor erector spinal muscles	−3,154 N	−3,220 N
F_C	Compression force acting on the lumbosacral discs L5/S1	−3,612 N	−3,560 N
F_S	Shear force acting on the lumbosacral discs L5/S1	−656 N	−660 N

the published literature are referred to a single posture of the human identified by torso and knee angle. Assuming the following values:
Posture: $h = 30$ cm; $b = 20$ cm; $E = 6.5$ cm; $T = 60°$; $K = 120°$; $\Theta_H = 110°$
Loads: $P_{BW} = -350\,N$; $P_L = -450\,N$.

We have compared the results obtained simulating the lifting task using the developed digital human with the data from the published literature [21].

In spite of the simplicity of the humanoid model proposed, the simulation results, for the posture examined, are in good agreement with the literature data, as shown in Table 1.

5 Applications

In this section, the results of several VR simulations are reported. They show how the proposed model can be used to make the digital human accomplish some basic as well as more complex tasks.

It is worth emphasizing that the CLIK algorithm always takes into account the constraint about the CoP.

5.1 Standing-up from a sitting position

In Fig. 9 different frames of a standing-up simulation are shown. This task has been achieved just by imposing a null velocity reference to the feet of the virtual humanoid and by giving a vertical velocity reference to its pelvis. As a further constraint, the CoP must always belong to the support plane

