A review on dynamic control of parallel kinematic machine: theory and experiments

Flavien Paccot\textsuperscript{1} Nicolas Andreff\textsuperscript{1,2} Philippe Martinet\textsuperscript{1,3}

\textsuperscript{1}LASMEA - UMR 6602 CNRS, Blaise Pascal University, Clermont-Ferrand, France
\textsuperscript{2}LAMI, IFMA, France
\textsuperscript{3}ISRC, Sungkyunkwan University, Suwon, South Korea

\textbf{Abstract}

In this article, a review on parallel kinematic machine dynamic control is performed. It is shown that the classical control strategies from serial robotics generally used for parallel kinematic machine have to be thought again. Indeed, it is first shown that the joint space control is not relevant for these mechanisms for several reasons such as mechanical behaviour or computational efficiency. Consequently, the Cartesian space control should be preferred over the joint space one. Nevertheless, some modification to the well-known Cartesian space control strategies of serial robotics are proposed to perfectly suit them to parallel kinematic machines, particularly a solution using an exteroceptive measure of the end-effector pose. The expected improvement in terms of accuracy, stability and robustness are discussed. A comparison between the main presented strategies is finally performed both in simulation and experiments.

\section{Introduction}

From a theoretical point of view, parallel kinematic machines allow for better dynamic performances than serial ones, in terms of speed, accuracy and stiffness [Merlet, 2000]. Therefore, they seem perfectly suitable for industrial high-speed applications, such as pick-and-place or high speed machining. However, experiments on parallel kinematic machines point out that these good dynamic properties are not always established [Wang and Masory, 1993, Tlusty et al., 1999, Pritschow, 2002, Brecher et al., 2006a, Denkena and Holz, 2006]. Consequently, the improvement of static and dynamic accuracy is still an up-to-date and prosperous research field.

On the one hand, recent machines allow for impressive maximal acceleration, such as 200\text{m.s}^{-2} for the high-speed manipulator PAR4 [Nabat et al., 2005], or 50\text{m.s}^{-2} for the Urane SX machine tool [Company and Pierrot, 2002]. Such high acceleration can not be achievable with serial kinematic machines. Consequently, the time gain is clearly established [Geldart et al., 2003, Terrier et al., 2004].

On the other hand, several works deal with issues on the accuracy and stiffness of parallel kinematic machines. Pritschow [Pritschow, 2002] presents a list of phenomena affecting the accuracy (Figure 1).

Two major issues are the object of numerous works.

First major issue, the presence of numerous

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{Figure1.png}
\caption{Causes of accuracy losses according to Pritschow [Pritschow, 2002]}
\end{figure}
Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X, \dot{X}, \ddot{X}$</td>
<td>Any representation of end-effector pose, velocity and acceleration</td>
</tr>
<tr>
<td>$q, \dot{q}, \ddot{q}$</td>
<td>Joint positions, velocities and accelerations</td>
</tr>
<tr>
<td>$FKM$</td>
<td>Forward kinematic model</td>
</tr>
<tr>
<td>$IKM$</td>
<td>Inverse kinematic model</td>
</tr>
<tr>
<td>$\dot{X} = D\dot{q}$</td>
<td>Forward instantaneous kinematics matrix, abusively called Jacobian matrix</td>
</tr>
<tr>
<td>$\dot{q} = D_{inv}\dot{X}$</td>
<td>Inverse instantaneous kinematics matrix, abusively called inverse Jacobian matrix</td>
</tr>
<tr>
<td>$\dot{D}$</td>
<td>Time derivative of the forward instantaneous kinematics matrix</td>
</tr>
<tr>
<td>$\dot{D}_{inv}$</td>
<td>Time derivative of the inverse instantaneous kinematics matrix</td>
</tr>
<tr>
<td>$FDM$</td>
<td>Forwards dynamic model</td>
</tr>
<tr>
<td>$IDM$</td>
<td>Inverse dynamic model</td>
</tr>
<tr>
<td>$\Gamma$</td>
<td>Actuation torques</td>
</tr>
<tr>
<td>$A$</td>
<td>Inertia matrix</td>
</tr>
<tr>
<td>$H$</td>
<td>Vector containing Coriolis, centrifugal and gravity forces</td>
</tr>
<tr>
<td>$M$</td>
<td>Inertia matrix of the actuated bodies, mapped into the active joint space, diagonal and constant</td>
</tr>
<tr>
<td>$I$</td>
<td>Inertia matrix of the end-effector, diagonal and constant</td>
</tr>
<tr>
<td>$\Gamma_f$</td>
<td>Friction forces</td>
</tr>
<tr>
<td>$F_v, F_s$</td>
<td>Viscous and dry friction parameters</td>
</tr>
<tr>
<td>$K_p, K_v, K_i$</td>
<td>Proportional, derivative and integral gain</td>
</tr>
<tr>
<td>$u_{PID}$</td>
<td>Signal generated by the PID controller</td>
</tr>
<tr>
<td>$u_{ff}$</td>
<td>Feedforward compensation term</td>
</tr>
<tr>
<td>$u_{comp}$</td>
<td>Compensation term</td>
</tr>
<tr>
<td>$v_d$</td>
<td>Desired variable $v$</td>
</tr>
<tr>
<td>$e = v_d - v$</td>
<td>Error signal between the desired and measured (or estimated) variable $v$</td>
</tr>
<tr>
<td>$s$</td>
<td>Laplace variable</td>
</tr>
<tr>
<td>$\hat{m}$</td>
<td>Numerical estimation of model $m$</td>
</tr>
<tr>
<td>$S(m)$</td>
<td>Skew matrix associated to the cross-product by vector $m$</td>
</tr>
</tbody>
</table>

Table 1: General notation used in this article
joints causes kinematic model errors because of clearances and assembly defects [Wang and Masory, 1993]. Moreover, the complex kinematics often leads to model simplifications decreasing accuracy [Pritschow, 2002]. The main solution to these problems is a performant kinematic identification [Wang and Masory, 1993]. It allows for matching as far as possible the machine model and its real behaviour. The measures used for identification are performed with various means [Daney, 1999, Besnard and Khalil, 1999, Renaud et al., 2006, Chanal et al., 2006]. Another solution to kinematic errors consists in using adapted modeling methods to increase the model accuracy while simplifying the algorithms [Merlet, 2000].

Second major issue, the actuators of a parallel kinematic machine tool do not apply a torque along the end-effector motion axis, contrary to a serial one [Tlusty et al., 1999]. It results in a decrease of stiffness leading to a lack of accuracy during machining process. In this way, a workspace can be determined where stiffness allows for sufficient accuracy [Chanal et al., 2006]. Moreover, improvements are sought throughout design of new structure [Tlusty et al., 1999, Liu et al., 2000].

In summary, it seems that a parallel kinematic machine is really faster than a serial one, but gains in terms of stiffness and accuracy are questionable. Actually, Merlet explains that the advantages of parallel kinematic machines can only be qualified as potential [Merlet, 2002]. To reach their theoretical performances, parallel kinematic machines still require improvements in design, modeling and control.

Nevertheless, the solutions presented above concern only mechanical design, kinematic modeling and identification. To our mind, the control of parallel kinematic machines is a field where great potential remains for improving accuracy. Indeed, most of the work in the literature only reuses the knowledge of serial robotics whereas control strategies have to be thought again to improve parallel kinematic machines performances. To illustrate this point, a state of the art on joint space control, simple control, is used [Khalil and Dombre, 2002, Brecher et al., 2006a, Denkena and Holz, 2006, Zhiyong and Huang, 2004, Yang and Huang, 2006]. Figure 2 reminds of this well-known control scheme. This is the conventional and simplest way to control a system. It can be used for serial and parallel kinematic machines. Therefore, controllers can be reused, from serial to parallel kinematic machines without major adaptation, making this control strategy interesting in an industrial context. In addition, it provides rather good performances with regards to its wide use. The tuning of such a control is well known from the classical robotics method [Khalil and Dombre, 2002] to more elaborated ones, adapted for parallel kinematic machines [Zhiyong and Huang, 2004, Yang and Huang, 2006].

