Research Article

INTERNATIONAL JOURNAL OF ADVANCED ROBOTIC SYSTEMS

International Journal of Advanced Robotic Systems
January-February 2017: 1–13

DOI: 10.1177/1729881416687112
journals.sagepub.com/home/arx
SAGE

Modeling and PDC fuzzy control of planar parallel robot: A differential–algebraic equations approach

Benyamine Allouche, Antoine Dequidt, Laurent Vermeiren and Michel Dambrine

Abstract

Many works in the literature have studied the kinematical and dynamical issues of parallel robots. But it is still difficult to extend the vast control strategies to parallel mechanisms due to the complexity of the model-based control. This complexity is mainly caused by the presence of multiple closed kinematic chains, making the system naturally described by a set of differential–algebraic equations. The aim of this work is to control a two-degree-of-freedom parallel manipulator. A mechanical model based on differential–algebraic equations is given. The goal is to use the structural characteristics of the mechanical system to reduce the complexity of the nonlinear model. Therefore, a trajectory tracking control is achieved using the Takagi-Sugeno fuzzy model derived from the differential–algebraic equation forms and its linear matrix inequality constraints formulation. Simulation results show that the proposed approach based on differential–algebraic equations and Takagi-Sugeno fuzzy modeling leads to a better robustness against the structural uncertainties.

Keywords

Parallel manipulator, Takagi-Sugeno model, parallel distributed compensation, differential–algebraic equations, LMI

Date received: 4 March 2016; accepted: 18 November 2016

Topic: Robot Manipulation and Control
Topic Editor: Andrey V Savkin
Associate Editor: Jayantha Katupitiya

Introduction

In modern societies, manipulators have become a key tool in industry due to their great versatility in repetitive task. The mechanical architectures traditionally used in automated production line are based on serial robots. These robots are made of a sequence of rigid links serially assembled through revolute and/or prismatic active joints.[1] In other words, the end effector is connected to the base via a single open-loop kinematic chain which gives them a large work space and high dexterity.[2] Despite these advantages, serial robot suffer from high inertia due to the position of actuators, which are located on the moving part, they also suffer from a low payload to weight ratio and a low precision due to the cumulative joint errors and link deflection. Owing to the drawback of serial robot, parallel manipulators have taken a great interest in many applications, such as high-speed machining, assembly and packaging task, flight simulators, and various medical and space applications.[3–7] Parallel robots are generally characterized by their nonanthropomorphic shape, they are composed of an end effector connected to a base by at least two separate and independent kinematic chain.[8] This multiple kinematic chain provides a high rigidity and agility with a high payload to weight ratio due to the deportation

Université de Valenciennes et du Hainaut Cambrésis, Valenciennes, France

Corresponding author:
Benyamine Allouche, LAMIH UMR CNRS 8201 / UVHC / Le Mont-Houy, Bât. Jonas-110.
Email: benyamine.allouche@univ-valenciennes.fr

of the actuators to the base and the distribution of the load between different chain.[9] However, they suffer from some drawbacks such as limited work space, abundance of singularities,[10] and complex dynamic model caused by the presence of multiple closed kinematic chains (CKCs).[11]

Many works discussed the kinematic or the dynamic issues of CKCs.[9,12] But it is still relatively difficult to extend the vast control theory developed for serial manipulator to the parallel one, and this difficulty is due to the complexity of the model-based control.[13] This complexity comes from the fact that parallel robot are described by a set of differential–algebraic equations (DAEs) of index-3.[14] This differential index represents the number of time differentiating the constraint equations to obtain a set of ordinary differential equations (ODEs).[15] The main difficulty here is the high differential index (index > 2).[16,17] This makes the DAEs not explicitly expressed in the state–space representation which is a suitable form for most of the control strategies.

In the case where the model of the robot is given on the ODEs form, the simplest way for the control is the famous linear single-axis proportional integral derivative (PID) controller in the joints space.[8,18–22] This method is well-known for its effectiveness under the assumption of locally linear dynamics. It is verified only for low-speed control and cannot be efficient over the whole work space with a single tuning.[23] An equivalent version of linear single-axis control in the Cartesian space is given in the works by Callegari and colleagues.[24–27] In this case many simplifications are assumed that leads to a lack of accuracy and stability.[27] Another famous approach based on feedback linearization is the computed torque control (CTC).[28] This technique is widely spread in serial robotics[8,29,30] and has been implemented on several parallel platforms.[23,31] From this, the literature on the control of parallel robot based on ODE models is very large and mainly depends on the complexity of the studied manipulator and the targeted control strategy such as the sliding mode control,[32,33] the H^{∞} control,[34,35] the robust TS control,[3] the adaptive control,[36,37] the passivity-based approach,[38] and other variances.[10,23,39–43] Despite this nonexhaustive scope, models of CKCs are still complex and highly nonlinear in most cases which makes the guarantee of stability in the Lyapunov sense quite difficult.

The aim of this work is to deal with the parallel robot on its DAE form, the difficulty here is that the DAE forms are not expressed explicitly in state–space representation.[44] From this, in the second section a DAE model is presented, then the ODE model is deduced to show how complex the ODE model could be. After that, a technique based on the input–output linearization (IOL) is used to write the DAEs model on the state–space representations with an algebraic equation helping to compute the nonlinear parameters that represent the internal forces of the robots. The originality here is that by computing the internal forces the robot can be fully described by two decoupled submodel and then, depending on the complexity of those models, the robot can be controlled either in the joints space or in the operational space. In the third section, a brief description of the Takagi-Sugeno (TS) modeling and control is given. The TS modeling is applied on the legs model of 2-degree-of-freedom (DOF) parallel manipulator by transforming the internal forces as premises for the control law. The interest of bonding the internal forces is to make the system naturally robust against the structured uncertainties just by solving some simple linear matrix inequality (LMI). In the last sections, a comparison between PDC and CTC control laws is presented and some concluding remarks are given.

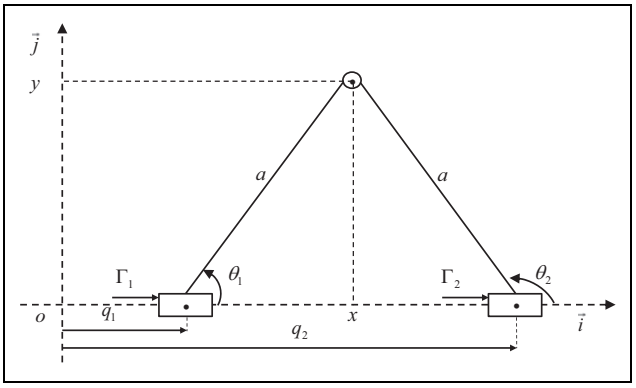

Figure 1. Kinematic diagram of Biglide robot.

Mechanical modeling of a 2-DOF parallel manipulator

DAE model of Biglide

The Biglide is a 2-DOF planar parallel manipulator[31,45] (Figure 1). This parallel kinematic machine consists of two bars mounted through a joint on two separate sliding blocks (active prismatic joints), the extremity of each bar is connected to an end effector, thereby forming a CKC. This configuration allows the positioning of the end effector (operational coordinates: $X = [x \quad y]^T$) at a specified point in the operational space by controlling the position $q = [q_1 \quad q_2]^T$ (active joint coordinates) of the prismatic joints.

The kinematic analysis of the Biglide leads to the following constraint of loop closure

$$\phi(q, X) = 0 \quad \text{with} \quad \phi(q, X) = \begin{pmatrix} (x - q_1)^2 + y^2 - a^2 \\ (q_2 - x)^2 + y^2 - a^2 \end{pmatrix} \tag{1}$$

The inverse kinematics can thereby be expressed as

$$q = g(X) \quad \text{with} \quad g(X) = \begin{pmatrix} x - C(y) \\ x + C(y) \end{pmatrix}, \; C(y) = \sqrt{a^2 - y^2} \tag{2}$$

Thanks to its simple mechanism, the forward kinematics of the Biglide can be easily derived from equation (2)

$$[x \quad y]^T = \left[\frac{q_1+q_2}{2} \sqrt{a^2 - \frac{(q_1-q_2)^2}{4}}\right]^T \tag{3}$$

Under the assumption that the mechanical system satisfies the loop-closure constraint, the kinematic relationship between end effector velocities and active joints velocities is computed by differentiating the constraint (1) with respect to time, and this relationship is conveniently described by two Jacobian matrices $J_x(X,q)$ and $J_q(X,q)$[9] as

$$J_x(X,q)\dot{X} = J_q(X,q)\dot{q} \tag{4}$$

with $J_x(X,q) = \begin{bmatrix} x-q_1 & y \\ x-q_2 & y \end{bmatrix}, J_q(X,q) = \begin{bmatrix} x-q_1 & 0 \\ 0 & x-q_2 \end{bmatrix}$

