

## A HYBRID FORCE FOLLOWING CONTROLLER FOR MULTI-SCALE MOTIONS

Philipp STOLKA and Dominik HENRICH

Embedded Systems and Robotics Lab. (RESY)  
Faculty of Informatics, Building 48  
University of Kaiserslautern, D-67653 Kaiserslautern, Germany  
E-Mail: philipp@stolka.de, <http://resy.informatik.uni-kl.de/>

**Abstract:** In many robotic applications, the teaching of points in space is necessary to register the robot coordinate system with the one of the application. Robot-human interaction is awkward and dangerous for the human because of the possibly large size and power of the robot, so robot movements must be predictable and natural. We present a novel hybrid control algorithm which provides the needed precision in small scale movements while allowing for fast and intuitive large scale translations. *Copyright © 2002 IFAC*

### 1. INTRODUCTION

When robot systems are confronted with changing positions of the objects to work with, it is necessary to implement some procedure to teach the robot the new respective locations. When the relative positions of objects in the robot's workspace are known, this teaching reduces to registering a task coordinate system which can be done by positioning the robot at three distinct points in space. Two possibilities to achieve this come to mind: First, the robot can be positioned through motion commands over a Manual Control Pendant (MCP) or a textual terminal. This method is rather cumbersome due to the need to mentally convert the next approximation step into movement commands. Or, secondly, the robot can be controlled by manual force-based following (*Force Following*) of the robot, effected by the user gripping and guiding the robot directly. This is of course more intuitive and very probably offers better results in terms of time and goal deviation through direct visual inspection of goal approximation.

However, when robot size and power increases, it becomes more dangerous for the human operator to work in the robot's workspace. Therefore, it is necessary to provide a control method which allows for precise and slow small-scale motions where necessary while at the same time offering convenient large-scale behavior, i.e., fast and easy translation over longer distances. Viewed in terms of dynamics, stability, and ergonomics, these two are conflicting

goals difficult to achieve with conventional control algorithms.

Many applications have a need for both types of motions. Among these could be: high-precision teaching of positions for repeated pick-and-place operations with large-scale motions, using the robot as an active positioning device (tool holder) for human operator support, or handling of sensitive components (e.g. glass plates) in robot-assisted assembly. In this paper, we propose a suitable control algorithm for these tasks.

The state of the art will be presented first, focusing on systems and concepts that we used as starting points. Then, we present two closely related control schemes and compare them with respect to their ease-of-use and their degree of fulfillment of the mentioned requirements. Safety considerations follow. Finally, the proposed control scheme is experimentally validated for its utility in the envisioned application types.

### 2. STATE OF THE ART

Some important considerations when deciding on a force following control algorithm are the following: First, the stability of the controller for the given inputs must be ensured, e.g. no oscillations during contact state changes are allowed. Second, the desired dynamics have to be evaluated, e.g. is a required steady force acceptable? Third, precision may be important. For many applications, e.g. robot-assisted surgery, sub-millimeter accuracy is required. And fourth, speed constraints may apply, so precise

but slow controllers are undesirable.

In our system for robot-based milling in the lateral skull base, it is necessary to provide three points that define a local coordinate system for the milling volume geometry. Previously, these have been taught by manual guidance or force following with an algorithm based on P (proportional) control [Waringo01]. Moving the robot in the direction of the measured force vector, a stable controller for registration was implemented. However, the remaining steady-state deviation under constant force makes this control scheme tedious to handle for the user. Furthermore, it is impossible to find a proportional gain factor that allows for both high-speed large scale motions and precise small scale positioning.

One system that has similar objectives to the presented one is the ROBODOC surgical robot system. A SCARA-type robot is used for the milling of implant beds in the human femur. Precision is paramount in this type of application, so the initial registration procedure has to be performed with only small target errors. When the taught positions deviate more than three millimeters from the expected ones, the registration is considered as failed. Technically, manual guidance is achieved via a zero force control scheme with subsequent tactile fine search, following a ball-in-cone algorithm. (Alternatively, marker-free registration can be achieved by touching 20...30 neighboring points on exposed bone.)

