

Trajectory Generation under the Least Action Principle for Physical Human-Robot Cooperation

Martin Lawitzky Melanie Kimmel Peter Ritzer Sandra Hirche

Institute for Information-Oriented Control

Technische Universität München

80290 Munich, Germany

Email: ml@tum.de, melanie.kimmel@mytum.de, peter.ritzer@mytum.de, hirche@tum.de

Abstract—Trajectory generation for active physical assistance to humans in cooperative haptic tasks gains increasing interest in recent literature. Planning-based approaches represent one class of trajectory synthesis methods for active robotic partners. To overcome the limitations of kinematic planning algorithms in dynamic tasks, we propose a three-step approach to the synthesis of trajectories under the principle of least action. This is motivated by neuroscientific findings on human effort minimization in motor tasks. A trajectory is generated by optimized sequencing of optimal motion primitives. The benefits of the proposed method for physical human-robot cooperation are demonstrated in human user studies in a 2D cooperative transport task in a virtual maze.

I. INTRODUCTION

Physical assistance to humans is of high relevance for many future industrial and domestic application scenarios of intelligent robotic systems. Whether fast and precise motion under direct human guidance is required in an assembly/construction setting or safe and intuitive strength assistance for elderly is the challenge, goal-directed assistance beyond pure force magnification is desirable. First works on physical robotic assistance consider passive support without an active urge towards task completion [1], [2]. More recent works including our own employ either learning-based [3]–[5] or planning-based [6]–[10] methods to implement a goal-directed assistive behavior. Alternatively, a common desired trajectory is given by an expert [11]. Planning-based approaches thus far operate purely on kinematic constraints neglecting task dynamics and require full explicit scene knowledge. Learning-based approaches require a training phase but generate implicit knowledge of the scene and the task dynamics. To the best of the authors’ knowledge, no state-of-the art method is able to generate a goal-directed assistive robot behavior in a complex environment utilizing explicit knowledge of the task dynamics.

In this work, we target the problem of trajectory generation for goal-directed physical assistance considering the dynamics of a commonly manipulated object. We propose a novel three-step scheme for the computation of start-to-goal motion trajectories. The resulting trajectory serves as prediction of the human’s desired trajectory during cooperation. Our approach aims to minimize the required human effort under kinodynamic constraints for complex scenarios beyond point-to-point primitive motions. Note, that mechanical effort in cooperation relates to interaction efficiency [12].

Different methods and paradigms for the estimation of human desired motion trajectories are found in the literature. The most widely used principle of jerk minimization is validated for free-space arm motion of humans [13]. A successful transfer of this principle to cooperative manipulation [14] is outperformed by a polynomial model targeting simple cooperative point-to-point movement [15].

In object transport tasks, the least-action principle becomes a more relevant aspect. The computation of an energy optimal trajectory even for a point mass subject to viscous friction in 2D is generally computationally intractable for the general case including non-convex constraints [16]. Biological findings inspire the deployment of motion primitives to synthesize more complex actions [17]. This is successfully applied also for the planning of a manipulator motion, where linear-quadratic (LQ) optimal motion primitives are combined with a heuristic planning approach in [18] leading to fast suboptimal results. A more general method for the derivation of motion primitives (maneuvers) is proposed in [19]. The authors propose an exploitation of inherent system symmetries to determine optimal maneuvers, both approaches [18], [19] are limited to kinematic constraints. Especially in the context of physical human-robot cooperation, however, constraints on acceleration and velocity are safety-relevant and must be incorporated in the trajectory-generation procedure.

In this paper, we propose a three-step optimization approach tailored towards the requirements in human-robot cooperative manipulation: Energy-optimal trajectory segments within given acceleration and velocity limits are sequenced based on an initial solution produced by a feedback motion plan. The sequence is further iteratively optimized towards a minimized overall energy consumption. The individual segments render straight lines or single curves, a reasonable optimization horizon, traceable for the human partner. Under these conditions, a scheme for trajectory generation under the least-action principle provides a desirable trajectory that obeys not only the environmental constraints but also velocity and acceleration limits that allow comfortable interaction with the human partner. We evaluate our proposed approach in an experimental human user study in a 2D cooperative transport task in a virtual environment. While the goal is known to the human and the robotic partner, no information on the robot’s desired trajectory is provided to the human

other than through haptic interaction. The results show significant benefits for cooperation efficiency over kinematic feedback plans investigated in our earlier works [10].

The remainder of this work is organized as follows: Section II describes our overall approach using an implementation of optimized motion primitives described in Section III. Our experimental evaluation is presented in Section IV.