In order to derive motion equations of the Biglide, the structural characteristics of the robot are exploited to write a model based on two independent subsystems: the end effector dynamics and the open-kinematic chains dynamics. The actuator forces (robot inputs) are denoted by $\Gamma = [\Gamma_1 \quad \Gamma_2]^T$ (Figure 1). The passive joint coordinates are denoted by $\theta = [\theta_1 \quad \theta_2]^T$. The robot is subject to gravity $\vec{g} = -g\vec{j}$ with $\|\vec{j}\| = 1$. The active and passive joint coordinates of each link are denoted by $\rho_i = [q_i \quad \theta_i]^T$, $i = 1,2$. The dynamics of each leg are denoted by H_i, while the dynamic of the end effector is denoted byF_P. Thereby, the dynamics can be easily obtained from different formalisms such as Newton–Euler equations, Lagrange equations, or virtual work[8] as

$$H_i = M_{\rho_i}(\rho_i)\ddot{\rho}_i + N_{\rho_i}(\rho_i,\dot{\rho}_i)\dot{\rho}_i + G_{\rho_i}(\rho_i)\rho_i \tag{5}$$

$$F_p = M_x(X)\ddot{X} + G_x(X)X \tag{6}$$

where $M_{\rho_i}(\rho_i) = \begin{bmatrix} m_i & -ms_i\sin\theta_i \\ -ms_i\sin\theta_i & J_i \end{bmatrix}$ and $M_x(X) = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix}$ represent the inertia matrices

$N_{\rho_i}(\rho_i,\dot{\rho}_i) = \begin{bmatrix} b_i & ms_i\cos\theta_i\dot{\theta}_i \\ 0 & 0 \end{bmatrix}$ represents the centrifugal and Coriolis matrix

$G_{\rho_i}(\rho_i) = \begin{bmatrix} 0 & 0 \\ 0 & ms_i g\cos\theta_i/\theta_i \end{bmatrix}$ and $G_x(X) = \begin{bmatrix} 0 & 0 \\ 0 & mg/y \end{bmatrix}$ represent the gravity matrices.

With m, m_i, ms_i, J_i, $i = 1,2$ are the inertial parameters of the robot, which are the end effector mass, the mass, and the first and second moments of link i related to revolute joint axis on the sliding blocks 1 and 2, respectively.

From the kinematic constraint it is easy to write passive joints coordinate functions of operational coordinates.

$$\theta = \mu(X) \quad \text{where} \quad \mu(X) = \tan^{-1}\left(\frac{C(y)}{y}\right)[1 \quad -1]^T \tag{7}$$

Thereby, the velocities of passive joints can be expressed as

$$\dot{\theta} = K_x(X)\dot{X} \quad \text{where} \quad K_x(X) = \begin{bmatrix} 0 & \frac{1}{C(y)} \\ 0 & \frac{-1}{C(y)} \end{bmatrix} \tag{8}$$

Using the equations (7) and(8), the dynamic of the open-kinematic chain H_i, $i = \{1,2\}$ given by equation (5) are reordered into active part H_q and passive part H_θ such that

$$\begin{bmatrix} H_q \\ H_\theta \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}\begin{bmatrix} \ddot{q} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} N_{11} & N_{12} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \dot{q} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & G_{22} \end{bmatrix}\begin{bmatrix} q \\ \theta \end{bmatrix} \tag{9}$$

with $M_{11} = \mathrm{diag}(m_1, m_2)$, $M_{12} = M_{21} = diag(-ms_1 sin\theta_1, -ms_2 sin\theta_2)$, $M_{22} = diag(J_1, J_2)$, $N_{11} = diag(b_1, b_2)$, $N_{12} = diag(ms_1 cos\theta_1\dot{\theta}_1, ms_2 cos\theta_2\dot{\theta}_2)$, $G_{22} = diag(ms_1 g\cos\theta_1/\theta_1, ms_2 g\cos\theta_2/\theta_2)$.

The virtual power P_r^* of the overall dynamic of the robot given by the equations 6 and 9[11,46] is first written with a dependent virtual velocity field $(\dot{q}^{*T}, \dot{\theta}^{*T}, \dot{X}^{*T})$ such that

$$P_r^* = \dot{q}^{*T}(H_q - \Gamma) + \dot{X}^{*T}F_p + \dot{\theta}^{*T}H_\theta = 0 \tag{10}$$

Using equation (8), the virtual passive velocities is eliminated with $\dot{\theta}^{*T} = \dot{X}^{*T}K_x^T$, leading to the following form

$$P_r^* = \dot{q}^{*T}(H_q - \Gamma) + \dot{X}^{*T}(F_p + K_x^T H_\theta) = 0 \tag{11}$$

The new virtual velocities field $(\dot{q}^{*T}, \dot{X}^{*T})$ is still dependent. Therefore, we introduce the Lagrange multipliers $\lambda = [\lambda_1 \quad \lambda_2]^T$ such that, the virtual power of the constraint (1) is expressed with equation (4) as follows

$$P_c^* = (\dot{q}^{*T}J_q^T - \dot{X}^{*T}J_x^T)\lambda = 0, \quad \forall\lambda = [\lambda_1 \quad \lambda_2]^T \in \mathbb{R}^2 \tag{12}$$

Finally, the virtual power with the Lagrange multipliers is expressed as follows

$$P^* = P_r^* + P_c^* = \dot{q}^{*T}(H_q + J_q^T\lambda - \Gamma) + \dot{X}^{*T}(F_p + K_x^T H_\theta - J_x^T\lambda) = 0 \tag{13}$$

Given that the Lagrange multipliers are unknown, the virtual velocities field $(\dot{q}^{*T}, \dot{X}^{*T})$ may be considered as independent and then the index-1 DAEs model of the constrained system in terms of (q, X) is derived from equation (13) as

$$P^* = 0 \quad \forall[\dot{q}^{*T}\dot{X}^{*T}]^T \in \mathbb{R}^4 \Rightarrow \begin{cases} M'_{11}\ddot{q} + M'_{12}\ddot{X} + N'_{11}\dot{q} + N'_{12}\dot{X} + J_q^T\lambda = \Gamma \\ M'_{21}\ddot{q} + M'_{22}\ddot{X} + N'_{22}\dot{X} + G'_{22}X - J_x^T\lambda = 0 \\ J_q\dot{q} - J_x\dot{X} = 0 \end{cases} \tag{14}$$

with $M'_{11} = M_{11}$, $M'_{12} = M_{12}K_x$, $M'_{21} = K_x^T M_{21}$, $M'_{22} = M_x + K_x^T M_{22} K_x$,

$N'_{11} = N_{11}$, $N'_{12} = M_{12}\dot{K}_x + N_{12}K_x$, $N'_{22} = K_x^T M_{22}\dot{K}_x + K_x^T N_{22} K_x$, $G'_{22} = G_x + K_x^T G_{22}\mu$.

Input–output linearization

The DAEs model given by equation (14), presents the coupled dynamics between the legs and the end effector. The algebraic variables (Lagrange multipliers λ_i) represent the internal forces between the legs and the end effector. The assembly of the two subsystems is ensured through the algebraic constraint. This model is singular, the difficulty here is that DAEs are not expressed explicitly in state–space representation.[44] From this, two main approaches have been developed. The first one is the approximation of the DAE model to a singularly perturbed model.[47] The idea is to substitute the constraint equation with a fast dynamics representing the violation of the constraint.[44,48–50] From a practical point of view, this approach is completely justified because the connections between joints and links are elastic.[51] This method helps to relax the complexity of the nonlinear model through the additional artificial dynamics, but at the same time it extends the state vector. This leads to an increase in the complexity of the control applications based on the resolution of the LMI problem. The second technique is the IOL,[52] and this approach consists of solving the algebraic equation by differentiating the constraint until the appearance of the algebraic variables. The solution is used in the differential equation of the DAE system and a state–space model-based control is given. The differentiation of the constraint (14) is written as

$$J_q\ddot{q} - J_x\ddot{X} + \dot{J}_q\dot{q} - \dot{J}_x\dot{X} = 0 \tag{15}$$

By combining (15) with the DAEs model (14), a redundant ODEs model is obtained:

$$\begin{cases} M_q\ddot{q} + N_q\dot{q} + J_q^T\lambda = \Gamma \\ M'_x\ddot{X} + N'_x\dot{X} + G'_xX - J_x^T\lambda = 0 \\ J_q\ddot{q} - J_x\ddot{X} + \dot{J}_q\dot{q} - \dot{J}_x\dot{X} = 0 \end{cases} \tag{16}$$

with $M_q = M'_{11} + M'_{12}J_x^{-1}J_q$, $M'_x = M'_{22} + M'_{21}J_q^{-1}J_x$,

$N_q = N'_{11} + N'_{12}\ J_x^{-1}J_q + M'_{12}J_x^{-1}\dot{J}_q - M'_{12}J_x^{-1}\dot{J}_xJ_x^{-1}J_q$, $N'_x = N'_{22} + M'_{21}J_q^{-1}\dot{J}_x - M'_{21}J_q^{-1}\dot{J}_qJ_q^{-1}J_x$, $G'_x = G'_{22}$.

