

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/361967066>

# An MPC Framework For Planning Safe & Trustworthy Robot Motions

Conference Paper · May 2022

DOI: 10.1109/ICRA46639.2022.9812160

---

CITATIONS

6

---

READS

48

5 authors, including:



Moritz Eckhoff

Technische Universität München

3 PUBLICATIONS 16 CITATIONS

SEE PROFILE



Robin Jeanne Kirschner

Technische Universität München

21 PUBLICATIONS 109 CITATIONS

SEE PROFILE

# An MPC Framework For Planning Safe & Trustworthy Robot Motions

Moritz Eckhoff, Robin Jeanne Kirschner, Elena Kern, Saeed Abdolshah and Sami Haddadin

**Abstract**—Strategies for safe human-robot interaction (HRI), such as the well-established Safe Motion Unit, provide a velocity scaling for biomechanically safe robot motion. In addition, psychologically-based safety approaches are required for trustworthy HRI. Such schemes can be very conservative and robot motion complying with such safety approaches should be time efficient within the robot motion planning. In this study, we improve the efficiency of a previously introduced approach for psychologically-based safety in HRI via a Model Predictive Control robot motion planner that simultaneously adjusts Cartesian path and speed to minimise the distance to the target pose as fast as possible. A subordinate real-time motion generator ensures human physical safety by integrating the Safe Motion Unit. Our motion planner is validated by two experiments. The simultaneous adjustment of path and velocity accomplishes highly time efficient robot motion, while considering the human physical and psychological safety. Compared to direct path velocity scaling approaches our planner enables 28% faster motion execution.

## I. INTRODUCTION

Safe human-robot interaction (HRI) requires physical integrity of the human and the state of the art offers several approaches to human physical safety [1]–[10]. Nevertheless, psychological safety limitations should also be considered in the context of safe and trustworthy HRI [11]. Generally, such psychological or physical safety constraints can be integrated into motion planning or control [11]. Recently, a control scheme for human physical and psychological safety considerations was introduced that uses the robot real-time control for velocity scaling [12]. While such control schemes only react in case of a motion conflict, e.g. by reducing the robot speed [1], safe online motion planning allows to proactively avoid such conflicts and may improve the human-robot team efficiency [11]. Model Predictive Control (MPC) is one method which enables proactive path planning. Based on a constrained optimisation problem, MPC predicts and optimises future states using the current state and a given system model [13]. Consequently, online MPC motion planning allows avoiding safety conflicts of the robot motion such as human-robot collisions [8], [9], [14]. The main challenges for integrating MPC planning to safe robot motion planning is the high degree of non-linearity which is required for most safety functions, as well as the computational load and ensuring reachable motions [8], [9], [15]–[18].

In our previous work, a distance-velocity based psychological safety law was introduced that shall reduce the number of triggered involuntary motions of the human [12]. Based on this approach, in our current paper we present a robot control loop featuring an MPC-based motion planner, minimising

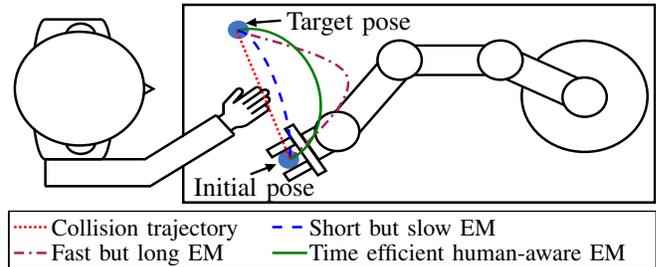


Fig. 1. Illustration of the planning problem. The objective of our proposed motion planner is to ensure compliance to the safety restrictions and avoid unnecessarily long (purple) or slow (blue) evasive motions (EMs). Instead it shall find and execute the green highly time efficient EM.

the distance to the goal pose with simultaneous adjustments of the Cartesian robot path and velocity profile. The combination of the distance-velocity correlating constraint and the simultaneous path-velocity adaption in an MPC optimisation problem leads to highly time efficient evasive motions (EMs) as depicted by Fig. 1. Using a physical safety motion generator subordinate to the planner, we guarantee human physical and psychological integrity.

The paper is structured as follows. Sec. II provides the state of the art in robotic safety and MPC motion planning. Subsequently, Sec. III addresses our approach in detail, including the safety concept and the control loop with our motion planner. Sec. IV describes and discusses the two validation experiments. Finally, Sec. V gives a conclusion to this work.

## II. STATE OF THE ART

For ensuring human physical safety in HRI various strategies were introduced, based on velocity scaling [1]–[7] or separation monitoring [1], [8]–[10]. As the strategies often maintain a predefined path which can become outdated and slow in dynamic environments [4], [6], [7] or consider exclusion zones which the robot may not enter [8], [19], they result in a reduction of the robot motion efficiency. A continuous and thus online (re)calculation of Cartesian path and velocity within an optimisation problem can resolve this reduction while safety is ensured by constraints. For constraint optimisation problems MPC is used e.g. in autonomous driving [20], [21] and mobile robot motion planning [14], [22]. Furthermore, for robot manipulators, it is often applied to so-called trajectory planners which require a fixed and predefined path to calculate a velocity profile [6], [7] or to task-specific control [23]. Full and general MPC motion planners adjusting the path and velocity simultaneously online, however, are less common in the literature due to the immense computational cost caused by the high number of degrees of freedom (DoFs), nonlinearities, and strict real-time constraints [19]. To the best of the authors’ knowledge such full MPC motion planners