Notation: By convention, in this paper bold symbols denote vectors. The Q -weighted norm of vector \mathbf{x} is denoted $\|\mathbf{x}\|_Q$ such that $\|\mathbf{x}\|_Q^2 = \mathbf{x}^\top Q \mathbf{x}$.

II. TRAJECTORY GENERATION SCHEME

In this work, we confine our problem to the following conditions: A virtual circular object with point-mass dynamics exposed to viscous friction is cooperatively moved through a cluttered environment. The sum of forces of human and the virtual robotic partner acts on the object. The common goal in terms of final configuration is known to both cooperation partners. A path from starting position to the goal exists.

In this section, an iterative method is derived, to derive a point mass trajectory from an admissible initial to a final state under given constraints aiming to minimize the required cost. We consider the dynamics of a 2D point mass m motion with isotropic viscous friction d described by

$$m\ddot{\mathbf{x}} + d\dot{\mathbf{x}} = \mathbf{u} + \mathbf{u}_h, \quad \mathbf{x}, \mathbf{u}, \mathbf{u}_h \in \mathbb{R}^2, \quad (1)$$

where \mathbf{x} denotes the object position, $\dot{\mathbf{x}}$ its velocity, and \mathbf{u} and \mathbf{u}_h the robot's and human's interaction force; $\boldsymbol{\xi} = [\mathbf{x}^\top \dot{\mathbf{x}}^\top]^\top \in \mathbb{R}^4$ denotes the object's full state on the map such that $\mathbf{x} \in \mathcal{C}_{\text{free}}$, the accessible configuration space $\mathcal{C}_{\text{free}} = \mathcal{C} \setminus \mathcal{C}_{\text{obs}}$. The determined trajectory should adhere to kinodynamic constraints given by an environment map and constraints on the object's velocity and the acceleration.

A constraint-admissible trajectory can be calculated by concatenating a sequence of constraint-admissible trajectories. Therefore, a set of intermediate states between initial and final state have to be determined which are then connected by optimal and admissible trajectories. The method used to calculate optimal motion primitives and the resulting optimal trajectories, which connect two given states, will be introduced in Section III.

The optimal trajectory with N intermediate states and therefore $N+1$ sub-trajectories under the constraints $g(\boldsymbol{\xi}, \mathbf{u})$ on position, velocity and force is determined by

$$\min_{\boldsymbol{\xi}, \mathbf{u}, N} J_{\text{traj}}(\boldsymbol{\xi}, \mathbf{u}, N) \quad \text{s.t.} \quad g(\boldsymbol{\xi}, \mathbf{u}) \leq 0 \quad (2)$$

with cumulative cost J_{traj} of the entire trajectory

$$J_{\text{traj}}(\boldsymbol{\xi}, \mathbf{u}, N) = \sum_{k=0}^N J_{\text{subtraj},k}(\boldsymbol{\xi}, \mathbf{u}).$$

Therefore, the subtrajectories, the number of subtrajectories and the connection states between the subtrajectories are varied in the following to minimize the overall cost J_{traj}

A. Derivation of initial trajectory segments

In order to derive a set of initial valid trajectory segments, the feedback-planning scheme described in [9] is deployed. The method provides an estimate of the shortest direction to the goal for all points contained in the admissible set of

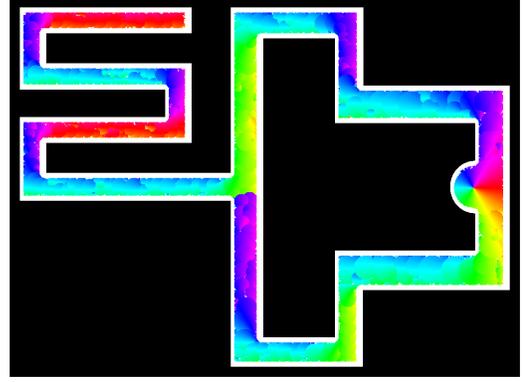


Fig. 1. Map of the environment with hue-encoded feedback plan

positions. Constructing a valid sequence of straight line segments from start to goal considering the computed direction as input provides an initial estimate for intermediate states. We propose the following algorithm:

- 1) Consider a point at the admissible state $\boldsymbol{\xi}_k$ at time k , the point moves at constant velocity $\dot{\mathbf{x}}_{\text{act}}$ during the time interval Δt in the direction determined using the feedback plan at position \mathbf{x}_k to determine the next position \mathbf{x}_{k+1} and next direction $\dot{\mathbf{x}}_{k+1}/\|\dot{\mathbf{x}}_{k+1}\|$.
- 2) The motion primitive between the current state $\boldsymbol{\xi}_k$ and the new state $\boldsymbol{\xi}_{k+1}$ is computed.

