Stiffness Analysis Of Multi-Chain Parallel Robotic Systems

Anatoly Pashkevich, Damien Chablat, Philippe Wenger

▶ To cite this version:

HAL Id: hal-00326672

https://hal.archives-ouvertes.fr/hal-00326672

Submitted on 4 Oct 2008

Stiffness Analysis Of Multi-Chain Parallel Robotic Systems

Anatol Pashkevich*, Damien Chablat**
Philippe Wenger***

*Ecole des Mines de Nantes,
Nantes, France, (e-mail: anatol.pashkevich@emn.fr)
** Institut de Recherches en Communications et Cybernétique de Nantes,
Nantes, France, (e-mail: damien.chablat@irccyn.ec-nantes.fr)
*** Institut de Recherches en Communications et Cybernétique de Nantes,
Nantes, France, (e-mail: philippe.wenger@irccyn.ec-nantes.fr)

Abstract: The paper presents a new stiffness modelling method for multi-chain parallel robotic manipulators with flexible links and compliant actuating joints. In contrast to other works, the method involves a FEA-based link stiffness evaluation and employs a new solution strategy of the kinetostatic equations, which allows computing the stiffness matrix for singular postures and to take into account influence of the internal forces. The advantages of the developed technique are confirmed by application examples, which deal with stiffness analysis of the Orthoglide manipulator. *Copyright © 2008 IFAC*

Keywords: Parallel robotic manipulators, stiffness analysis, kinetostatic modelling, Orthoglide robot.

1. INTRODUCTION

In modern manufacturing systems, parallel manipulators have become more and more popular for a variety of technological processes, including high-accuracy positioning and high-speed machining (Brogardh, 2007; Chanal et al., 2006). This growing attention is inspired by their essential advantages over serial manipulators, which have already reached the dynamic performance limits   In contrast, parallel manipulators are claimed to offer better accuracy, lower mass/inertia properties, and higher structural rigidity (i.e. stiffness-to-mass ratio) (Merlet, 2000).

These features are induced by their specific kinematic structure, which resists the error accumulation in kinematic chains and allows convenient actuators location close to the manipulator base. This makes them attractive for innovative robotic systems, but practical utilization of the potential benefits requires development of efficient stiffness analysis techniques, which satisfy the computational speed and accuracy requirements of relevant design procedures.

Generally, the stiffness analysis evaluates the effect of the applied external torques and forces on the compliant displacements of the end-effector. Numerically, this property is defined through the "stiffness matrix" K, which gives the relation between the translational/rotational displacement and the static forces/torques causing this transition. As follows from mechanics, K is 6 6 semi-definite non-negative matrix, where structure may be non-diagonal to represent the coupling between the translation and rotation (Duffy, 1996). Besides, this matrix may be not-symmetrical under the static load (Griffis & Duffy, 1993), but standard stiffness analysis focuses on the non-loaded structures. Similar to other manipulator properties (kinematical, for instance), the

stiffness essentially depends on the force/torque direction and on the manipulator configuration. Hence, to provide the designer with integrated performance criteria, various scalar indices are usually computed (such as the best/worst/average stiffness with respect to the rotation or translation). Besides, since the matrix K varies through the workspace, corresponding global benchmarks must be computed (Alici & Shirinzadeh, 2005).

Several approaches exist for the computation of the stiffness matrix, such as the Finite Element Analysis (FEA), the matrix structural analysis (MSA), and the virtual joint method (VJM). The FEA method is proved to be the most accurate and reliable, since the links/joints are modeled with its true dimension and shape. Its accuracy is limited by the discretisation step only. However, because of high computational expenses required for the repeated re-meshing, this method is usually applied at the final design stage.

The MSA method incorporates the main ideas of the FEA but operates with rather large flexible elements (beams, arcs, cables, etc.). This obviously yields reduction of the computational expenses and, in some cases, allows even obtaining an analytical stiffness matrix. This method gives a reasonable trade-off between the accuracy and computational time, provided that link approximation by the beam elements is realistic. Because it involves rather high-dimensional matrix operations, it is not attractive for the parametric stiffness analysis.