All authors are with Chair for Robotics and Systems Intelligence, Munich Institute of Robotics and Machine Intelligence, Technical University of Munich, 80797 Munich, Germany moritz.eckhoff@tum.de

TABLE I  
CONTRIBUTION COMPARED TO OTHER FULL MPC MOTION PLANNERS

Feature	[8]	[15]	[16]	[17]	[18]	[9], [19]	Our approach
Orientations are planned as well		✓	✓			✓	✓
Considers safety constraints in motion planning	✓		(✓)		(✓)	(✓)	✓
Smooth Cartesian EMs	✓						✓
Time efficiency optimisation			✓	✓			✓
Technical feasibility shown in demonstration	✓	✓		✓	✓	✓	✓

can only be found in [8], [9], [15]–[19]. While no safety aspects are considered in [15] and [17], [8] and [18] define safety exclusion zones where the robot may not intrude. The planners from [16] and [9], [19] feature a more dynamic obstacle avoidance based on non-linear distance constraints, which can be considered as a safety feature to some extent. In [16], these non-linear constraints are directly implemented to a non-linear MPC algorithm, whereas Avanzini et al. [9], [19] suggest an efficient linearisation strategy for them to enable the use of linear MPC. In this way, [9], [19] create a technically feasible system able to comply with the real-time constraints.

Besides the human physical safety, safe and trustworthy HRI requires psychological well-being [4], [24]. This holds particularly true for the emerging field of social robots. Previous research found that the robot velocity [25] and the distance between human and robot [26] influence the human perception of safety. Based on this, a safety approach named Expectable Motion Unit (EMU) was introduced to implement human expectation understanding to robot motion design [12]. Such a safety approach allows a robot path-velocity optimisation.

We use the psychological safety constraints from our earlier work and embed them into an online full MPC motion planner for robot manipulators. Table I summarises the contribution of our approach compared to the mentioned ones. The implementation and features of our approach are discussed in the next section.

### III. MPC APPROACH FOR SAFE AND TRUSTWORTHY HRI

In the following, we propose an online MPC motion planner for time efficient, safe, and human-aware robot motions. At each time step, the planner simultaneously recalculates the Cartesian robot path and speed while maintaining the desired safety constraints. The MPC motion planner results in human-aware robot EMs that reduce the robot motion time. To this end, we embed two safety approaches in a hierarchical structure:

- a) the psychologically-grounded distance-velocity mapping called EMU introduced in [12] which we embed in the MPC planner and
- b) the well-established real-time capable Safe Motion Unit (SMU) to ensure human physical safety.

By combining the velocity-distance mapping with the simultaneous Cartesian path-velocity recalculation in the optimisation problem we gain safe planning, smooth Cartesian EMs, and improve the robot motion time efficiency.

We use a linear MPC design to reduce the computational cost and thus linearise all non-linear constraints online at

each time step. However, the MPC planner is still computationally expensive and needs to run at lower frequency than real time. Since the EMU concept requires Cartesian position information, we apply a Cartesian motion planner. Joint space motion planners on the contrary require a non-linear coordinate transformation which conflicts with the Cartesian safety law and causes the optimiser to fail in deriving smooth Cartesian EMs.

To implement the real-time capable physical safety interface, we cascade the control loop structure with an outer and inner loop, as depicted by Fig. 2. At the end of each non-real-time motion planning step both loops are synchronised, enabling real-time motion generation. The inputs to the outer *human-aware Cartesian MPC motion planner* are the Cartesian target pose  $\mathbf{x}_{\text{tar}} \in \mathbb{R}^6$  and the human position matrix  $\mathbf{P}_h = [\mathbf{p}_{h,0|k}, \dots, \mathbf{p}_{h,N_p|k}] \in \mathbb{R}^{3 \times N_p + 1}$  containing current measured values  $\mathbf{p}_{h,0|k}$  and predicted values  $\mathbf{p}_{h,i|k}$  with  $i \in \mathbb{N}$  between 1 and  $N_p$  which is the length of the MPC prediction horizon. The current robot pose and velocity  $\mathbf{x}, \dot{\mathbf{x}} \in \mathbb{R}^6$  of the end effector are fed back to the motion planner which determines the desired pose and velocity  $\mathbf{x}_d, \dot{\mathbf{x}}_d \in \mathbb{R}^6$ . Subsequently, the *SMU motion generator* in the inner loop checks for physical safety and determines the desired safe velocities  $\dot{\mathbf{x}}_{d,\text{safe}}$ . The design of the motion planner and that of the motion generator are described in the following.