- If the resulting trajectory does not fulfill the environmental constraints, the computation is executed for half of the velocity $\dot{\mathbf{x}}_{\text{act}}$.
- If a feasible trajectory can be computed, but the difference between the angle between $\dot{\mathbf{x}}_k$ and $\dot{\mathbf{x}}_{k+1}$ is below a given $\Delta\phi_{\text{min}}$ which means that the two connected segments are approximately aligned, the time interval Δt is doubled and the computation is repeated.
- Otherwise, the new state $\boldsymbol{\xi}_{k+1}$ is added to the list of intermediate states and the computation is started with the maximum values of the constant speed and the time interval using this state as the current state.

- 3) The computation terminates as soon as a feasible trajectory from the current state to the final state exists. The distribution of the initial subtrajectories is influenced by the choice of the maximum constant velocity $\dot{\mathbf{x}}_{\text{max}}$, the maximum time interval Δt_{max} and the minimum angular difference $\Delta\phi_{\text{min}}$. Depending on these values, the cost of the initial path varies.

For the illustration of the approach we will use an example of a cooperative transport task in a 2D maze throughout the paper, which is also considered as scenario later in the experimental human user study. In Fig. 1 the maze is presented together with the feedback plan, where the direction towards the goal is hue-encoded. The influence of parameter variation is illustrated in Fig. 2.¹ However, depending on the choice of the parameters and the given map, the calculation can produce more than necessary intermediate states which

¹In the exemplary figures shown in this paper, the velocity and acceleration constraints are chosen to be $\dot{\mathbf{x}}_{\text{max}|\text{min}} = \pm 0.15$ m/s and $\ddot{\mathbf{x}}_{\text{max}|\text{min}} = \pm 0.12$ m/s².

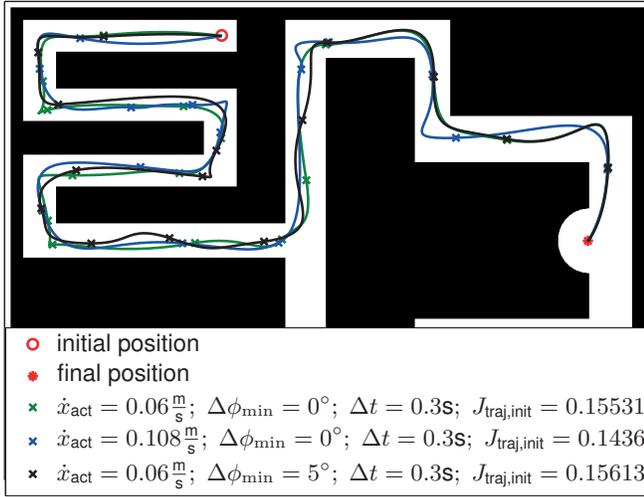


Fig. 2. Influence of parameter variation on the initial state distribution

results in higher cost than necessary and a higher effort for the local optimization. Therefore it is reasonable to reduce the number of subtrajectories. This can be done by consecutively replacing two connected subtrajectories by a single subtrajectory if the constraints remain satisfied. Note, that the cost of the replacing optimal subtrajectory is always lower than the sum of the cost of the replaced subtrajectories. It is apparent that the cost of the reduced path is significantly lower than the cost of the non-reduced path.

B. Optimization of segment sequencing

Since the initial set of subtrajectories is determined without the consideration of optimality principles, this set is optimized to reduce the cost of the resulting path. Therefore, the connecting states are altered. A simplex search method is used on the connecting states between the subtrajectories, which requires the computation of the numeric gradi-

$$\dot{x}_{max} = 0.06 \frac{m}{s}; \Delta\phi_{min} = 0^\circ; \Delta t_{min} = 0.5s$$

