https://doi.org/10.20965/jrm.2011.p0557

Paper:

Unified Robot Control Scheme for Cooperative Motion, Autonomous Motion and Contact Reaction

Vincent Duchaine* **and Clément Gosselin****

*Department of Automated Manufacturing Engineering, École de Technologie Supérieure
1100 Rue, Notre-Dame Ouest, Montreal, H3C 1K3, Canada
E-mail: vincent.duchaine@etsmtl.ca
**Department of Mechanical Engineering, Université Laval
1065 Avenue de la médecine, Québec, Qc, G1V 0A6, Canada
E-mail: gosselin@gmc.ulaval.ca

[Received October 25, 2010; accepted January 22, 2011]

While the majority of industrial manipulators currently in use only need to perform autonomous motion, future generations of cooperative robots will also have to execute cooperative motion and intelligently react to contacts. These extended behaviours are essential to enable safe and effective physical Human-Robot Interaction (pHRI). However, they will inevitably result in an increase of the controller complexity. This paper presents a single variable admittance control scheme that handles the three modes of operation, thereby minimizing the complexity of the controller. First, the adaptative admittance controller previously proposed by the authors for cooperative motion is recalled. Then, a novel implementation of variable admittance control for the generation of smooth autonomous motion including reaction to collisions anywhere on the robot is presented. Finally, it is shown how the control equations for these three modes of operation can be simply unified into a unique control scheme.

Keywords: physical human-robot interaction, admittance control, impedance control, collision avoidance

1. Introduction

Over the last decade, the dynamic performances of commercially available robots have increased while their cost has significantly reduced. However, their general capabilities have remained virtually unchanged. Indeed, they are still mainly confined to performing free-space motion such as pick and place or trajectory tracking. A next logical step in the path towards more advanced robots is to bring them into our own environment so that they can directly assist humans in the achievement of more complex tasks. Combining the robot's powerful and precise motion capabilities with the human intelligence and adaptability could lead to an efficient synergy that would benefit many areas such as surgery [1], health care for the elderly [2] and manufacturing [3].

However, there are still many challenges on the road to achieving an efficient physical Human-Robot Interaction (pHRI) [4]. These challenges are mainly related to safety [5] and to developing robot capabilities. The enhancement of the robot capabilities usually leads to an increase in the complexity of the robot controllers. Indeed, while conventional manipulators were mainly controlled to produce *autonomous motion*, a cooperative robot needs to be controlled in at least three operation modes. First, these robots need to perform *cooperative motion* with human beings [6]. Also, they need to be capable of performing *autonomous motion* in subtasks that do not involve a human [7]. Finally, with appropriate sensing, they have to provide *contact reaction* in order to safely and intuitively move away from unwanted contacts [8].

In this paper, we propose an approach to tackle these three operation modes with a single admittance control scheme. The main objective is to reduce the complexity of the controller and to avoid the transition of control schemes between the different operation modes. First, the adaptative admittance controller proposed by the authors in [9] for safe and intuitive *cooperative motion* is recalled. Then, admittance controllers are introduced for *autonomous motion* and *contact reaction*. Finally, the three control algorithms are combined into a single set of equations. **Fig. 1** provides a general idea of the final controller structure. In the last section, experimental results obtained with a parallel manipulator is presented in order to show the impact of the proposed controller.

2. Cooperative Motion

The most important mode of operation for a cooperative robot is clearly *cooperative motion*. In this mode, the robot is physically guided by a human in order to perform a joint task. A stable and intuitive admittance controller for cooperative motion was proposed by the authors in [9]. This section briefly recalls the latter control scheme, which will be integrated here alongside with a novel admittance controller for autonomous motion and contact reaction into a single control scheme. A more detailed presentation of the algorithm can be found in [9].

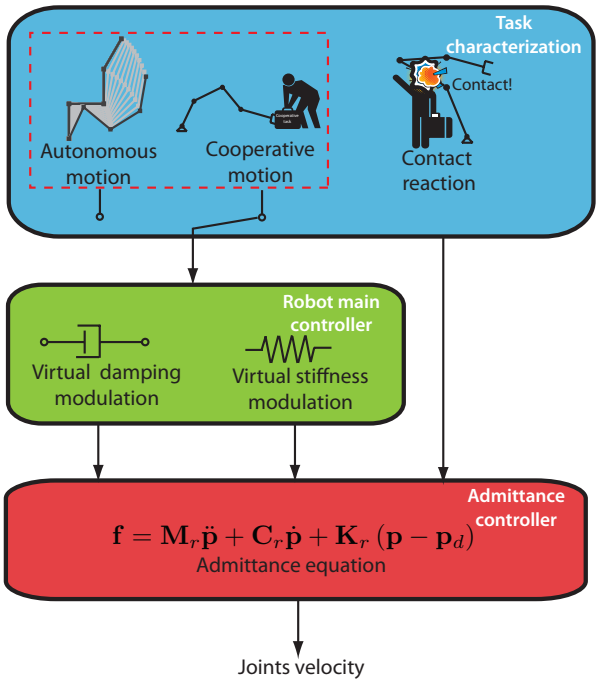

Fig. 1. Structure of the proposed robot controller.

Impedance/admittance control is a way of linking forces and motion via the introduction of a virtual model between the two. The general form used for this model is given by:

$$\mathbf{f} = \mathbf{M}_r\ddot{\mathbf{p}} + \mathbf{C}_r\dot{\mathbf{p}} - \mathbf{K}_r(\mathbf{p} - \mathbf{p}_d), \quad \ldots\ldots (1)$$

where $\mathbf{M}_r, \mathbf{C}_r$ and $\mathbf{K}_r$ are respectively the diagonal matrices of virtual mass, damping and stiffness, $\mathbf{f}$ is the force vector and $\mathbf{p}$, $\dot{\mathbf{p}}$, $\ddot{\mathbf{p}}$ are the position vector and its derivatives. If this equation is satisfied by feeding the force (impedance) or the displacement (admittance), the robot will exhibit a given behaviour prescribed by the parameters ($\mathbf{M}_r, \mathbf{C}_r$ and $\mathbf{K}_r$) of the model. Typically, in cooperative motion the virtual stiffness ($\mathbf{K}_r$) is set to zero so that the robot can freely follow humans without having a spatial equilibrium point.

There are two main drawbacks associated with the use of admittance control in the context of pHRI. First, using a fixed virtual damping can lead to an inefficient cooperation. Indeed, if the damping is set to a low value the robot will tend to easily accelerate but will be harder to stop. The case of a high damping we lead to the opposite effect. Therefore, it was proposed in [9] to adjust the virtual damping online according to the human intention of accelerating or decelerating. These intentions are inferred using the time derivative of the forces measured. The corresponding matrix of variable damping coefficients is referred to as $\mathbf{C}_{vi}$.

The second drawback of admittance control is related to safety. Admittance control is known to become unstable when facing a stiff environment [10]. This problems is harder to solve in the context of pHRI since humans, who act as the environment, have a variable stiffness. To solve this problem a variable damping, noted $\mathbf{C}_c$, that keeps the robot stable, was introduced in [9]. The variable damping is calculated using the equation of the stability frontier of the admittance control scheme.