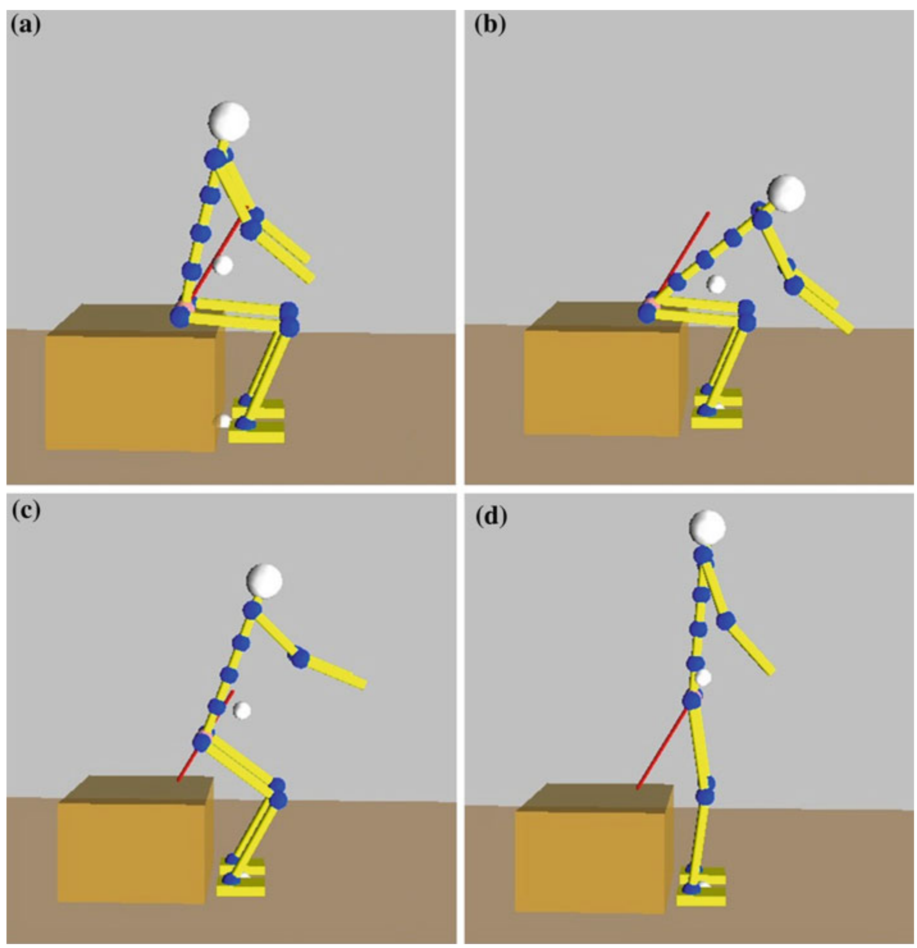


Fig. 9 Standing up from a sitting position

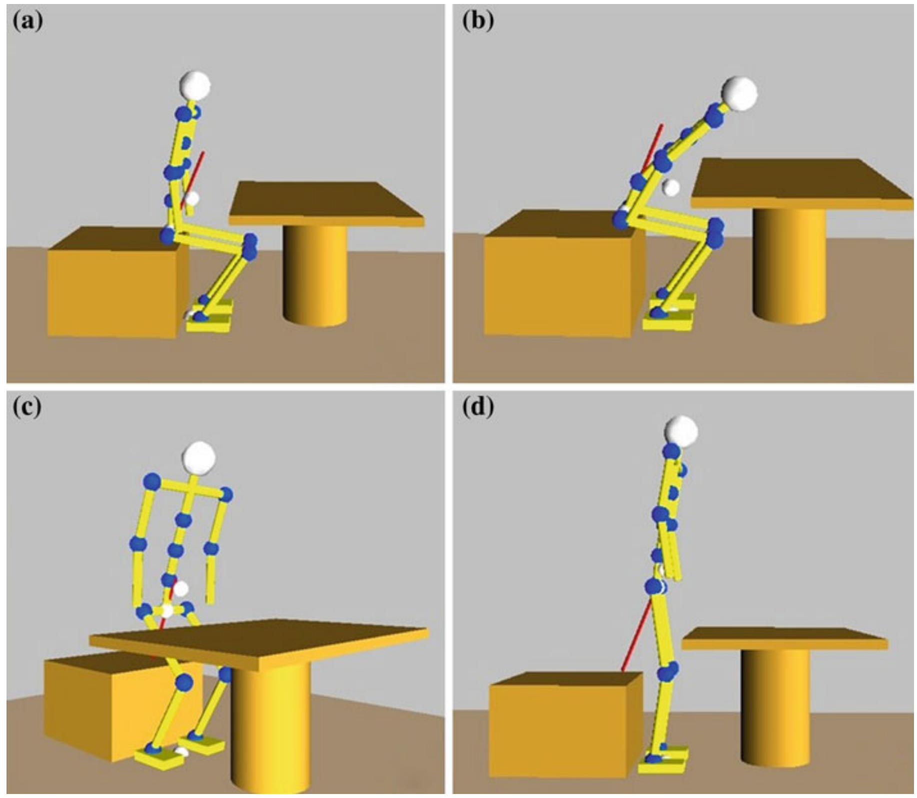


Fig. 10 Standing up from sitting position near a table

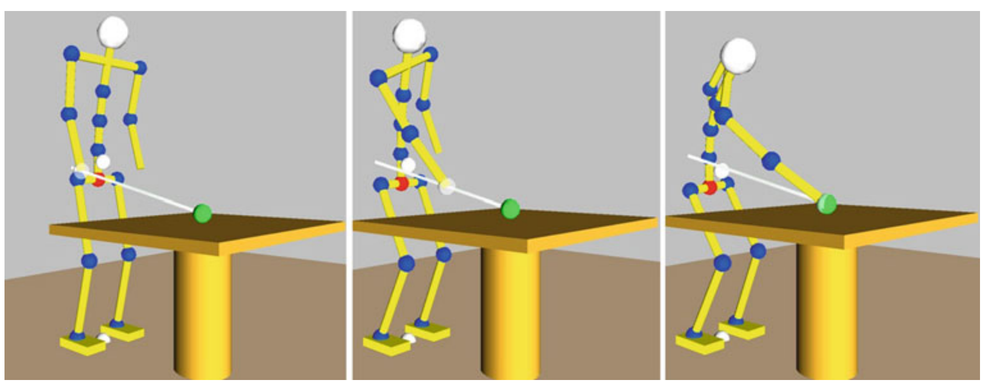


Fig. 11 The virtual humanoid grabbing an object on a table

(balance control). As shown, the virtual humanoid performs the assigned movement always keeping itself in balance.