One approach to force-based robot control is to make the tool movement as intuitive as possible, basing it on physical analogies. In [Hein00], an inertia-based control algorithm for force following of a hexapod robot has been derived, although it has been linearized at certain points. The tool is associated with a virtual mass and with simulated friction which behaves ideally in Newtonian physics. It is expected that the user can handle the attached tool as if it were free-floating. However, though intuitive, this approach brings about new problems, among them the need for collision detection, as the high forces resulting from stiff contact accelerate the robot rapidly.

With regard to safety, stand-alone force following systems may be insufficient for applications where human reaction possibilities (with respect to self-protection) are limited. Additional measures need to be taken to implement safe motion.

In [Heinzmann99], a system similar based on a serial kinematics robot arm is presented which is designed to move like in zero-gravity space (Zero-G). The single joints are backdrivable, so direct interaction with the kinematic structure is possible through torque measurements at the joints. The dynamics are modeled to allow constraining of the energy of the robot: Should the energy of any part of the structure exceed predefined thresholds, a *Maximum Impact Force* (MIF) Controller limits the motor torques.

When combining different needs into a control system like contact-based and contact-free or high- and low-speed motions, it may be necessary to use hybrid controllers. One type (*hybrid position/force control*) is described in [Craig86] which accommodates both position control along free (unconstrained) axes and force control along contact axes.

As to the usability of different force following control schemes in six degrees of freedom (6DOF) manipulation tasks, the literature is unclear. An overview in [Zhai95] shows no clear preferences between zero-order control (direct, *position control*) and first-order control (integrating, *rate or speed control*) for 2DOF. Position control is slightly superior in small scale positioning tasks with respect to time requirements, while larger scale motions call for rate control. However, this depends strongly on the type of input device. In general, higher-order control schemes (e.g. acceleration control) are deemed inappropriate for human use due to their more complex interaction patterns.

### 3. PHYSICALLY INSPIRED CONTROL ALGORITHMS

We assume that ease-of-use of a force-following robot is connected with intuitiveness, which in turn is determined by clear physical analogies. Therefore, we derived a control scheme based on Newtonian physics (therefore called *Newton Control* or *N control*) to be used as a positioning aid in the envisioned applications.

Inertia, acceleration and friction play an important role for modeling a moving mass. As the motion of a body with mass  $m$  and without external forces  $\mathbf{F}$  continues infinitely, its vector  $\mathbf{v}$  remains unchanged according to the first Newtonian law. Acceleration  $\mathbf{a}$  is given by the second law  $\mathbf{F} = m \cdot \mathbf{a}$ . Friction force  $\mathbf{F}_{\text{frict}}$  is assumed as inversely linear in the exerted force  $\mathbf{F}$  according to a friction coefficient  $f_{\text{frict}}$ . Having defined the virtual mass  $m > 0$  and a friction coefficient  $f_{\text{frict}}$ ,  $0 \leq f_{\text{frict}} < 1$ , the resulting control equation is as follows:

$$\mathbf{v}_{\text{comp}}(t) = \mathbf{v}_{\text{comp}}(t-1) + \left( \frac{\mathbf{F}_{\text{user}}(t)}{m} - \frac{f_{\text{frict}} \cdot \mathbf{v}_{\text{comp}}(t)^2}{2\Delta s} \right) \cdot \Delta t$$

The computed speed  $\mathbf{v}_{\text{comp}}(t)$  is defined with recourse to the speed from the last control cycle  $\mathbf{v}_{\text{comp}}(t-1)$ , thereby providing inertia. The force  $\mathbf{F}_{\text{user}}()$  exerted by the user is sampled at discrete time points and assumed constant over the past control cycle, and through scaling with the mass  $m$ , the user-effected acceleration is computed. Recoursing to kinetic energy, the last term defines the negative acceleration owing to friction effects over a path length of  $\Delta s$ . The duration  $\Delta t$  is the control cycle time which is constant due to the real-time operating system on the robot controller unit and amounts to  $\Delta t = 16\text{ms}$ .

