On-Line Computational Scheme for Dynamic Walking of Anthropomorphic Biped Robots

Guanzheng Tan, Feng Liang
College of Information Science and Engineering
Central South University
Changsha 410083, Hunan Province
P. R. China

Yuechao Wang
Robotics Laboratory
Chinese Academy of Sciences
Shenyang 110015, Liaoning Province
P. R. China

Abstract—Based on the Luh-Walker-Paul's algorithm, a new and more effective on-line computational scheme used for real-time control of dynamic walking of anthropomorphic biped robots is developed. This scheme includes two algorithms, one is the famous Luh-Walker-Paul's algorithm used for the single-foot supporting phase, and another is called IDADFS algorithm which is developed in this paper and used for the double-foot supporting phase. In IDADFS algorithm, the authors not only have given the recursive formulas for the kinematic and dynamic computations but also have proposed three criteria to examine the correctness of these computations. By means of this new computational scheme, one can perform precise real-time control for the dynamic walking of anthropomorphic biped robots. It should be pointed out that the research result of this paper can be also spread to multilegged robots. The more important is that the IDADFS algorithm also can be generalized to the manipulators with an open kinematic chain to realize the real-time control over this kind of manipulators when they execute closed-chain operating tasks or multirobots coordinative manipulations.

I. INTRODUCTION

As a kind of biped robots, anthropomorphic biped robots have the basic features of human being. The joint configurations and number of degrees-of-freedom of their legs are almost the same as the human being's. This type of biped robots is a very complicated dynamic system. To study and control them, first we must establish their dynamics equations. The dynamics problem of anthropomorphic biped robots includes two opposite aspects—the direct dynamics and inverse dynamics. The direct dynamics means determining the motion trajectory of each joint, given the joint drive forces or torques. While the inverse dynamics means determining the required drive force or torque of each joint, given the joint motion trajectories. Generally, a biped robot is always required to perform a walking movement with a definite purpose, and its walking path is designed in advance. So, the inverse dynamics problem is more important than the direct one.

This paper mainly deals with the on-line inverse dynamics computation for dynamic walking of anthropomorphic biped robots. From the published papers, so far, most of researchers have adopted the standard Lagrangian equation [1, 2] or the Newton-Euler equation [3] to establish the inverse dynamics models of biped robots. The greatest shortcoming of the models established by use of the Lagrangian equation is that the joint drive forces or torques can not be computed, given the positions, velocities, and accelerations of leg's components. So, this kind of models results in many difficulties in real-time control [4]. On the other hand, for the dynamic models developed by use of the Newton-Euler equation, they are effective on computing the drive forces or torques of joints when biped robots are in the single-foot supporting phase. But, this method is no longer effective on control when they are in the double-feet supporting phase. It is based on the above reasons that we determine to study the inverse dynamics computation of dynamic walking of anthropomorphic biped robots. The basis of this paper is the Luh-Walker-Paul's algorithm [5]. This algorithm is based on the Newton-Euler equations. Its computational formulas possess recursive forms and can be utilized in the real-time inverse dynamics computation of manipulators with an open kinematic chain. It takes only 4.5 milliseconds on average to compute six joint torques on a PDP 11/45 minicomputer using float-point assembly language [5]. However, if one wants to apply this algorithm to an anthropomorphic biped robot, he or she must consider a very important conceptual problem, i.e., originally the Luh-Walker-Paul's algorithm was established by taking manipulators with an open kinematic chain as the object of study. For an anthropomorphic biped robot, when it is in the single-foot supporting phase, this algorithm can be directly used to compute its inverse dynamics because the robot is an open kinematic chain in this case. But, when the robot is in the double-feet supporting phase, this algorithm no longer has any effect because the robot becomes a closed kinematic chain in this case. So, studying how to modify the Luh-Walker-Paul's algorithm so that it can be applied to the anthropomorphic biped robots is the main purpose of this paper.

Because the Luh-Walker-Paul's algorithm can be utilized directly in the single-foot supporting phase, the computation of inverse dynamics in this phase will not be discussed in the paper. Here, we will emphatically study and establish the more effective inverse dynamics computation method, which is used to control the dynamic walking of anthropomorphic biped robots in the double-feet supporting phase.

