Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 55-38 (2022) 55–60

Stiffness Optimized Multi-Robot Behavior Planning using Reduced Hessian Method ⋆

X. Ye, M. Schwartz, S. Hohmann

Institute of Control Systems, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
(e-mail: {xin.ye, manuel.schwartz, soeren.hohmann}@kit.edu)

Abstract: Stiffness is a critical weakness in robot-driven manufacturing. Following the idea of enhancing stiffness by physically coupling multiple robots, the cooperative behavior should be optimized. We present a reduced Hessian method to generate stiffness optimized placement and motion for coupled robots with given manufacturing processes. To improve efficiency, the search space dimension of the optimization problem is reduced while the high-dimensional path and coupling constraints are satisfied in each iteration. By integrating a stiffness model in the optimization, cooperative behavior with stiffness enhancement is generated. The results are validated by experiments for both 7-axis and 6-axis robots with physical coupling.

Keywords: Cooperative manipulation, cooperative manufacturing, force and compliance control, multi-cooperative-robot systems, path-constrained trajectory planning

1. INTRODUCTION

Utilizing robots in manufacturing processes has the potential to increase flexibility and reduce costs (Perzylo et al., 2019). In multi-robot manufacturing systems, physically coupled robots execute coordinated actions as shown in Fig. 1. The physical coupling can be realized by a rigid coupler connecting the flages of multiple robots. With tools attached to the coupler, various manufacturing tasks such as drilling, milling, or metal sheet bending can be carried out. Compared to a single robot, the coupled system is predicted to exhibit higher manufacturing quality due to enhanced stiffness in Mühlbeier et al. (2020). To establish such a manufacturing system with coupled robots, methods for planning cooperative behavior, i.e. placement and motion, need to be developed. An optimal cooperative behavior is characterized by a path tracking with sufficiently low error, a high stiffness against process forces, and a reasonable distribution of loads according to robots' capabilities. Excessive internal loading in the coupler is to be avoided, since it embodies a counter-cooperative behavior, causing unnecessary structural deformation and reduction of precision (Erhart and Hirche, 2015).

Compared to machine tools, stiffness is identified as robots' major weakness that leads to large errors in manufacturing (Abele et al., 2007). Therefore, finding stiff configurations has been the focus of research (Busson et al., 2017). Although the stiffness can be optimized by integrating the simplified joint stiffness model (Abele et al., 2007) and the relevant performance indexes in optimization problems (Chiu, 1988), these optimization approaches are not extended to multiple coupled robots.

For coupled robots, additional coupling constraints must be satisfied along with path tracking. In multi-robot path-constrained trajectory planning (Kabir et al., 2021), multiple stages are necessary to sequentially consider different objectives and constraints. Therefore, it relies on use-case dependent weight tuning for different stages. Placement variables, which provide additional degrees of freedom (DOF) in path-constrained problems, are crucial to robot performance. But placement is either not explicitly handled (Kabir et al., 2021) or separately considered apart from joint configurations, creating a nested optimization framework with inefficient iterations (Nicola et al., 2018).

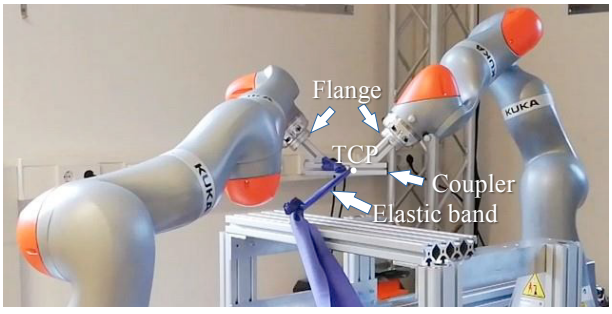

Fig. 1. Physical coupling by a rigid coupler with process force applied by an elastic band

For equality-constrained optimization, the method of reduced Hessian was proposed in Gabay (1982). In Schmid and Biegler (1994), the optimization is further developed by integrating a solver for successive quadratic programming (SQP) subproblems. There exist potential benefits of adopting such method to constrained optimization for multi-robot systems. On the one hand, the high dimension of variables describing robots' behavior and the low DOF due to strict coupling and path tracking constraints can be exploited with the low dimensional Hessian matrix, which reduces the computational complexity. On the other hand, the descent in null-space ensures the preservation

⋆ This work was conducted within the KIT Excellence Initiative FutureFields project "Wertstromkinematik".

10.1016/j.ifacol.2023.01.133

of constraints. However, these methods have not been applied to manipulators except point-to-point single-robot path planning in Schulz (1998). For path-constrained robot motions, there is yet no suitable procedure to establish the reduced Hessian matrix or to determine the null-space.