The control scheme proposed in [9] is an admittance controller that uses the above two dampings. In this scheme, the variable damping $\mathbf{C}_{vi}$ is used as long as it is larger than the critically stable damping $\mathbf{C}_c$. The decision on which damping matrix should be used is made based on a test on the positive definiteness of a matrix written as:

$$\mathbf{D} = \mathbf{G}^{\mathrm{T}}\mathbf{C}_{vi}\mathbf{G} - \mathbf{C}_c. \quad \ldots\ldots (2)$$

If this matrix is positive definite or positive semi-definite, the pre-assigned virtual dynamics allows the closed-loop system to remain dissipative and thus, the original parameters can be kept. However, if $\mathbf{D}$ becomes negative definite, the critical damping coefficients are used in the impedance damping matrix – instead of the pre-assigned coefficients – in order to prevent energy build-up.

However, verifying the positive definiteness of matrix $\mathbf{D}$ using the well-known Sylvester criterion can be time consuming since it requires computing the determinant of all the sub-matrices. A more efficient way to use this criterion is to compute a $\mathbf{LU}$ factorization of matrix $\mathbf{D}$ and to use the sign of the pivots to conclude on positive definiteness. Using this approach, two scalar indices are defined. When these scalar indices are equal, it can be concluded that matrix $\mathbf{D}$ is positive definite or at least semi-positive definite. Using Einstein's convention for summation, the two scalar indices can be written as:

$$e = L_{ij}\delta^{ij} + U_{ij}\delta^{ij} \quad \ldots\ldots (3)$$

and

$$e^* = \mathrm{abs}(L_{ij})\delta^{ij} + \mathrm{abs}(U_{ij})\delta^{ij}. \quad \ldots\ldots (4)$$

Applying the Kronecker delta on these indices gives us a simple and clear numerical answer on the positive definiteness of matrix $\mathbf{D}$. This answer takes the form:

$$\delta_{e,e^*} = \begin{cases} 1, & \text{if } e = e^* \\ 0, & \text{if } e \neq e^*. \end{cases} \quad \ldots\ldots (5)$$

Finally using these indices a complete, safe, stable and intuitive control solution for *cooperative motion* was proposed. The resulting control equation is:

$$\mathbf{f} = \mathbf{M}\ddot{\mathbf{p}} + [\delta_{e,e^*}\mathbf{C}_{vi} + (1 - \delta_{e,e^*})\mathbf{C}_s]\dot{\mathbf{p}} \quad \ldots\ldots (6)$$

where $\mathbf{C}_s$ is the stable damping matrix to be applied, which slightly differs from the critically stable damping matrix $\mathbf{C}_c$.

3. Autonomous Motion

Even if a robot has the capability of physically interacting with a human, it may be used to perform some tasks or subtasks autonomously. When performing autonomous motion, cooperative robots should behave similarly to conventional manipulators, i.e., to be position

controlled. One solution to this problem is to use conventional position control with trajectory planning for pick and place tasks. However, using this approach requires a sudden change of control scheme, going from admittance control (that is often implemented with a low-level velocity controller) to a path planning algorithm that uses a low level position control. In order to enable autonomous motion while keeping a unique control algorithm, this section proposes a new way of using the impedance virtual stiffness to effectively generate autonomous pick and place motion.

As shown in the preceding section, the virtual stiffness coefficient of the general impedance/admittance equation – Eq. (1) – is usually set to zero during cooperative motion. In doing so, the equation admits infinitely many spatial equilibrium points, which allows the robot to become a passive follower. This behaviour is exactly what is expected for robots in the cooperation mode. Indeed, during this kind of task, the human leads the motion. However, in autonomous mode, the robot has to be more active and generate the motion. Setting the stiffness term – $\mathbf{K}_r$ in Eq. (1) – to a nonzero value will add an equilibrium point at $\mathbf{p}_d$, which will generate an attractive field that will drive the robot motion in the appropriate direction.

Similar approaches have been widely used in robotics for decades. Indeed, when considered from a different perspective, this approach can be seen as a Cartesian PID controller. However, used with a single value $\mathbf{K}_r$ and with the final destination of the trajectory directly as the equilibrium point $\mathbf{p}_d$, this method will generate a rather awkward motion that will start with a very high initial velocity that will decrease along the trajectory. Instead, such a control approach is typically used with a trajectory planning algorithm that generates a successive series of equilibrium points $\mathbf{p}_d$ toward the final destination. In order to minimize the difference between each control mode and avoid trajectory planning *per se*, the proposed control equation allows to directly set $\mathbf{p}_d$ as the final destination in Eq. (1) but in conjunction with a variable virtual stiffness ($\mathbf{K}_r$). This has the effect of generating an effective and safe motion for any targeted posture of the robot.

Arimoto et al. presented a similar approach in [11], but where virtual stiffness was fixed. One thing they showed in their simulation is the importance of the joint damping ratio on the quality of the motion produced. Indeed, the ideal value of the stiffness $\mathbf{K}_r$ that will generate the desired motion is dependent on the virtual damping parameters. If the damping is too low, an oscillatory behaviour of the robot around the final destination point $\mathbf{p}_d$ will be observed. On the other hand, if the damping is too high, an overdamped motion will result. Humans have the natural ability to adjust the stiffness of their arms using different activation levels in antagonistic and agonistic muscles. Some studies [12, 13] have shown that viscosity properties of the arm also simultaneously increase or decrease with the stiffness to stabilize the motion. The investigation of the relationship between these two characteristics reported in [13, 14] led to the conclusion that the damping ratio is constant regardless of bi-articulate muscle activation level. This fact gives a direct relation between the damping effect and the square root of the stiffness. In the proposed approach we will use the same relation to adjust the virtual damping in order to stabilize the motion.

The equation that links the damping ratio to a given stiffness and damping is:

$$2\zeta\mathbf{I} = \mathbf{C}_r\left(\mathbf{K}_r\mathbf{M}_r\right)^{-\frac{1}{2}}, \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (7)$$

where $\mathbf{I}$ is the identity matrix of the appropriate dimension and ζ is the damping ratio. Based on the assumption that this ratio is kept constant for any value of the stiffness, the damping matrix is given by:

$$\mathbf{C}_r = 2\zeta\left(\mathbf{K}_r\mathbf{M}_r\right)^{\frac{1}{2}}. \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (8)$$

Using this value of the damping, Eq. (1) can be rewritten as:

$$\mathbf{f} = \mathbf{M}_r\ddot{\mathbf{p}} + 2\zeta\left(\mathbf{K}_r\mathbf{M}_r\right)^{\frac{1}{2}}\dot{\mathbf{p}} - \mathbf{K}_r\left(\mathbf{p}-\mathbf{p}_d\right) = 0. \quad (9)$$

The value of the stiffness $\mathbf{K}_r$ that will ensure that the velocity of the robot remains under some safety limit v_{max} is found when all the potential energy is dissipated by the viscous force, i.e., when:

$$\mathbf{M}_r\ddot{\mathbf{p}} = \mathbf{0} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (10)$$

and

$$\mathbf{f} = \mathbf{0}. \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (11)$$