II. INTRODUCTION TO LUH-WALKER-PAUL'S ALGORITHM

The Luh-Walker-Paul's algorithm[5] is a recursive computational algorithm, which is based on the Newton-Euler equations and was established originally aiming at the manipulators with an open kinematic chain. Its purpose is to compute the joint torques or forces applied at each instant to the actuators from
the given joint trajectories. Before using this algorithm, first we must attach an orthogonal coordinate frame \(O_2-X_2Y_2Z_2\) to each of links of a considered manipulator from its base link to its end-effector by means of the Denavit-Hartenberg theory. For example, we can attach coordinate frame \(O_2-X_2Y_2Z_2\) to the base link fixed to the ground and frame \(O_2-X_2Y_2Z_2\) to link \(i\), \(i = 1 \cdots n\), where \(n\) is the degrees of freedom of the manipulator. The Luh-Walker-Paul's algorithm includes the forward recursive computation of kinematics and backward recursive computation of dynamics. Its recursive formulas are shown as follows. In general, because every joint of an anthropomorphic biped robot is revolute, here we only give the Luh-Walker-Paul's algorithm for the manipulators with revolute joints.

\[
\begin{align*}
\alpha_{i+1} &= \omega_i + \dot{\alpha}_{i+1}\beta_i \\
\omega_{i+1} &= \omega_i + \dot{\omega}_{i+1} + \alpha_i \times (\omega_{i+1}\beta_i) \\
\nu_{i+1} &= \nu_i + \dot{\nu}_{i+1} + \alpha_i \times (\nu_{i+1}\beta_i) \\
a_{i+1} &= a_i + \dot{\alpha}_{i+1}\gamma_{i+1} + \alpha_i \times (\gamma_{i+1}\alpha_i) \\
\nu_i &= \nu_i + \dot{\alpha}_i \beta_i \\
\omega_i &= \omega_i + \dot{\omega}_i \beta_i \\
\alpha_i &= \alpha_i + \dot{\alpha}_i \beta_i
\end{align*}
\]

where \(\alpha_i\), \(\omega_i\), \(\nu_i\), \(a_i\) and \(\dot{\alpha}_i\), \(\dot{\omega}_i\), \(\dot{\nu}_i\), \(\dot{a}_i\), \(\ddot{\alpha}_i\), \(\ddot{\omega}_i\), \(\ddot{\nu}_i\) and \(\dddot{a}_i\) are the angular velocities and accelerations of link \(i\) and link \(i+1\) respectively; \(\beta_i\), \(\gamma_i\) are the unit vector of \(Z\) axis; \(\nu_i\) and \(\omega_i\) are the velocities at the origins \(O_i\) and \(O_{i+1}\) of coordinate frames \(O_i\)-\(X_iY_iZ_i\) and \(O_{i+1}\)-\(X_{i+1}Y_{i+1}Z_{i+1}\) respectively; \(a_i\) and \(a_{i+1}\) are the accelerations at the origins \(O_i\) and \(O_{i+1}\) respectively; \(\gamma_{i+1}\alpha_i\) is the position vector from \(O_i\) to \(O_{i+1}\); \(\nu_i\) and \(a_i\) are the velocity and acceleration at the centroid \(C_i\) of link \(i\) respectively; \(\gamma_{i+1}\) is the position vector from \(O_i\) to \(C_{i+1}\); \(\nu_i\) and \(a_i\) are the velocity and acceleration at the centroid \(C_i\) of link \(i\) respectively; \(m_i\) is the mass of link \(i\); \(\nu_{i+1}\) is the position vector from the origin \(O_i\) to the centroid \(C_i\); \(\nu_{i+1}\) is the unit vector of \(Z_{i+1}\) axis; \(M_i\) is the drive torque of joint \(i\); and \(I_i\) is the inertia tensor of the link \(i\). All the above vectors are expressed in the base frame \(O_2\)-\(X_2Y_2Z_2\). The inertia tensor \(I_i\) can be computed by the following formula:

\[
I_i = [R_i^o]I_i[R_i^o]^T
\]

where \(I_i\) is the inertia tensor expressed in the coordinate frame fixed to the link itself, and \([R_i^o]\) is the \(3 \times 3\) rotation matrix associated with the coordinate transformation from frame \(i\) to the base frame.