#### A. Human-Aware Cartesian Motion Planner

Our motion planner shall fulfill reachability constraints discussed later and the EMU-function

$$\|\dot{\mathbf{p}}\| \leq m\|\mathbf{p} - \mathbf{p}_h\| + n, \quad (1)$$

where  $m$  and  $n$  denote, based on [12], the adjustable slope and y-intercept of the safety function while  $\mathbf{p} \in \mathbb{R}^3$  represents the Cartesian position of the robot end effector,  $\dot{\mathbf{p}} \in \mathbb{R}^3$  its velocity, and  $\mathbf{p}_h \in \mathbb{R}^3$  is the position of the human wrist as observed in [12]. Since  $\mathbf{p}$ ,  $\dot{\mathbf{p}}$ , and  $\mathbf{p}_h$  vary over the MPC prediction horizon, the EMU constraint must be formulated flexibly. Therefore, we use the MPC index notation  $i|k$  which corresponds to the  $i^{\text{th}}$  predicted variable in the current time step  $k$ . The EMU constraint for the Cartesian space motion planner can now be written as

$$\|\dot{\mathbf{p}}_{i|k}\| \leq m\|\mathbf{p}_{i|k} - \mathbf{p}_{h,i|k}\| + n \quad \forall i = 1 \dots N_p. \quad (2)$$

The MPC discretisation step size  $\Delta t_{\text{MPC}}$  needs to be larger than  $\Delta t_{\text{real-time}} = 1 \text{ ms}$  of the SMU motion generator to cope with the computational cost of the optimisation problem and thus create a technically feasible system. In order to obtain a sufficiently long prediction horizon without generating a large amount of prediction steps  $N_p$ , we choose  $\Delta t_{\text{MPC}} =$

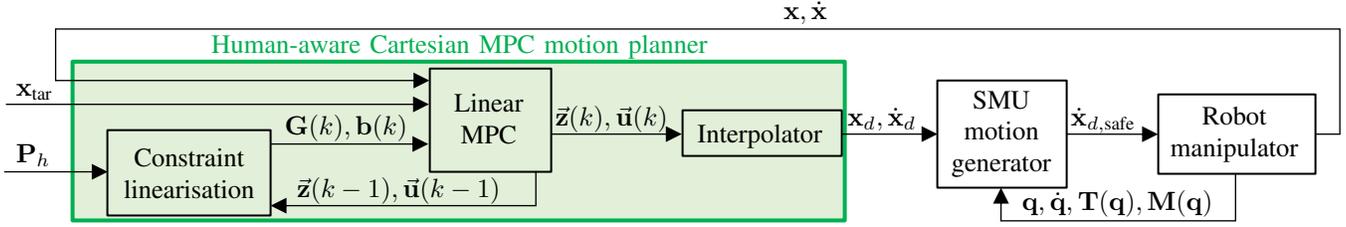


Fig. 2. Our cascaded control loop with the human-aware Cartesian MPC motion planner in the outer and the SMU motion generator in the inner one.

25 ms. The different step sizes of inner and outer loop, however, require a synchronisation and the nonlinear constraints require a linearisation. Thus, the resulting structure of the *human-aware Cartesian motion planner* is separated into three parts shown in the green box of Fig. 2.

At each time step  $k \in \mathbb{N}$ , the *constraint linearisation* block ensures linearity for all constraints based on the matrix  $\mathbf{P}_h$  and the latest MPC solution which consists of the optimal state series  $\bar{\mathbf{z}}(k-1)$  and the optimal input series  $\bar{\mathbf{u}}(k-1)$ , such that the constraints can be formulated as

$$\mathbf{G}(k)\bar{\mathbf{u}}(k) \leq \mathbf{b}(k). \quad (3)$$

Here,  $\mathbf{G}(k) \in \mathbb{R}^{N_c N_p \times N_p N_u}$  is the constraint matrix which needs to be determined and  $\mathbf{b}(k) \in \mathbb{R}^{N_c N_p}$  is the corresponding constraint vector.  $\mathbf{G}(k)$  and  $\mathbf{b}(k)$  are fed to the *linear MPC* block which determines  $\bar{\mathbf{z}}(k) = [\mathbf{z}_{1|k}^T, \dots, \mathbf{z}_{N_p|k}^T]^T \in \mathbb{R}^{N_z N_p}$  and  $\bar{\mathbf{u}}(k) = [\mathbf{u}_{0|k}^T, \dots, \mathbf{u}_{N_p-1|k}^T]^T \in \mathbb{R}^{N_u N_p}$ . The used variables  $N_c$ ,  $N_z$ , and  $N_u$  correspond to the number of constraints, the number of states, and the number of inputs. From the MPC solution continuous trajectories are generated and subsequently resampled with real-time frequency by the *interpolator* to obtain  $\mathbf{x}_d$  and  $\dot{\mathbf{x}}_d$ . A detailed description of the motion planner elements depicted by Fig. 2 is provided in the following sections.

1) *Linear MPC*: For the MPC optimisation problem, we first define the state vector  $\mathbf{z}_{i|k} \in \mathbb{R}^{12}$  and the input vector  $\mathbf{u}_{i|k} \in \mathbb{R}^6$  based on the Cartesian pose of the robot end effector  $\mathbf{x}_{i|k} \in \mathbb{R}^6$  as

$$\mathbf{z}_{i|k} = \begin{bmatrix} \mathbf{x}_{i|k} \\ \dot{\mathbf{x}}_{i|k} \end{bmatrix}, \quad \mathbf{u}_{i|k} = \ddot{\mathbf{x}}_{i|k} \quad \forall i, k \in \mathbb{N}_0. \quad (4)$$

Then, we describe the system model as a linear kinematics

$$\mathbf{z}_{i+1|k} = \begin{bmatrix} \mathbf{I}_6 & \mathbf{I}_6 \Delta t_{\text{MPC}} \\ \mathbf{0}_6 & \mathbf{I}_6 \end{bmatrix} \mathbf{z}_{i|k} + \begin{bmatrix} \frac{1}{2} \mathbf{I}_6 \Delta t_{\text{MPC}}^2 \\ \mathbf{I}_6 \Delta t_{\text{MPC}} \end{bmatrix} \mathbf{u}_{i|k}, \quad (5)$$

where  $\mathbf{I}_6 \in \mathbb{R}^{6 \times 6}$  is the identity matrix, and  $\mathbf{0}_6 \in \mathbb{R}^{6 \times 6}$  is the zero matrix.