2 Control of parallel kinematic machines in the joint space

The knowledge on parallel robotics comes directly from serial one. Therefore, parallel kinematic machines are mainly controlled with the same strategies as serial ones. Therefore, the main control methods met in the literature are linear single-axis and computed torque control, both in the joint space.

2.1 Linear single-axis control

In most industrial cases, a linear single-axis control, also called PID control or simple control, is used [Khalil and Dombre, 2002, Brecher et al., 2006a, Denkena and Holz, 2006, Zhiyong and Huang, 2004, Yang and Huang, 2006]. Figure 2 reminds of this well-known control scheme. This is the conventional and simplest way to control a system. It can be used for serial and parallel kinematic machines. Therefore, controllers can be reused, from serial to parallel kinematic machines without major adaptation, making this control strategy interesting in an industrial context. In addition, it provides rather good performances with regards to its wide use. The tuning of such a control is well known from the classical robotics method [Khalil and Dombre, 2002] to more elaborated ones, adapted for parallel kinematic machines [Zhiyong and Huang, 2004, Yang and Huang, 2006].
Let us remind the reader of the conventional method for a theoretical tuning [Khalil and Dombre, 2002]. Out of habit, the assumption of simple dynamics only with inertia forces associated with a single constant and diagonal inertia matrix $M$, and without centrifugal, Coriolis and gravity forces is made:

$$\Gamma = M \ddot{q}$$  \hspace{1cm} (1)

The PID controller generates directly a torque input, $u_{PID}$:

$$u_{PID} = K_v \dot{e} + K_p e + K_i \int e dt \hspace{1cm} (2)$$

The denominator of the closed loop transfer function can be thus expressed as:

$$B(s) = (Ms^3 + K_v s^2 + K_p s + K_i)$$  \hspace{1cm} (3)

The classical tuning aims at obtaining a third-order negative real poles system:

$$B(s) = M(s + \omega)^3$$ \hspace{1cm} (4)

with $\omega$ chosen with respect to the mechanical resonance frequency of the controlled machine. By matching each term of Equation 3 and Equation 4, the gain values are expressed as follows:

$$\begin{cases} K_v = 3M\omega \\ K_p = 3M\omega^2 \\ K_i = M\omega^3 \end{cases}$$ \hspace{1cm} (5)

This tuning gives theoretical values which have to be adapted in practice. Indeed, the integral gain $K_i$ is generally increased to compensate for the dry frictions, while the derivative gain $K_v$ is generally decreased to cope with measurement noise.

In most cases, the linear single-axis control is improved with a feedforward term, $u_{ff}$. The general formulation of this term is:

$$u_{ff} = M \ddot{q}_d \hspace{1cm} (6)$$

In this case, the error signal behaviour is fixed by a third-order ordinary differential equation:

$$Me^{(3)} + K_v \ddot{e} + K_p \dot{e} + K_i e = 0$$ \hspace{1cm} (7)

where $e^{(3)}$ is the third derivative with respect to time.

The gain values are the same as Equation 5, which allows for a performant error behaviour. Let us notice that the $M$ matrix is often used out of the feedforward and PID gain, like in Figure 3. It yields to the following error signal behaviour:

$$M(e^{(3)} + K_v \ddot{e} + K_p \dot{e} + K_i e) = 0$$ \hspace{1cm} (8)

and a gain tuning independent from the machine inertia:

$$\begin{cases} K_v = 3\omega \\ K_p = 3\omega^2 \\ K_i = \omega^3 \end{cases}$$ \hspace{1cm} (9)

Furthermore, in an industrial controller, some additional features are used, such as friction, gravity and backlash compensation, to improve accuracy. Such a control strategy ensures sufficiently good performances for serial kinematic machine tools to make it still widely used in the industry. Nevertheless, a machine tool is quite slow, very heavy and stiff. Therefore, a single-axis linear control ensures a efficient compensation of the small dynamics and the behaviour of the stiff mechanical structure ensures a good static accuracy.

On the opposite, the single-axis control is known to be weak with fast serial manipulators since it does not ensure a sufficient compensation of the nonlinear dynamics, leading to a poor dynamic accuracy [Khalil and Dombre, 2002]. Actually, reported experiments show that such a control can not either ensure a good accuracy for parallel kinematic machines [Brecher et al., 2006a, Denkena and Holz, 2006, Vivas et al., 2003, Ouyang et al., 2002, Honegger et al., 2000]. Indeed, the presence
of the complex inverse kinematics in the path planning is a first source of static accuracy lack, requiring adapted modeling and identification as stated above. However, the complex dynamic behaviour is also a very unfavourable phenomenon.

As a matter of fact, the dynamic behaviour of a parallel kinematic machine is strongly non linear due to a dynamic coupling between legs, which does not exist in the serial case. Furthermore, most of the parallel kinematic machines have an anisotropic behaviour. Therefore, the hypothesis of linear dynamics is only verified at low speed and very locally. Consequently, a linear single-axis control can not be efficient in the whole workspace with the same tuning, as established by Brecher [Brecher et al., 2006a]. A first solution is the determination of a restricted workspace with regard to maximal accelerations, as initiated by Barrette [Barrette and Gosselin, 2005]. This method could be extended with the determination of a workspace associated with a maximal speed and acceleration, where dynamics are fairly homogeneous with a low dynamic coupling. A second method is a path planning with dynamic consideration [Abdel- latif and Heiman, 2005, Oen and Wang, 2006]. In addition, the use of an adapted time interpolation can smooth the trajectory by limiting jerk or snap (respectively 4th and 5th order time derivative of the joint position) [Erkorkmaz and Altintas, 2001, Fleisig and Spence, 2001, Lambrechts et al., 2005]. It can be noticed that a feedforward compensation in terms of jerk and snap can thus be used [Lambrechts et al., 2005]. Such methods aim to ease controller action to ensure the required accuracy. However, the limitations of such methods are the decrease of the effective speed and workspace leading to a low use of the machines capabilities. Moreover, the real machine motion is not completely mastered since the heavy computation generally imposes an off-line path generation without any on-line corrections of this path.

To improve the real machine motion control, the control gain tuning can be optimised with dynamic considerations, as proposed by Zhiyong and Yang [Zhiyong and Huang, 2004, Yang and Huang, 2006]. Other solutions consist in control laws modification (not always achievable on an industrial controller). Some works deal with nonlinear gains [Ouyang et al., 2002], robust control [Kim et al., 2005, Fu and Mills, 2005] and inappreciable phenomena compensation [Brecher et al., 2006b]. Nevertheless, these strategies are still based on a single-axis control with various compensation in an external loop. There is no dynamics compensation in the control loop. Yet, a direct and simple way to compensate for the dynamic behaviour is the well known computed torque control.