From system of equations (16) and by placing the differential equations into the algebraic one, the Biglide parallel robot can be fully described by two different subsystems as follows

$$\text{Legs model} \begin{cases} M_q\ddot{q} + N_q\dot{q} + J_q^T\lambda = \Gamma \\ \mathcal{P}\lambda + \mathcal{L}\dot{q} - \mathcal{Q}q - \mathcal{W}\Gamma = 0 \end{cases} \tag{17}$$

$$\text{Endeffector model} \begin{cases} M'_x\ddot{X} + N'_x\dot{X} + G'_xX - J_x^T\lambda = 0 \\ \mathcal{P}\lambda + \mathcal{L}'\dot{X} - \mathcal{Q}'X - \mathcal{W}\Gamma = 0 \end{cases} \tag{18}$$

with $\mathcal{P} = J_qM_q^{-1}J_q^T + J_xM_x'^{-1}J_x^T$, $\mathcal{L} = J_qM_q^{-1}N_q - \dot{J}_q + \dot{J}_xJ_x^{-1}J_q - J_xM_x^{-1}N'_xJ_x^{-1}J_q$, $\mathcal{L}' = \mathcal{L}J_q^{-1}J_x$, $\mathcal{Q}' = J_xM_x^{-1}G'_x$, $\mathcal{Q} = \mathcal{Q}'g^{-1}$, $\mathcal{W} = J_qM_q^{-1}$.

This is the first main results of this article, the Biglide robot can be controlled by two different approaches depending on the targeted control law. The first approach is to only use the legs model equation (17). In this case, the internal forces λ are seen as nonlinear terms to be compensated by the controller. The second approach is to only consider the end effector model (18). In this case, the internal forces of the system are seen as a new input control defined by the algebraic equation.

ODEs model of Biglide

In order to get the ODEs model of Biglide in operational space, the virtual active joint velocities are eliminated from the virtual power equation (11) using the equation (4), $\dot{q}^{*T} = \dot{X}^{*T}J_x^TJ_q^{-T}$, the virtual power can be rewritten with the independent velocities $\dot{X}^*$ as

$$P_r^* = \dot{X}^{*T}\left(J_x^TJ_q^{-T}(H_q - \Gamma) + F_p + K_x^TH_\theta\right) = 0 \quad \forall\dot{X}^* \in \mathbb{R}^2 \tag{19}$$

Therefore, the ODEs model can be obtained from equation (19) as $\Gamma = H_q + J_q^TJ_x^{-T}(F_p + K_x^TH_\theta)$. Then to eliminate $\ddot{\theta}$ and $\ddot{q}$, the first derivative of equations (19) and (8) are used. Finally the ODE model of Biglide in operational space is given as

$$\Gamma = M(X)\ddot{X} + N(X,\dot{X})\dot{X} + G(X)X \tag{20}$$

with $M(X) = \begin{bmatrix} m_1 + \frac{1}{2}(m - \eta_1 + \eta_2) & f_1(X) \\ m_2 + \frac{1}{2}(m - \eta_2 + \eta_1) & f_2(X) \end{bmatrix}$, $N(X,\dot{X}) = \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix}$, $G(X) = \begin{bmatrix} 0 & gC(y)(m + \eta_1 + \eta_2)/2y^2 \\ 0 & -gC(y)(m + \eta_1 + \eta_2)/2y^2 \end{bmatrix}$, $f_1(X) = [(2m_1 - 3\eta_1 - \eta_2)y^2 + mC(y)^2 + J_1 + J_2]/2yC(y)$, $f_2(X) = -[(2m_2 - 3\eta_2 - \eta_1)\quad y^2 + mC(y)^2 + J_1 + J_2]/2yC(y)$, $\eta_{1,2} = ms_{1,2}/a$, $n_{11} = b_1$, $n_{21} = b_2$, $n_{12} = b_1y/C(y) - [(2m_1 - 3\eta_1 - \eta_2)y^2 + (2m_1 - 3\eta_1 - \eta_2)C(y)^2 + J_1 + J_2]\dot{y}/(2C(y)^3$, $n_{22} = -b_2y/C(y) + [(2m_2 - 3\eta_2 - \eta_1)y^2 + (2m_2 - 3\eta_2 - \eta_1)C(y)^2 + J_1 + J_2]\dot{y}/(2C(y)^3$.

In a more general case, it is not obvious to get the ODEs model of parallel robot in operational space. To get the model in the active joint space, the same logic is followed by eliminating the virtual operational velocities. Note that the ODEs model is given here just for a comparison purpose with the DAEs one. Unlike the DAEs model, where most of the nonlinear terms are hidden into the algebraic equation, the ODEs model presents a large number of nonlinear terms. For this reason, we thought

that it will be better to use the DAEs model to design nonlinear controllers.

PDC fuzzy control

TS modeling

The TS fuzzy model is a mathematical representation of systems, it belongs to the quasi LPV family.[53] Inside a compact set of state variables, TS fuzzy model can represent exactly a nonlinear system by a collection of linear models weighted together by a nonlinear function issued from nonlinearities of the system.[54] The conditions of stability and stabilization of these models are generally based on the Lyapunov theory.[55] The advantage of this representation is that it provides a systematic framework for designing control laws through the LMI constrain formulation.[56]

Let's consider the following nonlinear plant

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \tag{21}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the control input vector A, B, and C are nonlinear matrices of proper dimension.

Let r be the number of nonlinear terms of the system (21). A TS model is viewed as a convex sum of linear models via membership functions (MFs), it is defined by «*If*... *Then*» rules representing the local linear input–output relations of the plant.[53] From this, the nonlinear system (21) can be written on continuous-time TS fuzzy model as follows[56]

$$\begin{cases} \dot{x}(t) = \sum_{i=1}^{r} h_i(z(t))[A_i x(t) + B_i u(t)] \\ y(t) = \sum_{i=1}^{r} h_i(z(t)) C_i x(t) \end{cases} \tag{22}$$

where *z(t)* is the premise vector (vector of nonlinear terms of the model). It may depend on the state, input, exogenous parameters, or time, on measurable and/or unmeasurable variables. Matrices (A_i, B_i), $i \in \{1, 2, ..., r\}$ represent the ith linear model of the TS form (22). The MFs $h_i(z(t)) \geq 0$, $i \in \{1, ..., r\}$ are nonlinear functions and satisfy the convex property as follows $\sum_{i=1}^{r} h_i(z(t)) = 1$

TS model of the legs

For the following, let's consider the dynamical model of the Biglide given by equation (17). In order to decompose the internal forces λ into a convex sum, the term $J_q^T \lambda$ is written as

$$J_q^T \lambda = \psi(\lambda) q \quad \text{with} \quad \psi(\lambda) = \begin{bmatrix} \frac{-\lambda_1}{2} & \frac{\lambda_1}{2} \\ \frac{\lambda_2}{2} & \frac{-\lambda_2}{2} \end{bmatrix} \tag{23}$$

Then, the state–space representation of the Biglide robot in term of joint variables $(q, \dot{q})$ is given as

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & M_q^{-1} \end{bmatrix} \left(\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\lambda_1}{2} & \frac{-\lambda_1}{2} & -b_1 & 0 \\ \frac{-\lambda_2}{2} & \frac{\lambda_2}{2} & 0 & -b_2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \Gamma \right) \tag{24}$$

where $I \in \mathbb{R}^{2\times 2}$ is an identity matrix.

From the state representation (24), the TS fuzzy model of the Biglide can be obtained by considering the following two nonlinearities $\lambda_1(\cdot)$ and $\lambda_2(\cdot)$. From this it yields a 4-rule (2^r) TS fuzzy model. Thanks to the DAE model, the number of local models have been reduced from 16-rule TS descriptor model[3] to a set of 4-rule classical TS models. Note that, $\lambda_i(\cdot)$, $i = \{1, 2\}$ are computed via the algebraic equation (17). A small delay ε_t is introduced to break the algebraic loop. This delay can be justified by supposing that the dynamics of the system are slow enough to not react between t and $(t + \varepsilon_t)$. It is also known that the control inputs are subject to saturation $|\Gamma_{1,2}| \leq \Gamma_{\max}$, the working space is restrained with the range of $q_i \in [\underline{q_i}, \bar{q}_i]$, $i = \{1, 2\}$, and the velocities are clamped $|\dot{q}_{1,2}| \leq \dot{q}_{\max}$. From this consideration, the premises $\lambda_i(\cdot)$, $i = \{1, 2\}$ can be bounded as $\lambda_i(\cdot) \in [\underline{\lambda_i}\ \bar{\lambda}_i]$. The linear submodels are obtained using the sector nonlinearity transformation[56] by writing λ as

$$\lambda(.) = w_0^i(.) \cdot \underline{\lambda_i} + w_1^i(.) \cdot \bar{\lambda}_i \tag{25}$$

with $w_0^i(.) = (\bar{\lambda}_i - \lambda(.))/(\bar{\lambda}_i - \underline{\lambda_i})$, $w_1^i(.) = (\lambda(.) - \underline{\lambda_i})/(\bar{\lambda}_i - \underline{\lambda_i})$. Note that the functions $h_i(z(t)) \geq 0$ are expressed as products between the w_j^i, $i \in \{1, ..., r\}$, $j \in \{1, 0\}$.