Finally, the VJM method, which is also referred to as the "lumped modeling", is based on the expansion of the traditional rigid model by adding virtual joints, which describe the elastic deformations of the manipulator components (links, joints and actuators). This approach originates from the work of Gosselin (1990), who evaluated

parallel manipulator stiffness taking into account only the actuators compliance. At present, there are a number of variations and simplifications of the VJM method, which differ in modelling assumptions and numerical techniques. Generally, the lumped modelling provides acceptable accuracy in short computational time. However, it is very hypothetic and operates with simplified stiffness models that are composed of one-dimensional springs that do not take into account the coupling between the rotational and translational deflections.

This paper presents a new stiffness modelling method, which is based on a multidimensional lumped-parameter model that replaces the link flexibility by localized 6-dof virtual springs that describe both the linear/rotational deflections and the coupling between them. The spring stiffness parameters are evaluated using FEA modelling to ensure higher accuracy. In addition, it employs a new solution strategy of the kinetostatic equations, which allows computing the stiffness matrix for the overconstrained architectures, including the singular manipulator postures. This gives almost the same accuracy as FEA but with essentially lower computational effort because it eliminates the model re-meshing through the workspace.

2. STIFFNESS MODEL

2.1 Manipulator Architecture

Let us consider a general n-dof parallel manipulator, which consists of a mobile platform connected to a fixed base by n identical kinematics chains. Each chain includes an actuated joint "Ac" (prismatic or rotational) followed by a "Foot" and a "Leg" with a number of passive joints "Ps" inside. Generally, certain geometrical conditions are assumed to be satisfied with respect to the passive joints to eliminate the undesired platform rotations and to achieve stability of desired motions. Typical examples of such architectures include 3-PUU translational parallel kinematic machine; (Li & Xu,, 2008), Delta parallel robot (Clavel, 1988), Orthoglide parallel manipulator that implements the 3-PRPaR architecture with parallelogram-type legs and translational active joints (Chablat & Wenger, 2003). Here R, P, U and Pa denote the revolute, prismatic, universal and parallelogram joints, respectively.

2.2 Basic Assumptions

To evaluate the manipulator stiffness, let us apply a modification of the virtual joint method (VJM), which is based on the lump modeling approach (Gosselin, 1990). According to this approach, the original rigid model should be extended by adding the virtual joints (localized springs), which describe elastic deformations of the links. Besides, virtual springs are included in the actuating joints to take into account stiffness of the control loop. . Under such assumptions, each kinematic chain of the manipulator can be described by a serial structure, which includes sequentially:

(a) a rigid link between the manipulator base and the ith actuating joint (part of the base platform) described by the constant homogenous transformation matrix $\mathbf{T}_{base}^i$;

(b) a 1-d.o.f. actuating joint with supplementary virtual spring, which is described by the homogenous matrix function $\mathbf{V}_a(q_0^i + \theta_0^i)$ where q_0^i is the actuated coordinate and θ_0^i is the virtual spring coordinate;

(c) a rigid "Foot" linking the actuating joint and the leg, which is described by the constant homogenous transformation matrix $\mathbf{T}_{foot}$;

(d) a 6-d.o.f. virtual joint defining three translational and three rotational foot-springs, which are described by the homogenous matrix function $\mathbf{V}_s(\theta_1^i, \ldots \theta_6^i)$, where $\{\theta_1^i, \theta_2^i, \theta_3^i\}$ and $\{\theta_4^i, \theta_5^i, \theta_6^i\}$ correspond to the elementary translations and rotations respectively;

(e) a 2-d.o.f. passive U-joint at the beginning of the leg allowing two independent rotations with angles $\{q_1^i, q_2^i\}$, which is described by the homogenous matrix function $\mathbf{V}_{u1}(q_1^i, q_2^i)$;