The cost function  $J$  of the motion planner is defined as

$$J(\mathbf{z}_{0|k}, \bar{\mathbf{u}}(k)) = \frac{1}{2} \|\bar{\mathbf{z}}(k) - \bar{\mathbf{z}}_{\text{ref}}\|_{\mathbf{Q}_s}^2 + \frac{1}{2} \|\bar{\mathbf{u}}(k)\|_{\mathbf{R}_s}^2 + \frac{1}{2} \|\bar{\mathbf{u}}(k) - \bar{\mathbf{u}}_{\ll N_u}(k-1)\|_{\mathbf{S}_s}^2, \quad (6)$$

with the weighting matrices  $\mathbf{Q}_s$ ,  $\mathbf{R}_s$ , and  $\mathbf{S}_s$ . Here,  $\mathbf{Q}_s$  and  $\mathbf{R}_s$  contain cost terms penalising the state error and the input, while  $\mathbf{S}_s$  penalises the difference between the the current optimal series and the one from the latest time step. It aims at reducing oscillations caused by a possible cost minimum that exploits the linearisation of the constraints by choosing strongly divergent consecutive input variables. To enable this operation, the input series of the latest time step is shifted

by the size of  $N_u$  such that e.g.  $\mathbf{u}_{1|k}$  meets  $\mathbf{u}_{2|k-1}$ . This is indicated by  $\ll$ . By repeatedly inserting the system model (5) into the state  $\mathbf{z}$ , one obtains the cost function which depends only on the known current state  $\mathbf{z}_{0|k}$  and the input series  $\bar{\mathbf{u}}(k)$  to be determined.

To ensure reachable robot motion, reachability constraints are formulated as

$$\begin{aligned} p_{lb} &\leq \|\mathbf{p}_{i|k}\| &&\leq p_{ub}, \\ \varphi_{lb} &\leq \|\boldsymbol{\varphi}_{i|k}\| &&\leq \varphi_{ub}, \\ \dot{p}_{lb} &\leq \|\dot{\mathbf{p}}_{i|k}\| &&\leq \dot{p}_{ub}, \\ \dot{\varphi}_{lb} &\leq \|\dot{\boldsymbol{\varphi}}_{i|k}\| &&\leq \dot{\varphi}_{ub}, \\ \ddot{p}_{lb} &\leq \|\ddot{\mathbf{p}}_{i|k}\| &&\leq \ddot{p}_{ub}, \\ \ddot{\varphi}_{lb} &\leq \|\ddot{\boldsymbol{\varphi}}_{i|k}\| &&\leq \ddot{\varphi}_{ub}, \\ \ddot{p}_{lb} &\leq \frac{\|\ddot{\mathbf{p}}_{i|k}\| - \|\ddot{\mathbf{p}}_{i-1|k}\|}{\Delta t_{\text{MPC}}} &&\leq \ddot{p}_{ub}, \\ \ddot{\varphi}_{lb} &\leq \frac{\|\ddot{\boldsymbol{\varphi}}_{i|k}\| - \|\ddot{\boldsymbol{\varphi}}_{i-1|k}\|}{\Delta t_{\text{MPC}}} &&\leq \ddot{\varphi}_{ub}. \end{aligned} \quad \forall i = 1 \dots N_p, \quad (7)$$

The next section describes how the constraints (2) and (7) are linearised for our linear MPC.

2) *Constraint Linearisation*: The linearisation assumptions of the motion planner are based on those used in [9], [19] and defined as

$$\begin{aligned} \begin{bmatrix} \mathbf{z}_{0|k-1} \\ \bar{\mathbf{z}}(k-1) \end{bmatrix}_{\ll N_z} &= \begin{bmatrix} \mathbf{z}_{1|k-1} \\ \mathbf{z}_{2|k-1} \\ \vdots \\ \mathbf{z}_{N_p+1|k-1} \end{bmatrix} \approx \begin{bmatrix} \mathbf{z}_{0|k} \\ \mathbf{z}_{1|k} \\ \vdots \\ \mathbf{z}_{N_p|k} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_{0|k} \\ \bar{\mathbf{z}}(k) \end{bmatrix}, \quad (8) \\ \bar{\mathbf{u}}_{\ll N_u}(k-1) &= \begin{bmatrix} \mathbf{u}_{1|k-1} \\ \mathbf{u}_{2|k-1} \\ \vdots \\ \mathbf{u}_{N_c|k-1} \end{bmatrix} \approx \begin{bmatrix} \mathbf{u}_{0|k} \\ \mathbf{u}_{1|k} \\ \vdots \\ \mathbf{u}_{N_c-1|k} \end{bmatrix} = \bar{\mathbf{u}}(k). \quad (9) \end{aligned}$$