Before the computation, three kinds of parameters must be determined in advance. The first kind of the parameters are the angular velocities \(\dot{q}_i\) and accelerations \(\ddot{q}_i\) of all joints, \(i=1, 2, \cdots, n\), which can be obtained from the joint trajectories. The second kind of parameters are the drive force \(f_{\alpha i+1}\) and moment \(N_{\alpha i+1}\) exerted on the environment (denoted by \(n+1\)) by the end-effector (link \(n\)), whose values are generally specified or can be measured using the force and moment sensors mounted on the end-efflector. The third kind of parameters are velocity \(v_0\), acceleration \(a_0\), angular velocity \(\omega_0\), and angular acceleration \(\dot{\omega}_0\) of the base. This algorithm is fully suitable for the base whether at rest or in motion. The Luh-Walker-Paul's algorithm can be summarized into the following four steps.

**Forward recursive computation for kinematic variables:**

**Step 1:** Let \(i\) increase consecutively in the order from 0 to \(n-1\) and compute \(\dot{\omega}_i\), \(\dot{\omega}_i\), \(\nu_i\), \(\omega_i\) respectively by (1) ~ (4).

**Step 2:** Let \(i\) increase consecutively in the order from 1 to \(n\) and compute \(v_1\) and \(a_1\) respectively by (5) and (6).

**Backward recursive computation for Dynamic variables:**

**Step 3:** Let \(i\) decrease consecutively in the order from \(n\) to 1 and compute \(f_{\alpha i}\) and \(N_{\alpha i}\) respectively by (7) and (8). The third kind of parameters are velocity \(v_0\), acceleration \(a_0\), angular velocity \(\omega_0\), and angular acceleration \(\dot{\omega}_0\) of the base. This algorithm is fully suitable for the base whether at rest or in motion. The Luh-Walker-Paul's algorithm can be summarized into the following four steps.

**Forward recursive computation for kinematic variables:**

**Step 1:** Let \(i\) increase consecutively in the order from 0 to \(n-1\) and compute \(\dot{\omega}_i\), \(\dot{\omega}_i\), \(\nu_i\), \(\omega_i\) respectively by (1) ~ (4).

**Step 2:** Let \(i\) increase consecutively in the order from 1 to \(n\) and compute \(v_1\) and \(a_1\) respectively by (5) and (6).

**Backward recursive computation for Dynamic variables:**

**Step 3:** Let \(i\) decrease consecutively in the order from \(n\) to 1 and compute \(f_{\alpha i}\) and \(N_{\alpha i}\) respectively by (7) and (8). It should be noted that in the Luh-Walker-Paul's algorithm, to increase computational speed, all the vectors and variables are expressed with reference to the coordinate frame fixed to each link, instead of representing them with reference to the base coordinate frame. The detailed introduction to the Luh-Walker-Paul's algorithm can be read in [5].

**III. CHARACTERISTICS OF ANTHROPOMORPHIC BIPED ROBOTS IN DOUBLE-FEET SUPPORTING PHASE**

Assuming an anthropomorphic biped robot has \(n\) revolute joints, it then consists of \(n+1\) links (each of its feet is also regarded as a link). In double-feet supporting phase, the biped robot becomes a closed kinematic chain, as shown in Fig. 1.

In this case, we can assume any one of the two feet to be the base (called the base foot, or link \(0\)). For example, we assume the front foot is the base foot. Then, the back foot is called the end foot (the last link, or link \(n\)). In the order of links from the base foot to the end foot, each link of the robot can be denoted by a sign from 0 to \(n\) successively.

Generally, in the double-feet supporting phase, a biped robot can not make a turning motion and can only move its body either forward or backward in sagittal plane or towards the left or right in frontal plane so as to adjust its center of gravity. The
sagittal and frontal planes are shown in Fig. 2.