(f) a rigid "Leg" linking the foot to the movable platform, which is described by the constant homogenous matrix transformation $\mathbf{T}_{leg}$;

(g) a 6-d.o.f. virtual joint defining three translational and three rotational leg-springs, which are described by the homogenous matrix function $\mathbf{V}_s(\theta_7^i, \ldots \theta_{12}^i)$, where $\{\theta_7^i, \theta_8^i, \theta_9^i\}$ and $\{\theta_{10}^i, \theta_{12}^i, \theta_{12}^i\}$ correspond to the elementary translations and rotations, respectively;

(h) a 2-d.o.f. passive U-joint at the end of the leg allowing two independent rotations with angles $\{q_3^i, q_4^i\}$, which is described by the homogenous matrix function $\mathbf{V}_{u2}(q_3^i, q_4^i)$;

(i) a rigid link from the manipulator leg the end-effector (part of the movable platform) described by the homogenous matrix transformation $\mathbf{T}_{tool}^i$.

The expression defining the end-effector location subject to variations of all coordinates of a single kinematic chain may be written as follows

$$\mathbf{T}_i = \mathbf{T}_{base}^i \cdot \mathbf{V}_a(q_0^i + \theta_0^i) \cdot \mathbf{T}_{foot} \cdot \mathbf{V}_s(\theta_1^i, \ldots \theta_6^i) \cdot$$
$$\cdot \mathbf{V}_{u1}(q_1^i, q_2^i) \cdot \mathbf{T}_{leg} \cdot \mathbf{V}_s(\theta_7^i, \ldots \theta_{12}^i) \cdot \mathbf{V}_{u2}(q_3^i, q_4^i) \cdot \mathbf{T}_{tool}^i \quad (1)$$

where matrix function $\mathbf{V}_a(.)$ is either an elementary rotation or translation, matrix functions $\mathbf{V}_{u1}(.)$ and $\mathbf{V}_{u2}(.)$ are compositions of two successive rotations, and the spring matrix $\mathbf{V}_s(.)$ is composed of six elementary transformations.

2.3 Compliance and stiffness models

For each ith manipulator chain, the differential kinematic equation can be written as follows

$$\delta\mathbf{t}_i = \mathbf{J}_\theta^i \cdot \delta\mathbf{\theta}_i + \mathbf{J}_q^i \cdot \delta\mathbf{q}_i, \quad i = 1,2,3, \qquad (2)$$

where vector $\delta\mathbf{t}_i = (\delta p_{xi}, \delta p_{yi}, \delta p_{zi}, \delta\varphi_{xi}, \delta\varphi_{yi}, \delta\varphi_{zi})^T$ describes the translation $\delta\mathbf{p}_i = (\delta p_{xi}, \delta p_{yi}, \delta p_{zi})^T$ and the rotation $\delta\mathbf{\varphi}_i = (\delta\varphi_{xi}, \delta\varphi_{yi}, \delta\varphi_{zi})^T$ of the end-effector with respect to the Cartesian axes; vector $\delta\mathbf{\theta}_i = (\delta\theta_0^i, \ldots \delta\theta_{12}^i)^T$ collects all virtual joint coordinates, vector $\delta\mathbf{q}_i = (\delta q_1^i, \ldots \delta q_4^i)^T$ includes all passive joint coordinates, symbol 'δ' stands for the variation with respect to the rigid case values. The desired matrices $\mathbf{J}_\theta^i$, $\mathbf{J}_q^i$, which are the only parameters of the differential model (2), may be computed from (1) analytically or semi-numerically, using the tree-term fractioning where the first and the third multipliers are the constant homogenous matrices, and the second multiplier is the elementary translation or rotation.