In this case, Eq. (9) is reduced at:

$$2\zeta\left(\mathbf{K}_r\mathbf{M}_r\right)^{\frac{1}{2}}\dot{\mathbf{p}}_{max} - \mathbf{K}_r\left(\mathbf{p}-\mathbf{p}_d\right) = 0. \quad . \quad . \quad . \quad (12)$$

This equation can be rewrite as:

$$2\zeta\left(\mathbf{K}_r\mathbf{M}_r\right)^{\frac{1}{2}}\dot{\mathbf{p}}_{max} = \mathbf{K}_r\left(\Delta\mathbf{p}\right), \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (13)$$

with

$$\Delta\mathbf{p} = \mathbf{p} - \mathbf{p}_d \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (14)$$

and

$$\dot{\mathbf{p}}_{max} = \frac{v_{max}}{\|\Delta\mathbf{p}\|}\Delta\mathbf{p}. \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (15)$$

Eq. (15) is key to appropriately scale the motion of each axis. Setting each of the components of vector $\dot{\mathbf{p}}_{max}$ to the velocity limit v_{max}, will result in a non-synchronized motion on each axis. In other words, the total trajectory time on each axis will not be the same and the norm of the robot velocity will be above the value of v_{max}. **Fig. 2** shows that using Eq. (15), the proposed algorithms have the capability to synchronize the motion on all axes without requiring any specific reference to trajectory times.

Using the square of Eq. (13), the stiffness matrix that generated motion where velocity is always under v_{max} can be found as:

$$\mathbf{K}_r = 4\zeta\dot{\mathbf{P}}_{max}^T\dot{\mathbf{P}}_{max}\mathbf{M}_r\left(\Delta\mathbf{P}^T\Delta\mathbf{P}\right)^{-1}, \quad . \quad . \quad . \quad . \quad . \quad . \quad (16)$$

where

$$\Delta\mathbf{P} = \mathrm{diag}(\Delta\mathbf{p}) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (17)$$

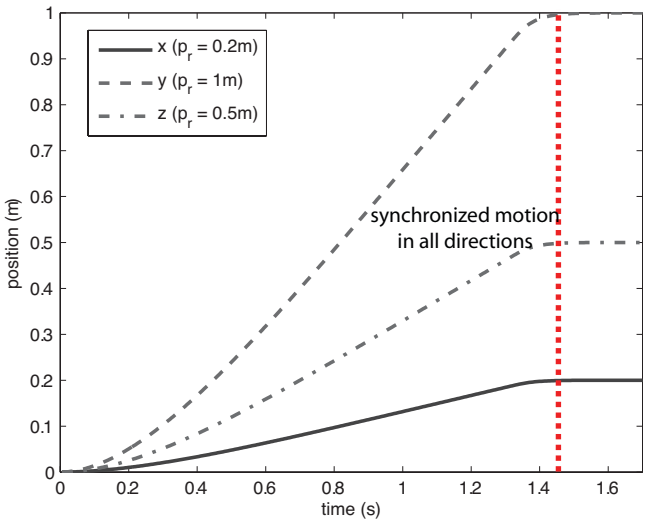

Fig. 2. Automatic multi axis synchronisation.

and

$$\dot{\mathbf{P}}_{max} = \mathrm{diag}(\dot{\mathbf{p}}_{max}). \quad \ldots\ldots\ldots\ldots (18)$$

Since $\mathbf{p}$ is the current position of the robot, the virtual stiffness will be constantly changing. It will reach infinity at the desired position $\mathbf{p}_r$, i.e., when $\Delta\mathbf{p} = 0$. Therefore a maximum value of stiffness should be defined. In fact, as demonstrated in [15], the variation of the stiffness is similar to how humans control their own stiffness. When a human is holding his hands far from the desired destination point, stiffness is typically low. However, the latter will increase when the hands are getting closer to the final destination.

Figure 3 shows an example of motion resulting from the use of the proposed approach based, as well as two other examples given by using a trapezoidal trajectory and 5$^{\text{th}}$ order polynomial. These curves are not intended to show any advantage of one methods over the others, but rather to show that motions are continuous and similar to those produced by other approaches. Indeed, the advantage of the proposed method do not rely on any performance improvement, but rather on the fact that it uses the same control equation to address cooperative and autonomous motion. This has the impact of greatly simplifying the complexity of the final controller. The proposed method has also an interesting feature that makes the robot's behavior potentially safer for humans. Indeed, this algorithm generates its own motion in function of the actual position and velocity of the robot, without ever referring to time. Conventional trajectory generation typically refers to a trajectory time. In most of these methods, if an unwanted contact would temporarily immobilize the robot, the reference point would continue its motion towards the final destination. This would thus generating a growing force that will result in a brutal motion when the contact would be released. However, a robot whose motions would be driven by the proposed control method would just resume smoothly the movement using its actual position and velocity as initial condition.

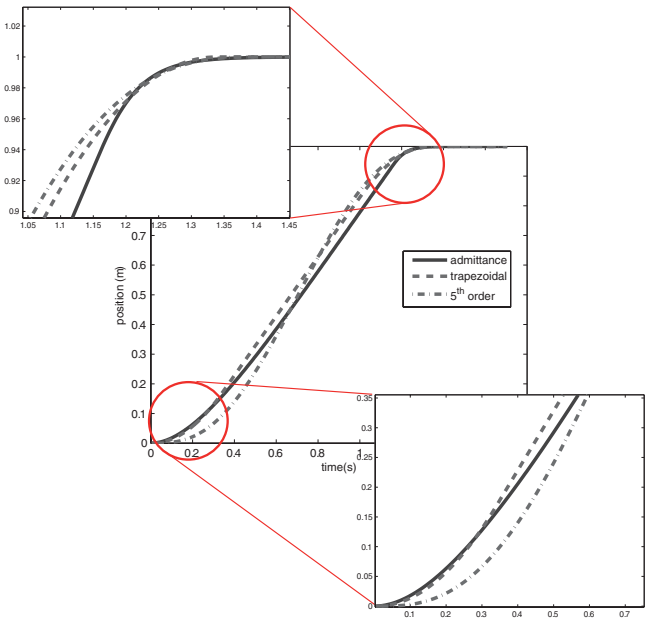

Fig. 3. Comparison of the motion obtained with the novel approach with conventional position control based on trapezoidal and 5$^{\text{th}}$ order polynomial trajectories.

4. Contact Reaction

As mentioned in the introduction, an appropriate level of safety is a necessary condition for the coexistence of humans and robots in the same workspace. Cooperation robots should in fact be designed so that they cannot seriously hurt humans. However, from a control perspective, it is essential that cooperation robots have the capability to sense their surrounding environment and to *react* in accordance.

This section presents a way to react to contacts using Cartesian admittance control at any point on the robot. The capability of reacting to contacts should always have the priority over the other two operation modes, namely *cooperation motion* and *autonomous motion*. However, for effectiveness, the controller should, as much as possible, try to continue to perform the task while reacting to contacts. This will lead to a more efficient behaviour that resembles that of humans, who can naturally adapt their kinematics and dynamics to the surrounding environment while performing a given task.