The N controller can be transformed into an I (integral) controller when neglecting friction:

$$\begin{aligned}\mathbf{v}_{\text{comp}}(t) &= \mathbf{v}_{\text{comp}}(t-1) + \left( \frac{1}{m} \cdot \frac{\Delta t}{T_N} \mathbf{F}(t) \right) \\ &= \left( \frac{V_I}{T_N} \sum_{t=0}^{t-1} \mathbf{F}(t) \Delta t \right) + \left( \frac{V_I}{T_N} \mathbf{F}(t) \Delta t \right) \\ &= \frac{V_I}{T_N} \sum_{t=0}^t \mathbf{F}(t) \Delta t\end{aligned}$$

Having set the gain factor  $V_I = m^{-1}$  and the follow-up time  $T_N = 1$ , the N control equation is equal to the discretized I control equation.

While this N control scheme proved to be appropriate for large scale and high speed motions in initial experiments [Stolka02], it is not suited for high precision tasks. There, the human operator grips the tool tighter, resulting in higher stiffness and possible oscillations. As the user tries to dampen the oscillation by gripping even tighter, the resonance frequency goes up, and the system becomes unstable. Even when the user does not exert high forces, the system is very sensitive during exact positioning, effectively rendering high precision impossible.

Thus, we developed a control scheme combining both the convenient large scale behavior of N control with the high precision of P control without including their respective drawbacks. We settled on a hybrid controller that switches between P control at low speeds to N control at higher speeds. Experiments with users of different experience levels with this controller showed that a useful threshold speed for switching is about  $v_{\text{thresh}} = 10\text{mm/sec}$  for the task described in “Experiments”. As a sudden control change between P and N control is undesirable, interpolation between both was implemented, governed by the current absolute speed  $v_{\text{abs}} = |\mathbf{v}| = |\mathbf{v}(t)|$ , allowing for a gradual, sliding mixing of both controllers’ effects according to the formula

$$\frac{N}{P} = r(|\mathbf{v}|) = \begin{cases} \left( \frac{|\mathbf{v}|}{v_{\text{thresh}}} \right)^{\alpha}, & \text{if } |\mathbf{v}| < v_{\text{thresh}} \\ 1, & \text{else} \end{cases}$$

The exponential factor  $\alpha > 0$  determines the slope behavior of the mixing (linear, quadratic, ...). User interviews showed that linear interpolation ( $\alpha = 1$ ) is the easiest to master for most users [Stolka02].

The P and N controllers are giving two parallel sets of speed components,  $\mathbf{v}_{\text{Pcomp}}(t)$  and  $\mathbf{v}_{\text{Ncomp}}(t)$ . The final speed is a linear combination of both:

$$\begin{aligned}\mathbf{v}_{\text{N/Pcomp}}(t) &= (1 - r(|\mathbf{v}|)) \cdot \mathbf{v}_{\text{Pcomp}}(t) \\ &\quad + r(|\mathbf{v}|) \cdot \mathbf{v}_{\text{Ncomp}}(t)\end{aligned}$$

The presented controllers are implemented on a Stäubli RX130 industrial robot (serial 6DOF kine-

matics) with an Adept CS7 control unit (real time operating system V+ 12.3, CPU: 68040, 40MHz, 32MB RAM). All of the experiments were conducted on this system.

Plotting exerted absolute force  $|\mathbf{F}|$  vs. absolute speed  $|\mathbf{v}|$  from an experiment (approximating a single point five times in a row in contact-free motion) results in the following graphs. Figure 1 shows the linear relationship with a simple P controller. High forces result in only small speeds. The slope can be influenced by changing the controller gain, but obviously choosing too high a value results in coarse small scale motions.

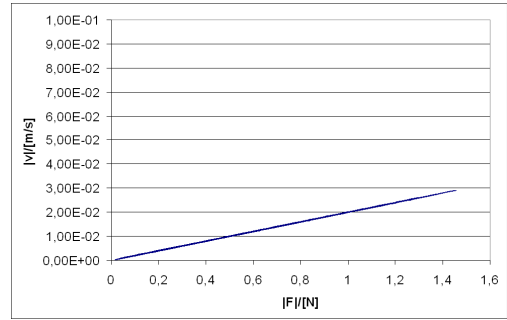


Figure 1: Absolute speed  $|\mathbf{v}|$  vs. absolute force  $|\mathbf{F}|$  with P control

Figure 2 shows the same for the N controller. No direct relation between  $|\mathbf{v}|$  and  $|\mathbf{F}|$  exists anymore, so different speed levels may correspond to a single force level and vice versa. Especially irritating is the large variation in speeds for small forces, reflecting the controller’s unsteadiness in exact positioning tasks.