For the kinetostatic model, which describes the force-and-motion relation, it is necessary to introduce additional equations that define the virtual joint reactions to the corresponding spring deformations. In accordance with the adopted stiffness model, three types of virtual springs are included in each kinematic chain: (i) 1-d.o.f. virtual spring describing the actuator compliance K_{act}; (ii) 6-d.o.f. virtual spring describing compliance of the foot $\mathbf{K}_{Foot}$; (iii) 6-d.o.f. virtual spring describing compliance of the leg. For analytical convenience, all relevant expressions may be collected in a single matrix equation

$$\mathbf{\tau}_\theta^i = \mathbf{K}_\theta \cdot \delta\mathbf{\theta}_i, \quad i = 1,2,3 \qquad (3)$$

where $\mathbf{\tau}_\theta^i = (\tau_{\theta 0}^i, \ldots \tau_{\theta 12}^i)^T$ is the aggregated vector of the virtual joint reactions, and $\mathbf{K}_\theta = diag(K_{act}, \mathbf{K}_{Foot}, \mathbf{K}_{Leg})$ is the aggregated spring stiffness matrix of the size 13×13. Similarly, one can define the aggregated vector of the passive joint reactions $\mathbf{\tau}_q^i = (\tau_{q1}^i, \ldots \tau_{q4}^i)^T$ but all its components must be equal to zero:

$$\mathbf{\tau}_q^i = \mathbf{0}, \quad i = 1,2,3 \qquad (4)$$

To find the static equations corresponding to the end-effector motion $\delta\mathbf{t}_i$, let us apply the principle of virtual work assuming that the joints are given small, arbitrary virtual displacements $(\Delta\mathbf{\theta}_i, \Delta\mathbf{q}_i)$ in the equilibrium neighborhood. Then, for the *"unloaded static equilibrium"*, the virtual work of the external force $\mathbf{f}_i$ applied to the end-effector along the corresponding displacement $\Delta\mathbf{t}_i = \mathbf{J}_\theta^i \cdot \Delta\mathbf{\theta}_i + \mathbf{J}_q^i \cdot \Delta\mathbf{q}_i$ is equal to the sum $(\mathbf{f}_i^T \mathbf{J}_\theta^i) \cdot \Delta\mathbf{\theta}_i + (\mathbf{f}_i^T \mathbf{J}_q^i) \cdot \Delta\mathbf{q}_i$. For the internal forces,

the virtual work is $-\mathbf{\tau}_\theta^{i^T} \cdot \Delta\mathbf{\theta}_i$ since the passive joints do not produce the force/torque reactions (the minus sign takes into account the adopted directions for the virtual spring forces/torques). Therefore, because in the static equilibrium the total virtual work is equal to zero for any virtual displacement, the equilibrium conditions may be written as

$$\mathbf{J}_\theta^{i^T} \cdot \mathbf{f}_i = \mathbf{\tau}_\theta^i$$

$$\mathbf{J}_q^{i^T} \cdot \mathbf{f}_i = \mathbf{0} \qquad (5)$$

This gives additional expressions describing the force/torque propagation from the joints to the end-effector.

Hence, the complete kinetostatic model consists of four matrix equations (2) ...(5) where either $\mathbf{f}_i$ or $\delta\mathbf{t}_i$ are treated as known, and the remaining variables are considered as unknowns. Since the matrix $\mathbf{K}_\theta$ is non-singular (it describes the stiffness of the virtual sprigs), the variable $\delta\mathbf{\theta}_i$ can be expressed via $\mathbf{f}_i$ using equations $\mathbf{\tau}_\theta^i = \mathbf{K}_\theta \cdot \delta\mathbf{\theta}_i$ and $\mathbf{J}_\theta^{i^T} \cdot \mathbf{f}_i = \mathbf{\tau}_\theta^i$. This yields substitution $\delta\mathbf{\theta}_i = (\mathbf{K}_\theta^{-1} \mathbf{J}_\theta^{i^T}) \cdot \mathbf{f}_i$ allowing reducing the kinetostatic model to system of two matrix equations