4.1. Contact Admittance for Passive Reaction

Even if admittance control is already used for controlling the end-effector behaviour of a robot, it should not be directly extended to the entire body of the robot for dealing with contact. Instead, it would be preferable to define a new distinct admittance that would lead to a safe behaviour during collisions. Indeed, contact admittance should be specified according to safety considerations – such as the maximum resulting virtual inertia during a collision – while end-effector admittance/impedance is generally tuned to be more intuitive and effective for collab-

orative manipulation.

Based on this premise, the expression for contact admittance is given by:

$$\mathbf{f}_c = \mathbf{M}_c\ddot{\mathbf{p}}_c + \mathbf{C}_c\dot{\mathbf{p}}_c, \quad \ldots \ldots \ldots \ldots \quad (19)$$

where $\mathbf{M}_c$ and $\mathbf{C}_c$ are respectively the virtual mass and damping that the robot should exhibit at a contact point and where $\mathbf{p}_c$ and its time derivatives are the Cartesian position, velocity, and acceleration at this point.

In admittance control, the forces exerted by the user are measured and it is the motion that is fed back to the user. Therefore Eq. (19) can be solved in order to provide the reference position $\mathbf{p}_c$ or its first or second derivative $(\dot{\mathbf{p}}_c, \ddot{\mathbf{p}}_c)$. In this work the robot velocity will be commanded in order to exploit the linear relation between Cartesian and joint velocity:

$$\dot{\mathbf{p}} = \mathbf{J}\dot{\mathbf{q}}. \quad \ldots \ldots \ldots \ldots \quad (20)$$

Using a zero-order hold with a sampling rate of T_s, the discrete state resolution of Eq. (1) becomes

$$\dot{\mathbf{p}}_c(k+1) = \mathbf{A}\dot{\mathbf{p}}_c(k) + T_s\mathbf{M}_c^{-1}\left(\mathbf{f}_c(k)\right), \quad \ldots \quad (21)$$

where

$$\mathbf{A} = \left(\mathbf{1} - T_s\mathbf{M}_c^{-1}\mathbf{C}_c\right). \quad \ldots \ldots \quad (22)$$

This equation of the Cartesian velocity that the robot should exhibit at the point of contact to follow the desired admittance, has the advantage of not involving the measurement of $\ddot{\mathbf{p}}_c$, which can be very noisy and hard to obtain.

In current robotics, the control being typically made at the robot joint, the updated desired Cartesian velocity vector at the contact point given by Eq. (21) could be projected directly into this space. However, depending on if the robot is redundant or not at the contact point regarding the contact constraints, different methods can be used to find the joint space velocity associated with this Cartesian speed. This problem will be included directly in the control approaches presented above.

4.1.1. Admittance Parameters

Satisfying Eq. (21) at each point of the manipulator would theoretically allow a passive and natural reaction to any contact. Considering the complexity and the wide range of reactions that humans exhibit for some given contact, having a passive reaction based on an admittance model is probably not sufficient for covering all ideal reactions to the complete spectrum of possible collision scenarios. However, using only the sense of touch, it is probably a good intuitive reaction to which other active and nonlinear behaviours could be added.

Nevertheless, the resulting behaviour in terms of safety and intuitiveness of the reaction will greatly depend on the choice of the admittance parameters. It is not straightforward to define what should be the value of these parameters to maintain the contact forces under a given limitation. On one hand, the parameters should reflect the dynamic capabilities of the robot to render adequately the virtual admittance during the impact. On the other hand, the parameters will depend also on the mechanical properties (stiffness, coefficient of restitution) of the object or human part in contact. However, using the physical meaning of Eq. (21), it is possible to estimate the mass (m_c) and damping (c_c) parameters that should be used. For a mass moving at a certain velocity, the lower the mass is, the more it will be damped, thereby reducing the kinetic energy transferred at the impact.

4.1.2. Force Vector

Depending on the technique used to sense contact (e.g., robot skin, torque sensor at the joint, current at the actuator), in some cases it will be possible to sense multiple contact points and in other cases, only the resultant force. The processing of the output signals of such sensors is outside the scope of this paper, but it can easily be assumed that in both cases it will be possible to find an equivalent force vector at an equivalent location.

Therefore, in this work, for a given contact on the i^{th} member of a robot, an equivalent force vector at the joint connecting the i^{th} and $(i+1)^{\text{th}}$ members will be calculated. The contact admittance will thereafter be calculated for this point using this equivalent force vector and Eq. (1), from which the proper reference velocity that the robot should satisfy is found.

4.2. Satisfying Contact Admittance and End-Effector Task

A straightforward approach would be to simply control the robot at the contact point to satisfy the required velocity vector given by the admittance model. However, depending on where the contact occurs on the robot, it is very likely that there will be several degrees of freedom (*DOFs*) above the contact point. In order to improve effectiveness during human-robot cooperative tasks, it is suggested here to use these extra *DOFs* to continue, as much as possible, to perform the end-effector task. The potential benefit of the application of this concept will be to no longer consider contacts between a human and a robot as an unwanted situation that requires an emergency action, thereby enabling a more proactive cooperation.

4.2.1. Projection Matrix

Before going any further in the derivation of the control equations, some projection matrices need to be defined. Regarding contact reaction, we aim at satisfying a given admittance at a random contact point while trying to continue a given end-effector task given from one of the other two modes of operation. Except for the case in which the contact occurs exactly at the end-effector point, working with the constraints implies working in a subspace of the operational space. The first projection matrix is used to project a vector or matrix from a space of dimension equal to the joint space of the robot to a space of the dimension of the joint space of the contact manipulator, i.e., the equivalent manipulator that goes from the

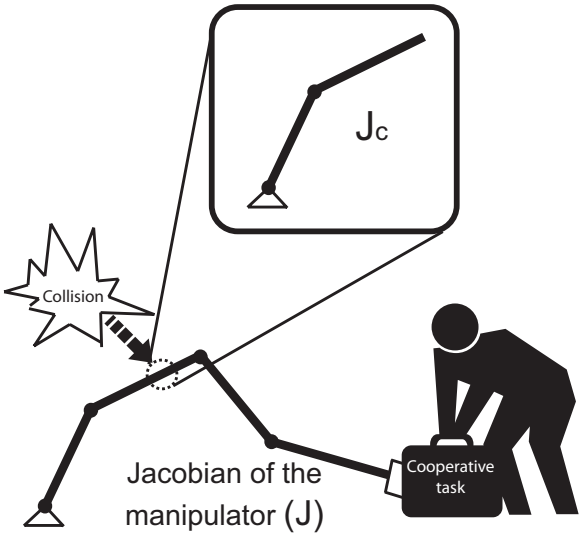

Fig. 4. Definition of the contact Jacobian.

base of the robot to the contact point. This matrix, noted **B**, is defined as:

$$B_{ij} = \delta_{ij}, \quad \ldots \quad (23)$$

where the maximum value of i and j are respectively the dimension of the joint space of the robot and of the number of joints located before the contact point.