Accordingly, we assume that a state or input predicted for a certain point in time will not change in the next time step. By shifting all entries of the state and input series by  $N_z$  and  $N_u$  respectively, a vector  $\mathbf{z}_{N_p+1|k-1}$  is created at the end of the state series and a vector  $\mathbf{u}_{N_p|k-1}$  at the end of the input series for which no values have yet been predicted. To resolve this problem, we assume no change between  $\mathbf{u}_{N_p|k-1}$  and the previous known input  $\mathbf{u}_{N_p-1|k-1}$ . Subsequently, we determine  $\mathbf{x}_{N_p+1|k-1}$  based on the assumed values for  $\mathbf{u}_{N_p|k-1}$ , the known state  $\mathbf{x}_{N_p|k-1}$  and the system model (5). The assumptions (8) for  $\bar{\mathbf{z}}(k)$  and (9) for  $\bar{\mathbf{u}}(k)$  shall be inserted into the non-linearities of the constraints as often as necessary until they become linear. To do so, we change the notation of the norm function to a product formulation as

$$\|\mathbf{a}\| = \sqrt{a_1^2 + a_2^2 + a_3^2} = \mathbf{a}_o^T \mathbf{a}, \quad (10)$$

with the index  $o$  indicating the normalisation of the vector to the length of one. We now insert (8), (9) and (10) into the constraints (2) and (7) to obtain their linearisation. As an example, the linearised absolute translational velocity  $\|\dot{\mathbf{p}}_{i|k}\|$  results in

$$\|\dot{\mathbf{p}}_{i|k}\| \approx \dot{\mathbf{p}}_{o,i+1|k-1}^T \dot{\mathbf{p}}_{i|k}, \quad (11)$$

where the first part of the product  $\dot{\mathbf{p}}_{o,i+1|k-1}^T$  is constant and known from the previous solution of the MPC algorithm. The second part  $\dot{\mathbf{p}}_{i|k}$  is the yet unknown and to be optimised end effector velocity. The approximate absolute end effector velocities, thus, depend linearly on the vectors  $\dot{\mathbf{p}}_{i|k}$  which are part of  $\bar{\mathbf{z}}(k)$ . To obtain linear constraints depending on  $\bar{\mathbf{u}}(k)$  only, the system model (5) is substituted into the constraints as e.g.

$$\begin{aligned} \dot{p}_{lb} &\leq \dot{\mathbf{p}}_{o,i+1|k-1}^T \dot{\mathbf{p}}_{i|k}, \\ \dot{p}_{lb} &\leq \dot{\mathbf{p}}_{o,i+1|k-1}^T \mathbf{E}_{\dot{\mathbf{p}},i} (\mathbf{A}_s \mathbf{z}_{1|k-1} + \mathbf{B}_s \bar{\mathbf{u}}(k)), \quad \forall i = 1 \dots N_p \end{aligned} \quad (12)$$

where  $\mathbf{A}_s$  and  $\mathbf{B}_s$  are stacked versions of  $\mathbf{A}$  and  $\mathbf{B}$  satisfying  $\mathbf{A}_s \mathbf{x}_{0|k} + \mathbf{B}_s \bar{\mathbf{u}}(k) = \bar{\mathbf{x}}(k)$  and  $\mathbf{E}_{\dot{\mathbf{p}},i}$  is an extraction matrix fulfilling  $\dot{\mathbf{p}}_{i|k} = \mathbf{E}_{\dot{\mathbf{p}},i} \bar{\mathbf{x}}(k)$ . A final rearrangement of (12) and a subsequent comparison of coefficients provides the entries of  $\mathbf{G}(k)$  and  $\mathbf{b}(k)$  for the  $N_p$  columns constraining the lower velocity bound. The described procedure needs to be repeated for all  $N_c = 17$  constraints. Then, the resulting columns are stacked to obtain the full constraint matrix  $\mathbf{G}(k)$  and the constraint vector  $\mathbf{b}(k)$ . Note that the number of constraints of the optimisation problem is  $N_p$  times higher than the number of constraints of the MPC algorithm  $N_c$ . Since the constant terms as  $\dot{\mathbf{p}}_{o,i+1|k-1}^T$  change in each time step, the linearisation has to be performed online at each iteration.

3) *Interpolation*: The different frequencies of the motion planner and the SMU motion generator require a synchronisation which our control loop performs by interpolation of the discrete MPC solution  $\bar{\mathbf{u}}(k)$ ,  $\bar{\mathbf{z}}(k)$  and a subsequent resampling with the frequency of the SMU motion generator. To ensure reachability of the continuous trajectory, we use two times continuously differentiable cubic splines for each of the six dimensions of the end effector pose. The discrete MPC solutions serve as grid points. After each MPC iteration, new splines are calculated that update the old ones. To ensure a smooth transition between the old and the new splines, the current measured end effector pose  $\mathbf{x}$  and velocity  $\dot{\mathbf{x}}$  are defined as constraints for the first spline. Since the splines are updated in each MPC iteration it is sufficient to only calculate two consecutive splines for each dimension. Thus, only the first  $N_i = 3$  states and inputs of the MPC solution are used as grid points and fewer splines are required.