$$(\mathbf{J}_\theta^i \mathbf{K}_\theta^{-1} \mathbf{J}_\theta^{i^T}) \cdot \mathbf{f}_i + \mathbf{J}_q^i \cdot \delta\mathbf{q}_i = \delta\mathbf{t}_i$$

$$\mathbf{J}_q^{i^T} \cdot \mathbf{f}_i = \mathbf{0} \qquad (6)$$

with unknowns $\mathbf{f}_i$ and $\Delta\mathbf{q}_i$. This system can be also rewritten in a matrix form

$$\begin{bmatrix} \mathbf{S}_\theta^i & \mathbf{J}_q^i \\ \mathbf{J}_q^{i\,T} & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{f}_i \\ \delta\mathbf{q}_i \end{bmatrix} = \begin{bmatrix} \delta\mathbf{t}_i \\ \mathbf{0} \end{bmatrix} \qquad (7)$$

where the sub-matrix $\mathbf{S}_\theta^i = \mathbf{J}_\theta^i \mathbf{K}_\theta^{-1} \mathbf{J}_\theta^{i^T}$ describes the spring compliance relative to the end-effector, and the sub-matrix $\mathbf{J}_q^i$ takes into account the passive joint influence on the end-effector motions. Therefore, for a separate kinematic chain, the desired stiffness matrix $\mathbf{K}_i$ defining the motion-to-force mapping

$$\mathbf{f}_i = \mathbf{K}_i \cdot \delta\mathbf{t}_i, \qquad (8)$$

can be computed by direct inversion of relevant 10×10 matrix in the left-hand side of (7) and extracting from it the 6×6 sub-matrix with indices corresponding to $\mathbf{S}_\theta^i$. It is also worth mentioning that computing $\mathbf{S}_\theta^i$ requires 6×6 inversions only, since $\mathbf{K}_\theta^{-1} = diag(K_{act}^{-1}, \mathbf{K}_{Foot}^{-1}, \mathbf{K}_{Leg}^{-1})$.

The described technique can be also generalized for the case

of the "*loaded static equilibrium*", which produces the stiffness matrix that consists of two components: (i) the symmetrical part, which describes the manipulator intrinsic properties in the neighborhood of the equilibrium; and (ii) the skew-symmetrical part that takes into account changes in the manipulator Jacobian due to the equilibrium shift caused by the externally applied force (Chen & Kao, 2000). It can be proved, that in the presence of the external force system (7) should be rewritten as

$$\begin{bmatrix} \mathbf{S}_\theta^i & \mathbf{J}_q^i \\ \mathbf{J}_q^{i\,T} & \mathbf{S}_q^i \end{bmatrix} \cdot \begin{bmatrix} \mathbf{f}_i \\ \delta\mathbf{q}_i \end{bmatrix} = \begin{bmatrix} \delta\mathbf{t}_i \\ \mathbf{0} \end{bmatrix} \qquad (9)$$

where the matrices $\mathbf{S}_q^i$ and $\mathbf{S}_\theta^i$ are expressed via the derivative of the product of the corresponding Jacobian and the external load as follows

$$\mathbf{S}_\theta^i = \mathbf{J}_\theta^i \left(\mathbf{K}_\theta - \frac{\partial(\mathbf{J}_\theta^i \mathbf{f}_i)}{\partial\boldsymbol{\theta}} \right)^{-1} \mathbf{J}_\theta^{i\,T}$$

$$\mathbf{S}_q^i = \frac{\partial(\mathbf{J}_q^i \mathbf{f}_i)}{\partial\mathbf{q}} \qquad (10)$$

After the stiffness matrices $\mathbf{K}_i$ for all kinematic chains are computed, the stiffness of the entire manipulator can be found by simple addition

$$\mathbf{K}_m = \sum_{i=1}^{3} \mathbf{K}_i \qquad (11)$$

This follows from the superposition principle, because the total external force corresponding to the end-effector displacement $\delta\mathbf{t}$ (the same for all kinematic chains) can be expressed as $\mathbf{f} = \sum_{i=1}^{3} \mathbf{f}_i$ where $\mathbf{f}_i = \mathbf{K}_i \cdot \delta\mathbf{t}$. It should be stressed that the resulting matrix $\mathbf{K}_i$ is not invertible, since some motions of the end-effector do not produce the virtual spring reactions (because of passive joints influence). However, for the entire manipulator, the stiffness matrix $\mathbf{K}_m$ is s positive definite and invertible for all non-singular (for the rigid model) postures.