Hence, in the double-feet supporting phase, the motion of an anthropomorphic biped robot can be resolved into two motions, which are in the sagittal plane and frontal plane respectively. Each of the two motions possesses a motion model and each model has its corresponding degrees of freedom. Assuming the degrees of freedom in the sagittal plane is denoted by \( n_s \) and in the frontal plane by \( n_f \). According to the structure symmetry of anthropomorphic biped robots, \( n_s \) and \( n_f \) generally are even numbers. Because the two feet are all in contact with the ground on this condition, the robot motion possesses the following characteristics: (1) If \( i < n_s / 2 \) or \( n_f / 2 \), link \( i \) will rotate around the axis of joint \( i \); (2) If \( i > n_s / 2 \) or \( n_f / 2 \), link \( i \) will rotate around the axis of joint \( i + 1 \); (3) If \( i = n_s / 2 \) or \( n_f / 2 \), i.e., for the upper part of the body, which is linked with the waist, when the robot walks in the sagittal plane, its upper body must rotate around the axis of joint \( n_s / 2 \) and \( (n_f / 2) + 1 \) simultaneously.

On this condition, the axes of the two joints must not only lie on the same line but also have the same rotational speed and direction, that is, their angular velocity vectors must be equal to each other. However, when the robot walks in the frontal plane, in order to ensure that the motion of the upper body is both stable and anthropomorphic, we can assume that the upper body remains straight posture in walking process, which means that joint \( n_f / 2 \) and joint \( (n_f / 2) + 1 \) must have the same velocities. Just because there are these characteristics, the Luh-Walker-Paul's algorithm established for manipulators with an open kinematic chain is no longer suitable for the double-feet supporting phase of anthropomorphic biped robots. For this reason, we have made modification to this algorithm so that it can be generalized to the inverse dynamics computation of anthropomorphic biped robots in the double-feet supporting phase. The modified algorithm still consists of kinematics and dynamics computations, which will be given as follows.

### IV. ON-LINE COMPUTATIONAL METHOD OF KINEMATICS FOR ANTHROPOMORPHIC BIPED ROBOTS IN DOUBLE-FEET SUPPORTING PHASE

Before starting the computation of kinematics, first we must attach an orthogonal coordinate frame to each link of the robot considered in the order of links from the base foot to the end foot by means of the Denavit-Hartenberg theory. The coordinate frames attached to the base foot and the end foot are denoted by \( O_xO_yO_z \) and \( O_{e_x}O_{e_y}O_{e_z} \), respectively.

During the single-foot supporting phase in which the robot is an open kinematic chain, by means of the Luh-Walker-Paul's algorithm the kinematics computation starts with the first link, i.e., the base foot, and then is performed recursively in the order of links until the last link. But during the double-feet supporting phase, this computation procedure cannot be realized because of the reason mentioned in the above section. To this end, we propose a modified algorithm, which consists of a forward recursive algorithm and a backward recursive algorithm. This modified algorithm is called the forward-backward recursive kinematics algorithm and given as follows.

**Step 1:** For \( i \leq n_s / 2 \) (or \( n_f / 2 \)), adopt the forward recursive algorithm.

First assuming that there is no relative movement between the base foot and the ground, i.e., the base foot does not slide on the ground, then the velocities and accelerations of the base foot are equal to zero: \( v_{b} = 0 \), \( a_{b} = 0 \), \( \omega_{b} = 0 \), \( \alpha_{b} = 0 \). Secondly, the joint displacements \( q_j \), velocities \( \dot{q}_j \), and accelerations \( \ddot{q}_j \) can be obtained from the joint planning, where \( i = 1, 2, \ldots, n_s / 2 \). Thirdly, substituting \( \dot{v}_{b}, \dot{a}_{b}, \omega_{b}, \alpha_{b} \) and \( q_j \) into formulas (1)-(4) and the motion of link \( i \) in the formulas (1) and (4) are to be equal to 0, 1, 2, ..., \( n_s / 2 \) consecutively and the \( i \) in formulas (5) and (6) are to be equal to 1, 2, ..., \( n_s / 2 \), then the angular velocity \( \omega_{i} \) and acceleration \( \alpha_{i} \), linear centroidal velocity \( v_{i} \) and acceleration \( a_{i} \) of link \( i \) can be computed in the forward recursive order of variable \( i \), where \( i = 1, 2, \ldots, n_s / 2 \).