2.2 Computed Torque Control

The computed torque control is a widespread control strategy for serial manipulators [Khalil and Dombre, 2002, Luh et al., 1980]. Figure 4 reminds of this control scheme. Let us remind the reader of how the classical computed torque control works [Khalil and Dombre, 2002]. The control law is based on the Lagrange formulation of the machine inverse dynamic model:

\[ \Gamma = A(q)\ddot{q} + H(q, \dot{q}) \]  

(10)

By replacing \( \ddot{q} \) in Equation 10 by an adapted control signal \( u \), an exact linearisation of the dynamics is ensured. Indeed, there is only a double integrator between control signal and joint variables. The following control signal is used:

\[ u = \dot{q}_d + K_v \dot{e} + K_a e \]  

(11)

In this case, the error signal has a second order behaviour:
\[
\dot{e} + K_v \dot{e} + K_p e = 0
\]  
(12)

The gain tuning is, as it is well known, fixed by a cut-off frequency and a damping:

\[
\begin{align*}
K_v &= 2\xi \omega \\
K_p &= \omega^2
\end{align*}
\]  
(13)

The damping \( \xi \) is generally fix between 0.9 and 1 to avoid overshoot while yielding a good establishing time. The cut-off frequency \( \omega \) is fixed to the highest value with respect to the mechanical resonance frequency. It can be noticed that the integral gain is useless because the linearisation of the dynamics leads to a double integrator system. However, the integral gain is generally employed in practice. It allows for improving accuracy by compensating for the light unmodeled phenomena. Nevertheless, this control strategy can be improved with friction and backlash compensation, like the linear single-axis control.

With such a control scheme, the nonlinear dynamic behaviour of the machine is compensated for in the whole workspace. In this way, the linear controller is associated with an exactly linearised system. Therefore, the controller performances are maximal and homogeneous in the whole workspace. Nevertheless, these great performances are only achievable with a dynamic model reflecting perfectly the real machine behaviour. Indeed, the computed torque control does not cope very well with modeling errors [Khalil and Dombre, 2002]. They create a perturbation on the error behaviour which may lead to a lack of stability and accuracy. Since a model almost never reflects exactly the real machine behaviour, modeling errors are nearly unavoidable. Consequently, a minimisation of these modeling errors is required. In this way, dynamic identification is generally performed [Swevers et al., 1997, Gautier and Poignet, 2001, Olsen and Peterson, 2001]. Alternately, a more complex model can be used. A flexible body dynamic model [Kock and Shimacher, 2000b], instead of a rigid body one, allows for taking into account deformations, increasing model accuracy while increasing on-line computation. A model taking into account task influence [Oen and Wang, 2006], instead of a model of the stand alone mechanical structure, can cope with external torques applied on the end-effector, which are specific for the application (cutting, loads carrying...). Furthermore, if the influence of the modeling errors is still interfering, robust control technique can be employed [Vivas et al., 2003, Lee et al., 2003, Honegger et al., 2000]. Actually, robust techniques are used here for compensating for the phenomena which can not be modeled, as they were originally designed for, and not for compensating for insufficient modelling, as it is too often seen.

### 2.3 Discussion

The control strategies exposed above are performed in the joint space. Practically, the actuators encoders are generally the only available measurement mean. Theoretically, a serial kinematic machine is completely defined by its joint configuration, in terms of kinematics and dynamics [Khalil and Dombre, 2002]. The joint configuration reflects thus the state of the machine. Consequently, the joint space control is a state feedback control. As it is generally admitted, a state feedback control allows for ensuring the best accuracy. Therefore, the joint space control is relevant for serial kinematic
machine, provided that solving for the inverse kinematic problem is accurate enough to translate the desired Cartesian path into the correct joint reference path.

On the opposite, a parallel kinematic machine is completely defined by its end-effector pose, except in some rare cases (3RRR for example [Chablat and Wenger, 1998]). Actually, this is generally admitted for the kinematics [Waldron and Hunt, 1991, Merlet, 2002, Dallej et al., 2006] and it is being extended to the dynamics [Dasgupta and Choudhury, 1999, Khalil and Ibrahim, 2004, Callegari et al., 2006]. The end-effector pose can thus be considered as the state of a parallel kinematic machine [Dallej et al., 2006]. Therefore, a joint space control is not a state feedback control but a biased observer feedback control. Consequently, the best performances in term of accuracy can not be ensured which such a control.

Moreover, the instantaneous kinematics and dynamics depend on the end-effector pose as stated above. Consequently, a joint space model-based control, such as the computed torque control, should include the forward kinematic model. To illustrate this point, we propose an explicit form of the computed torque control in the joint space which includes these forward transformations (see Figure 5). In general, the forward kinematics of a parallel kinematic machine do not have a closed-form expression contrary to a serial one. A joint configuration can thus lead to several end-effector poses (namely up to 40 for the Gough-Stewart platform [Merlet, 1990, Husty, 1994]). Some solutions can be removed since they are complex or mechanically inadmissible, but the end-effector pose can not be estimated only from the active joint configuration with reliability. Indeed, the forward kinematic problem is a square model since it has exactly the same amount of equations and unknowns. Hence, it is sensitive to any measurement noise, not even to mention the kinematic model and calibration errors. In addition, the on-line computation of the end-effector pose leads to a lack of speed, accuracy and stability. Consequently, the performances of the control are limited. Furthermore, the implicit presence of on-line numerical transformations leads, in practice, to model simplifications thus increasing modeling errors. As reminded above, the computed torque control has a weak robustness with regards to the modeling errors. Thus, the joint space computed torque control is often unusable alone. Some solutions are reported such as simplified dynamics and robust control [Vivas et al., 2003, Lee et al., 2003] or nonlinear feedforward compensation with robust control [Honegger et al., 2000]. Nevertheless, the mastery of robust control and the perfectible accuracy make the joint space computed torque control not welcome in an industrial context.

As a conclusion, the joint space control seems to be inherently imperfect and unadapted for parallel kinematic machines. Since the latter are completely defined by their end-effector pose, improvements could be found in the Cartesian space ($SE_3$) control.

3 Control of the parallel kinematic machines in the Cartesian space

To our knowledge, the control of a parallel kinematic machine in the Cartesian space, or the task space, is often mentioned in the literature [Lee et al., 2003, Kock and Shimacher, 2000a, Marquet et al., 2001, Beji et al., 2001, Beji et al., 1998, Yamane et al., 1998, Callegari et al., 2006, Caccavale et al., 2003]. Nevertheless, only few experiments are performed. When it is used, many model simplifications are done decreasing accuracy and stability [Lee et al., 2003]. The purpose of this section, which the essential theoretical contribution of this paper, is therefore to propose a complete revisiting of the Cartesian space control strategies. It is based on the serial-parallel duality [Waldron and Hunt, 1991] and thus requires the following assumption:

**Assumption 1** The end-effector pose can be measured accurately at control frequency.

This assumption will be discussed in Section 3.3.

3.1 Equivalent of single-axis linear control in the Cartesian space

3.1.1 With the end-effector dynamics only

The Cartesian space equivalent of the linear single-axis control given by Figure 6 is generally used [Callegari et al., 2006]. However, it is shown that the
transposition from joint to Cartesian space is not completely straightforward. In the control scheme of Figure 6, the simplified dynamics is expressed as:

\[ \Gamma = \hat{D}^T(X)I \ddot{X} \]  

(14)

where only the inertia of the end-effector \( I \) is taken into account. The latter is mapped into the active joint space with the forward instantaneous kinematic matrix.