3. MODEL PARAMETERS

3.1 Actuator compliance

The actuator compliance, described by the scalar parameter K_{ctr}^{-1} and 6×6 matrix $\mathbf{K}_{act}^{-1}$,, depends on both the servomechanism mechanics and the control algorithm. Since most modern actuators implement a digital PID control, the main contribution to the compliance is done by the mechanical transmissions. The latter are usually located outside the feedback-control loop and consist of screws,

gears, shafts, belts, etc., whose flexibility is comparable with the flexibility of the manipulator links. Because of the complicated mechanical structure of the servomechanisms, these parameters are usually evaluated from static load experiments, by applying the linear regression to the experimental data.

3.2 Link compliance

Following a general methodology, the compliance of a manipulator link (foots and legs) is described by 6×6 symmetrical positive definite matrices $\mathbf{K}_{leg}^{-1}$, $\mathbf{K}_{foot}^{-1}$ corresponding to 6-d.o.f. springs with relevant coupling between translational and rotational deformations. This distinguishes our approach from other lumped-based techniques, where the coupling is neglected and only a subset of deformations is taken into account (presented by a set of 1-d.o.f. springs).

The simplest way to obtain these matrices is to approximate the link by a beam element for which the non-zero elements of the compliance matrix may be expressed analytically. However, for certain link geometries, the accuracy of a single-beam approximation can be insufficient. In this case the link can be approximated by a serial chain of the beams, whose compliance is evaluated by applying the same method (i.e. considering the kinematic chain with 6-d.o.f. virtual springs, but without passive joints). This leads to the resulting compliance matrix $\mathbf{K}_{Link}^{-1} = \mathbf{J}_b \mathbf{K}_b^{-1} \mathbf{J}_b^T$, where $\mathbf{J}_b$ and $\mathbf{K}_b^{-1}$ incorporate the Jacobian and the compliance matrices for all virtual springs.

3.3 FEA-based evaluation of model parameters

For complex link geometries, the most reliable results can be obtained from the FEA modeling. To apply this approach, the CAD model of each link should be extended by introducing an auxiliary 3D object, a "pseudo-rigid" body, which is used as a reference for the compliance evaluation. Besides, the link origin must be fixed relative to the global coordinate system. Then, sequentially and separately applying forces F_x, F_y, F_z and torques M_x, M_y, M_z to the reference object, it is possible to evaluate corresponding linear and angular displacements, which allow computing the stiffness matrix columns. The main difficulty here is to obtain accurate displacement values by using proper FEA-discretization ("mesh size"). As follows from our study, the single-beam approximation of the Orthoglide links gives accuracy of about 50%, and the four-beam approximation improves it up to 30% only (compared to the FEA-based method that is proved producing very accurate results).