We use the signs \( o_{c}^{i}, o_{c}^{i'}, a_{c}^{i}, a_{c}^{i'} \) to denote the corresponding velocities and accelerations of link \( n_f / 2 \) (i.e., the upper body) respectively, which are obtained in Step 1, where \( c \) in the subscripts denotes its centroid.

**Step 2:** For \( i > n_s / 2 \) (or \( n_f / 2 \)) adopt the backward recursive algorithm.

Similarly, first assuming that there is no relative movement between the end foot and the ground, then the velocities and accelerations of the end foot are equal to zero: \( v_{e} = 0 \), \( a_{e} = 0 \), \( \omega_{e} = 0 \), \( \alpha_{e} = 0 \). Secondly, the joint displacements \( q_j \), velocities \( \dot{q}_j \), and accelerations \( \ddot{q}_j \) can be obtained from the joint planning, where \( i = (n_f / 2) + 1, (n_f / 2) + 2, \ldots, n_f \). However, in this step, kinematic variables cannot be computed in the same forward recursive order as the step 1 but must be computed in the backward recursive order of links from the end foot (link \( n_f \)) to the upper body (link \( n_s / 2 \)), and the recursive formulas must be modified as follows:

\[
\begin{align*}
\dot{q}_{i+1} &= \dot{q}_{i} + \ddot{q}_{i} t_{i+1} \\
\dot{v}_{i+1} &= \dot{v}_{i} + \omega_{i} \times t_{i+1} \\
\dot{a}_{i+1} &= \dot{a}_{i} + \alpha_{i} \times t_{i+1} \\
v_{i+1} &= v_{i} + \omega_{i} \times t_{i+1} \\
a_{i+1} &= a_{i} + \alpha_{i} \times t_{i+1}
\end{align*}
\]
where \( i = n, n-1, \ldots, (n/2) + 1 \) consecutively in formulas (11)–(14), and \( i = n, n-1, \ldots, n/2 \) consecutively in formulas (15) and (16). Thirdly, substituting \( v_n, \omega_n, a_n, a_n, \dot{q}_i \), and \( \ddot{q}_i \) into formulas (11) through (16) and in the backward recursive order of variable \( i \), then the angular velocity \( \omega_i \) and acceleration \( a_i \), the linear centroidal velocity \( v_i \), and acceleration \( a_i \) of link \( i \) can be computed, where \( i = n, n-1, \ldots, n/2 \), consecutively.

We use the signs \( e_i^b \), \( e_i^p \), \( v_i^b \), \( v_i^p \), \( a_i^b \), \( a_i^p \) to denote the corresponding velocities and accelerations of the link \( n/2 \) (i.e., the upper body) respectively, which are obtained in step 2.

After the above two steps, we can complete the computation of all the kinematic variables of an anthropomorphic biped robot in the double-feet supporting phase. In this modified algorithm, one can find that the kinematic variables of the upper body (i.e., link \( n/2 \)) are computed either in step 1 or step 2, and their values are denoted by the superscripts \( \alpha \) and \( \beta \) respectively. The following criterion can be proposed.

[Criterion 1]: When an anthropomorphic biped robot is in the double-feet supporting phase, the computational values of kinematic variables of its upper body obtained from the forward recursive algorithm must be equal to those obtained from the backward recursive algorithm respectively, that is, \( e_i^\alpha = e_i^\beta \), \( e_i^\beta = e_i^\alpha \), \( v_i^\alpha = v_i^\beta \), \( v_i^\beta = v_i^\alpha \), \( a_i^\alpha = a_i^\beta \), \( a_i^\beta = a_i^\alpha \).

Criterion 1 is very important. Its direct function is to examine whether the kinematic computation for an anthropomorphic biped robot in the double-feet supporting phase is correct, whereas its implied function is to judge whether the gait planning for the robot in the double-feet supporting phase is reasonable, because the joint velocities \( \dot{q}_i \) and accelerations \( \ddot{q}_i \) obtained from the gait planning will be used in the above kinematic computation.