Moreover, matrix **L** is defined as a matrix containing the 'missing' columns of **B**, i.e., it forms a basis of the left nullspace of **B**.

Using these projection matrices, a new Jacobian matrix can be defined to be useful in what follows. This matrix is the Jacobian of the manipulator from the based to the contact point (contact Jacobian) and is given by:

$$\mathbf{J}_c = \mathbf{J}^* \mathbf{B} \quad \ldots \quad (24)$$

where $\mathbf{J}^*$ is the Jacobian matrix obtained by setting to zero the length of all the robot links located downstream from the contact point in the Jacobian **J** of the manipulator and where the sub-index c is used henceforth to denote the membership of a parameter to the joint space of the contact manipulator. **Fig. 4** illustrates the meaning of the contact Jacobian matrix.

Depending of the sensing capabilities of the robot, it is possible that the dimensions of the Cartesian contact space and the end-effector task space will differ. Indeed, using for instance the output signals of many currently proposed robot skin technologies [16–19], it would be very difficult to infer a force/torque vector of more than three components. If such sensing devices are mounted on a robot with more than three degrees of freedom at the end-effector, it is obvious that the dimension of the Cartesian contact space and the operational space will not be the same.

Therefore, a projection matrix that projects the operational space of the robot onto the Cartesian contact space should be defined. The components of this matrix, noted **V**, are defined as:

$$V_{ij} = \delta_{ij}, \quad \ldots \quad (25)$$

where the maximum value of i and j are respectively the dimension of the operational space ($\dim(\dot{\mathbf{p}}_o)$) of the robot and the dimension of the contact space ($\dim(\dot{\mathbf{p}}_c)$) and δ_{ij} is the Kronecker delta. Therefore, **V** is the matrix of the spanning set of vectors in $\mathbb{R}^{\dim(\dot{\mathbf{p}}_o)}$ of the contact space and it will be used as a projection between both spaces.

Matrix **W** is the matrix of the 'missing' columns of **V** and therefore it forms a basis of the left nullspace of **V**.

For example, for a 4-DOF planar manipulator, if contact occurs at the third member and the contact space is of dimension 1, the four matrices defined above will have the following form:

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \ldots \quad (26)$$

$$\mathbf{L} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad \ldots \quad (27)$$

$$\mathbf{V} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \ldots \quad (28)$$

and

$$\mathbf{W} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \ldots \quad (29)$$

where $\text{range}(\mathbf{L}) = \ker(\mathbf{B}^T)$ and $\text{range}(\mathbf{W}) = \ker(\mathbf{V}^T)$.

4.2.2. Satisfying Contact Constraints While Minimizing Deviation from End-Effector Task

Assuming that:

$$\dim(\dot{\mathbf{p}}_c) \leq \dim(\dot{\mathbf{p}}_o) \quad \ldots \quad (30)$$

and that:

$$\text{rank}(\mathbf{J}_c) \geq \dim(\dot{\mathbf{p}}_c), \quad \ldots \quad (31)$$

the joint velocity vector of the contact manipulator that satisfies the contact admittance can be separated into two distinct parts, i.e., the part of the known components given by a solution to the contact admittance velocity vector $\dot{\mathbf{p}}_c$ and the part resulting from the other $(\dim(\dot{\mathbf{p}}_0) - \dim(\dot{\mathbf{p}}_c))$ unknown constraints **a**:

$$\dot{\mathbf{q}}_c = \mathbf{J}_c^I (\mathbf{V}\dot{\mathbf{p}}_c + \mathbf{W}\mathbf{a}). \quad \ldots \quad (32)$$

Using this definition of $\dot{\mathbf{q}}_c$, the complete joint velocity vector of the manipulator can be expressed as:

$$\dot{\mathbf{q}} = \mathbf{B}\dot{\mathbf{q}}_c + \mathbf{L}\dot{\mathbf{q}}_r = \mathbf{B}\mathbf{J}_c^I (\mathbf{V}\dot{\mathbf{p}}_c + \mathbf{W}\mathbf{a}) + \mathbf{L}\dot{\mathbf{q}}_r, \quad \ldots \quad (33)$$

where $\dot{\mathbf{q}}_r$ is the vector in $\mathbb{R}^{\dim(\mathscr{R}(\mathbf{J}_r))}$ of the residual joint velocity components. Projecting the joint velocity vector given by Eq. (33) onto the operational space using the Jacobian of the manipulator **J**, it is possible to define an optimization index that will be the norm of the difference

between the desired operational velocity and the best end-effector velocity vector that satisfies the contact requirements. The resulting index is written as:

$$\begin{aligned}\min \chi^2_{(q_r,a)} &= \left[\mathbf{J}\dot{\mathbf{q}}_{(q_r,a)} - \dot{\mathbf{p}}_o\right]^T \left[\mathbf{J}\dot{\mathbf{q}}_{(q_r,a)} - \dot{\mathbf{p}}_o\right] \\ &= \left[\mathbf{c}_{(q_r,a)}\right]^T \left[\mathbf{c}_{(q_r,a)}\right], \quad \ldots\ldots \quad (34)\end{aligned}$$

with

$$\mathbf{c} = \mathbf{J}\mathbf{B}\mathbf{J}_c^I(\mathbf{V}\dot{\mathbf{p}}_c + \mathbf{W}\mathbf{a}) + \mathbf{J}\mathbf{L}\dot{\mathbf{q}}_r - \dot{\mathbf{p}}_o. \quad \ldots\ldots \quad (35)$$

This function, to be minimized, contains two unknown vectors. To simplify the minimization procedure, an augmented vector of unknowns is defined as:

$$\mathbf{u} = \begin{bmatrix}\mathbf{a} \\ \mathbf{q}_r\end{bmatrix}. \quad \ldots\ldots\ldots\ldots\ldots \quad (36)$$

Therefore, Eq. (35) can be rewritten as:

$$\mathbf{c} = \left(\mathbf{J}\mathbf{B}\mathbf{J}_c^I\mathbf{V}\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_o\right) + \mathbf{M}\mathbf{u}, \quad \ldots\ldots\ldots \quad (37)$$

with

$$\mathbf{M} = \begin{bmatrix}\mathbf{J}\mathbf{B}\mathbf{J}_c^I\mathbf{W} & \mathbf{J}\mathbf{L}\end{bmatrix}. \quad \ldots\ldots\ldots \quad (38)$$

This equation being linear in $\mathbf{u}$, its minimum can be found as:

$$\text{grad}\ \chi^2_{\mathbf{u}} = 2\mathbf{M}^T\left[\left(\mathbf{J}\mathbf{B}\mathbf{J}_c^I\mathbf{V}\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_o\right) + \mathbf{M}\mathbf{u}\right] = 0. \quad (39)$$

With some manipulations, the latter equation leads to an expression of vectors $\mathbf{u}$, namely:

$$\mathbf{u} = -\left(\mathbf{M}^T\mathbf{M}\right)^I\mathbf{M}^T\left(\mathbf{J}\mathbf{B}\mathbf{J}_c^I\mathbf{V}\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_o\right). \quad \ldots \quad (40)$$