### B. SMU Motion Generator

We ensure physical safety with a real-time motion generator that implements the SMU introduced in [2] and, thus, checks whether the desired motion  $\mathbf{x}_d$ ,  $\dot{\mathbf{x}}_d$  is safe. In case the velocity is unsafe, the motion generator scales down the velocity to obtain  $\dot{\mathbf{x}}_{d,\text{safe}}$ . Fig. 3 gives a visual summary to our safety concept, in which the EMU function considered in motion planning is illustrated by blue circles and the physical safety considered in the SMU motion generator during HRI

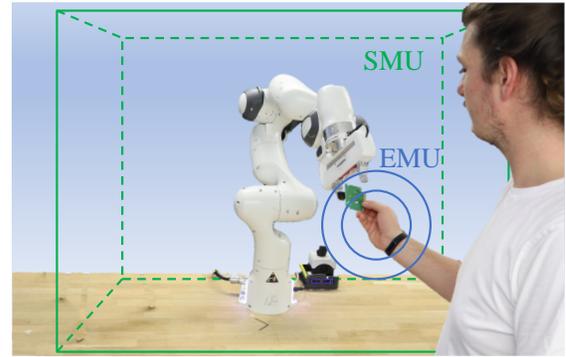


Fig. 3. Visualisation of the safety concept. When the human enters the shared workspace, the SMU checks the planned trajectory for safety. The blue circles represent the contour lines from the EMU considered in motion planning.

TABLE II  
PARAMETERS USED IN THE EXPERIMENTS

Parameter	Symbol	Value
Number of states	$N_z$	12
Number of inputs	$N_u$	6
Number of constraints	$N_c$	17
Discretisation step size	$\Delta t_{\text{MPC}}$	0.025 s
Length of the prediction horizon	$N_p$	18
Slope of the safety inequality	$m$	$0.8 \frac{1}{\text{s}}$
y-intercept of the safety inequality	$n$	$0.01 \frac{\text{m}}{\text{s}}$
State cost terms	$[c_x \quad c_{\dot{x}}]$	$[100 \quad 0]$
Input cost	$c_{\ddot{x}}$	0
Oscillation cost	$c_{\text{osc}}$	15

tasks is highlighted by the green box representing the shared workspace.

The control scheme is embedded in C++ to the Franka Emika (FE) Panda robot using qpOASES [27] and experimentally evaluated in the following section.

## IV. EXPERIMENTAL EVALUATION

This section describes the validation experiments for the proposed MPC motion planner and SMU motion generator.

### A. Experimental Procedure

We use a FE Panda robot and conduct two experiments using the experimental setup depicted by Fig. 3. The human hand position information for the EMU concept and the SMU safety control law inside the defined human-robot workspace are applied. The weighting matrices  $\mathbf{Q}_s$ ,  $\mathbf{R}_s$  and  $\mathbf{S}_s$ , validated in simulation, are chosen to be diagonal with the corresponding four different costs  $c_x$ ,  $c_{\dot{x}}$ ,  $c_{\ddot{x}}$ , and  $c_{\text{osc}}$  for the position error, velocity, acceleration, and oscillation reduction. For stability, the terminal stage cost at the end of  $\mathbf{Q}_s$  is chosen to be the solution to the algebraic Riccati equation [13]. Table II lists the planner parameters used in our experiments.

The experiments serve to verify the compliance to the defined safety thresholds, execution time, computational time, and compare the time efficiency with a velocity-scaled direct path based on real-time velocity scaling. The first experiment evaluates the compliance of our MPC planner to the EMU, its time efficiency, the generated EMs, and the required computational time. Four handovers between robot and human at different workspace positions are performed where the robot passes the human in close proximity and, thus, applies

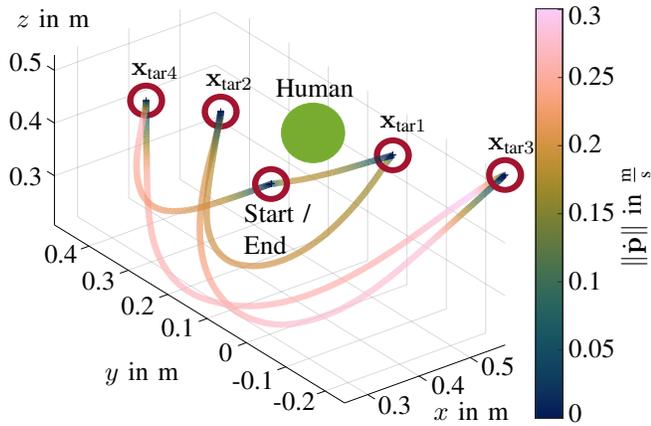


Fig. 4. Logged end effector motion of the first experiment. The traversed paths are dyed according to the absolute velocity.

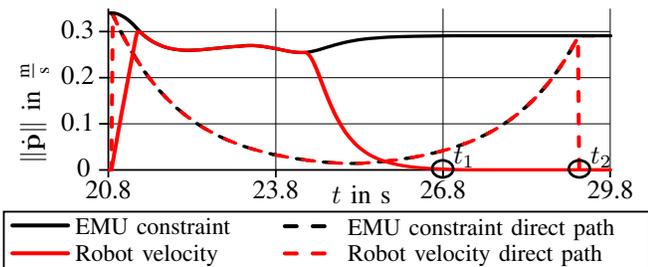


Fig. 5. Illustration of the EMU constraint during the motion from  $\mathbf{x}_{tar3}$  to  $\mathbf{x}_{tar4}$ . The result of our experiment (solid lines) is compared with a velocity scaled straight-line motion. The advantage of the simultaneous adjustment of path and velocity over velocity scaling is shown by the more constant velocity and the shorter path execution time.

EMs. The second experiment is a collaborative assembly task where the robot supplies the human with components to assemble and our proposed MPC planner is cascaded with the safe SMU motion generator. In this experiment, we observe the feasibility of the cascaded control loop for a collaborative pick-and-place task and its compliance to the SMU.