V. ON-LINE COMPUTATIONAL METHOD OF INVERSE DYNAMICS FOR ANTHROPOMORPHIC BIPED ROBOTS IN DOUBLE-FEET SUPPORTING PHASE

Based on the kinematic computation, now we can consider the computational method of the inverse dynamics for anthropomorphic biped robots in the double-feet supporting phase. We also propose a modified algorithm for the inverse dynamics computation. It also consists of two steps: forward recurrence and backward recurrence. We call it the forward-backward recursive dynamics algorithm and give it as follows.

Step 1: For \( i = n/2 \) (\( n = n_1 \), or \( n_2 \)), adopt the forward recursive algorithm.

In this case, first the unknown force \( f_{2n} \) and moment \( N_{2n} \) applied to the base foot (link 0) by the ground (the ground is denoted by sign \( \pm \) with reference to the base foot) can be determined based on the measurement values from the pressure sensors fixed on the sole of the base foot. Then, letting variable \( i \) be equal to \( 0 \), \( 1, \ldots, (n/2)-1 \) consecutively in the forward recursive order and using the following recursive formulas:

\[
f_{i+1} = f_{i+1} + m_i g - m_i a_i,
\]

where \( i = n, n-1, \ldots, (n/2) + 1 \) consecutively in formulas (11)–(14), and \( i = n, n-1, \ldots, (n/2) + 1 \) consecutively in formulas (15) and (16). Thirdly, substituting \( v_n, \omega_n, a_n, a_n, \dot{q}_i \), and \( \ddot{q}_i \) into formulas (11) through (16) and in the backward recursive order of variable \( i \), then the angular velocity \( \omega_i \) and acceleration \( a_i \), the linear centroidal velocity \( v_i \), and acceleration \( a_i \) of link \( i \) can be computed, where \( i = n, n-1, \ldots, n/2 \), consecutively.

We use the signs \( e_i^b \), \( e_i^p \), \( v_i^b \), \( v_i^p \), \( a_i^b \), \( a_i^p \) to denote the corresponding velocities and accelerations of the link \( n/2 \) (i.e., the upper body) respectively, which are obtained in step 2.

After the above two steps, we can complete the computation of all the kinematic variables of an anthropomorphic biped robot in the double-feet supporting phase. In this modified algorithm, one can find that the kinematic variables of the upper body (i.e., link \( n/2 \)) are computed either in step 1 or step 2, and their values are denoted by the superscripts \( \alpha \) and \( \beta \) respectively. The following criterion can be proposed.

[Criterion 1]: When an anthropomorphic biped robot is in the double-feet supporting phase, the computational values of kinematic variables of its upper body obtained from the forward recursive algorithm must be equal to those obtained from the backward recursive algorithm respectively, that is, \( e_i^\alpha = e_i^\beta \), \( e_i^\beta = e_i^\alpha \), \( v_i^\alpha = v_i^\beta \), \( v_i^\beta = v_i^\alpha \), \( a_i^\alpha = a_i^\beta \), \( a_i^\beta = a_i^\alpha \).

Criterion 1 is very important. Its direct function is to examine whether the kinematic computation for an anthropomorphic biped robot in the double-feet supporting phase is correct, whereas its implied function is to judge whether the gait planning for the robot in the double-feet supporting phase is reasonable, because the joint velocities \( \dot{q}_i \) and accelerations \( \ddot{q}_i \) obtained from the gait planning will be used in the above kinematic computation.

V. ON-LINE COMPUTATIONAL METHOD OF INVERSE DYNAMICS FOR ANTHROPOMORPHIC BIPED ROBOTS IN DOUBLE-FEET SUPPORTING PHASE

Based on the kinematic computation, now we can consider the computational method of the inverse dynamics for anthropomorphic biped robots in the double-feet supporting phase. We also propose a modified algorithm for the inverse dynamics computation. It also consists of two steps: forward recurrence and backward recurrence. We call it the forward-backward recursive dynamics algorithm and give it as follows.

Step 1: For \( i = n/2 \) (\( n = n_1 \), or \( n_2 \)), adopt the forward recursive algorithm.