The feedforward compensation term can be then expressed as:

\[ u_{ff} = \ddot{X}_d \]  

(15)

In this case, the error signal behaviour is fixed by the following equation:

\[ \hat{D}^T(X)I(\varepsilon^{(3)} + K_v \ddot{e} + K_p \dot{e} + K_i e) = 0 \]  

(16)

Consequently, the following tuning should be used:

\[
\begin{align*}
K_v &= 3\omega \\
K_p &= 3\omega^2 \\
K_i &= \omega^3
\end{align*}
\]  

(17)

Consequently, a similar running between single-axis linear control and this Cartesian space control strategy is retrieved here, with similar dynamics, feedforward compensation term and PID tuning. The only difference is the presence of the transposed forward instantaneous kinematic matrix in the control loop. However, we can make some remarks here. First, the presence of a numerically estimated model in the control loop can lead to a lack of stability and accuracy, and increases the complexity of the control scheme. Second, the dynamics in Equation 14 only concerns the end-effector inertia and the legs inertia is neglected. However, this assumption seems to be too restrictive, particularly in the machine-tool case where legs are generally heavier than the end-effector. The compensation of the machine dynamic behaviour might hence not be efficiently achieved. Thus, the accuracy of this control strategy is questionable. Let us see whether it is more relevant to take into account the legs simplified dynamics.

### 3.1.2 With the leg dynamics only

The simplified inverse dynamics in Equation 1 is reused here. The joint acceleration \( \ddot{q} \) are expressed as a function of the end-effector pose with the second order inverse instantaneous kinematics:

\[ \Gamma = M \left(D_{inv}(X)\dot{X} + \dot{D}_{inv}(X, \dot{X})\dot{X}\right) \]  

(18)

The feedforward term is also expressed in as a function of the end-effector pose:

\[ u_{ff} = M \left(D_{inv}(X)\ddot{X}_d + \dot{D}_{inv}(X, \dot{X})\dot{X}_d\right) \]  

(19)

In this case, the error signal behaviour is fixed by the following equation:

\[ MD_{inv}(X)e^{(3)} + (K_v + MD_{inv}(X, \dot{X}))\ddot{e} + K_p \dot{e} + K_i e = 0 \]  

(20)

Consequently, the following tuning has to be used:

\[
\begin{align*}
K_v &= 3MD_{inv}(X)\omega - MD_{inv}(X, \dot{X}) \\
K_p &= 3MD_{inv}(X)\omega^2 \\
K_i &= MD_{inv}(X)\omega^3
\end{align*}
\]  

(21)

In a first approach, with this formulation, the tuning is not constant and thus difficult to set up in an industrial context. Nevertheless, we can rearrange each term to propose a lighter control scheme, as illustrated by Figure 7. In this case, the feedforward term is:

\[ u_{ff} = \ddot{X}_d \]  

(22)

A compensation term is added. It is expressed as:

\[ u_{comp} = MD_{inv}(X, \dot{X})\dot{X} \]  

(23)
In this case, the error signal behaviour is fixed by the following equation:

\[ MD_{\text{inv}}(X)(e^{(3)} + K_v \dot{e} + K_p \ddot{e} + K_i e) = 0 \] (24)

Consequently, the gain tuning becomes:

\[
\begin{align*}
K_v &= 3\omega \\
K_p &= 3\omega^2 \\
K_i &= \omega^3
\end{align*}
\] (25)

This proposed control strategy is the direct Cartesian space equivalent of the single-axis control in the joint space. Yet, it can be noticed that the transposition between Cartesian and joint space is not as straightforward as it could have seemed at first glance. The control scheme complexity has clearly increased whereas the dynamics compensation issues listed previously are still present since the same simplified dynamics is used. Moreover, the end-effector dynamics is neglected.

### 3.1.3 With the legs and end-effector dynamics

Now, the two simplified dynamics used above can be grouped together to take into account both legs and end-effector inertia (Figure 8). Thus, a more elaborated expression of the simplified dynamics can be:

\[ \Gamma = M\ddot{q} + D^T I \dddot{X} \] (26)

The control law is nearly the same as the one presented above (compare Figure 7 and Figure 8). The tuning is the same as Equation 25. Nevertheless, a more efficient compensation of the machine dynamic behaviour can be performed here with comparison to the the two previous cases (3.1.1 and 3.1.2) since more phenomena are taken into account. However, the numerical issues of the forward instantaneous kinematic matrix are retrieved here and thus impose some care.

Let us remark that this formulation is used by Marquet and Vivas directly in a computed torque control [Marquet et al., 2001, Vivas et al., 2003]. Nevertheless, since these simplified dynamics presents inevitably important modeling errors, a predictive control, asking for heavy computation, is thus employed to ensure good performances at high speed. Actually, this approach stands on the border between simple control and computed torque control: the control is almost considered as a single-axis control when a simple PID controller is used, and as a computed torque control when a more complex controller is used.

Consequently, the interest of change from the joint to Cartesian space could be questionable, in the simple PID control case. Nevertheless, we show in Section 3.3 some mechanical advantages for the Cartesian space control. Now, instead of using simplified dynamics and time consuming complex controller, the use of the complete dynamics in a computed torque control can allow for using simpler controller with lighter computational burden, while ensuring equivalent or better performances.

### 3.2 Computed Torque Control in the Cartesian Space

The Cartesian space Computed Torque Control is well known for serial kinematic machines [Khalil and Dombre, 2002]. The presence of the numerical inverse instantaneous kinematic matrix \( \tilde{D}_{\text{inv}} \) (see Figure 9) make this control strategy rarely used for serial kinematic machines. Indeed, the forward instantaneous kinematic matrix of a serial kine-
matic machine is generally composed of trigonometric functions, thus making the numerical inversion all the more difficult because of the existence of numerous singularities and nonlinear dependence upon noise.

On the opposite, in the parallel kinematic case, this control scheme is perfectly relevant when it encloses an inverse dynamic model depending on the end-effector pose and time derivatives [Caccavale et al., 2003, Callegari et al., 2006]. Indeed, there is a minimal use of numerical transformations when the end-effector pose and speed are measured (see Figure 10). Actually, the only used numerical transformation is the transposed forward instantaneous kinematics matrix used to map the Cartesian space dynamics into the active joint space [Dasgupta and Choudhury, 1999, Khalil and Ibrahim, 2004, Callegari et al., 2006]. Since inverting this matrix consists only of a numerical inversion of a quite simple matrix, the computational burden is less important than for solving for the forward kinematics problem. Moreover, the Cartesian space Computed Torque Control for parallel kinematic machines is dual with the joint space Computed Torque Control for the serial kinematic machines (see Figure 4 and Figure 10). Consequently, the behaviour of the joint space computed torque control described above is retrieved here, namely the error behaviour in Equation 12 and the tuning in Equation 13, which we recall here:
\[ \ddot{e} + K_v \dot{e} + K_p e = 0 \quad (27) \]

and
\[ \begin{cases} 
K_v = 2 \xi \omega \\
K_p = \omega^2 
\end{cases} \quad (28) \]

Consequently, the known performances of the computed torque control could be thus expected, with the prerequisite of a good dynamic modeling, a good dynamic identification and a good algorithm for the remaining numerical transformation.

3.3 Discussion

3.3.1 Joint space or Cartesian space control?

The Cartesian space control is particularly relevant for parallel kinematic machines. Theoretically, since the end-effector pose is the state of a parallel kinematic machine, the Cartesian space control ensures a state feedback control leading to a better accuracy than a joint space control which is not a state feedback control any more. Moreover, by using a Cartesian space inverse dynamic model in a Cartesian space computed torque control, a minimal use of numerical transformations is required leading to a fast, stable and accurate control, when a good model is used and a good dynamic identification is performed. Furthermore, some additional advantages can be noticed when a fast and accurate end-effector pose measure is available (Assumption 1).

First of all, in a joint space control, the regulated error is the error between a transformed desired trajectory, thus biased by the modeling errors, and a measure not reflecting the real end-effector pose, insensitive to backlashes or deformation. On the opposite, in the Cartesian space control case, the regulated error is the error between the measured and desired end-effector trajectories. Consequently, a Cartesian space control ensures a direct task control and thus can be more accurate than a joint space one.