In a similar way, it is possible to simulate the virtual humanoid sitting down from a standing position.

5.2 Collision avoidance

The approach that has been presented in this paper, can also be used to take into account possible obstacles inside the virtual workspace. Figure 10 shows again the simulation of a standing up, but this time there is a table. This task has been achieved by assigning to the control points velocity references coming from repulsive potential fields.

5.3 Reaching an object on a table

Figure 11 shows some frames of a simulation related to the task of reaching an object on a table. The reference motion is given by simply imposing a null velocity reference to the feet and a non zero velocity reference to the hand, which

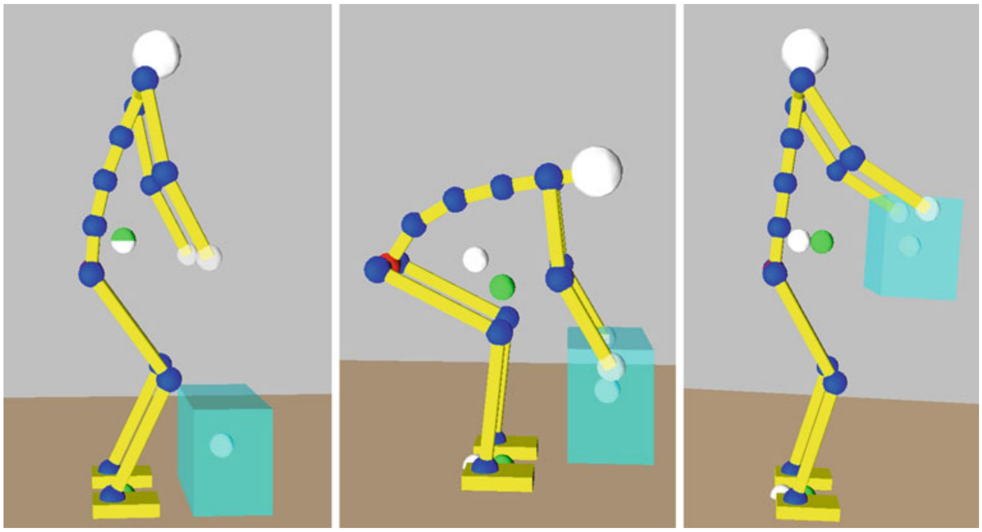


Fig. 12 The humanoid lifts a weight

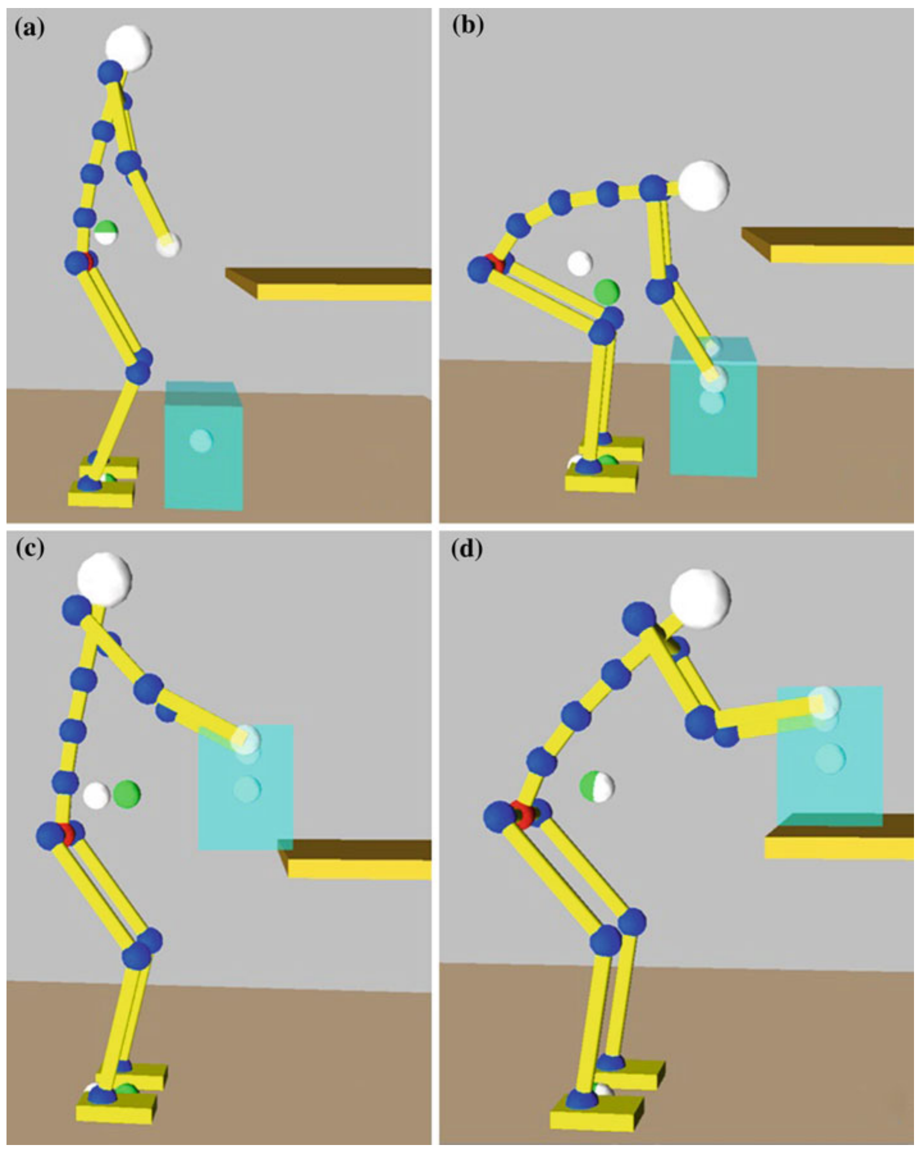


Fig. 13 The human lifts a weight and releases it on a plane

in this case is the “task related control point”. Even for this simulation the resulting whole-body motion is quite natural despite the simplicity of motion planning process.

5.4 More complex tasks

In this section frames of simulation results of quite complex tasks related to possible work activities are reported. The CLIK algorithm has always taken into account the constraints about the CoP, as before mentioned.

The results shown in the following figures can be quite interesting in the field of product and workplace design as well as virtual training environment.

Figure 12 shows frames of simulation result related to the task of lifting a weight.