It worth mentioning that here, in contrast to the straightforward FEA-modeling of the entire manipulator, which requires re-computing for each manipulator posture, the proposed technique involves a single evaluation of link stiffness. The latter essentially improves the computational speed while preserving accuracy of the FEA method.
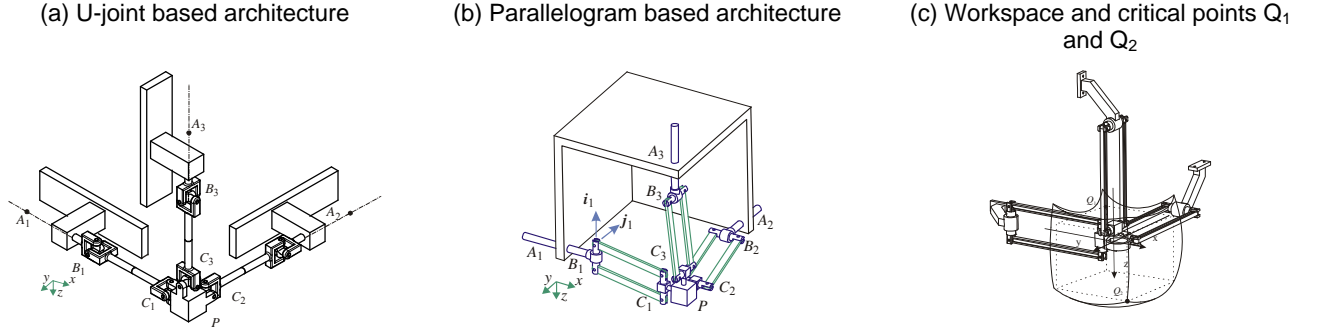
(a) U-joint based architecture   (b) Parallelogram based architecture   (c) Workspace and critical points Q_1 and Q_2

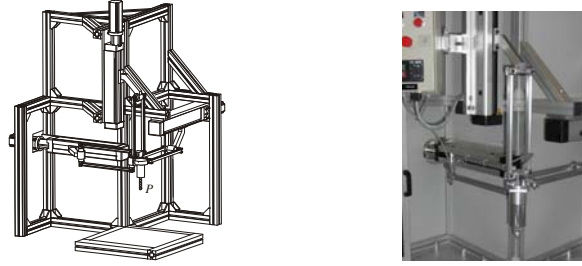Fig. 1. Kinematics of two 3-dof translational mechanisms employing the Orthoglide architecture

Fig. 2 CAD model of and Orthoglide and its prototype

4. APPLICATION EXAMPLES

To demonstrate efficiency of the proposed methodology, let us apply it to the comparative stiffness analysis of two 3-d.o.f. translational mechanism, which employ Orthoglide architecture. CAD models of these mechanisms are presented in Figs. 1 and 2.

First, let us derive the stiffness model for the simplified Orthoglide mechanics (3-*PUU*), where the legs are comprised of equivalent limbs with U-joints at the ends. Accordingly, to retain major compliance properties, the limb geometry corresponds to the parallelogram bars with doubled cross-section area. The geometrical models of separate kinematic chains can be described by the expression (1), where the product components are defined via the standard translational/rotational operators. Because for the rigid manipulator the end-effector moves with only translational motions, the nominal values of the passive joint coordinates are subject to the specific constrains $q_3 = -q_2$; $q_4 = -q_1$, which are implicitly incorporated in the direct/inverse kinematics. The modelling results are presented in Table 1. Below, they are compared with the compliance of the parallelogram-based manipulator.

For the second architecture (3-*PRPrP*) it is necessary to derive first the stiffness matrix of the parallelogram. Using the adopted notations, the parallelogram equivalent model may be written as

$$\mathbf{T}_{Plg} = \mathbf{R}_y(q_2) \cdot \mathbf{T}_x(L) \cdot \mathbf{R}_y(-q_2) \cdot \mathbf{V}_s(\theta_7, \ldots \theta_{12}) \tag{12}$$

where, compared to the above case, the third passive joint is eliminated (it is implicitly assumed that $q_3 = -q_2$). On the other hand, the original parallelogram may be split into two serial kinematic chains (the "upper" and "lower" ones). Hence, the parallelogram compliance matrix may be also derived using the proposed technique that yields an analytical expression.

Using this model and applying the proposed technique, we computed the compliance matrices for three typical manipulator postures (see table Table 1). As follows from the comparison with the U-joint case, the parallelograms allow increasing the rotational stiffness roughly in 10 times. This justifies application of this architecture in the Orthoglide prototype design (Chablat & Wenger, 2003).