Secondly, since the inverse kinematic model is not used to compute the joint reference path (see Figure 5 and Figure 10), the constraints on kinematic identification could be released. Indeed, without any kinematic identification, the Cartesian control performs an accurate positioning of the end-effector, when a point to point task is desired, since the reference trajectory is not biased by the inverse kinematic model errors. Furthermore, as far as the trajectory tracking is concerned, the dynamic identification prevails against the kinematic one in the minimisation of the dynamic modeling errors. In addition, a dynamic identification, which is linear, is easier to set up than a kinematic one, which is nonlinear.

Thirdly, a Cartesian space control is more interesting in the neighbourhood of singularities. Indeed, one joint variable configuration leads to several end-effector poses [Husty, 1994]. In the worst cases, a disturbance on joint trajectory can thus shift the end-effector pose without changing joint configuration. This can happen especially in the neighborhood of singularities (assembling mode changing trajectory [Chablat and Wenger, 1998]) or in cups points (non-singular posture changing trajectory [Zein et al., 2006]). This change of the end-effector pose is not observed by a joint space control whereas a Cartesian space one is able to do so (see Figure 11). Consequently, the Cartesian space control tries to bring back the end-effector pose to its reference or fails when the singularity can not be crossed again. On the contrary, a converging joint space control can not tell whether the Cartesian reference tracking fails or not. Consequently, a Cartesian space control can ensure a more reliable tracking than a joint space control.

Last but not least, even on planned path taking into account kinematic and dynamic constraints,
the joint position errors are independent from each other when using a joint space control. Therefore, the kinematic constraints can not be ensured and two types of defects may appear: uncontrolled parasite end-effector moves or internal torques if these moves are impossible, thus degrading passive joints. Like two-arm robot control [Dauchez et al., 1989], Cartesian space control can minimize, or cancel in the best cases, internal torques [Marquet et al., 2001]. Indeed, the regulated errors, which are end-effector pose errors, are naturally compatible with the end-effector moves.

Consequently, the theoretical advantages of the Cartesian space control over the joint space one are now undoubtful. Therefore, the Cartesian space control seems perfectly relevant for the parallel kinematic machines and should always be used. However, the discussion made above assumes to have an available fast and accurate observation of the end-effector pose. This point remains the main issue making the Cartesian space control use occasional. Indeed, the measure of the end-effector pose is not an easy deal.

### 3.3.2 Comments on Assumption 1

In the literature, the observation is generally indirect: the end-effector pose is estimated throughout the forward kinematics problem solving [Lee et al., 2003, Kock and Shimacher, 2000a, Marquet et al., 2001, Beji et al., 1998, Yamane et al., 1998, Callegari et al., 2006, Caccavale et al., 2003]. Thus, the numerical estimation issues, such as computation time, stability, reliability and accuracy, are retrieved here. In such a case, the property of a stable and accurate control is called into question and should be investigated. Nevertheless, adapted algorithms [Merlet, 2004] or metrological redundancy [Baron and Angeles, 2000, Marquet et al., 2002] can decrease the forward kinematics complexity and computation. Thus, it can improve the accuracy and stability of the control. However, the

---

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X = [X_e \ Y_e \ Z_e \ \theta]^T$</td>
<td>Cartesian variables describing the end-effector pose $X$</td>
</tr>
<tr>
<td>$X_0, Y_0, Z_0, \delta Z$</td>
<td>Constant parameters describing the reference position of each leg in the fixed basis</td>
</tr>
<tr>
<td>$q_{i1} = q_i, q_{2i}, q_{3i}$</td>
<td>Joint variables describing the leg $i$</td>
</tr>
<tr>
<td>$q_{jki} = q_{ji} + q_{ki}$</td>
<td>Joint variables sum</td>
</tr>
<tr>
<td>$s_{ji} = \sin q_{ji}, c_{ji} = \cos q_{ji}$</td>
<td>Trigonometric operations on joint variables</td>
</tr>
<tr>
<td>$\Omega = [0 \ \dot{\theta} \ 0]^T$</td>
<td>Angular velocity of the end-effector</td>
</tr>
<tr>
<td>$L$</td>
<td>Length of the end-effector</td>
</tr>
<tr>
<td>$d_3, d_4$</td>
<td>Length of arm and forearm of a leg</td>
</tr>
<tr>
<td>$MR_1$</td>
<td>Mass of a leg</td>
</tr>
<tr>
<td>$MXR_2, MYG_2$</td>
<td>First moments of the arm of each leg around $X$ (grouped with other terms) and $Y$ axis</td>
</tr>
<tr>
<td>$MXG_3, MYG_3$</td>
<td>First moments of the forearm of each leg around $X$ and $Y$ axis</td>
</tr>
<tr>
<td>$ZZ_2, ZZR_3$</td>
<td>Inertia term of the forearm and the arm (grouped with other terms) of each leg around $Z$ axis</td>
</tr>
<tr>
<td>$M_{comp_1}, M_{comp_2}$</td>
<td>Equivalent mass of the forces applied by the compensator mounted on the vertical legs</td>
</tr>
<tr>
<td>$M_P$</td>
<td>Mass of the end-effector</td>
</tr>
<tr>
<td>$ML = MR_1 + MP$</td>
<td>Mass of the end-effector and a leg</td>
</tr>
<tr>
<td>$MS_P = [MPXP \ MPYP \ MPZP]$</td>
<td>Vector of the first moments of the end-effector around the fixed basis frame</td>
</tr>
<tr>
<td>$I_P$</td>
<td>Inertia matrix of the end-effector</td>
</tr>
<tr>
<td>$YY_P$</td>
<td>Inertia term of the end-effector around $Z$-axis</td>
</tr>
<tr>
<td>$g$</td>
<td>Acceleration of gravity</td>
</tr>
</tbody>
</table>

Table 2: Notation for the modeling of the Isoglide-4 T3R1
use of a kinematic model imposes a heavy modeling and an accurate kinematic identification since the measure is biased by the kinematic errors.

On the opposite, instead of using a mechanical model, a direct measure can be used. To our knowledge, the means to measure the end-effector pose are few and far between. The laser-tracker and computer vision are the main. On the one hand, the laser tracker allows for a very accurate and fast Cartesian position measure (20µm and 3kHz [Far, ]). Nevertheless, it is very expensive and hard to use. In addition, the orientation measure is not mastered. To our knowledge, it is only used for kinematic identification [Newman et al., 2000] and was never used in the control loop.

On the other hand, computer vision is no as accurate and fast but is very easy to implement in a control scheme. It is a well known solution for the kinematic control of serial kinematic machine, namely visual servoing [Weiss et al., 1987, Espiau et al., 1992, Hutchinson et al., 1996]. Recent works deal with visual servoing of parallel kinematic machine and show good properties [Kino et al., 1999, Dallej et al., 2006]. However, only kinematic control is concerned. On the opposite, Ginhoux and Gangloff proposed a fast visual servoing of serial kinematic machine [Ginhoux et al., 2004]. However, the dynamics are compensated for with a robust controller and not with a computed torque control. To our mind, the application of such a control to parallel kinematic machines is not relevant according to what we stated above. A more relevant solution could be a visual computed torque control as initiated by Fakhry for serial kinematic machine [Fakhry and Wilson, 1996]. To our knowledge, there is no work on the fast visual servoing of parallel kinematic machine whereas good performances could be expected [Ait-Aider et al., 2006, Paccot et al., 2006].

To conclude, the Cartesian space control of parallel kinematic machines seems to be a relevant solution improving accuracy, stability, speed, reliability and mechanical behaviour. Let us now validate experimentally the theoretical discussion above.

4 Modeling of the test-bed

The notation used is this section is described in Table 2.