The CoM control is always active. It is worth noticing some aspects: first, the motion is imposed only at the hands while holding the feet and keeping the CoP belonging to the support polygon; second, it is interesting to see how, although few control points are considered, the resulting motion is quite natural.

The algorithm takes into account the load, when lifted, and the humanoid changes its postures accordingly with CoM variation, due to the weight lifted, in such a way that a total CoP still belongs to the support polygon.

Even when more complex task is simulated, where a humanoid lifts a weight from the ground and releases it on a table (Fig. 13) the resulting motion is quite natural and it is evident how the humanoid modifies its posture according to the CoM variation in order to keep the balance.

In this way, different kinematic behaviours can be obtained for the same planned task (Fig. 14). In particular, it can be accomplished either through some optimal criteria or by using a weighted pseudo-inverse or both. This characteristic is very useful in the simulation of assembly/disassembly operations where a cost function based on ergonomic factors could be optimized in order to show the best postural sequence [26,27].

Finally, to show the power and the versatility of the algorithm as well as the ease of motion planning, simulations of walking task have been implemented. It is worst emphasizing that the planning of the motion has been achieved simply imposing a reference velocity motion to the feet of the humanoid while the whole body motion is automatically generated by the algorithm taking into account imposed constraints (e.g. CoP’s position).