Controller design

For the stabilization of system (22), a parallel distributed compensation (PDC) control law is proposed[57]

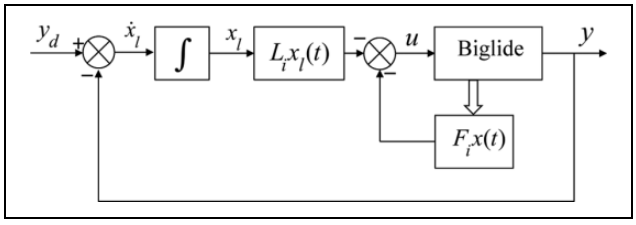

Figure 2. Extended PDC control scheme.

$$u(t) = \sum_{i=1}^{r} h_i\Big(z(t)\Big) F_i x(t) \tag{26}$$

where the matrices $F_i \in \mathbb{R}^{m\times n}$ need to be fixed to ensure the stabilization of the system.[58]

Theorem 1: The fuzzy TS model (22) with the PDC control law (26) is asymptotically stable if there exist a matrix $P = P^T > 0$ such as[56,59]

$$\begin{cases} Y_{ii} < 0 & \forall i \in \{1, ..., r\} \\ \dfrac{1}{r-1} Y_{ii} + \dfrac{1}{2}(Y_{ij} + Y_{ji}) < 0 & \forall i, j \in \{1, ..., r\}\ i < j \end{cases} \tag{27}$$

with $Y_{ij} = A_i P + P A_i^T - B_i M_j - M_j^T B_i^T$. If the LMI problem is feasible, the PDC controller gain matrices are given by $F_i = M_i P^{-1}$. □

An integral part is added (Figure 2) in order to compensate the stationary error. Consider the extended state vector $\bar{x}^T = [x^T \quad x_l^T]$, then[3]

$$\dot{x}_l = y_d - \sum_{i=1}^{r} h_i(z(t)) C_i \, \mathrm{x}(t) \tag{28}$$

where y_d is the desired input vector. The extended system is expressed as

$$\begin{cases} \dot{\bar{x}}(t) = \sum_{i=1}^{r} h_i(z(t))[\bar{A}_i \bar{x}(t) + \bar{B}_i u(t)] + B_0 y_d(t) \\ y(t) = \sum_{i=1}^{r} h_i(z(t)) \bar{C}_i \bar{x}(t) \end{cases} \tag{29}$$

with $\bar{A}_i = \begin{bmatrix} A_i & 0 \\ -C_i & 0 \end{bmatrix}$, $\bar{B}_i = \begin{bmatrix} B_i \\ 0 \end{bmatrix}$, $\bar{C}_i = [C_i \quad 0]$, $B_0 = \begin{bmatrix} 0 \\ I \end{bmatrix}$; where 0 and I are matrices of proper dimension. Therefore, the extended PDC control law can be written as

$$u(t) = \sum_{i=1}^{r} h_i(z(t)) \bar{F}_i \bar{x}(t) \tag{30}$$

with the extended gain vector $\bar{F}_i = [F_i \quad L_i]$.

Simulation results

The model of the parallel robot used for numerical simulations includes structured and unstructured uncertainties (Appendix 1). The structured uncertainties represent the variation in the end-effector mass, this variation is given by $\Delta m \in [0\ 0.816]\ kg$. To simulate a realistic behavior, The numerical model includes unmodeled dynamics such as elasticity between joints[60] and the Stribeck friction[61] (Appendix 2). Furthermore, two resonant modes are added to simulate the elastic joints, the lower value of the resonant frequency is $\omega \simeq 29\ rad/s$. To compare, a CTC controller is designed. It is a well-known approach for serial robotics.[8,29,30] It has also been implemented on several platforms of parallel kinematic machines.[23,31,62] Usually, the tuning of the CTC controller with a PID control action uses a pole placement technique for robot manipulators.[23] The gains are adjusted in order to get a negative real triple pole with a frequency less than the half of the lower resonant frequency.[8] The suitable value is used for simulation. For the TS approach, the PDC controller was tuned in the same manner as the CTC with a convex optimization algorithm via the LMI solver of Matlab LMI Toolbox.[63]

In order to get the TS numerical model, the joint positions were bounded in a manner to avoid singularities and to cover a large area of the working space $q_1 \in [54.5,\ 89.5]$ mm and $q_2 \in [129.5,\ 164.5]$ mm. The velocity of each joint was bounded as $|\dot{q}_{1,2}| \leq 428$ mm/ s. The control inputs was saturated as $|\Gamma_{1,2}| \leq 50$ N. The internal forces λ were bounded using the algebraic equation (17). To guarantee the robustness against the structured uncertainty, the internal forces were bounded in the worst case (where the payload is equal to the global mass of the robot).

For the simulation, we chose a circular trajectory in the operational space. The response of both controllers, CTC and TS, are depicted in Figures 3 and 4 for the model without payload ($\Delta m = 0$ kg), Figures 5 and 6 for a payload representing 50% of the global mass ($\Delta m = 0.816$ kg), and Figures 7 and 8 for a payload representing 100% of the global mass ($\Delta m = 1.632$ kg).

In the case where the robot does not carry any payload, both control laws ensure a good tracking with a slight advantage to the TS control law, so we can say that the TS and CTC are robust against the unmodeled dynamics (stick/slip friction). Moreover, from Figure 3 (c and d) we can note that the control inputs are almost similar, this indicates that the two controllers were tuned in a manner to get similar performances in the nominal case. In the second case, ($\Delta m = 0.816$ kg), the TS control law ensures good tracking performances, while the CTC controller loses the desired path but the robot remains stable. In the latter case, ($\Delta m = 1.632$ kg), the TS-based control law presents the best results exhibiting its robustness according

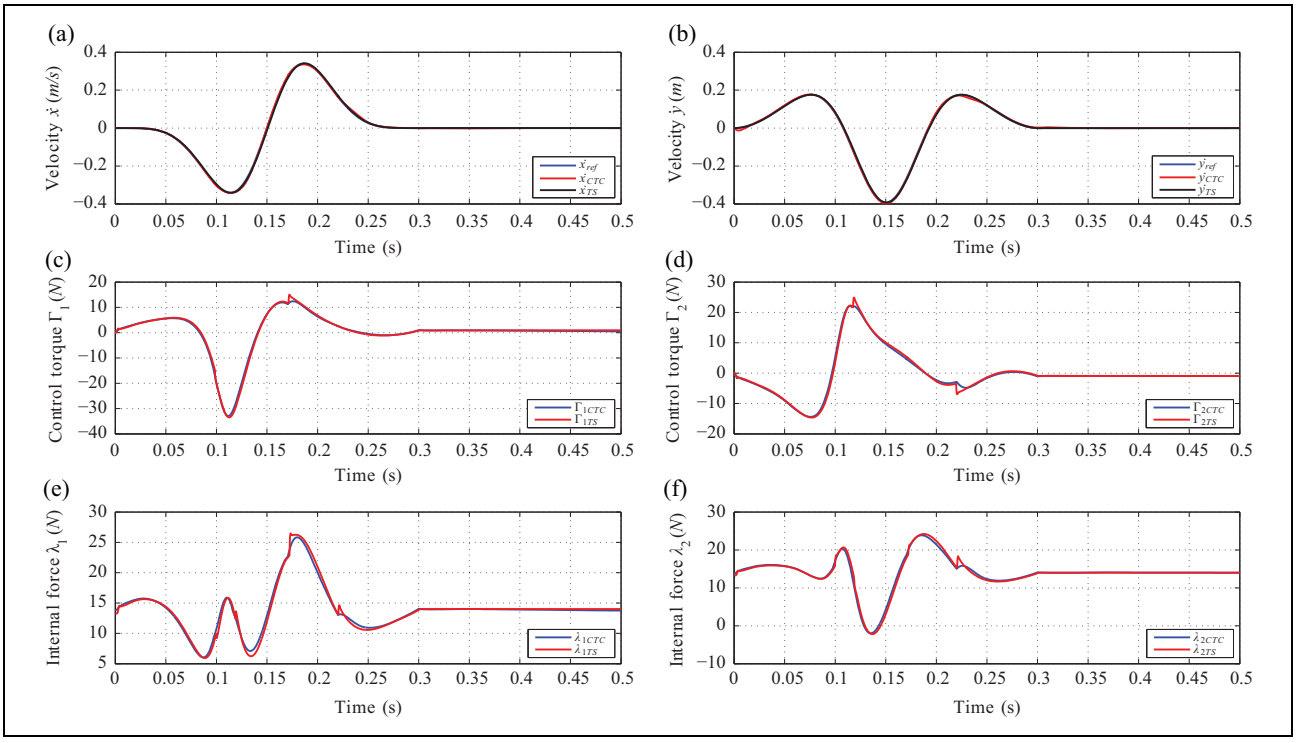

Figure 3. Simulation results for $\Delta m = 0\,kg$: (a) velocity profile of $\dot{x}$, (b) velocity profile of $\dot{y}$, (c) torque profile of joint 1, (d) torque profile of joint 2, (e) internal force profile of leg 1, and (f) internal force profile of leg 2.