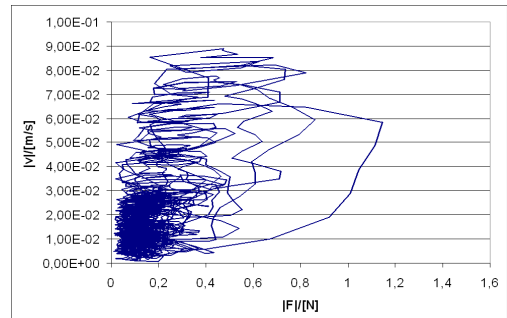


Figure 2: Absolute speed  $|\mathbf{v}|$  vs. absolute force  $|\mathbf{F}|$  with N control

Figure 3, finally, shows a clear and repeatable hysteresis cycle for the Hybrid N/P controller. Starting from small forces and small speeds, increasing forces map quite linearly to increasing speeds until the speed threshold  $v_{\text{thresh}}$  is reached. Then, forces can be lowered until a steady high speed is attained with only a comfortably small effort (still, friction requires a tiny net force). Decelerating towards the goal is achieved by zeroing the exerted force, which reduces

speed slowly through friction until  $v_{\text{thresh}}$  is reached again from above. This results in a rapid rise of P control influence, decelerating nearly instantly. Final exact positioning occurs within P control bounds.

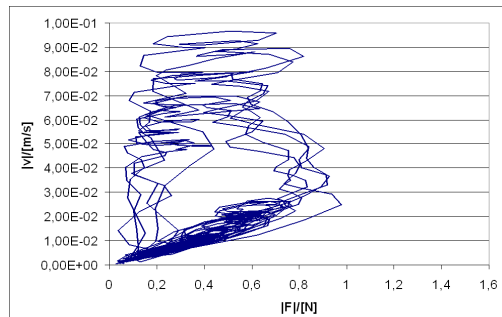


Figure 3: Absolute speed  $|\mathbf{v}|$  vs. absolute force  $|\mathbf{F}|$  with Hybrid N/P control

#### 4. SAFETY CONSIDERATIONS

One aspect that is especially important for positioning tasks that include contact-state changes is the susceptibility of the control scheme to collisions between the tool (e.g. gripper) and the environment. These collisions are associated with sudden changes of environment stiffness. Due to the control cycle time lag, high amplitudes of the measured forces may result upon contact. With integral controllers like the N controller, the simulated inertia aggravates the problem, as the mass moves further into the obstacle until integrated acceleration results in a net reverse movement that may be uncontrollable.

Usually, contact state changes in general can be detected by analyzing the measured moments and forces. In our system, however, the user holds the tool combination itself. Therefore, attempts on modeling forces are futile, as user-effected forces interfere with the contact forces. In one experiment, the N-controlled robot was manually guided into collisions with a rigid obstacle four times, yielding a relationship between collisions, forces and the computed speed of the robot (Figure 4).

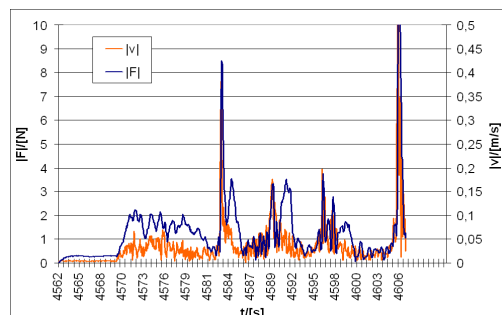


Figure 4: Absolute forces  $|\mathbf{F}|$  and speeds  $|\mathbf{v}|$  during collisions with a rigid obstacle (N control)

Obviously, both absolute forces  $|\mathbf{F}(t)|$  and absolute speed  $|\mathbf{v}(t)|$  peak during and after a collision. Taking only one of these as a collision indicator is obviously insufficient. Thus, a combination of both indicators was used as follows:

$$\left( |\mathbf{v}(t)| > 0.05 \frac{m}{s} \right) \wedge \left( |\mathbf{v}(t)| > 3 \cdot |\mathbf{v}(t-3) \right) \wedge \left( |\mathbf{F}(t)| > |\mathbf{F}|_{\max} \right)$$