Going back from $\mathbf{u}$ to the original vectors $\mathbf{a}$ and $\dot{\mathbf{q}}_r$, the value of the joint velocity vector can be found as:

$$\dot{\mathbf{q}} = \mathbf{B}\mathbf{J}_c^I(\mathbf{V}\dot{\mathbf{p}}_c + \mathbf{W}\mathbf{a}) + \mathbf{L}\dot{\mathbf{q}}_r \quad \ldots\ldots\ldots \quad (41)$$

with

$$\mathbf{a} = \begin{bmatrix}u_1 & \cdots & u_{\text{rank}(\mathbf{W})}\end{bmatrix}^T \quad \ldots\ldots\ldots \quad (42)$$

and

$$\dot{\mathbf{q}}_r = \begin{bmatrix}u_{\text{rank}(\mathbf{W})+1} & \cdots & u_{\text{end}}\end{bmatrix}^T. \quad \ldots\ldots \quad (43)$$

Eq. (41) is a solution that satisfies the velocity vector at the contact point given by the contact admittance equation and that minimizes the deviation from the end-effector task. With the introduction of matrices $\mathbf{W}$ and $\mathbf{V}$, this equation can also handle a contact velocity vector and an end-effector task velocity vector of different dimensions. This approach to solve the problem of contact reaction is what we refer to as a *safety priority* approach as opposed to a *task priority* approach that would satisfy the end-effector task but would minimize the deviation from the desired velocity at the contact point. It should be noted that a redundant manipulator is the ideal case for these algorithms since with a high level of redundancy both constraints could be satisfied. In fact, the equation presented above is applicable to both redundant or non-redundant manipulators, but with different capabilities to continue the end-effector task while reacting to the contact.

5. Unification of the Three Operative Mode

Control algorithms for the three modes of operation of a cooperative manipulator have been independently defined above. Since all these modes are controlled based on the same control scheme, i.e., admittance control, this section presents how they can be integrated into a unique set of equations in order to simplify the controller.

As shown in **Fig. 1**, the cooperative motion mode and the autonomous mode require the exact same admittance equation but with different values for the damping and stiffness parameters. To integrate both controller in the same equation, a Heaviside function (H) is used. This function will have a value of 0 when the robot is in the cooperative mode and 1 when it is in autonomous motion mode, i.e.:

$$H = \begin{cases}0 & \text{if} \quad \text{cooperative} \\ 1 & \text{if} \quad \text{autonomous.}\end{cases} \quad \ldots\ldots\ldots \quad (44)$$

Using this Heaviside function, Eq. (6) can be modified to become:

$$\mathbf{f} = \mathbf{M}\ddot{\mathbf{p}} + \left[\delta_{e,e^*}\mathbf{C} + (1-\delta_{e,e^*})\mathbf{C}_s\right]\dot{\mathbf{p}} - H\mathbf{K}(\mathbf{p} - \mathbf{p}_d), \quad (45)$$

where

$$\mathbf{C} = \left(H2\zeta\sqrt{\mathbf{K}_r\mathbf{M}_r} + (1-H)\mathbf{C}_{vi}\right). \quad \ldots \quad (46)$$

Eq. (45) can be solved similarly to what is presented in Eq. (21), to find the end-effector velocity vector $\dot{\mathbf{p}}_o$ that should be followed. Once this vector is known, it should be integrated into Eq. (40) to calculate Eq. (41) and obtain the resulting joint vector.

If there are no contacts on the body of the manipulator, by definition, matrices $\mathbf{B}$ and $\mathbf{J}_c$ are zero while matrix $\mathbf{L}$ is the identity matrix. In this case, the result of Eq. (41) simply becomes:

$$\dot{\mathbf{q}} = \left(\mathbf{J}^T\mathbf{J}\right)^{-1}\mathbf{J}^T\dot{\mathbf{p}}_o = \mathbf{J}^I\dot{\mathbf{p}}_o. \quad \ldots\ldots \quad (47)$$

Since the presented approach uses the Jacobian matrix or a variation of the latter to generate the robot joint velocities, problematic behaviour should be expected around singularities. It is outside the scope of this paper to provide robust solutions for the control of the manipulator in singular configurations and many papers cover the latter topic. However, in our implementation of the algorithms, the so-called *singularity-robust inverse* proposed by Nakamura [20] was used to avoid singularity problems.

6. Experimentation

The presented control algorithm is a solution to pHRI that can produce autonomous motions as well as perform cooperative manipulation. In addition, this new controller also leads to a safer physical interaction. Indeed, assuming appropriate sensing, this algorithm provides a safe and intuitive robot reaction to a contact by rendering a given Cartesian admittance at a contact point while trying to continue to perform a given end-effector task. Al-

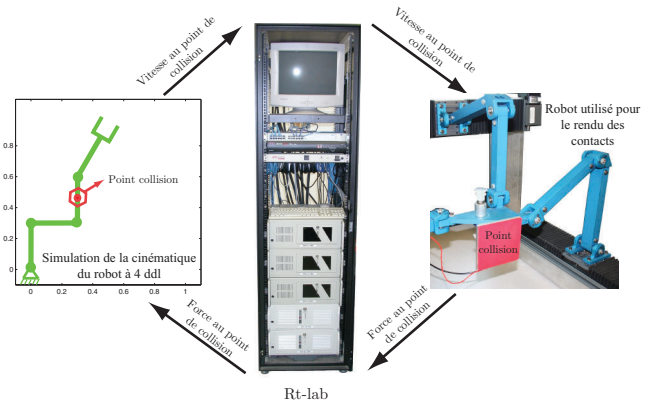

Fig. 5. Experimental setup: interaction between the simulated robot and the real robot.

though, experiments presented in [9, 21, 22] demonstrated the potential of the control solution to generate cooperative motion and also simulations were presented in Section 3 for autonomous motions, no experimental results have yet demonstrated the capability of the proposed algorithm to handle contact reaction. Therefore, in this section, it is shown that the derived equations are effectively capable of improving safety during human-robot physical interaction.

6.1. Experimental Setup

Simulation can be sufficient to verify if the kinematic reaction is as expected. However, part of this work implies interaction forces and the associated human perception that need to be experimentally verified. Therefore, the following experiment is a combination of both: a simulation and a real human-robot interaction. A 4-DOF planar serial robot is simulated using RT-Lab, a real-time control software based on Simulink. In this simulation, the robot is controlled at the end-effector to follow an arbitrary task. One of the points of this simulated robot is being linked with a real manipulator that will follow the exact kinematic displacement and that will also sense the force and render the desired contact admittance. The robot used to render this force is the Tripteron [23], a 3-DOF parallel robot that has excellent dynamic capabilities (acceleration up to 7 G). This robot is a fully decoupled translational parallel manipulator and hence the Jacobian matrix is the identity matrix in all configurations. In the present context, this characteristic has the advantage of limiting the potential pathological cases to the singularities associated with the simulated 4-DOF serial robot. The contact sensing is provided by a 6-DOF force sensor mounted between the parallel robot and an impactor. **Fig. 5** shows the simulated robot and its interaction with the real robot.