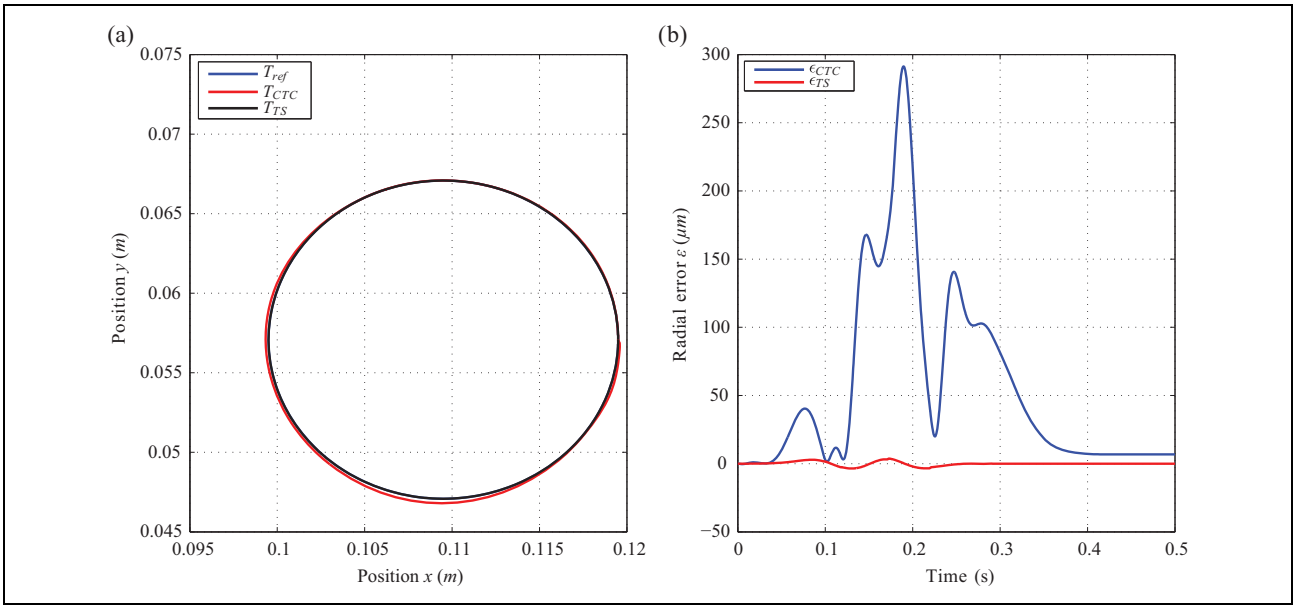

Figure 4. Simulation results for $\Delta m = 0\,kg$: (a) trajectory profile and (b) radial error.

to structured uncertainties. The CTC, as expected, presents an unstable trajectory.

Finally, two well-known criteria are computed over the simulation time ($T = 0.5$ s) in order to quantify the behavior of both controllers (Table 1): the first criterion is the integral of absolute error: $J_{IAE} = \int_0^T |\varepsilon| dt$. where $\varepsilon(t)$ is the radial error. The second criterion is the integral of square value of the control: $J_{ISV} = \int_0^T \sum_{i=1}^{2} \Gamma_i(t)^2 dt$.

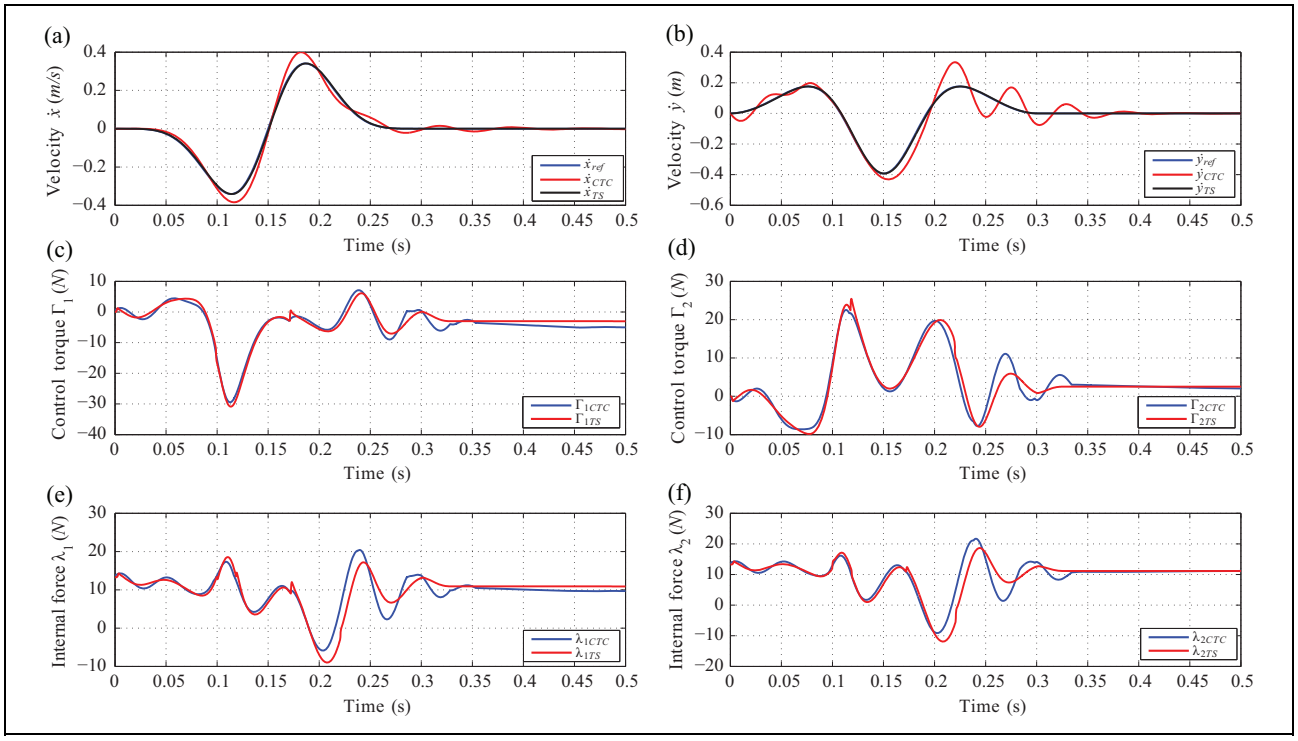

Figure 5. Simulation results for $\Delta m = 0.816\,kg$: (a) velocity profile of x, (b) velocity profile of y, (c) torque profile of joint 1, (d) torque profile of joint 2, (e) internal force profile of leg 1, and (f) internal force profile of leg 2.

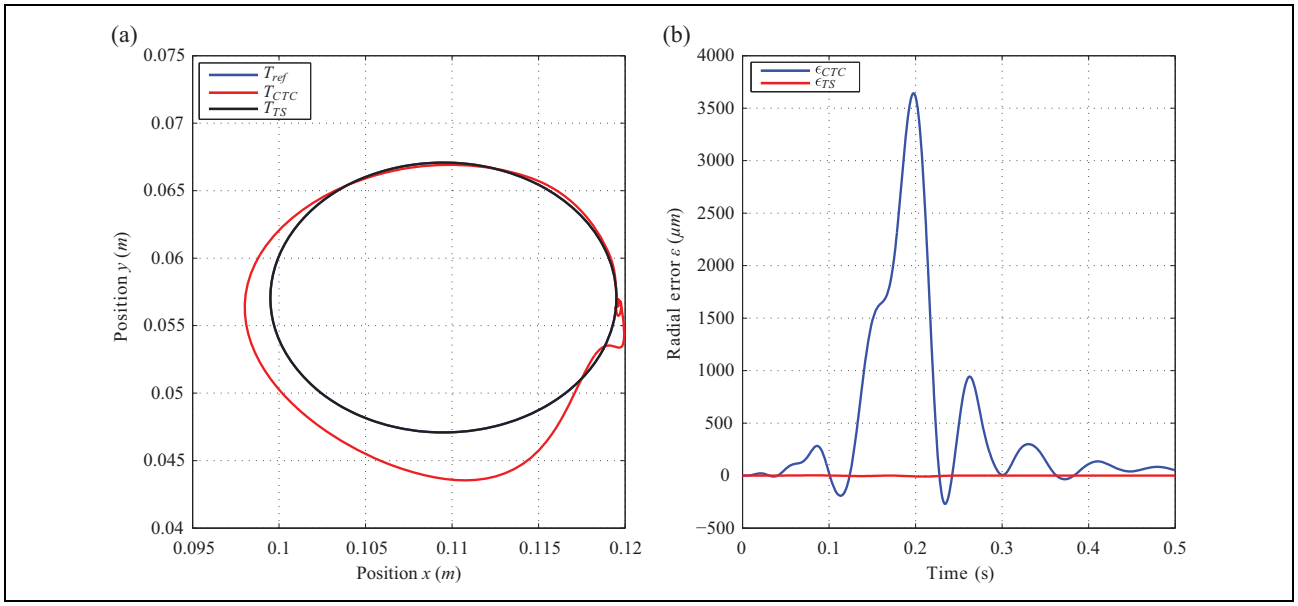

Figure 6. Simulation results for $\Delta m = 0.816\,kg$: (a) trajectory profile and (b) radial error.