4.1 Presentation of the test-bed

The test-bed is the Isoglide-4 T3R1 (see Figure 12 and [Gogu, 2004]). This parallel kinematic machine...
is a fully-isotropic one with decoupled motion. It is a four degrees of freedom machine with three translations and one rotation. This machine is composed of four identical legs. Each leg contains one actuated prismatic joint and two passive revolute joint, linked to the end-effector by one universal joint (see Figure 13 and Figure 14). The actuation is performed with linear motors.

This machine is designed for high speed machining. Hence, the structure weight is important to meet the stiffness requirements: 31 kg for each leg and 14 kg for the end-effector. Therefore, the leg dynamics have a great influence and can thus not be neglected, contrary to common light parallel kinematic machines for pick-and-place. Consequently, a complete dynamic modeling and a performant dynamic control is required to ensure the high accuracy required in machining. This test-bed is thus relevant for validating the assumption made on the weakness of single-axis linear control.

The main advantage of the Isoglide-4 T3R1, as far as control is concerned, is to have a closed-form expression of the forward kinematic and instantaneous kinematic models:

\[
\begin{align*}
X_e &= q_1 - X_0 \\
Y_e &= q_2 - Y_0 \\
Z_e &= q_3 - Z_0 \\
sin\theta &= \frac{q_4 - q_3 + \delta Z}{L}
\end{align*}
\]

and

\[
D(X) = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & \frac{-1}{L \cos \theta} & \frac{1}{L \cos \theta}
\end{bmatrix}
\]

Therefore, the simple and closed-form expression of the forward kinematics are interesting for validating the proposed Cartesian space control schemes. Indeed, it removes from the numerical estimation issues and the lack of fast and accurate end-effector pose measure. Therefore, a more fairly comparison between forward kinematic model based control and exteroceptive measure based control could be achieved. Actually, the numerical estimations issues influence on control behaviour, which are hard to quantify, are removed. The comparison can only be achieved in term of sensor and identification accuracy with regards to the control accuracy.

4.2 Dynamic modeling

Achieving a performant computed torque control requires a Cartesian space dynamic modeling method with an easy implementation, a low computation cost and minimal simplifications. In this way, Khalil's method [Khalil and Ibrahim, 2004] is preferred on other known methods [Dasgupta and Choudhury, 1999, Tsai, 2000, Callegari et al., 2006]. Indeed, it is based on the Newton-Euler algorithm, known to be relevant in a control context. Moreover, the application of the method is easy due a very simple formulation.

According to Khalil, the inverse dynamic model can be simply expressed as [Khalil and Ibrahim, 2004]:

\[
\Gamma = \hat{D}^T \left( F_p + \sum_{i=1}^{n} J_{pi}^T D_i^T (H_i + G_i) \right) + \Gamma_f \tag{31}
\]

where:

- \( F_p \) are the dynamics of the end-effector
- \( n \) is the number of legs
- \( D_i \) is the inverse instantaneous kinematic matrix of leg \( i \)
- \( J_{pi} \) is a Jacobian matrix linking the Cartesian coordinates of the end of the leg \( i \) to the Cartesian coordinates of the end-effector.
- \( H_i \) are the dynamics of the leg \( i \).
- \( G_i \) is the gravity vector of the leg \( i \).

This modeling method is thus achieved through a complete modeling of each leg and a determination of the end-effector dynamics.

4.2.1 Modeling of each leg

A leg can be seen as a stand-alone 3-PRR serial kinematic machine. The modeling of such a serial kinematic machine is well known. Consequently, we only give the obtained models without details on the method. The kinematics are determined with Khalil-Kleinfinger notation and the dynamics with the Newton-Euler algorithm and the notation in [Khalil and Dombre, 2002].

The inverse instantaneous kinematic matrix of leg 1 expresses as:
Each term of the inverse dynamics of the first leg, $H_1 = [H_{11} \ H_{12} \ H_{13}]^T$ are detailed below:

$$H_{11} = MR_1 \ddot{q}_{11}$$

$$H_{12} = (Z Z_3 + 2d_3 MXG_3 c_{31} - 2d_3 MYG_3 s_{31}) \ddot{q}_{21} + (Z Z_3 + d_3 MXG_3 c_{31} - d_3 MYG_3 s_{31}) \ddot{q}_{31} + (d_3 MXG_3 s_{31} + d_3 MYG_3 c_{31}) \ddot{q}_{21} - (d_3 MXG_3 s_{31} + d_3 MYG_3 c_{31}) \ddot{q}_{31}$$

$$H_{13} = (Z Z_3 + d_3 MXG_3 c_{31} - d_3 MYG_3 s_{31}) \ddot{q}_{21} + (Z Z_3) \ddot{q}_{31} + (d_3 MXG_3 s_{31} + d_3 MYG_3 c_{31}) \ddot{q}_{21}$$

The passive joint variables are expressed as function of the end-effector pose with simple trigonometric relations. Other legs have similar models. The change comes from the position of the legs in the Cartesian space, modifying the gravity terms and Jacobian matrix organisation. The gravity vector, $G_i$, for each leg, are detailed below:

$$G_1 = \begin{bmatrix} 0 \\ \Omega \times (\Omega \times M_1 S_1) \\ -S_1 \end{bmatrix} (40)$$

$$G_2 = \begin{bmatrix} 0 \\ \Omega \times (\Omega \times M_2 S_2) \\ -S_2 \end{bmatrix} (41)$$

$$G_3 = \begin{bmatrix} -MR_1 + M_{comp3} g \\ 0 \\ 0 \end{bmatrix} (42)$$

$$G_4 = \begin{bmatrix} -MR_1 + M_{comp4} g \\ 0 \\ 0 \end{bmatrix} (43)$$

Two gravity compensators are used for the vertical legs, namely the third and fourth, explaining the presence of terms $M_{comp3}$ and $M_{comp4}$. Their influence is approximated as constant.

The Jacobian matrices $J_{pi}$ have a very simple expression. Indeed, the end of each leg is defined by its Cartesian position. The end-effector is defined by four Cartesian variables $[X_e \ Y_e \ Z_e \ \theta]^T$. For the first and third legs, there is only an offset between the Cartesian position of the legs and the end-effector and the orientation $\theta$ has no influence. For the two other legs, the orientation $\theta$ has an influence which is easily determined with trigonometric relations. Consequently, the matrices have the following expressions:

$$J_{p1} = J_{p3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} (40)$$

$$J_{p2} = J_{p4} = \begin{bmatrix} 1 & 0 & 0 & Ls_\theta \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & Lc_\theta \end{bmatrix} (41)$$

4.2.2 Dynamics of the end-effector

The dynamics of the end-effector are determined with the Newton-Euler equation [Khalil and Dombre, 2002, Khalil and Ibrahim, 2004]:

$$\mathbf{F}_p = \gamma_p \ddot{X} + \left[ \Omega \times (\Omega \times M S_p) \right] - \frac{M P I_3}{MS_p} g (42)$$

where $\gamma_p$ is the inertia tensor of the end-effector expressed as:

$$\gamma_p = \begin{bmatrix} M P I_3 \\ S(M S_p) \\ I_p \end{bmatrix} (43)$$

Only the terms along the end-effector degrees of freedom are retained which yields the final expression of the end-effector dynamics in Equation 44.