6.2. Experimenting Collision Between a Human and a Robot

Using the experimental testbed described above, the capability of the presented approach to reduce collision impact was investigated. The end-effector task was to maintain a velocity vector of 2.5 m/s in the x direction and 0 m/s in the y direction. During this task, a human subject was asked to maintain his arm in a given position and at a given place in the workspace of the robot to serve as an obstacle for the robot. From previous tests presented in [24] it was *a priori* known that the subject could not suffer any injury from the experiment and that the contact in the worst case would be perhaps unpleasant but not dangerous. The collision was set-up to occur in the simulation at the extremity of the second member. At the moment of impact, the real robot was travelling at 0.89 m/s, following exactly the velocity of the contact point of the simulated robot. The robot was controlled using Eqs. (41) and (45). Many collisions have been performed in order to test various values of admittance parameters.

6.3. Results

Figure 6 shows three of the force curves obtained: one corresponding to the case in which the robot does not react to the collision and two others corresponding to a reaction produced with two different sets of admittance parameters. As expected, the collision force peak for the case in which there are no collision sensing and reaction is above the unified pain tolerance threshold defined in [25] as the force limit at which humans start to feel discomfort. The peak force in this case is 67.78 N. Moreover, on this curve it can be also seen that after the collision there is a remaining force that opens the door to the unwanted clamping effect.

The other two curves show that using the approach presented in this paper, the collision force can be significantly reduced if the admittance is tuned properly. In both cases the resulting force peak is considerably lower than the unified pain tolerance threshold. Also, like the results with the admittance schemes reported in [26], the curves clearly show that after the impact peak, the force quickly decreases to return to zero, therefore cancelling the risk of clamping.

On the human side, the feeling associated with this reaction was way more intuitive and predictable than during the experiment reported in [24] where the reaction was simply to go as fast as possible in the direction opposite to the contact.

As it was also predicted in the first section of this paper, a low virtual mass and a high virtual damping produce a lower collision force peak. **Fig. 7** – where multiple tests are reported with different sets of contact admittance – clearly illustrates this effect. One may notice that below a certain virtual mass or above a certain damping value, the peak force does not continue to drop. This limitation appears to correspond to the maximum dynamic capabilities of the robot used in the experiment. Beyond this limit, the robot can no longer adequately render the desired ad-

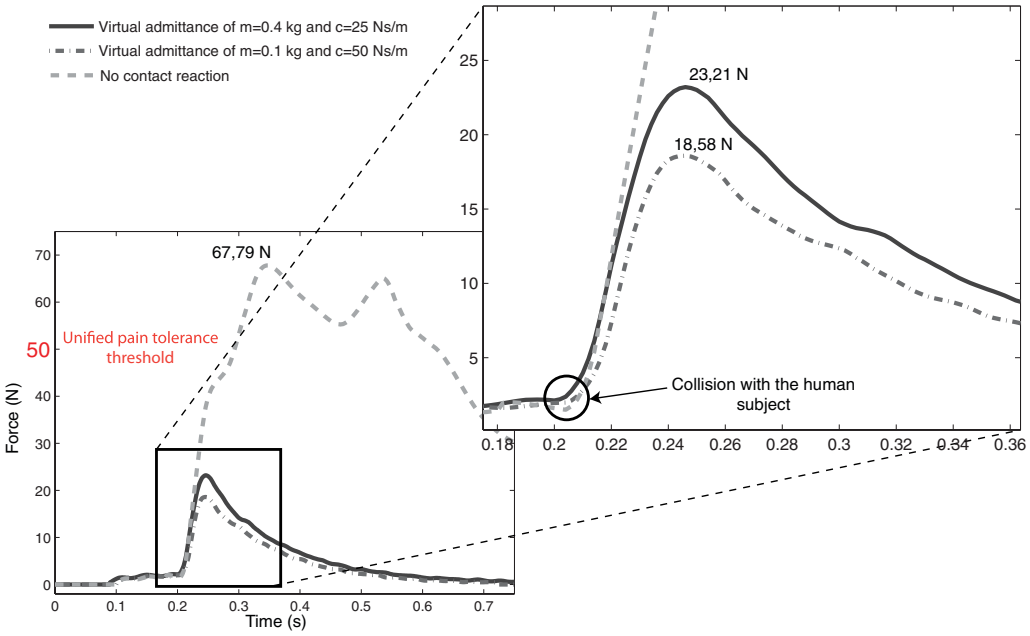

Fig. 6. Force curve for different contact reactions.

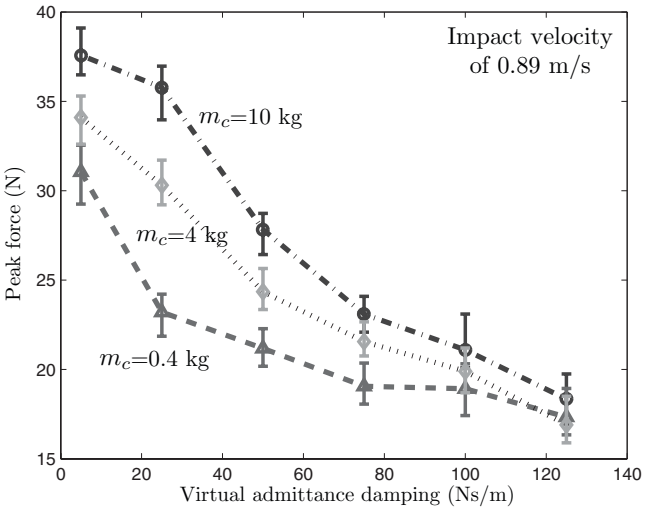

Fig. 7. Peak impact force at 0.89 m/s for different sets of admittance parameters.

mittance during the impact. This limit on the admittance should be known when implementing the proposed control scheme on a robot in order to adequately set the robot behaviour. From a safety point of view, exceeding this limit is not an issue. However, if the virtual mass is too small or the virtual damping too large, the interaction will be less effective. Therefore, the optimal contact admittance would be a set that leads to an acceptable impact force for the worst case and that will still be efficient and intuitive during interaction. This observation opens the door to variable contact admittance adjusted according to the contact properties but this is outside the scope of this paper.

7. Conclusion

In order to decrease the controller complexity of cooperative robots, we have proposed a single set of equations based on admittance control that can dictate the robot behaviour in its three typical modes of operation, namely, *cooperation motion*, *autonomous motion*, and *contact reaction*. An approach proposed in a prior publication for the control in the cooperation mode was first recalled. Then a novel approach was presented to produce autonomous motion and to efficiently react to contacts. Finally, it was shown how the equations to handle these three modes can be combined to obtain a simple unified control scheme.

Acknowledgements

This work was supported by The Natural Sciences and Engineering Research Council of Canada (NSERC) and by General Motors (GM) of Canada.