The first term clips absolute speeds below 5cm/sec, as collisions below that speed are considered uncritical. The empirically derived second term checks for high accelerations: If the current speed  $|\mathbf{v}(t)|$  is more than three times higher than the one three cycles before, it is likely that a collision-reaction movement is occurring. And lastly, the third term checks for inappropriately high forces  $|\mathbf{F}(t)|$ : An empiric threshold  $|\mathbf{F}|_{\max} = 10$  Newton (N) was defined before; greater input forces are scaled down to this magnitude. Combining these indicators (by logical AND) allows safe and easy approaching of rigid target points, as potential collisions are detected and handled. The latter occurs by stopping the robot and moving out of the collision area by lifting the robot over  $s = 10\text{mm}$ . This approach is feasible in the investigated application, as the robot and the tool combination do not change their orientation, resulting in a collision-free retraction path outwards along the z axis.

Another problem identified in the experimental phase are singularities of the robot kinematics. Certain configurations of the robot cause very large movements in configuration (joint) space, although only small Cartesian distances are covered. The currently investigated application requires the robot to be bent in such a way that allows only the wrist-type singularity to occur (caused by collinear alignment of joints four and five). Moving through this singularity results in rapid rotation of the forearm and causes an envelope error, i.e. having the actual joint positions lag grossly behind the planned ones, thus calling for robot shutdown.

In our case, the only safety-relevant quantity is (rotational) speed, as the rotating mass itself cannot possibly be dangerous to the user. User interviews showed that the rotational speed of joint four should be thresholded to  $v_{\text{rot4}} = 45^\circ/\text{sec}$  for perceived safety and convenient use.

Two ways to detect and handle such singularities are possible: First, to analyze the difference between a (saved) past joint configuration  $\theta(t-n)$  and the current one  $\theta(t)$ . This *ex post* analysis is easy and cheap, but does not avoid hitting singularities completely. (For a single path segment, it is possible that a singularity exists between the start and the end point, so that a discrete *ex post* checking method cannot show the existence of the singularity before hitting it.) Second, one can extrapolate the current trajectory

into the future and compare the projected joint configuration  $\theta(t+n)$  with the current one  $\theta(t)$  (*ex ante* approach). While this allows to check for fast rotations in the near future, it doubles the computation load on the trajectory generator as it has to calculate the target joint positions twice. Besides, the extrapolation has to be justified by a reasonable model – which is difficult to define in the light of unpredictable user force inputs. However, assuming that the single path segments are only very small (as holds for the presented system with mean = 1,3mm, maximum  $\approx 8$ mm), the *ex post* approach is valid as well. In the experimental system, Cartesian speeds that resulted in excessive rotational speeds of joint 4 were scaled down until  $v_{\text{rot}4} = 45^\circ/\text{sec}$ . With this method, singularities are not dangerous anymore and the user is not surprised by sudden rotational speed changes. This method obviously violates the simple physical analogy of the N controller, but the effects are clearly preferable to a sudden shutdown.

## 5. EXPERIMENTAL RESULTS

The three implemented control algorithms (P, N, and Hybrid N/P control) were compared through the following experiment. The setup consisted of a fixed table with two marked spots  $P_0, P_1$  (distance  $l = 1$  m) within the robot’s working space. Both points were taught manually (via the MCP) before the experiment took place in order to have as exact position data as possible. The task was to approach each point and to confirm the position, whereupon automatic force-based approximation along the z-axis took place. This was repeated  $n = 14$  times in a row for each control algorithm (N, P, N/P). The time of each target point approximation  $t_i$  and the remaining position deviation  $d$  was recorded. Besides, during each control cycle, forces  $|\mathbf{F}(t)|$  and the current position  $\mathbf{P}(t)$  were logged.