In this case, first the unknown force \( f_{2n+1} \) and moment \( N_{2n+1} \) applied to the base foot (link 0) by the ground (the ground is denoted by sign \( \pm \) with reference to the base foot) can be determined based on the measurement values from the pressure sensors fixed on the sole of the base foot. Then, letting variable \( i \) be equal to \( 0 \), \( 1, \ldots, (n/2)-1 \) consecutively in the forward recursive order and using the following recursive formulas:

\[
f_{i+1} = f_{i+1} + m_i g - m_i a_i,
\]

where \( i = n, n-1, \ldots, (n/2) + 1 \) consecutively in formulas (11)–(14), and \( i = n, n-1, \ldots, (n/2) + 1 \) consecutively in formulas (15) and (16). Thirdly, substituting \( v_n, \omega_n, a_n, a_n, \dot{q}_i \), and \( \ddot{q}_i \) into formulas (11) through (16) and in the backward recursive order of variable \( i \), then the angular velocity \( \omega_i \) and acceleration \( a_i \), the linear centroidal velocity \( v_i \), and acceleration \( a_i \) of link \( i \) can be computed, where \( i = n, n-1, \ldots, n/2 \), consecutively.

We use the signs \( e_i^b \), \( e_i^p \), \( v_i^b \), \( v_i^p \), \( a_i^b \), \( a_i^p \) to denote the corresponding velocities and accelerations of the link \( n/2 \) (i.e., the upper body) respectively, which are obtained in step 2.

After the above two steps, we can complete the computation of all the kinematic variables of an anthropomorphic biped robot in the double-feet supporting phase. In this modified algorithm, one can find that the kinematic variables of the upper body (i.e., link \( n/2 \)) are computed either in step 1 or step 2, and their values are denoted by the superscripts \( \alpha \) and \( \beta \) respectively. The following criterion can be proposed.

[Criterion 1]: When an anthropomorphic biped robot is in the double-feet supporting phase, the computational values of kinematic variables of its upper body obtained from the forward recursive algorithm must be equal to those obtained from the backward recursive algorithm respectively, that is, \( e_i^\alpha = e_i^\beta \), \( e_i^\beta = e_i^\alpha \), \( v_i^\alpha = v_i^\beta \), \( v_i^\beta = v_i^\alpha \), \( a_i^\alpha = a_i^\beta \), \( a_i^\beta = a_i^\alpha \).

Criterion 1 is very important. Its direct function is to examine whether the kinematic computation for an anthropomorphic biped robot in the double-feet supporting phase is correct, whereas its implied function is to judge whether the gait planning for the robot in the double-feet supporting phase is reasonable, because the joint velocities \( \dot{q}_i \) and accelerations \( \ddot{q}_i \) obtained from the gait planning will be used in the above kinematic computation.
obtained from the forward-backward recursive kinematics algorithm, must satisfy the balance equations (23) and (24).

The direct function of the criterion 2 is to examine whether the dynamics computation for an anthropomorphic biped robot in the double-feet supporting phase is correct, while its implied function is to judge whether the ground reaction forces and moments \( f_{i,0}, -f_{i,0}, N_{i,0}, \) and \(-N_{i,0}\), which are computed from the measurement values of the pressure sensors fixed on the soles of the robot, are correct. If criterion 1 is satisfied but criterion 2 is not, that means there are mistakes in the measurements or computations of the ground reaction forces and moments. If criterion 1 and criterion 2 are all satisfied, that means the dynamics and kinetics computations for the anthropomorphic biped robot in the double-feet supporting phase are totally correct.

In addition, another method for the inverse dynamics computation of an anthropomorphic biped robot in double-feet supporting phase can be considered. This method is similar to the Luh-Worker-Paul’s algorithm and called the backward recursive dynamics algorithm. It is given as follows.