References:

[1] R. Kumar, P. Berkelman, P. Gupta, A. Barnes, P. Jensen, L. Whitcomb, and R. Taylor, "Preliminary experiments in cooperative human/robot force control for robot assisted microsurgical manipulation," in IEEE Int. Conf. on Robotics and Automation, Vol.1, pp. 610-617, 2000.

[2] N. Roy, G. Baltus, D. Fox, F. Gemperle, J. Goetz, T. Hirsch, D. Magaritis, M. Montemerlo, J. Pineau, J. Schulte et al., "Towards personal service robots for the elderly," in Proc. of the Workshop on Interactive Robotics and Entertainment (WIRE), 2000.

[3] J. Colgate, M. Peshkin, and S. Klostermeyer, "Intelligent assist devices in industrial applications: a review," in 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems 2003 (IROS 2003), Proc., Vol.3, 2003.

[4] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human-robot interaction," Mechanism and Machine Theory, Vol.43, No.3, pp. 253-270, 2008.

[5] S. Haddadin, A. Albu-Schaffer, and G. Hirzinger, "Safety evaluation of physical human-robot interaction via crash-testing," in Robotics: Science and Systems Conf. (RSS2007), 2007.
[6] R. Ikeura, H. Monden, and H. Inooka, "Cooperative motion control of a robot and a human," in 3rd IEEE Int. Workshop on Robot and Human Communication 1994 (RO-MAN'94), Nagoya, Proc., pp. 112-117, 1994.
[7] A. Edsinger and C. Kemp, "Human-robot interaction for cooperative manipulation: Handing objects to one another," in Proc. of the 16th IEEE Int. Symposium on Robot and Human Interactive Communication (RO-MAN), Citeseer, 2007.
[8] A. De Luca, A. Albu-Schaffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," in 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1623-1630, 2006.
[9] V. Duchaine and C. Gosselin, "Safe, Stable and Intuitive Control for Physical Human-Robot Interaction," in IEEE Int. Conf. on Robotics and Automation, 2009 (ICRA 2009), pp. 3606-3611, 2009.
[10] D. Surdilovic, "Contact stability issues in position based impedance control: Theory and experiments," Proc. of the Int. Conf. on Robotics and Automation, Vol.4, pp. 1675-1681, 1996.
[11] S. Arimoto, M. Sekimoto, H. Hashiguchi, and R. Ozawa, "Natural resolution of ill-posedness of inverse kinematics for redundant robots: a challenge to Bernstein's degrees-of-freedom problem," Advanced Robotics, Vol.19, No.4, pp. 401-434, 2005.
[12] H. Gomi and R. Osui, "Task-dependent viscoelasticity of human multijoint arm and its spatial characteristics for interaction with environments," J. Neuroscience, Vol.21, pp. 8965-8978, 1998.
[13] F. Lacquaniti, F. Licata, and J. Soechting, "The mechanical behavior of the human forearm in response to transient perturbations," Biological Cybernetics, Vol.44, No.1, pp. 35-46, 1982.
[14] R. Kearney and I. Hunter, "System identification of human joint dynamics," Crit. Rev. Biomed. Eng., Vol.18, No.1, pp. 55-87, 1990.
[15] M. Rahman, R. Ikeura, and K. Mitzutani, "Investigating the impedance characteristic of human arm for development of robots to cooperate with human operators," Proc. IEEE Int Conf. on Systems, Man and Cybernetics, Vol.2, pp. 676-681, 1999.
[16] K. Yamada, K. Goto, Y. Nakajima, N. Koshida, and H. Shinoda, "A sensor skin using wire-free tactile sensing elements based on optical connection," 41st SICE Annual Conf., Vol.1, 2002.
[17] M. Shimojo, A. Namiki, M. Ishikawa, R. Makino, and K. Mabuchi, "A tactile sensor sheet using pressure conductive rubber with electrical-wires stitched method," Sensors Journal, IEEE, Vol.4, No.5, pp. 589-596, 2004.
[18] J. Engel, J. Chen, and C. Liu, "Development of polyimide flexible tactile sensor skin," J. of Micromechanics and Microengineering, Vol.13, No.3, pp. 359-366, 2003.
[19] T. Someya, T. Sekitani, S. Iba, Y. Kato, H. Kawaguchi, and T. Sakurai, "A large-area, flexible pressure sensor matrix with organic field-effect transistors for artificial skin applications," Proc. of the National Academy of Sciences, Vol.101, No.27, p. 9966, 2004.
[20] Y. Nakamura, "Advanced robotics: redundancy and optimization." Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1990.
[21] V. Duchaine and C. Gosselin, "General Model of Human-Robot Cooperation Using a Novel Velocity Based Variable Impedance Control," World Haptics 2007, pp. 446-451, 2007.
[22] V. Duchaine and C. Gosselin, "Investigation of human-robot interaction stability using Lyapunov theory," Int. Conf. on Robotics and Automation 2008, pp. 2189-2194, 2008.
[23] C. M. Gosselin, X. Kong, S. Foucault, and B. Ilian, "A fully decoupled 3-dof translational parallel mechanism," Parallel Kinematic Machines Int. Conf., pp. 595-610, 2004.
[24] V. Duchaine, N. Lauzier, M. Lacasse, M. Baril, and C. Gosselin, "A Flexible Robot Skin for Safe Physical Human Robot Interaction," in IEEE Int. Conf. on Robotics and Automation 2009 (ICRA 2009), pp. 3606-3611, 2009.
[25] Y. Yamada, Y. Hirasawa, S. Huang, Y. Umetani, and K. Suita, "Human-robot contact in the safeguarding space," IEEE/ASME Trans. on Mechatronics, Vol.2, No.4, pp. 230-236, 1997.
[26] S. Haddadin, A. Albu-Schaffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction," in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Nice, France, 2008.

Name:
Vincent Duchaine

Affiliation:
Professor, Department of Automated Manufacturing Engineering, École de Technologie Supérieure

Address:
1100 Rue, Notre-Dame Ouest, Montreal, H3C 1K3, Canada

Brief Biographical History:
2010 Received Ph.D. in Mechanical Engineering from Université Laval
2010- Post-Doctoral Fellow, Stanford University
2010- École de Technologie Supérieure

Main Works:
• control and sensing for physical human-robot interaction

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE)

Name:
Clément Gosselin

Affiliation:
Département de Génie Mécanique, Université Laval

Address:
1065 Avenue de la médecine, Québec, Qc, G1V 0A6, Canada

Brief Biographical History:
1989- Joined Laval University
1995- Visiting Researcher, RWTH, Aachen, Germany
1999- Visiting Researcher, IRCyNN, Nantes, France
2001- Canada Research Chair, Laval University

Main Works:
• "Type Synthesis of Parallel Mechanisms," Springer Tracts in Advanced Robotics, 2007.
• "Underactuated Robotic Hands," Springer Tracts in Advanced Robotics, 2008.

Membership in Academic Societies:
• Fellow, The American Society of Mechanical Engineers (ASME)
• Senior Member, The Institute for Electrical and Electronics Engineers (IEEE)
• Member, The Canadian Committee for the Theory of Machines and Mechanisms (CCToMM)