In particular Fig. 15 shows a simulation task related to the walking up and the walking down the stairs.

It is worth emphasizing that the algorithm developed allows plotting the efforts, and any other parameter of

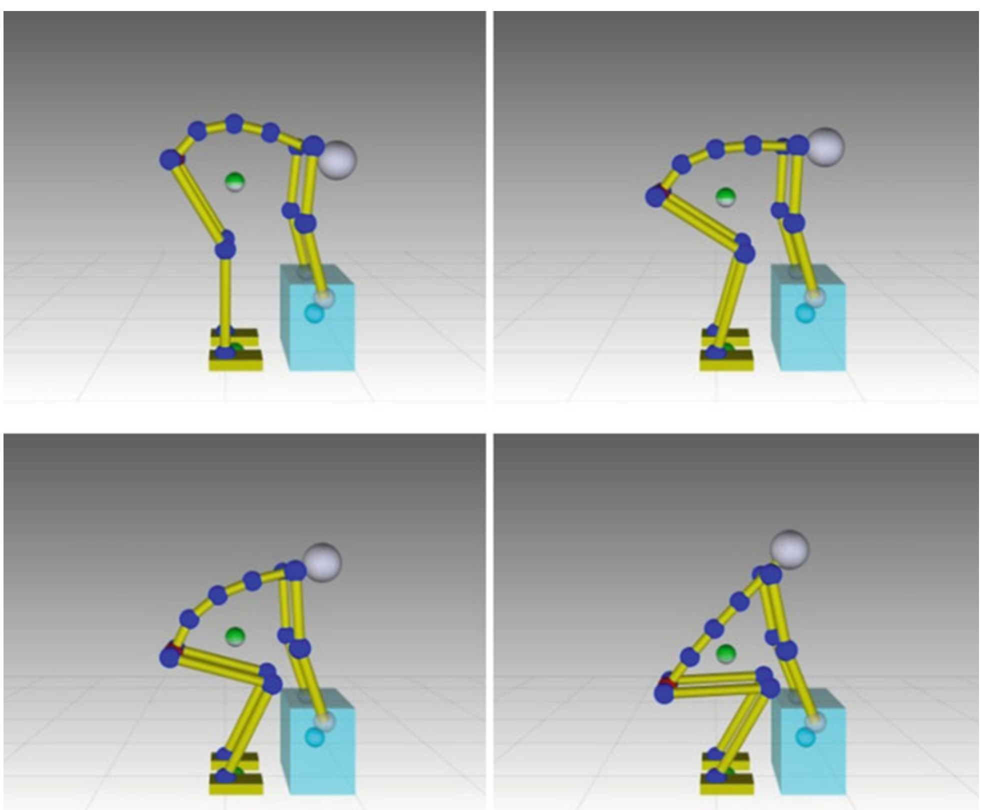


Fig. 14 Different behaviours related to the task of lifting a weight

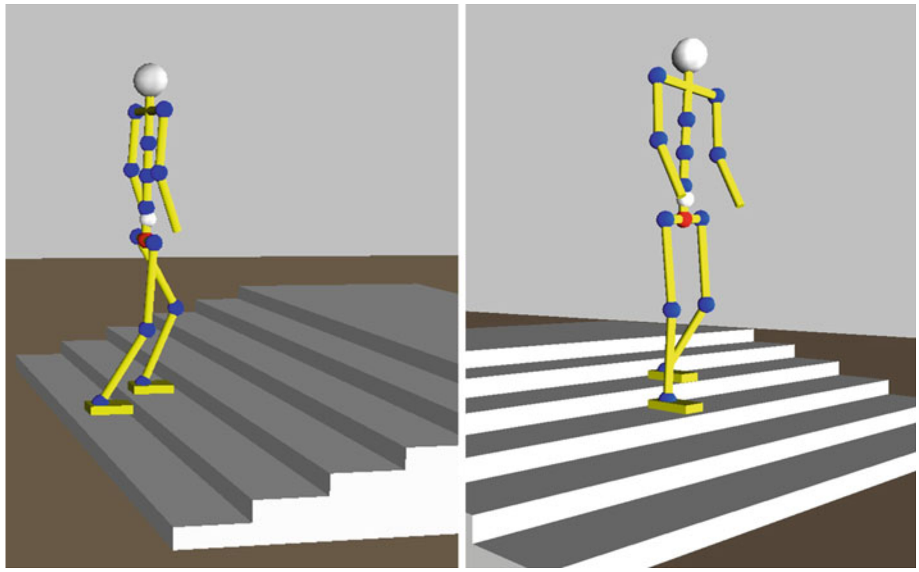


Fig. 15 The humanoid walks up and down the stairs

interest, throughout the duration of the whole simulation task.

6 Conclusion and future works

A biomechanically-consistent motion control for a digital human has been developed through the use of techniques derived from robotics. Unlike most of the commercially available software tools, only few control points are needed to control the motion of the whole body.

The main contribution of the proposed approach is the introduction of an augmented Jacobian matrix that allows specifying trajectories for different control points and solving the inverse kinematics with a single CLIK algorithm.

The introduction of the kinematic control for the CoM of the structure has granted quite natural movement to the digital human in spite of the limited number of considered control points. Moreover, virtually every point of its kinematic structure can be used as a control point.

The symbolic implementation leads to a very fast response of the algorithm with respect to complex simulations in the humanoid configuration space. Moreover, the algorithm can be adopted for a very huge set of applications.

Although the described model is advanced in terms of quality of analysis, it is also computationally efficient. Specifically, a symbolic representation for the kinematics of the digital humanoid has been derived. In this way, it is possible to change at run-time several characteristic parameters of the kinematic chain, such as the applied loads, without further computational overload.