5. CONCLUSIONS

The paper proposes a new systematic method for computing the stiffness matrix of multi-chain parallel robotic manipulators for both unloaded and loaded equilibriums. It is

TABLE 1
TRANSLATIONAL AND ROTATIONAL STIFFNESS OF THE 3-PUU AND 3-PRPAR MANIPULATORS

MANIPULATOR ARCHITECTURE	Point Q_0 $x, y, z = 0.00$ *mm*		Point Q_1 $x, y, z = -73.65$ *mm*		Point Q_2 $x, y, z = +126.35$ *mm*	
	k_{tran} [N/mm]	k_{rot} [N·mm/rad]	k_{tran} [N/mm]	k_{rot} [N·mm/rad]	k_{tran} [N/mm]	k_{rot} [N·mm/rad]
3-PUU manipulator	$2.78 \cdot 10^{-4}$	$20.9 \cdot 10^{-7}$	$10.9 \cdot 10^{-4}$	$24.1 \cdot 10^{-7}$	$71.3 \cdot 10^{-4}$	$25.8 \cdot 10^{-7}$
3-PRPaR manipulator	$2.78 \cdot 10^{-4}$	$1.94 \cdot 10^{-7}$	$9.86 \cdot 10^{-4}$	$2.06 \cdot 10^{-7}$	$21.2 \cdot 10^{-4}$	$2.65 \cdot 10^{-7}$

based on multidimensional lumped model of the flexible links, whose parameters are evaluated via the FEA modeling and describe both the translational/rotational compliances and the coupling between them. In contrast to previous works, the method employs a new solution strategy of the kinetostatic equations and allows computing the stiffness matrices for any given manipulator posture and. Another advantage is computational simplicity that requires low-dimensional matrix inversion compared to other techniques.

The efficiency of the proposed method was demonstrated through application examples, which deal with comparative stiffness analysis of two parallel manipulators of the Orthoglide family. Relevant simulation results have confirmed essential advantages of the parallelogram based architecture and validated adopted design of the Orthoglide prototype. In future work, the method will be extended to other parallel architectures composed of several identical kinematic chains.

REFERENCES

Alici, G. and Shirinzadeh, B. (2005). Enhanced Stiffness Modeling, Identification and Characterization for Robot Manipulators. *IEEE Transactions on Robotics*, **21**(4), 554 – 564.

Brogardh, T. (2007). Present and future robot control development - An industrial perspective. *Annual Reviews in Control*, **31**(1), 69-79.

Chablat D., Wenger, P. (2003). Architecture Optimization of a 3-DOF Parallel Mechanism for Machining Applications, the Orthoglide. *IEEE Transactions on Robotics and Automation*, **19**(3), 403-410.

Chanal, H., Duc, E. and Ray, P. (2006). A study of the impact of machine tool structure on machining processes. *International Journal of Machine Tools and Manufacture*, **46**(2), 98-106.

Chen, S.F. and Kao, I. (2000). Conservative congruence transformation for joint and Cartesian stiffness matrices of robotic hands and fingers. *International Journal of Robotics Research*, **19**(9), 835-847.

Clavel, R. (1988). DELTA, a fast robot with parallel geometry. *Proceedings, of the 18th International Symposium of Robotic Manipulators*, IFR Publication, 91–100.

Duffy, J. (1996). *Statics and Kinematics with Applications to Robotics*. Cambridge University Press, New-York.

Gosselin, C.M. (1990). Stiffness mapping for parallel manipulators. *IEEE Transactions on Robotics and Automation*, **6**(3), 377–382.

Griffis, M. and Duffy, J. (1993). Global stiffness modeling of a class of simple compliant couplings, *Mechanism and Machine Theory*, **28**(2), 207–224.

Li, Y. and Xu, Q. (2008). Stiffness analysis for a 3-PUU parallel kinematic machine. *Mechanism and Machine Theory*, 43(2), 186-200.

Merlet, J.-P. (2000). *Parallel Robots*, Kluwer Academic Publishers, Dordrecht.