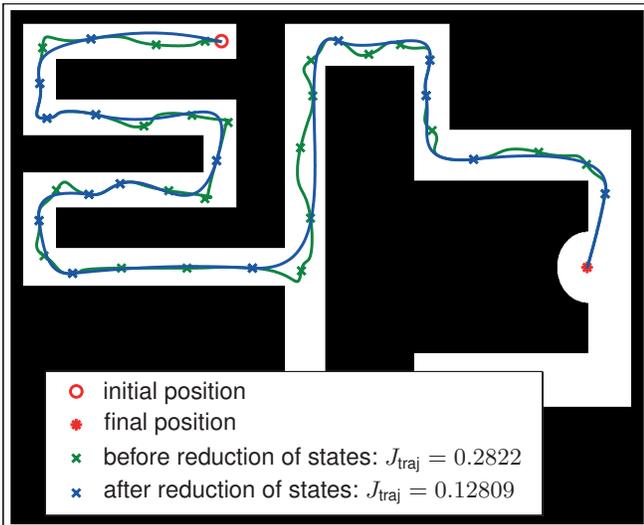


Fig. 3. Comparison of the path created by a reduced number of intermediate states with the non-reduced path

$$\dot{x}_{max} = 0.06 \frac{m}{s}; \Delta\phi_{min} = 0^\circ; \Delta t_{min} = 0.3s$$

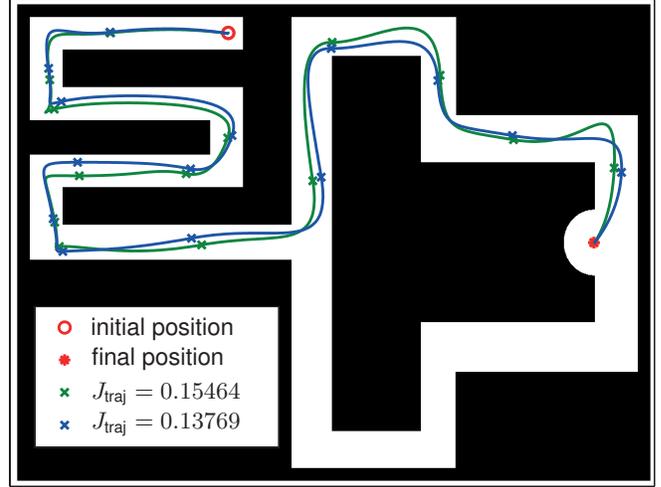


Fig. 4. Result of local optimization procedure

ent $\nabla_{\xi,k} J_{subtraj}$ in each connecting state ξ_k .

A reduction of the cost can be achieved by a local optimization of the connecting states, e.g. using a simplex search method. The cost function from (2) has to be slightly modified such that the cost is increased if a constraint is violated due to a change caused by the optimization process. This ensures that the states and the trajectories are not moved onto obstacles during optimization. This results in the new cost function

$$J_{traj} = \sum_{k=0}^N (1 + \gamma_k) \cdot J_{subtraj,k}, \quad (3)$$

where $\gamma_k = 0$ if all kinodynamic constraints are fulfilled and $\gamma_k \gg 1$ otherwise. The result of such a local optimization for our example is depicted in Fig. 4. It can be observed that the optimized path passes close to the obstacles. In the follow section, a method for the computation of optimal motion primitives used for sequencing is introduced.

III. GENERATION OF OPTIMAL MOTION PRIMITIVES

The previous section introduced a solution to the kinodynamic motion planning problem based on the combination of optimal trajectory segments, henceforth referred to as motion primitives. In this section we propose a method for derivation of such motion primitives in an energy optimal manner.

The motion primitive planning is based on decoupled one-dimensional optimal control. Therefore the planning process is performed separately for one-dimensional subproblems along each dimension of the overall planning problem. Considering relationship (1), two optimal-control problems remain: One along each axis of $x = [x_1 \ x_2]^T$ in the plane.

The key steps for computing the motion primitives are the calculation of the energy optimal control of the one-dimensional subproblems and the synchronization of these independently planned motions. For that reason we divide the planning process into four steps.

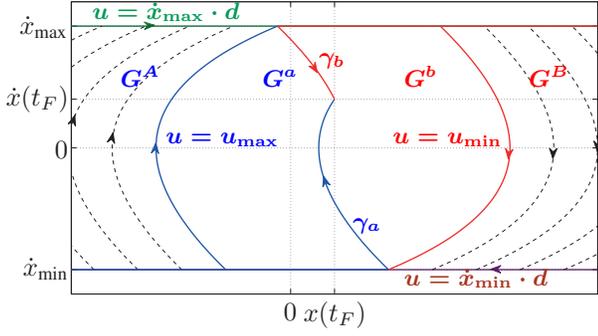


Fig. 5. Visualization of switching law in state space

- 1) Minimum-duration computation through solving time optimal control problem along each axis
- 2) Synchronization of 1-D motions via common final time t_{base}
- 3) Calculation of the energy optimal control
- 4) Calculation of the system trajectory