4.2.3 Inverse dynamic model of the Isoglide-4 T3R1

The inverse dynamic model is obtained by the Equation 31. For conciseness sake, the global expression is not mentioned here. Nevertheless, the obtained model has a closed-form expression allowing for an interpretation of each term. The main term is the inertia of the legs and the end-effector in translation, this term thus has a great influence on the tracking performances. Then the other main terms concern the coupling between legs due to the heavy inertia. Even if a kinematic decoupling is ensured, the dynamic coupling stays present and can not be compensated for with a linear control. The last term concerns the rotation inertia of the end-effector, this term is not preponderant and can be neglected if necessary.
The reference force $F_p$ is given by:

$$F_p = \begin{pmatrix}
M_P \ddot{X}_e + (M_P X_P s_\theta + M_P Z_P c_\theta) \ddot{\theta} + (-M_P X_P c_\theta + M_P Z_P s_\theta) \dot{\theta}^2 \\
M_P (\ddot{Z}_e - g) + (-M_P X_P c_\theta + M_P Z_P s_\theta) \ddot{\theta} - (M_P X_P s_\theta + M_P Z_P c_\theta) \dot{\theta}^2 \\
YY_P \ddot{\theta} + (M_P X_P s_\theta + M_P Z_P c_\theta) \dot{X}_e + (-M_P X_P c_\theta + M_P Z_P s_\theta) (\ddot{Z}_e - g)
\end{pmatrix} \quad (44)$$

Figure 15: Dynamic model of the machine used in simulation

A simple friction forces model is implemented to compensate for the latter and improve accuracy:

$$\Gamma_f = (F_v \dot{q} + F_s) \text{sign}(\dot{q}) \quad (45)$$

5 Results

5.1 Simulation

First, the improvement of the joint space computed torque control (Figure 5) over the single-axis (Figure 2) control will be shown. A comparison will be achieved in term of straightness error and tracking error on a relevant trajectory. Second, a comparison between computed torque control in the joint space and in the Cartesian space (Figure 5 and Figure 10) will be performed. The Cartesian space computed torque control will be performed with the forward kinematics and with a direct measure to emphasis the improvement when using the direct measure rather than the forward kinematics.

The machine behaviour is simulated with the forward dynamic model obtaining by inverting Equation 31 which allows for computing the end-effector acceleration (see Figure 15). Realistic noises and errors are used, such as a 10% error on dynamic parameters, 50 $\mu$m accuracy for geometric parameters (required manufacturing and assembly tolerance for the Isoglide-4 T3R1). Let us stress out that neither deformations nor assembly errors (such as non perpendicular axis) are simulated. A 1$\mu$m accuracy is fixed for the joint sensors, and a 20$\mu$m and 10$^{-4}$ $rad$ accuracy for the direct measure (laser tracker performances). The tuning is done with $\omega = 5$Hz and the control and sensors have a 1kHz sampling rate.

The reference trajectory, Figure 16, is composed of one translation along the X axis, one translation along X, Y and Z axes and one translation along the three axes with a rotation. The first part of the trajectory allows for pointing out the ability of the control strategy to compensate for the dynamic coupling between legs. The second part allows for comparing the compensation of the inertia forces. And the last part allows for comparing joint space control and Cartesian space one performed with forward kinematics and direct end-effector pose measure. Indeed, there is only a difference, between forward and inverse kinematics, on the rotation since the Isoglide-4 T3R1 has decoupled translations (see Equation 29).

A fifth degree polynomial point to point interpolation is used to have a smooth trajectory. The maximal acceleration is fixed to 3m.s$^{-2}$ to simulate a machining operation. This low speed is far from high-speed pick-and-place one.
5.1.1 Joint space computed torque control versus single axis control

Figure 17 shows the trajectory in the XY plane performed by the two control strategies, single-axis

---

<table>
<thead>
<tr>
<th></th>
<th>Single-axis</th>
<th>Joint space CTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>First segment</td>
<td>0.759mm</td>
<td>0.026mm</td>
</tr>
<tr>
<td>Second segment</td>
<td>1.900mm</td>
<td>0.089mm</td>
</tr>
<tr>
<td>Third segment</td>
<td>3.758mm</td>
<td>0.180mm</td>
</tr>
</tbody>
</table>

Table 3: Straightness error on each segment measured in simulation

Figure 18: Tracking error on X axis for the single-axis linear control and joint space computed torque control, and bias between reference and performed trajectory
control and the computed torque control in the joint space, and the reference. On Figure 17(a), the two control strategies are biased with regards to the reference trajectory. Figure 17(b) shows the details of a sharp corner crossing, with zero speed, and reveals a difference between single-axis and computed torque control. Indeed, the trajectory followed with the single-axis control is not completely straight and presents some oscillations in the corner, contrary to the joint space computed torque control. This is numerically verified in Table 3. The computed torque control allows for straightness errors about twenty times less important than single-axis control does. Figure 18 shows the tracking error on X axis for the same two control strategies. The single-axis control presents important tracking errors with a maximum along the four axes displacement (2.5 mm peak to peak). On the opposite, the computed torque control allows for small tracking errors. These errors are distributed around the constant bias, 318 µm, between the performed trajectory and the reference trajectory.

Thus, using a computed torque control instead of simple control improves dynamic accuracy of the machine. Indeed, the high dynamic coupling between legs, due to important masses, is clearly not negligible, even at machining speed often considered as quasi-static. Therefore, the complete machine dynamics should be taken into account in the control loop. Moreover, the strong influence of the kinematic identification is retrieved here. Actually, there is a bias between the Cartesian reference and the joint reference, due to remaining kinematic errors. In the Isoglide-4 T3R1 case, this bias is constant and can be seen as an adjustable offset. However, in more complex cases, this bias is not constant along the workspace and can thus not be compensated for without a more accurate kinematic identification asking for more accurate measure and heavier computation.

5.1.2 Joint space computed torque versus Cartesian one

Figure 19 show the performed trajectory in the XY plane by the joint space computed torque control, the Cartesian space computed torque control with forward kinematics and the Cartesian space computed torque control with direct end-effector pose measure. The first two control strategies have exactly the same behaviour and bias with regards to the reference trajectory. On the opposite, the trajectory performed by the computed torque control with direct end-effector pose measure does not present a bias and is mixed with the reference trajectory. Figure 20 shows the orientation tracking error for the computed torque control in the joint and the Cartesian space. Joint space and Cartesian space, with forwards kinematics, computed torque control have exactly the same behaviour. The computed torque control with direct end-effector pose measure allows for better accuracy than the two other control strategies, since there is no bias and there are better tracking performances during the rotation at the end of the trajectory, between times 1.3 and 1.8 seconds. The tracking performances are numerically summarized in Table 4 and Table 5. The joint space computed torque and the Cartesian space computed torque control with forward kinematics have strictly the same performances. On the opposite, the use of a direct end-effector measure allows for better mean errors (13µm versus 322µm) and similar standard deviation for X-axis and straightness errors. Concerning the orientation, the direct measure allows for a tracking error thirty
Figure 20: Comparison, on orientation error tracking, between joint space and Cartesian space control, with forward kinematics and direct measure of the end-effector pose.

Table 4: Straightness error on each segment measured in simulation

<table>
<thead>
<tr>
<th></th>
<th>joint space CTC</th>
<th>CTC with FKM</th>
<th>CTC with direct measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>First segment</td>
<td>0.026mm</td>
<td>0.026mm</td>
<td>0.040mm</td>
</tr>
<tr>
<td>Second segment</td>
<td>0.089mm</td>
<td>0.089mm</td>
<td>0.083mm</td>
</tr>
<tr>
<td>Third segment</td>
<td>0.180mm</td>
<td>0.180mm</td>
<td>0.147mm</td>
</tr>
</tbody>
</table>

The use of the Cartesian space control with a direct measure of the end-effector pose instead of a joint space control does improves static and dynamic accuracy, whereas a forward kinematics based control only allows for good dynamic performances but reduced geometric accuracy. Indeed, the use of a direct measure allows for compensating for the kinematic errors without extremely accurate kinematic identification concern. The performed trajectory is not shifted with respect to the reference one. Furthermore, it needs to be underlined that even though the direct measure is less accurate than the joint sensors, using it for control ensures equivalent translation tracking performances and better orientation ones. Actually, these performances are closed to the direct measure accuracy. In this case, the sensor accuracy has thus more influence on the control accuracy than the modeling errors. Finally, the numerical results shows that a 100µm accuracy, which is the minimum required in machining, can be achieved with the computed torque control without particular care. Nevertheless, this can be still improved with less error on parameters (in other words a better identification), a more elaborated gain tuning and a more accurate sensor.