Conclusion

This article proposes a novel approach for the control of parallel robots. Those systems are particularly difficult to control due to the presence of multiple CKCs. These mechanical loops lead to a model naturally described by a set of DAEs of index-3. Designing a controller based on the DAE model is not an easy task, it requires a particular knowledge on the control theory of singular systems. For this reason, most of the time we transform the DAE model into an ODE one by differentiating the constraint twice. This manipulation is appropriate for mechanical systems because if the system is well designed, the initial conditions

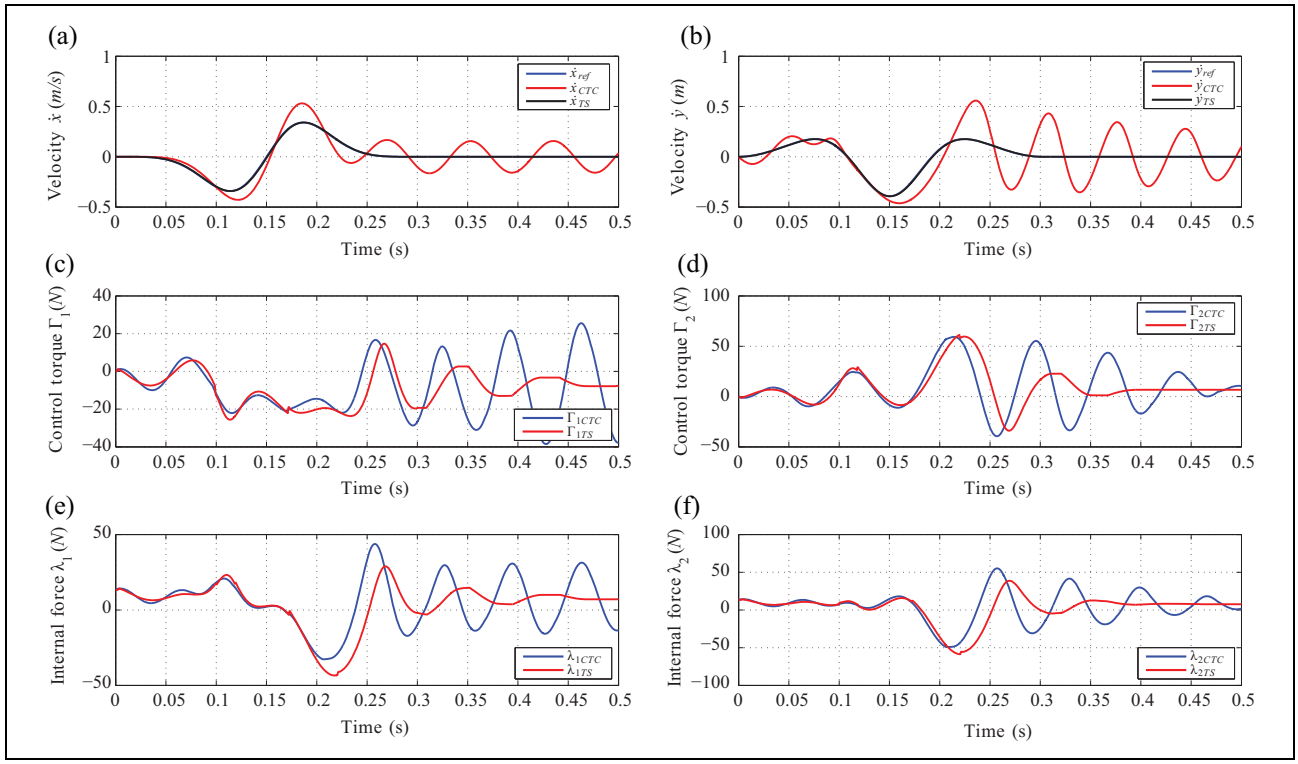

Figure 7. Simulation results for $\Delta m = 1.632\,kg$: (a) velocity profile of x, (b) velocity profile of y, (c) torque profile of joint 1, (d) torque profile of joint 2, (e) internal force profile of leg 1, and (f) internal force profile of leg 2.

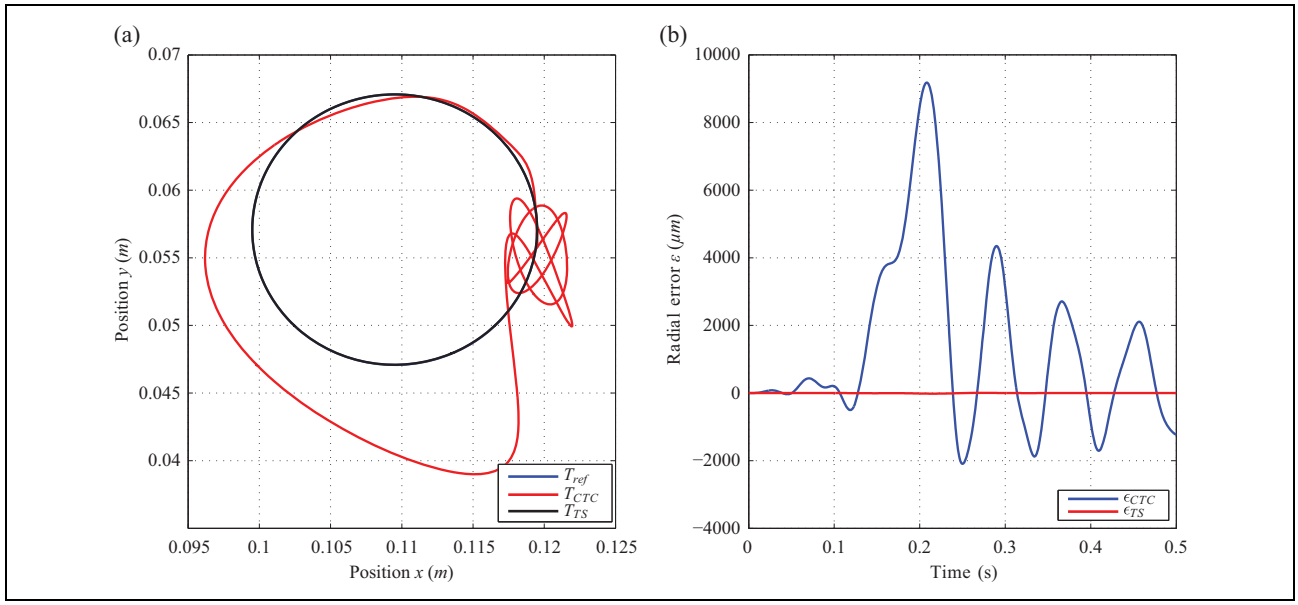

Figure 8. Simulation results for $\Delta m = 1.632\,kg$: (a) trajectory profile and (b) radial error.

should respect the algebraic constraint. Unfortunately, the obtained ODE model is most of the time complex and highly nonlinear. This makes the design of nonlinear controllers that requires the use of convex optimization techniques such as TS modeling and LMIs quite difficult. Because, in this case, the complexity of the controller is an exponential function of the number of nonlinear terms. For this reason, this article proposes to keep the DAE form of the system in order to decouple it into two independent subsystems. The idea here is to collect most of the nonlinear terms inside the Lagrange multipliers $\lambda_i(\cdot)$. According to our way of decomposing, the Lagrange multipliers

Table 1. ISV and IAE criteria.

	J_{ISV}		J_{IAE}	
	TS	CTC	TS	CTC
Δm=0 kg	50.05	55.2	0.4287	28.06
Δm=0.816 kg	58.84	58.23	0.6281	253.7
Δm=1.632 kg	277.33	446.6	1.458	943.6

ISV: integral of square value of the control; IAE: integral of absolute error; TS: Takagi-Sugeno; CTC: computed torque control.

represent the internal forces of the system. Knowing that all the physical parameters are bounded, the internal forces were bounded according to the range of the state variables and the saturation of the control inputs. From those bounds, the internal forces were decomposed into a convex sum and used as premises to establish the TS model. A PDC control law was designed. An extended integral part was added in order to compensate the stationary errors. Finally a comparison with the well-known CTC control law has been presented. The robustness of the PDC controller has been well checked against the structural uncertainties.