A. Time optimal control

The first step of our approach acts as preparation to the synchronization of the 1-D motions, which is based on the shortest possible time t_{opt} needed to reach the desired final values. To calculate t_{opt} we apply the minimum principle of Pontryagin to the one-dimensional subsystems of (1) considering

$$J(x_i, \dot{x}_i, u_i) = \int_0^{t_{\text{opt}}} 1 dt \quad i \in \{1, 2\} \quad (4)$$

as cost function and

$$u_{\min, i} \leq u_i \leq u_{\max, i} \quad i \in \{1, 2\} \quad (5)$$

as control constraints. This yields, due to the special structure of system (1), a switching law in state space variables of the system. For this control law an enhancement is possible such that velocity constraints

$$\dot{x}_{\min, i} \leq \dot{x}_i \leq \dot{x}_{\max, i} \quad i \in \{1, 2\} \quad (6)$$

can be considered. The enhanced switching law, expressed in state space variables, is given by:

- For $x(0) \in G^A$

$$u(t) = \begin{cases} u_{\max} & 0 < t < t_{\dot{x}, \max} \\ d \cdot \dot{x}_{\max} & t_{\dot{x}, \max} < t < t_{\text{switch}} \\ u_{\min} & t_{\text{switch}} < t < t_f \end{cases}$$

- For $x(0) \in G^B$

$$u(t) = \begin{cases} u_{\min} & 0 < t < t_{\dot{x}, \min} \\ d \cdot \dot{x}_{\min} & t_{\dot{x}, \min} < t < t_{\text{switch}} \\ u_{\max} & t_{\text{switch}} < t < t_f \end{cases}$$

- For $x(0) \in G^a$

$$u(t) = \begin{cases} u_{\max} & 0 < t < t_{\text{switch}} \\ u_{\min} & t_{\text{switch}} < t < t_f \end{cases}$$

- For $x(0) \in G^b$

$$u(t) = \begin{cases} u_{\min} & 0 < t < t_{\text{switch}} \\ u_{\max} & t_{\text{switch}} < t < t_f \end{cases}$$

With G^A , G^B , G^a and G^b subsets of the state space bounded by system trajectories as shown in Fig. 5, $t_{\dot{x}, \min}$ and $t_{\dot{x}, \max}$ as points in time when the velocity is equal to \dot{x}_{\min} and \dot{x}_{\max} , respectively; t_{switch} denotes the time when the trajectory reaches the switching curves γ_a or γ_b .

B. Synchronization of 1-D motions

Since the calculation of motion primitives incorporates only the one-dimensional case we need to ensure that both planned motions reach the desired final values at the same time t_{base} . Furthermore it is necessary to ensure that t_{base} is large enough such that motions are realizable considering acceleration and velocity constraints. In order to fulfill these requirements we define

$$t_{\text{base}} = k_{\text{slowdown}} \cdot \max(t_{\text{opt}, x}, t_{\text{opt}, y}). \quad (7)$$

where $t_{\text{opt}, x}$ and $t_{\text{opt}, y}$ denote the amount of time which is needed by time optimal control to reach the desired final state. The user-defined parameter $k_{\text{slowdown}} > 1$ determines the degree of slowdown relative to the time optimal case. Using this definition assures that the planned motions are feasible with velocity and acceleration within the constraints. Moreover it is possible to influence the completion time of the planned motion via k_{slowdown} .

C. Energy optimal control

In order to generate motion primitives, an energy-optimal control problem is solved. The cost function given by

$$J(\xi(t), \mathbf{u}(t)) = \int_0^{t_{\text{base}}} \|\xi\|_Q^2(t) + \|\mathbf{u}\|_R^2(t) dt \quad (8)$$

s.t. $g(\xi, \mathbf{u}) \leq 0$

where Q can be interpreted as mechanical friction

$$Q = \begin{pmatrix} 0_2 & 0_2 \\ 0_2 & q \cdot I_2 \end{pmatrix}$$

and R as electro-mechanic resistance $R = r \cdot I_2$ of a drive train. In our optimal control problem we consider the cost function (8) and furthermore (5) and (6) as constraints. Due to computational complexity of this optimal control problem and its non-convex nature, we approximate the solution. For this purpose we divide the planning problem into 1-D subproblems, sample the continuous dynamics of the subsystem of (1) and solve the resulting discrete approximations of the problems. These approximated optimal control problems can be formulated as quadratic programs such that efficient computation of reliable solutions is feasible.