First, we can compute the forces and moments \( f_{1,0}, f_{2,0}, N_{1,0}, -f_{2,0}, -N_{2,0} \) applied to the base foot and the end food by the ground respectively according to the values measured by the pressure sensors fixed on the soles. Then, based on the results of the kinematic computation and the values of \(-f_{2,0}\) and \(-N_{2,0}\), the drive torque \( M_i \) of each joint \( i (i=1, 2, \ldots, n) \) can be computed according to the computation procedure of the Luh-Walker-Paul’s algorithm. In the last computation of this method, we can also obtain the force \( f_{0,0} \) and moment \( N_{0,0} \) applied to link 1 by the base foot (link 0). The forces \( f_{i,0}, -f_{i,0}, \) and moment \( N_{i,0}, -N_{i,0} \) should satisfy the following balance equations:

\[
\begin{align*}
 f_{1,0} - f_{0,0} + m_y g - m_y a_y &= 0 \quad (25) \\
 N_{1,0} - N_{0,0} + r_{1,0} \times f_{0,0} - r_{1,0} \times a_y (x_1, y_1) &= 0 \quad (26)
\end{align*}
\]

where \( r_{1,0} \) represents the position vector from the action point of \( f_{1,0} \) and moment \( N_{1,0} \) to the centroid \( C_0 \). The following criterion can be proposed to examine whether the dynamics computation using this method is correct.

[Criterion 3]: When an anthropomorphic biped robot is in double-feet supporting phase, the force \( -f_{0,0} \) and moment \( N_{0,0} \) applied to the base foot by link 1 and obtained according to the Luh-Walker-Paul’s algorithm, and the force \( f_{1,0} \) and moment \( N_{1,0} \) applied to the base foot by the ground and computed by means of the measurement values from the pressure sensors fixed on the sole of the base foot, should satisfy (25) and (26), in which \( \omega_y \), \( a_y \), and \( a_0 \) are the velocity and accelerations of the base foot respectively and generally are prescribed to be equal to zero.

VI. ON-LINE COMPUTATIONAL SCHEME FOR DYNAMIC WALKING OF ANTHROPOMORPHIC BIPED ROBOTS

The above-established on-line inverse dynamics computation method is called IDADFS algorithm (Inverse Dynamics Algorithm for Double-Feet Supporting), which is used in the double-feet supporting phase of anthropomorphic biped robots. Fig.3 shows its computational structure (see next page). The structure consists of the forward-backward recursive kinematics algorithm and forward-backward recursive dynamics algorithm.

Combining the IDADFS algorithm with the Luh-Walker-Paul’s algorithm can constitute a complete on-line control algorithm for controlling the dynamic walking of anthropomorphic biped robots. In the concrete, we can directly use the Luh-Walker-Paul’s algorithm to compute the drive torque of each joint at each sampling instant when an anthropomorphic biped robot is in the single-foot supporting phase, while use the IDADFS algorithm to compute the drive torque of each joint at each sampling instant when the robot is in the double-feet supporting phase. The switch time between the two supporting phases can be detected using the pressure sensors or the micro-switches fixed on the soles of the robot’s feet, and this time can be used to realize the switch between the two algorithms. Thus, the accurate real-time dynamic control can be executed for the dynamic walking of an anthropomorphic biped robot.

VII. SUMMARY AND CONCLUSIONS

In this paper, through a deep-going analysis and study on the characteristics of dynamic walking of anthropomorphic biped robots and based on the Luh-Walker-Paul’s algorithm, a more effective on-line computational scheme for the real-time control of dynamic walking of this kind of biped robots is established. This scheme includes two algorithms, one is the Luh-Walker-Paul’s algorithm used in the single-foot supporting phase, and another is the IDADFS algorithm used in the double-feet supporting phase. By means of the new on-line computational scheme, we can execute an accurate real-time dynamic control for the dynamic walking of an anthropomorphic biped robot. It is worthy to be pointed out that the research result of this paper can also be spread to multilegged robots. The more important is that the IDADFS algorithm can be generalized to the manipulators with an open kinematic chain structure to realize the real-time dynamic control over this kind of manipulators when they perform various kinds of closed-chain operating tasks or multitask coordinative manipulations.

VIII. REFERENCES


Proceedings of the 4th World Congress on Intelligent Control and Automation June 10-14, 2002, Shanghai, P.R.China


---

Fig.3 The structure of the IDADFS algorithm for anthropomorphic biped robots