Acknowledgement

This research was supported by International Campus on Safety and Intermodality in Transportation, the Nord/Pas-de-Calais Region, the European Community, the Regional Delegation for Research and Technology, the Ministry of Higher Education and Research, the French National Research Agency, the National Center for Scientific Research and the Arts Carnot Institute. The authors gratefully acknowledge the support of these institutions, in addition to all partners of the VHIPOD project (Self-balancing transport vehicle for handicapped person in standing position with support for verticalization, ref : ANR-12-TECS-0001) : Laboratory of Medical information Processing (LaTIM-INSERM UMR 1101), National Centre of Resources and Innovation with a mission to promote mobility for all persons at any age (CEREMH), Kerpape Mutualistic Functional Reeducation and Rehabilitation Center (CMRRF of Kerpape) and BA systmes corporation.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

1. He JF, Jiang HZ, Cong DC, et al. A survey on control of parallel manipulator. *Key Eng Mater* 2007; 339: 307–313.
2. Simaan N. *Analysis and synthesis of parallel robots for medical applications*. PhD thesis, Technion-Israel Institute of Technology, Faculty of Mechanical Engineering, Israel, 1999.
3. Vermeiren L, Dequidt A, Afroun M, et al. Motion control of planar parallel robot using the fuzzy descriptor system approach. *ISA Trans* 2012; 51(5): 596–608.
4. Abdellatif H, Grotjahn M and Heimann B. High efficient dynamics calculation approach for computed-force control of robots with parallel structures. In: *Proceedings of the 44th IEEE conference on decision and control*, Sevilla, Spain, December 2005, pp. 2024–2029. IEEE.
5. Weck M and Staimer D. Parallel kinematic machine tools–current state and future potentials. *CIRP Ann-Manuf Technol* 2002; 51(2): 671–683.
6. Jamwal PK, Xie SQ, Tsoi YH, et al. Forward kinematics modelling of a parallel ankle rehabilitation robot using modified fuzzy inference. *Mech Mach Theory* 2010; 45(11): 1537–1554.
7. Abdellatif H, Heimann B, Kotlarski J, et al. Practical model-based and robust control of parallel manipulators using passivity and sliding mode theory. In: *Robotics 2010 current and future challenges*, 1st ed, Vukovar, Croatia: InTech, 2010.
8. Dombre E and Khalil W. *Modeling, performance analysis and control of robot manipulators*, Hoboken: Wiley, 2010.
9. Merlet J-P. *Parallel robots*. 2nd ed, Vol. 74. Dordrecht: Springer, 2012.
10. Natal GS, Chemori A and Pierrot F. Dual-space control of extremely fast parallel manipulators: payload changes and the 100g experiment. *IEEE Trans Control Syst Technol* 2015; 23(4): 1520–1535.
11. Khalil W and Ibrahim O. General solution for the dynamic modeling of parallel robots. *J Intell Robot Syst* 2007; 49(1): 19–37.
12. Tsai L-W. *Robot analysis: the mechanics of serial and parallel manipulators*. New York: John Wiley & Sons, 1999.
13. Wang Z and Ghorbel FH. Control of closed kinematic chains: a comparative study. In: *Proceedings of the american control conference*, Minneapolis, June 2006, p. 6. IEEE.
14. Gear CW, Leimkuhler B and Gupta GK. Automatic integration of Euler-Lagrange equations with constraints. *J Comput Appl Math* 1985; 12: 77–90.
15. Kumar A and Daoutidis P. Control of nonlinear differential algebraic equation systems: an overview. In: Ridvan Berber and Costas Kravaris (eds) *Nonlinear model based process control*, pp. 311–344. Dordrecht: Springer, 1998.
16. Gordon BW. Dynamic sliding manifolds for realization of high index differential-algebraic systems. *Asian J Control* 2003; 5(4): 454–466.
17. Kumar A and Daoutidis P. Feedback regularization and control of nonlinear differential-algebraic-equation systems. *AIChE J* 421996; (8): 2175–2198.
18. Brecher C, Ostermann T and Friedrich DA. Control concept for PKM considering the mechanical coupling between actors. *Int J Mach Tool Manu* 2008; 48(3): 427–436.
19. Denkena B and Holz C. Advanced position and force control concepts for the linear direct driven hexapod PaLiDa. In: *Proceedings of the 5th chemnitz parallel kinematics seminar*, Chemnitz, April 2006, pp. 359–378.
20. Zhiyong Y and Tian H. A new method for tuning PID parameters of a 3-DOF reconfigurable parallel kinematic machine. In: *Proceedings of the IEEE international conference on robotics and automation,* New Orleans, April 2004, Vol. 3, pp. 2249–2254. IEEE.

21. Amirat Y, Francois C, Fried G, et al. Design and control of a new six-DOF parallel robot: application to equestrian gait simulation. *Mechatronics* 1996; 6(2): 227–239.
22. Tadokoro S. Control of parallel mechanisms. *Adv Robot* 1994; 8(6): 559–571.
23. Paccot F, Andreff N and Martinet P. A review on the dynamic control of parallel kinematic machines: theory and experiments. *Int J Robot Res* 2009; 28(3): 395–416.
24. Callegari M, Palpacelli M-C and Principi M. Dynamics modelling and control of the 3-RCC translational platform. *Mechatronics* 2006; 16(10): 589–605.
25. Beji L, Abichou A and Pascal M. Tracking control of a parallel robot in the task space. In: *Proceedings of the IEEE international conference on robotics and automation*, Leuven, Belgium, May 1998, Vol. 3, pp. 2309–2314. IEEE.
26. Kock S and Schumacher W. Control of a fast parallel robot with a redundant chain and gearboxes: experimental results. In: *Proceedings of the international conference on robotics and automation*, California, USA, April 2000, Vol. 2, pp. 1924–1929. IEEE.
27. Lee S-H, Song J-B, Choi W-C, et al. Position control of a Stewart platform using inverse dynamics control with approximate dynamics. *Mechatronics* 2003; 13(6): 605–619.
28. Siciliano B and Khatib O. *Springer handbook of robotics*. Berlin Heidelberg: Springer Science & Business Media, 2008.
29. Spong MW and Vidyasagar M. *Robot dynamics and control*. New York: John Wiley & Sons, 2008.
30. Craig JJ. *Introduction to robotics: mechanics and control*. Vol 3. Upper Saddle River: Pearson Prentice Hall, 2005.
31. Cheung JWF and Hung YS. Modelling and control of a 2-dof planar parallel manipulator for semiconductor packaging systems. *In: Proceedings of the IEEE/ASME international conference on advanced intelligent mechatronics*, Monterey, USA, July 2005, pp. 717–722. IEEE.
32. Jafarinasab M, Keshmiri M, Azizan H, et al. Sliding mode control of a novel 6-DOF parallel manipulator with rotary actuators. In: *16th international conference on methods and models in automation and robotics*, Miedzyzdroje, Poland, August 2011, pp. 218–223. IEEE.
33. Litim M, Allouche B, Omari A, et al. Sliding mode control of biglide planar parallel manipulator. In: *11th international conference on informatics in control, automation and robotics*, Vienna, Austria, September 2014, Vol. 2, pp. 303–310. IEEE.
34. Becerra-Vargas M and Belo EM. Application of h∞ theory to a 6-DOF flight simulator motion base. *J Braz Soc Mech Sci Eng* 2012; 34(2): 193–204.
35. Rachedi M, Hemici B and Bouri M. Design of an H controller for the delta robot: experimental results. *Adv Robot* 2015; 29(18): 1165–1181.
36. Chemori A, Natal GS and Pierrot F. Control of parallel robots: towards very high accelerations. In: *SSD'2013: 10th international multi-conference on systems, signals and devices*, Hammamet, Tunisia, March 2013, p. 8. IEEE.
37. Shang W-W, Cong S and Ge Y. Adaptive computed torque control for a parallel manipulator with redundant actuation. *Robotica* 2012; 30(3): 457–466.
38. Abdellatif H, Heimann B and Kotlarski J. Passivity-based observer/controller design with desired dynamics compensation for 6 DOFs parallel manipulators. In: *IEEE/RSJ international conference on intelligent robots and systems*, Nice, France, September 2008, pp. 2392–2397. IEEE.
39. Ghorbel FH, Chételat O, Gunawardana R, et al. Modeling and set point control of closed-chain mechanisms: theory and experiment. *IEEE Trans Control Syst Technol* 2000; 8(5): 801–815.
40. Ghorbel FH, Gunawardana R and Dabney JB. Experimental validation of a reduced model based tracking control of parallel robots. In: *Proceedings of the international conference on control applications,* Mexico City, Mexico, September 2001, pp. 375–382. IEEE.
41. Zhengsheng C, Minxiu K, Ming L, et al. Dynamic modelling and trajectory tracking of parallel manipulator with flexible link. *Int J Adv Robot Syst* 2013; 10: 1.
42. Asgari M and Ardestani MA. Dynamics and improved computed torque control of a novel medical parallel manipulator: applied to chest compressions to assist in cardiopulmonary resuscitation. *J Mech Med Biol* 2015; 15(04): 1550051.
43. Bennehar M, Chemori A, Pierrot F, et al. Extended model-based feed forward compensation in l1 adaptive control for mechanical manipulators: design and experiments. *Front Robot AI* 2015; 2: 32.
44. Gordon BW, Liu S and Asada HH. Realization of high index differential-algebraic systems using singularly perturbed sliding manifolds. In: *American control conference, 2000. Proceedings of the 2000*, Chicago, Illinois, USA, June 2000, Vol. 2, pp. 752–756. IEEE.
45. Majou F, Wenger P and Chablat D. A novel method for the design of 2-DOF parallel mechanisms for machining applications. *arXiv preprint arXiv:0705.1280*. 2007.
46. Afroun M, Vermeiren L and Dequidt A. Revisiting the inverse dynamics of the Gough–Stewart platform manipulator with special emphasis on universal–prismatic–spherical leg and internal singularity. *P I Mech Eng C-J Mec* 2012; 226(10): 2422–2439.
47. Naidu D. Singular perturbations and time scales in control theory and applications: an overview. *Dynam Cont Dis Ser B* 2002; 9: 233–278.
48. Gordon BW and Liu S. A singular perturbation approach for modeling differential-algebraic systems. *J Dyn Syst Meas Control* 1998; 120(4): 541–545.
49. Dabney JB, Ghorbel FH and Wang Z. Modeling closed kinematic chains via singular perturbations. In: *Proceedings of the American control conference*, Anchorage, USA, May 2002, Vol. 5, pp. 4104–4110. IEEE.
50. Wang Z and Ghorbel FH. Control of closed kinematic chains using a singularly perturbed dynamics model. *J Dyn Syst Meas Control* 2006; 128(1): 142–151.
51. Khan WA, Krovi VN, Saha SK, et al. Modular and recursive kinematics and dynamics for parallel manipulators. *Multibody Syst Dyn* 2005; 14(3–4): 419–455.
52. Krishnan H and McClamroch NH. Tracking in nonlinear differential-algebraic control systems with applications to