The optimized sequence of the resulting optimized subtrajectories is forms the suboptimal overall solution \mathbf{u} applied to the targeted cooperative transport scenario.

IV. EXPERIMENTAL EVALUATION

The proposed optimization scheme is evaluated in a human-robot cooperative setup in virtual reality. Therefore, quantitative measures are used to rate the performance of the approach in comparison to a feedback planning scheme proposed in [10]. A virtual circular object with point-mass dynamics and isotropic viscous friction described by (1) is cooperatively moved to a virtual maze, visually and haptically rendered to the human user, see Fig. 6. The effective force on the virtual object is composed by the robot’s input \mathbf{u} and the human input \mathbf{u}_h .

The trajectory generation approach provides a reference trajectory $\mathbf{X} = [\mathbf{x}_d(\tau = 0) \dots \mathbf{x}_d(\tau = \tau_f)]$ and its first and second time derivatives. The robot’s feedforward force contribution \mathbf{u} is generated by

$$\mathbf{u} = m\ddot{\mathbf{x}}_d(\tau(\mathbf{x}_m)) + d\dot{\mathbf{x}}_d(\tau(\mathbf{x}_m))$$

where $\tau(\mathbf{x})$ denotes the reference-adaptation parameter, given by the closest $\mathbf{x}_d(\tau)$ to the current position \mathbf{x}_m

$$\mathbf{x}_m = \arg \min_{\mathbf{x}_d \in \mathbf{X}} \|\mathbf{x}_d - \mathbf{x}_m\|.$$

This accommodates the expected deviations from the computed trajectory due to the human interaction force \mathbf{u}_h .

A. Quantitative performance measures

We evaluate the following criteria in order to rate the performance of the proposed approaches:

- Mean disagreement \mathbf{u}_D :

$$\mathbf{u}_D = \begin{cases} \frac{-\mathbf{u}_h}{\|\mathbf{u}_h\|} \cdot \mathbf{u}_r, & \text{if } \mathbf{u}_h \cdot \mathbf{u}_r < 0 \wedge \mathbf{u}_h \neq \mathbf{0} \\ 0, & \text{otherwise.} \end{cases}$$

Larger values of disagreement \mathbf{u}_D indicate that the human and the robotic partner produce a higher amount of counteracting inefficient forces.

- Mean energy contributed by the human as measure of effort $E = \int_0^T \mathbf{u}_h^\top \dot{\mathbf{x}} dt$ indicates what share of the task effort is taken over by the robotic assistant.
- The mean completion time T is a measure for the speed-up gained through interaction. Completion time

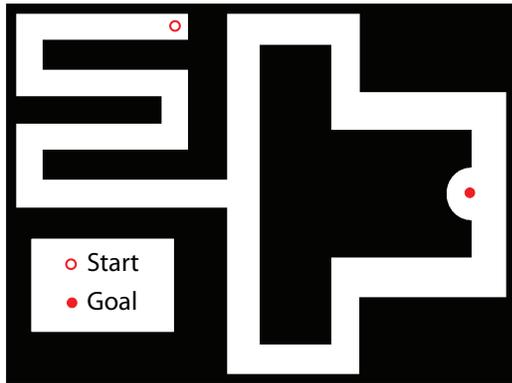


Fig. 6. Example task: Moving a point mass object from start to goal position through a maze

Constant	Equation	Value
Simulated object mass m	(1)	100 kg
Simulated viscous friction d	(1)	400 $\frac{\text{Ns}}{\text{m}}$
Maximum force $u_{\min} _{\max}$	(5)	± 12 N
Maximum velocity $\dot{x}_{\min} _{\max}$	(6)	± 0.15 m/s
Slowdown factor k_{slowdown}	(7)	2

TABLE I

CONTROL PARAMETERS USED IN 2-DOF EXPERIMENT

indicates the average velocity of the object, necessary for the interpretation of the energy contribution.

B. Experimental evaluation setup

The virtual-reality interface consists of a two degrees-of-freedom (anteroposterior and mediolateral plane of the user standing in front) linear-actuated device (*ThrustTube*) with a free-spinning handle (superoinferior direction of the user) at the grasp point. The control algorithm is implemented in *Matlab’s Simulink Coder* and executed on *Linux Preempt/RT* at a sampling rate of 1 kHz. Attached to the handle, a force sensor (*JR3*) measures the human interaction force. The virtual scene is visualized on a display placed on top of the interface. The displayed task to transport a virtual object is visually represented by a filled red circle and the target position in the upper left corner of the maze (blue dot). Collisions with the virtual walls should be avoided. Table I exhibits the constants used to parameterize the experiment.