Finally, the consistency of the simulation results with the biomechanical data from the published literature raises the interest in a deeper investigation about its use for the computation of the physical effort and the muscle strain of a human being, at least for static analyses. In this scenario, the developed model is a very powerful and easy to use tool for biomechanical as well as ergonomic analyses.

A future field of application could be marker-based motion capture [28], where the algorithm can be used to limit the number of markers needed to capture human movements.

Acknowledgments The present paper was developed with the economic support of MIUR (Italian Ministry of University and Research) performing the activities of the project PON01_01268 “Digital Pattern”. The authors thank Dr. Agostino De Santis for his helpful discussions and support.

References

1. Di Gironimo, G., Lanzotti, A.: Designing in VR. Int. J. Interact. Design Manuf. **3**(2), 51–53. ISSN:1955-2513 (2009)
2. Caputo, F., Di Gironimo, G., Marzano, A.: Ergonomic optimization of a manufacturing system work cell in a virtual environment. Acta Polytechnica. **46**(5), 21–27. ISSN:1210-2709 (2006)
3. Caputo, F., Di Gironimo, G., Papa, S.: A virtual reality system for ergonomics and usability validation of equipment controls. Anales de Ingegnería Gráfica. **18**, 47–64. ISSN:1137-7704 (2006)
4. Di Gironimo, G., Patalano, S.: Re-design of a railway locomotive in virtual environment for ergonomic requirements. Int. J. Interact. Design Manuf. **2**(1), 47–57. ISSN:1955-2513 (2008)
5. Moes, N.C.: Digital human models: an overview of development and applications in product and workplace design. In: Proceedings of TMCE (Tools and Methods of Competitive Engineering) International Conference, Ancona (2010)
6. Di Gironimo, G.: Study and Development of Ergonomic Design Methodologies in Virtual Environment. Ph.D. Thesis, Naples (2002)
7. Shaikh, I., Jayaram, U., Jayaram, S., Palmer, C.: Participatory ergonomics using VR integrated with analysis tool. 2004 Winter Simulation Conference, pp 1746–1754 (2004)