5.2 Experiments

5.2.1 Dynamic identification

In order to fit the inverse dynamic model to the real dynamics of the machine and ensure the best performances for computed torque control, dynamic identification was realized (see Table 6). The method used here was proposed by Guégan [Guégan et al., 2003]. The chosen exciting trajectory is composed of axis-by-axis displacements with acceleration ranging from 0.5 to 3 m.s\(^{-2}\). A simulation on different trajectories (circles, axis-by-axis displacements, coupled axis displacement and random trajectory) shows that the axis-by-axis displacements obtained the best condition number. Indeed, it allows for having both free and constrained moves on each axis. The parameters concerning the dynamic coupling are thus determined with the torques recorded during free moves. Those concerning the inertia terms are determined during constrained moves.

<table>
<thead>
<tr>
<th>Control</th>
<th>Mean of tracking error on X</th>
<th>Standard deviation of tracking error on X</th>
<th>Mean of tracking error on θ</th>
<th>Standard deviation of tracking error on θ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joint space CTC</td>
<td>0.322mm</td>
<td>0.024mm</td>
<td>2°</td>
<td>0.079°</td>
</tr>
<tr>
<td>CTC with FKM</td>
<td>0.322mm</td>
<td>0.024mm</td>
<td>2°</td>
<td>0.079°</td>
</tr>
<tr>
<td>CTC with direct measure</td>
<td>0.013mm</td>
<td>0.026mm</td>
<td>0°</td>
<td>0.006°</td>
</tr>
</tbody>
</table>

Table 5: Tracking errors along the trajectory
Results lead to an observation matrix condition number of 355.56 which is relatively good. Inertia parameters ($M XR_3$, $ZZR_3$, $ZZR_2$, $M_t$, $M R_1$) are identified with a standard deviation from 0.40% to 1.29%, friction terms ($F s_i$ and $F v_i$) from 1.07% to 6.34%. Let us remark that some parameters describing the end-effector can not be identified because the end-effector is lighter than the legs, thus having a little influence on dynamics. Anyhow, the good results of the identification process allows for the fulfilment of the small modeling errors condition, necessary to ensure a stable and accurate computed torque control.

### 5.2.2 Experiments

The simulation showed that a Cartesian space computed torque control with forward kinematics and a joint space computed torque control have the same behaviour. The expected improvements could only be established with a control using a direct measure of the end-effector pose. At the moment, the computer vision is not accurate and fast enough to set up relevant experiments and a laser tracker is too expensive. Consequently, we can only propose an experimental comparison between single-axis control and computed torque control in the Cartesian space with the forward kinematics. To achieve this comparison, the end-effector trajectory is measured with a 512×512 camera as an exteroceptive measure running at 250 Hz. This provides us with a measure of the real end-effector trajectory instead of a model biased estimation. A comparison between the camera and a laser interferometer is performed (see Figure 21). Figure 22 shows that the camera has an average accuracy of 26 µm compared to the interferometer measure thus validating further results.

Both control schemes have the same gain tuning with same cut-off frequency ($\omega_c$) of 5 Hz. Nevertheless, the derivative gain in the single-axis controller can not be set at its theoretical value because the linear actuators we use do not cope with noise, even filtered. The reference trajectory is a simple 100 mm square in the XY frame. A fifth degree path generation with a $3 m.s^{-2}$ maximal acceleration is used. The trajectory is executed segment by segment.

Figure 23 shows the performed trajectory in the XY plane for the two control strategies, single-
axis and computed torque control, compared with the reference trajectory. Let us stress out that the Figure 23 only represents the relative displacements. Thus, the bias due to the geometrical errors is not measured. The computed torque control achieves an accurate tracking while the single-axis can not. Indeed, the computed torque control performs straight displacements whereas the single-axis control presents some oscillations around the reference. Numerically, the straightness error are divided by 7 for X-axis displacement and 10 for Y-axis displacement (see Table 7). Figure 24 shows the time evolution of the end-effector position along the X axis for the reference trajectory, the single axis control and the computed torque control. It can be noticed that the computed torque control presents good tracking performances whereas the single-axis control presents important tracking errors and overshoot.

Thus, these experiments validate the simulation results above. In other words, using computed torque control instead of a linear single-axis control improves tracking. It can be noticed that the obtained results are worse than those expected. Indeed, simulations do not take into account assembly defects, making the difference between the simulated model and the control model smaller than the difference between the real machine and the control model. The assembly defects are treated in the kinematic models [Rizk et al., 2006] and should
Table 7: Measured straightness error on the square segments with a high speed camera

<table>
<thead>
<tr>
<th></th>
<th>PID</th>
<th>CTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left edge</td>
<td>0.733mm</td>
<td>0.154mm</td>
</tr>
<tr>
<td>Right edge</td>
<td>2.255mm</td>
<td>0.330mm</td>
</tr>
<tr>
<td>Bottom edge</td>
<td>3.318mm</td>
<td>0.443mm</td>
</tr>
<tr>
<td>Top edge</td>
<td>3.143mm</td>
<td>0.293mm</td>
</tr>
</tbody>
</table>

be extended to dynamics. In addition, a more accurate identification with exteroceptive measure, such as computer vision [Renaud et al., 2006], could be used to improve control accuracy.

**Conclusion**

In this article, we aimed at showing that control of parallel kinematic machine should be thought again. To our mind, a computed torque control in the Cartesian space, with an exteroceptive end-effector pose measure and a Cartesian space dynamic model, is the relevant solution to ensure the best performances and machine capabilities use.

Indeed, the inherent complexity of the closed mechanical structure leads to highly nonlinear dynamics with dynamic leg coupling. Therefore a single-axis control can not ensure correct performances
while using the whole workspace and the machine maximal speed capabilities. In this way, the computed torque control is known to be a relevant solution for serial kinematic machines. However, the computed torque control is often forsaken by the parallel kinematic machine community since it often requires robust control. Nevertheless, these control schemes of serial robotics are classically performed in the joint space and thus should not be reused directly for parallel kinematic machines.

Actually, since a parallel kinematics machine is defined by its end-effector configuration, using a Cartesian space control is more relevant than using the classical joint space one. In addition, when it is performed with an exteroceptive end-effector pose measure, the modeling errors sources are minimised leading to a more stable control than the joint space one, making the robust control useless or, at least, used only for unknown disturbances rejection. Furthermore, we showed that a Cartesian space control allows for a better mechanical structure handling than the joint space one.

Simulations were performed to validate the above discussion and some of them were validated experimentally. Nevertheless, further experiments should be performed to validate the improvement of the use of an exteroceptive measure of the end-effector, such as computer vision or laser tracker. These experiments should be preceded by a more performant identification. The test-bed is very particular, therefore experiments on other structures, such as Gough-Stewart platform, should be done to validate the genericity of the approach and validate the internal torque minimisation and the behaviour in the neighborhood of singularities. Last but not least, a theoretical demonstration of the control accuracy, stability and robustness in regards with measure and modeling errors could be performed as initiated in [Paccot et al., 2007].

References