The resulting data for each controller consisted of  $n$  2-tuples  $(\Delta t, d)$  ( $\Delta t$  being the duration of each approximation) and of the *perceived total energy*

$$\begin{aligned} E_{\text{FF}} &= \sum_t \Delta E_{\text{FF}}(t) \\ &= \sum_t |\mathbf{F}(t)| \cdot |\mathbf{P}(t) - \mathbf{P}(t-1)| \end{aligned}$$

which is an approximation of the energy along the way. Obviously, for  $\Delta E_{\text{FF}}(t)$  to stand for the physical energy expended along  $\mathbf{s} = (\mathbf{P}(t) - \mathbf{P}(t-1))$ , it would be necessary to scale each term with  $\cos(\alpha)$ , where  $\alpha$  is the angle between the force vector  $\mathbf{F}(t)$  and the motion vector  $\mathbf{s}$ . However, we are not interested in the physical energy that is needed for translating the robot from  $\mathbf{P}_i$  to  $\mathbf{P}_{i+1}$ , but rather in the “perceived effort” that the user has expended during the motion, independent of the relative orientation of the forces to the respective path segment’s motion vector. Fur-

thermore, to normalize the  $E_{\text{FF}}$  values with respect to the needed total time  $t_{\text{ges}}$ , we computed the *perceived total power*  $P_{\text{FF}}$  for each controller as

$$P_{\text{FF}} = \frac{E_{\text{FF}}}{t_{\text{ges}}}$$

The results confirmed the expectations. While P control proved to be exact and slow, N was fast, albeit with large end deviations. The Hybrid N/P controller combined the others’ advantages (fast, precise) without their drawbacks (Figure 5).

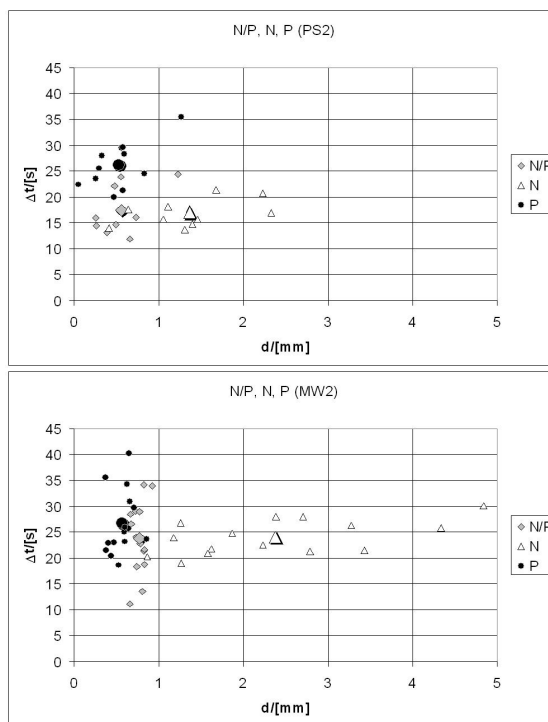


Figure 5: Final position deviation  $d$  vs. approximation duration  $\Delta t$  (test persons PS2 and MW2) (larger symbols: mean for each controller)

The results of two test persons show the effects of learning the different controllers’ behavior. Person PS2 already knew the switching behavior of the Hybrid N/P controller, so large scale motions were done swiftly, with exact positioning within P control bounds, this being reflected in high speed and high precision. With the P controller, positioning was similarly precise, but with longer duration for the motions; and while N control was as quick as N/P, it showed lower precision and greater variation in  $d$ .

The second person MW2 was not yet acquainted with the N/P switching, thus movements were slower, resulting in a longer mean duration  $\Delta t$  with N/P. Results for the P controller are comparable with PS2’s. Finally, N control was very unreliable, taking longer, being imprecise and having a great variation in  $d$ .

The total time  $t_{ges}$  was lower for N and N/P with both users, with N/P being relatively even better for inexperienced users (Figure 6). Total energy  $E_{FF}$  was very high for both kinds of users with P control (which might be reduced by setting a higher gain factor, but then precision would suffer). N and N/P are comparable in this respect. Normalizing  $E_{FF}$  to  $P_{FF}$  by the total time shows results very much like with  $E_{FF}$ , i.e. higher values for P and much lower, comparable values for N and N/P.