### B. Results

The trajectories traversed by the robot end effector in our first experiment are depicted by Fig. 4. All trajectories are reachable and feature EMs which are executed with higher velocities for greater human-robot distances according to the EMU function. The yellow colored motion in close proximity and the red ones with further distance illustrate this behaviour vividly. In this experiment, the average computational time for each motion planner iteration is 2.26 ms. The maximum observed computation time of 6.91 ms is still significantly smaller than the discretisation step size of  $\Delta t_{MPC} = 25$  ms, proving our motion planner’s feasibility. Also, all generated trajectories comply with the EMU constraint as exemplary depicted by Fig. 5 for the motion  $\mathbf{x}_{tar3}$  to  $\mathbf{x}_{tar4}$ .

In our second experiment the MPC planner is cascaded by the SMU and applied for a collaborative pick-and-place task. The safe and trustworthy EMs with time efficient paths and velocities traversed by the robot end effector are depicted by Fig. 6. We can claim again that all trajectories are reachable. The experiment shows that our cascaded control loop complies with the SMU.

Both experimental results indicate the practical stability of our cascaded MPC planner for real-world applications.

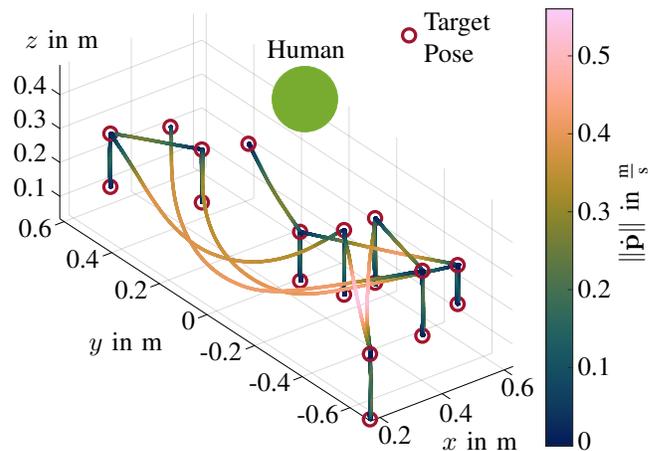


Fig. 6. Logged end effector motion of the second experiment. The traversed paths are dyed according to the absolute velocity.

### C. Discussion

The motion generated by our proposed motion planner finishes after about 6 s while a simulated direct path motion with velocity scaling takes 8.44 s; see Fig. 5. As a result, our MPC motion planner for online generation of safe human-aware motion enables 28% faster motions than a straightforward velocity scaling approach, due to the simultaneous path-velocity adjustment during optimisation.

Our MPC planner constrains absolute values for the physical limits (see Eq. (7)). On the contrary, previous research [8], [17] constrains each dimension or direction separately. As a result, our definition of the reachability constraints allows the maximum reachable motion speed during the whole motion even for space-diagonal motions where the distances to the target in  $x$ -,  $y$ -, and  $z$ -directions are not equal, while the previous approach applies only a fraction of the desired motion speed in these cases. Consequently, our approach results in faster motion and greater time efficiency.

As all robot motions are reachable and compliant with the EMU constraint, the applied linearisation strategy does not influence the effect of the constraints and our linearisation assumptions are acceptable. Although the linearisation leads to short calculation times enabling feasibility of the motion planner, they are not short enough for the robot real-time frequency of 1 kHz. Thus, to guarantee physical safety we cascaded our MPC planner with a real-time capable implementation of the SMU.

In the experiments we successfully embedded our proposed MPC planner to a Panda robot and observed stable task execution. However, an analytical stability analysis is prospect for future work. For real-time human detection, the system needs to be expanded e.g. with a 3D camera measuring the absolute position and an inertial measurement unit (IMU) to obtain additional acceleration data of the human hand. For this, we propose fusing both sensors for a more accurate human position and velocity estimation and apply predictor approaches like [24] or [28].

Lastly, the positive effect of the embedded EMU principle in our MPC planner on human involuntary motion occurrence [12] needs to be verified in user studies.

## V. CONCLUSION

In this paper, we proposed a general Cartesian MPC motion planner for robot manipulators which allows safe and trustworthy robot motion by applying constraints for psychologically-grounded safety. The planner was cascaded with the SMU in a robot motion generator and implemented to the FE Panda robot. Our experiments show that the closed-loop online motion planner generates reachable trajectories, which fulfill the given constraints and can be applied e.g. for pick-and-place applications. By simultaneous adjustments of Cartesian path and velocity in the constrained and predictive optimisation problem of the MPC planner, highly time efficient and human-aware motions are obtained, which are 2.44 s faster in the experiment than a common direct path-velocity scaling approach using the same constraints. Future work will focus on integration of the planner with human observers and validation studies on the planner's positive effect on team efficiency in HRI.

## ACKNOWLEDGMENT

The authors would like to thank Nico Mansfeld for lecturing. This work was supported by the Bavarian Institute for Digital Transformation (bidt-Project Responsible Robotics RR-AI), the European Union's Horizon 2020 research and innovation programme as part of the project Darko under grant no. 101017274 and project I.A.M. under Grant agreement no. 871899. We gratefully acknowledge the funding of the Lighthouse Initiative KI.FABRIK Bayern by StMWi Bayern, Forschungs- und Entwicklungsprojekt, grant no. DIK0249 and the funding of the Lighthouse Initiative Geriatrics by StMWi Bayern (Project X, grant no. IUK-1807-0007//IUK582/001). Funded by the Federal Ministry of Education and Research (BMBF) and the Free State of Bavaria under the Excellence Strategy of the Federal Government and the Länder. Please note that S. Haddadin has a potential conflict of interest as shareholder of Franka Emika GmbH.