We conducted a small pilot study in a VR scenario to evaluate the performance of our proposed approach. Six non-paid participants (age mean: 28.0, std: 2.36) were asked to move a virtual object through the maze. Each participant repeated the task without assistance for five times. Consecutively, five trials with kinematic feedback-planning-based assistance (see [10]) were performed. Finally, five trials with our proposed novel approach followed.

C. Quantitative performance results

The quantitative results of our pilot study are presented in Fig. 7. Figure 7(a) shows, that the mean energy induced to the point mass by the human subjects was significantly lower with optimized assistance considering the object dynamics than in the trials with assistance by the feedback-planning scheme. However, the participants chose a higher execution speed as depicted in Figure 7(b). The completion time in the feedback-planning assisted case decreased over trials. Figure 7(c) shows the occurrence of disagreement, hence counteracting, forces between the subject and the robotic assistant. Note, that the disagreement decreased significantly over the first trials in the optimized-trajectory assistance case, outperforming the feedback-planning assistance for all trials other than the first.

Given the target of human effort minimization, our novel proposed approach greatly outperforms our previous method for goal-directed assistance. Our hypothesis is, that the subjects decided to maintain a slower execution time in the optimized-trajectory case due to the fact that the optimized trajectory runs as close as possible to walls and corners.

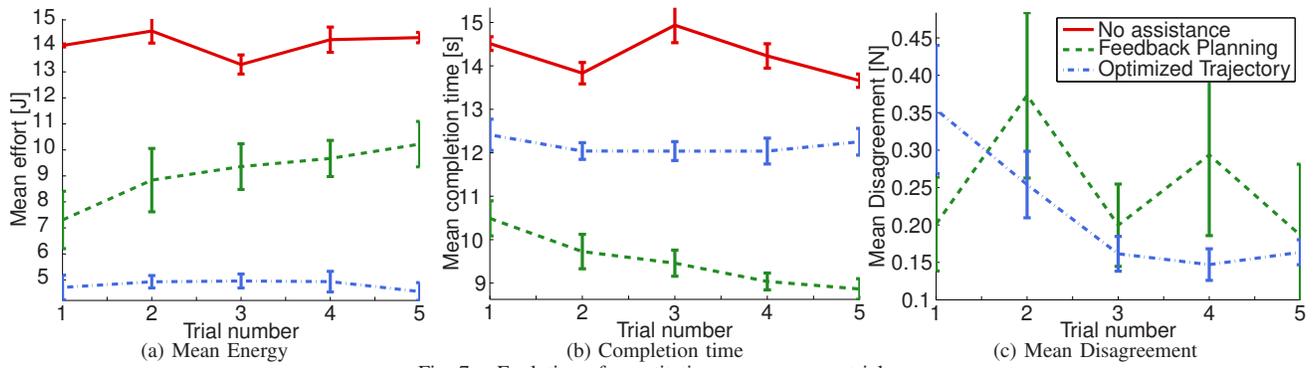


Fig. 7. Evolution of quantitative measures over trials

The human subject avoiding collisions is therefore more cautious, hence accepts higher execution times. Furthermore, the subjects gained practise over trials, visible in the feedback-planning assisted case. The execution velocity is increased from trial to trial, an indication of confidence. A learning curve of the subjects is observed in the disagreement forces over trials. Counteracting forces decrease significantly over the first three trials as the human participants gain trust in the optimized trajectory of the robotic assistant.

D. Computational tractability

In our prototypical implementation, the proposed optimization procedure requires the following computation times: The generation of the feedback plan for the given map example requires an approximate 100s to compute. The initial solution for a trajectory in the given example with non-optimized *Matlab* code requires 25.7s on an *Intel Core i7* at 2.3GHz on OS X. The optimization of the primitive sequencing (35 initial segments, 14 segments after state reduction) requires an approximate 2,700 iterations of which each takes an average of 2s, resulting in an overall computation time of 1.5h. Note, that our proposed approach is suitable for a parallel implementation for significantly increased computation speeds.

V. CONCLUSION

In this work, we propose a novel approach for trajectory generation under the least-action principle for active robotic assistance to humans. The three-step optimization scheme generates an iteratively optimized sequence of energy-optimal motion primitives also derived in this paper. The overall scheme obeys not only environmental constraints on the object position but also on its velocity and acceleration. It greatly outperforms our previously proposed feedback-planning approach as it considers inertia and friction effects in trajectory generation. Results show that the human effort is significantly reduced by the novel scheme. However, our findings also show that human caution/trust influences the result which is subject to future research.