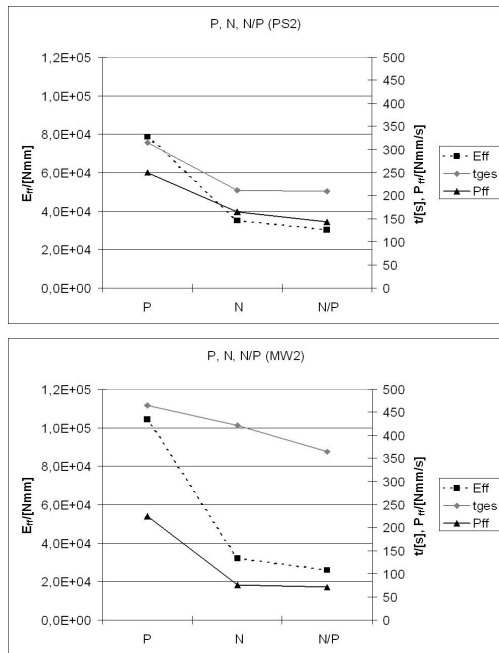


Figure 6: Perceived total energy  $E_{FF}$ , total experiment duration  $t_{ges}$ , and total perceived power  $P_{FF}$  for controllers P, N, N/P (test persons PS2 and MW2)

## 6. CONCLUSIONS

We have described a novel controller (Hybrid N/P control) for force following tasks based on the concept of hybrid control. Using P control for slow, small scale motions together with N control for fast, large scale motions, the goals of accuracy and speed were combined in one hybrid controller. Switching between them was accomplished through linear combination of their outputs, governed by a sliding, absolute speed-based variable. Preliminary experiments yielded a reasonably useful set of parameters (N control parameters like mass and friction, Hybrid N/P control parameters like threshold speed and interpolation type). Furthermore, it was proved experimentally that the Hybrid N/P controller provided the advantages of both primitive controllers (accuracy, speed) without introducing their drawbacks (high force requirements, instability).

Singularities of the robot kinematics are detected with an *ex post* strategy of checking joint speeds and handled by scaling the Cartesian tool tip speed down until rotational speed constraints are satisfied. Collisions that can lead to unstable behavior are detected with a simple force- and speed-based filter, thus avoiding jerky reactions.

Future work will embrace functional aspects like the integration of physical-based orientation control, but also safety features like constraining joint speeds for all joints (thus checking for all singularities).

## ACKNOWLEDGEMENTS

This work is a part of the project “*Robotergestützte Navigation zum Fräsen an der lateralen Schädelbasis (RONAF)*” (“Robot-based Navigation for Milling at the Lateral Skull Base”) of the special research cluster “*Medizinische Navigation und Robotik*” funded by the Deutsche Forschungsgemeinschaft (DFG).

## REFERENCES

- [Craig86] Craig, J. (1986). *Introduction to Robotics, Mechanics and Control*. Addison-Wesley Publishing Company.
- [Hein00] Hein, A. (2002). *Eine interaktive Robotersteuerung für chirurgische Applikationen*. Dissertation, Technische Universität Berlin, Fachbereich 13 – Informatik.
- [Heinzmann99] Heinzmann, J.; Zelinsky, A. (1999). *Building Human-Friendly Robot Systems*. In: Proceedings of International Symposium of Robotics Research ISRR’99, Salt Lake City, USA, 9-12 October 1999.
- [Mittelstadt93] Mittelstadt, B.; Kazanzides, P.; Zuhars, J.; Cain, P.; Williamson, B. (1993). *Achieving Predictable Results in an Unpredictable Environment*. In: Proceedings ICAR ’93.
- [Stolka02] Stolka, Ph. (2002). *Robotergestützte Navigation zum Fräsen an der lateralen Schädelbasis (RONAF): Auswertung von Kraftsensordaten*. Diploma Thesis, Universität Kaiserslautern, Fachbereich Informatik, AG Robotik und Eingebettete Systeme (RESY).
- [Waringo01] Waringo, M. (2001). *Robotergestützte HNO-Chirurgie (ROTOS – Robotics in Otolaryngology Surgery): Fräsalgorithmus*, Project Thesis. Universität Kaiserslautern, Fachbereich Informatik, AG Robotik und Eingebettete Systeme (RESY).
- [Zhai95] Zhai, Shumin (1995). *Human Performance in Six Degree of Freedom Input Control*, Dissertation. University of Toronto, Ergonomics in Teleoperation and Control Lab, Department of Industrial Engineering.