## REFERENCES

- [1] DIN ISO/TS 15066:2016-02, Robots and robotic devices — Collaborative robots (ISO/TS 15066:2016).
- [2] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "On making robots understand safety: Embedding injury knowledge into control," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1578–1602, 2012.
- [3] N. Lucci, B. Lacevic, A. M. Zanchettin, and P. Rocco, "Combining speed and separation monitoring with power and force limiting for safe collaborative robotics applications," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6121–6128, 2020.
- [4] P. A. Lasota, G. F. Rossano, and J. A. Shah, "Toward safe close-proximity human-robot interaction with standard industrial robots," *2014 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 339–344, 2014.
- [5] D. Beckert, A. Pereira, and M. Althoff, "Online verification of multiple safety criteria for a robot trajectory," *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 6454–6461, 2017.
- [6] A. Palleschi, M. Hamad, S. Abdolshah, M. Garabini, S. Haddadin, and L. Pallottino, "Fast and safe trajectory planning: Solving the robot performance/safety trade-off in human-robot shared environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5445–5452, 2021.
- [7] M. Faroni, M. Beschi, and N. Pedrocchi, "An mpc framework for online motion planning in human-robot collaborative tasks," *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1555–1558, 2019.
- [8] H. Ding, K. Wijaya, G. Reißig, and O. Stursberg, "Optimizing motion of robotic manipulators in interaction with human operators," *International Conference on Intelligent Robotics and Applications 2011*, vol. 7101, pp. 520–531, 2011.
- [9] G. B. Avanzini, A. M. Zanchettin, and P. Rocco, "Constraint-based model predictive control for holonomic mobile manipulators," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1473–1479, 2015.
- [10] S. B. Liu, H. Roehm, C. Heinzemann, I. Lütkebohle, J. Oehlerking, and M. Althoff, "Provably safe motion of mobile robots in human environments," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1351–1357, 2017.
- [11] P. A. Lasota, T. Fong, and J. A. Shah, "A survey of methods for safe human-robot interaction," *Foundations and Trends in Robotics*, vol. 5, no. 4, pp. 261–349, 2017.
- [12] R. J. Kirschner, H. Mayer, L. Burr, N. Mansfeld, S. Abdolshah, and S. Haddadin, "Expectable motion unit: Avoiding hazards from human involuntary motions in human-robot interaction," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2993–3000, 2022.
- [13] J. B. Rawlings, M. M. Diehl, and D. Q. Mayne, *Model predictive control: Theory, computation and design*, 2nd ed. Madison: Nob Hill, 2017.
- [14] D. Lee, C. Liu, Y.-W. Liao, and J. K. Hedrick, "Parallel interacting multiple model-based human motion prediction for motion planning of companion robots," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 52–61, 2017.
- [15] M. M. G. Ardakani, B. Olofsson, A. Robertsson, and R. Johansson, "Real-time trajectory generation using model predictive control," *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 942–948, 2015.
- [16] A. Tika, N. Gafur, V. Yfantis, and N. Bajcinca, "Optimal scheduling and model predictive control for trajectory planning of cooperative robot manipulators," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9080–9086, 2020.
- [17] L. Paparusso, N. Kashiri, and N. G. Tsagarakis, "A disturbance-aware trajectory planning scheme based on model predictive control," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5779–5786, 2020.
- [18] J. Nubert, J. Kohler, V. Berenz, F. Allgower, and S. Trimpe, "Safe and fast tracking on a robot manipulator: Robust mpc and neural network control," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3050–3057, 2020.
- [19] G. B. Avanzini, A. M. Zanchettin, and P. Rocco, "Constrained model predictive control for mobile robotic manipulators," *Robotica*, vol. 36, no. 1, pp. 19–38, 2018.
- [20] H. Waschl, *Optimization and optimal control in automotive systems*, ser. Lecture notes in control and information sciences. Cham: Springer, 2014, vol. 455.
- [21] T. Brüdigam, F. Di Luzio, L. Pallottino, D. Wollherr, and M. Leibold, "Grid-based stochastic model predictive control for trajectory planning in uncertain environments," *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–8, 2020.
- [22] M. Svenstrup, T. Bak, and H. J. Andersen, "Trajectory planning for robots in dynamic human environments," *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4293–4298, 2010.
- [23] T. Gold, A. Völz, and K. Graichen, "Model predictive position and force trajectory tracking control for robot-environment interaction," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7397–7402, 2020.
- [24] P. A. Lasota and J. A. Shah, "A multiple-predictor approach to human motion prediction," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2300–2307, 2017.
- [25] D. Kulić and E. Croft, "Physiological and subjective responses to articulated robot motion," *Robotica*, vol. 25, no. 1, pp. 13–27, 2007.
- [26] L. Takayama and C. Pantofaru, "Influences on proxemic behaviors in human-robot interaction," *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5495–5502, 2009.
- [27] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
- [28] W. Sheng, A. Thobbi, and Y. Gu, "An integrated framework for human-robot collaborative manipulation," *IEEE transactions on cybernetics*, vol. 45, no. 10, pp. 2030–2041, 2015.