8. Van der Meulen, P., Seidl, A.: Ramsis—the leading CAD tool for ergonomic analysis of vehicles, 1st International Conference on Digital Human Modeling, Beijing, China (2007)
9. Naval Biodynamics Laboratory. Antrophometry and Mass Distribution for Human. vol. 1. Military Male Aviators, New Orleans (1988)
10. Kroemer, K.H., Snook, S.H., Meadows, S.K., Deutsch, S.: Ergonomic Models of Anthropometry, Human Biomechanics and Operator-Equipment Interfaces. National Press Academy, Washington, D.C. (1998)
11. De Santis, A., Di gironimo, G., Pelliccia, L., Siciliano, B., Tarallo, A.: Human-like motion generation for a virtual human. In: Proceedings of IDMME (Integrated Design and Manufacturing in mechanical engineering)—Virtual Concept 2010, October 20th–22th, Bordeaux, France (2010)
12. De Santis, A., Albu-Shaeffer, A., Ott, C. Siciliano, B., Hirzinger, G.: The skeleton algorithm for self-collision avoidance of a humanoid manipulator. 2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Zürich (2007)
13. De Santis, A., Pierro, P., Siciliano, B.: The multiple virtual end-effectors approach for human–robot interaction. In: 10th International Symposium on Advances in Robot Kinematics, Ljubljana (2006)
14. De Santis, A., Siciliano, B.: Inverse kinematics of robot manipulators with multiple moving control points. In: 11th International Symposium an Advances in Robot Kinematics, Batz-sur-Mer (2008)
15. De Santis, A., Di Gironimo, G., Pelliccia, L., Siciliano, B., Tarallo, A.: Multiple-point kinematic control of a humanoid robot. In: 12th International Symposium on Advances in Robot Kinematics, Piran-Portorož (2010)
16. Yamane, K.: Simulating and Generating Motions of Human Figure. Springer, Heidelberg (2004)
17. Siciliano, B., Khatib, O. (eds.): Springer Handbook of Robotics. Springer, Berlin (2008)
18. Nakamura, Y.: Advanced Robotics: Redundancy and Optimization. Addison-Wesley, Reading (1991)
19. Siciliano, B., Slotine, J.-J.E.: A general framework for managing multiple tasks in highly redundant robotic systems. In: 5th International Conference on Advanced Robotics, Pisa (1991)
20. Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G.: Robotics: Modelling, Planning and Control. Springer, Berlin (2009)
21. Chaffin, D.B., Anderson, G.B.J., Martin, B.J.: Occupational Biomechanics. 3rd edn. Wiley, New York (1999)
22. Tichauer, E.R.: A pilot study of the biomechanics of lifting in simulated industrial work situations. J. Saf. Res. **3**, 98–115 (1971)
23. Morris, J.M., Lucas, D.B., Bressler, B.: Role of the trunk in stability of the spine. J. Bone Joint Surg. **43**, 327–351 (1961)
24. Bartelink, D.L.: The role of abdominal pressure in relieving the pressure on the lumbar intervertebral discs. J. Bone Joint Surg. **39**, 718–725 (1957)
25. Nachemson, A., Elfstrom, G.: Intravital dynamic pressure measurements in lumbar disc. Scand. J. Rehabil. Med. **1**, 1–39 (1970)
26. Di Gironimo, G.: Design for maintainability using virtual manikins: a case study on the air-conditioning system of a regional train. In: Proceedings of 5th CIRP International Conference on Intelligent Computation in Manufacturing Engineering. Ischia (Naples), 26th – 28th July 2006, pp. 225-230. ISBN 88-95028-01-5; 978-88-95028-01-9 (2006)
27. Di Gironimo, G., Monacelli, G., Patalano, S.: A design methodology for maintainability of automotive components in virtual environment. In: Proceedings of International Design Conference—Design 2004, pp. 723–734, May 18–21, Dubrovnik. ISBN: 953-6313-61-8 (2004)
28. Magnenat-Thalman, N., Thalman, D.: Modelling and Motion Capture Tecniques for Virtual Environments. Springer, Geneva (1998)