ACKNOWLEDGEMENTS

This work is supported in part within the DFG excellence initiative research cluster Cognition for Technical Systems - CoTeSys (www.cotesys.org). The authors would like to acknowledge valuable advice by José Ramón Medina Hernández regarding data analysis.

REFERENCES

- [1] K. Kosuge and Y. Hirata, "Human-Robot Interaction," in *Proc. IEEE ROBOTICS*, 2004, pp. 8–11.
- [2] M. Lawitzky, A. Mörtl, and S. Hirche, "Load Sharing in Human-Robot Cooperative Manipulation," in *Proc. IEEE Ro-Man*, 2010, pp. 185–191.
- [3] J. Medina, M. Lawitzky, A. Mörtl, D. Lee, and S. Hirche, "An Experience-Driven Robotic Assistant Acquiring Human Knowledge to Improve Haptic Cooperation," in *Proc. IEEE/RSJ IROS*, 2011, pp. 2416–2422.
- [4] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans," in *Proc. IEEE ICRA*, 2011.
- [5] A. Thobbi, Y. Gu, and W. Sheng, "Using Human Motion Estimation for Human-Robot Cooperative Manipulation," in *Proc. IEEE/RSJ IROS*, 2011, pp. 2873–2878.
- [6] A. Kirsch, T. Kruse, E. Sisbot, R. Alami, M. Lawitzky, D. Brščić, S. Hirche, P. Basili, and S. Glasauer, "Plan-Based Control of Joint Human-Robot Activities," *Künstl. Intell.*, vol. 24, no. 3, pp. 223–231, 2010.
- [7] M. Taix and D. Flavigné, "Motion Planning with interactive devices," in *Proc. IEEE ECMS*, 2011, pp. 1–6.
- [8] C. Vázquez, J. Rosell, L. Chirinos, and O. Domínguez, "Haptic Primitives Guidance Based on the Kautham Path Planner," in *Proc. IEEE/RSJ IROS*, 2010, pp. 4686–4691.
- [9] M. Lawitzky, J. Medina Hernandez, and S. Hirche, "Rapid prototyping of planning, learning and control in physical human-robot interaction," in *13th International Symposium on Experimental Robotics*, June 2012.
- [10] M. Lawitzky, J. Medina, D. Lee, and S. Hirche, "Feedback Motion Planning and Learning from Demonstration in Physical Robotic Assistance: Differences and Synergies," in *Proc. IEEE/RSJ IROS*, 2012.
- [11] A. Mörtl, M. Lawitzky, A. Küçükylmaz, M. Sezgin, C. Basdogan, and S. Hirche, "The Role of Roles: Physical Cooperation between Humans and Robots," *Int. J. Robot. Res.*, 2012.
- [12] D. Feth, R. Groten, A. Peer, S. Hirche, and M. Buss, "Performance Related Energy Exchange in Haptic Human-Human Interaction in a Shared Virtual Object Manipulation Task," in *Proc. EHS EuroHaptics'09*, 2009, pp. 338–343.
- [13] T. Flash and N. Hogan, "The coordination of arm movements: An experimentally confirmed mathematical model," *J. Neurosci.*, vol. 5, pp. 1688–1703, 1985.
- [14] Y. Maeda, T. Hara, and T. Arai, "Human-robot cooperative manipulation with motion estimation," in *Proc. IEEE/RSJ IROS*, vol. 4, 2001, pp. 2240–2245 vol.4.
- [15] S. Miossec and A. Kheddar, "Human motion in cooperative tasks: Moving object case study," in *Proc. IEEE ROBOTICS*, 2008, pp. 1509–1514.
- [16] S. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.
- [17] F. Nori and R. Frezza, "A Control Theory Approach to the Analysis and Synthesis of the Experimentally Observed Motion Primitives," in *Biological Cybernetics*, vol. 93, no. 5, 2005, pp. 323–342.
- [18] M. Steinegger, B. Passenberg, M. Leibold, and M. Buss, "Trajectory Planning for Manipulators based on the Optimal Concatenation of LQ Control Primitives," in *Proc. IEEE CDC-ECC*, 2011, pp. 2837–2842.
- [19] K. Flaßkamp, S. Ober-Blöbaum, and M. Kobilarov, "Solving Optimal Control Problems by Exploiting Inherent Dynamical Systems Structures," *J Nonlinear Sci*, vol. 22, pp. 599–629, 2012.