constrained robot systems. *Automatica* 1994; 30(12): 1885–1897.
53. Takagi T and Sugeno M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans Syst Man Cybern Syst* 1985; (1): 116–132.
54. Vermeiren L, Dequidt A, Guerra TM, et al. Modeling, control and experimental verification on a two-wheeled vehicle with free inclination: an urban transportation system. *Control Eng Pract* 2011; 19(7): 744–756.
55. Sala A, Guerra TM and Babuška R. Perspectives of fuzzy systems and control. *Fuzzy Sets Syst* 2005; 156(3): 432–444.
56. Tanaka K and Wang HO. *Fuzzy control systems design and analysis: a linear matrix inequality approach*. New York: John Wiley & Sons, 2004.
57. Wang HO, Tanaka K and Griffin MF. An approach to fuzzy control of nonlinear systems: stability and design issues. *IEEE Trans Fuzzy Syst* 1996; 4(1): 14–23.
58. Tanaka K, Ikeda T and Wang HO. Fuzzy regulators and fuzzy observers: relaxed stability conditions and LMI-based designs. *IEEE Trans Fuzzy Syst* 1998; 6(2): 250–265.
59. Tuan HD, Apkarian P, Narikiyo T, et al. Parameterized linear matrix inequality techniques in fuzzy control system design. *IEEE Trans Fuzzy Syst* 2001; 9(2): 324–332.
60. Spong MW. Modeling and control of elastic joint robots. *J Dyn Syst Meas Control* 1987; 109(4): 310–318.
61. Andersson S, Söderberg A and Björklund S. Friction models for sliding dry, boundary and mixed lubricated contacts. *Tribol Int* 2007; 40(4): 580–587.
62. Yang Z, Wu J, Mei J, et al. Mechatronic model based computed torque control of a parallel manipulator. *Int J Adv Robot Syst* 2008; 5(1): 123–128.
63. Gahinet PM, Nemirovskii A, Laub AJ, et al. The LMI control toolbox. In: *Proceedings of the conference on decision and control*, Florida, USA, Vol. 2, pp. 2038–2038. IEEE.

Appendix 1

Table 1A. Parameters of the nominal model.

Nominal parameters	Values	Additional simulation parameters	Values
Leg length (m)		Mass (kg)	
a	0.07	Δm	0.816
Mass (kg)		m_a	0.7
m	0.034	m_{L1}	0.104
$m1$	0.8040	m_{L2}	0.094
$m2$	0.7940	Stiffness constant (N/m)	
First moment of links (kgm)		k_t	3.88×10^3
ms_1	0.0045	Damping constant (Ns/m)	
ms_2	0.0043	b_t	17.48
Second moment of links (kgm^2)		Dry friction force (N)	
J_1	222.643×10^{-4}	Γ_{fs}	1.5
J_2	2.539×10^{-4}	Γ_{fc}	1
Gravity acceleration (ms^2)		Sliding speed coefficient (m/s)	
g	9.81	v_s	0.1

Appendix 2

Dynamic model with mass uncertainty, elastic joints, and Stribeck friction

Numerical simulations include a model with structured and unstructured uncertainties based on the nominal model used to design the controller. Unmodeled dynamics such as elastic joints[44,8] between actuators and linkages and Stribeck friction[45,8] applied on prismatic joints appear in this augmented model to provide more realistic simulations.

The dynamics of the actuator writes

$$\Gamma = M_a \ddot{q}_a + b\dot{q}_a + \Gamma_t + \Gamma_f \tag{31}$$

with $q_a = [\,q_{a1} \quad q_{a2}\,]^T$, $M_a = diag(m_{a1}, m_{a2})Z$, $\Gamma_f = [\,\Gamma_{f1} \quad \Gamma_{f2}\,]Z$, the elastic joint model

$$\Gamma_t = k_t(q_a - q) + b_t(\dot{q}_a - \dot{q}) \tag{32}$$

and the Stribeck friction model of the dry friction

$$\Gamma_{fi} = \begin{cases} \left[\Gamma_{fc} + (\Gamma_{fs} - \Gamma_{fc})e^{-\left(\frac{\dot{q}_{ai}}{v_s}\right)^2}\right] \operatorname{sign}(\dot{q}_{ai}) & if\ |\dot{q}_{ai}| > 0\ (\text{slip}) \\ \min(|\Gamma_i - \Gamma_{ti}|, \Gamma_{fs}) \operatorname{sign}(\Gamma_i - \Gamma_{ti}) & if\ \dot{q}_{ai} = 0\ (\text{stick}) \end{cases} \tag{33}$$

where m_a is the actuator mass, k_t the stiffness of the joint, b_t the damping of the joint, Γ_{fs} the static friction force, Γ_{fc} the Coulomb friction force, and v_s the sliding speed coefficient.

The linkage and effector dynamics are

$$\Gamma_t = \hat{M}(X)\ddot{X} + \hat{N}(X,\dot{X}) + \hat{G}(X)X \tag{34}$$

$$\hat{M}(X) = \begin{bmatrix} m_{L1} + \frac{1}{2}(m - \eta_1 + \eta_2) & \frac{[(2m_{L1} - 3\eta_1 - \eta_2)y^2 + mC(y)^2 + J_1 + J_2]}{2yC(y)} \\ m_{L2} + \frac{1}{2}(m - \eta_2 + \eta_1) & \frac{[(m_{L2} - 3\eta_2 - \eta_1)y^2 + mC(y)^2 + J_1 + J_2]}{2yC(y)} \end{bmatrix}$$

$$\hat{N}(X,\dot{X}) = \begin{bmatrix} 0 & \frac{-[(2m_{L1} - 3\eta_1 - \eta_2)y^2 + (2m_{L1} - 3\eta_1 - \eta_2)C(y)^2 + J_1 + J_2]\dot{y}}{2C(y)^3} \\ 0 & \frac{[(2m_{L2} - 3\eta_2 - \eta_1)y^2 + (2m_{L2} - 3\eta_2 - \eta_1)C(y)^2 + J_1 + J_2]\dot{y}}{2C(y)^3} \end{bmatrix}$$

$$\hat{G}(X) = \begin{bmatrix} 0 & gC(y)\frac{(m + \Delta m + \eta_1 + \eta_2)}{2y^2} \\ 0 & -gC(y)\frac{(m + \Delta m + \eta_1 + \eta_2)}{2y^2} \end{bmatrix}$$

where the linkage mass m_{Li} satisfies: $m_i = m_{ai} + m_{Li}$, $i = 1, 2$.