To plan the cooperative behavior in multi-robot manufacturing systems, the following challenges are to be tackled:

(a) Modeling of the stiffening effect of physically coupled robots against given external process forces
(b) Optimization of the stiffness of coupled robots along a path by adjusting the cooperative behavior
(c) Satisfaction of synchronization and path tracking constraints throughout the optimization
(d) Generation of smooth motion trajectories

This paper mainly focuses on (b) and (c) where a reduced Hessian optimization method is developed that involves both the robot-workpiece placement and the joint configurations as behavioral variables, and considers coupled stiffness objective under coupling constraints. This paper not only presents a robotic application of Schmid and Biegler (1994), but also proposes innovative methods for generating search spaces and finding the descending direction with reduced dimension in context of robot kinematics and B-spline motion path.

2. STIFFNESS MODEL OF COUPLED ROBOTS

Stiffness and compliance at a tool attached to the coupler are modeled in this section, before a performance index for stiffness is presented.

2.1 Stiffness and Compliance of a Coupled System

In manufacturing processes, a serial industrial robot is under external force $\boldsymbol{F} \in \mathbb{R}^3$ and moment $\boldsymbol{M} \in \mathbb{R}^3$, which are combined as process wrench $\boldsymbol{w} = [\boldsymbol{M}^\intercal, \boldsymbol{F}^\intercal]^\intercal$ at the tool center point (TCP). TCP of the robot exhibits deviation $\delta\boldsymbol{p} \in \mathbb{R}^6$ in three rotational and three translational directions. The compliance is a mapping from the wrench to the deviation: $\delta\boldsymbol{p} = \boldsymbol{C}(\boldsymbol{w})$. According to Abele et al. (2007), the compliance of gears on joints contributes 50% to 75% of the overall compliance. For simplication, we assume that the only compliant elements of the robotic system are the joints which only deform in their rotatable direction. The angular joint deviation δq under joint torque τ is assumed to comply to Hooke's Law: $\tau = k_q \delta q$, whereby k_q is the joint stiffness analog to that of the linear torsion spring. Therefore, the compliance of a single robot can be described by matrix $\boldsymbol{C} \in \mathbb{R}^{6\times6}$, derived as follows in Abele et al. (2007):

$$\delta\boldsymbol{p} = \boldsymbol{C}\boldsymbol{w} = \boldsymbol{J}(\boldsymbol{q})\boldsymbol{K}_q^{-1}\boldsymbol{J}(\boldsymbol{q})^\intercal\boldsymbol{w} \tag{1}$$

whereby $\boldsymbol{J}(\boldsymbol{q})$ is the configuration dependent Jacobian and $\boldsymbol{K}_q$ the diagonal joint stiffness matrix. TCP-stiffness is the inverse of compliance, so it applies:

$$\boldsymbol{w} = \boldsymbol{C}^{-1}\delta\boldsymbol{p} \tag{2}$$

Suppose A is a set of robots. A rigid coupler connects the flanges of the robots. They share a common TCP on the coupler so that the TCP-deviations $\delta\boldsymbol{p}_r, r \in A$ are equal:

$$\delta\boldsymbol{p}_r = \delta\boldsymbol{p}^{\mathrm{TCP}}, \forall r \in A \tag{3}$$

The external process wrench on TCP of the coupler is balanced by the counter-acting wrenches $\boldsymbol{w}_r$ provided by the robots: $\boldsymbol{w}^{\mathrm{TCP}} = \sum_{r\in A}\boldsymbol{w}_r = \sum_{r\in A}\boldsymbol{C}_r^{-1}\delta\boldsymbol{p}_r$, whereby (2) is used in the last equality. Under the assumption that the compliance $\boldsymbol{C}^{\mathrm{TCP}}$ of the coupled system is also linear, it applies that $\boldsymbol{w}^{\mathrm{TCP}} = \left(\boldsymbol{C}^{\mathrm{TCP}}\right)^{-1}\delta\boldsymbol{p}^{\mathrm{TCP}} = \sum_{r\in A}\boldsymbol{C}_r^{-1}\delta\boldsymbol{p}_r$. Considering the last equality and (3), and that the compliance matrices have full rank, the additivity of stiffness is derived:

$$\left(\boldsymbol{C}^{\mathrm{TCP}}\right)^{-1} = \sum_{r\in A}\boldsymbol{C}_r^{-1} \tag{4}$$

An intuitive interpretation is that a parallel connection of compliant serial industrial robots is analog to the parallel connection of springs.

By applying (2), (3), and (1) sequentially, the wrench $\boldsymbol{w}_r$ distributed on robot r is expressed as follows:

$$\boldsymbol{w}_r = \boldsymbol{C}_r^{-1}\delta\boldsymbol{p}_r = \boldsymbol{C}_r^{-1}\delta\boldsymbol{p}^{\mathrm{TCP}} = \boldsymbol{C}_r^{-1}\boldsymbol{C}^{\mathrm{TCP}}\boldsymbol{w}^{\mathrm{TCP}} \tag{5}$$

Expression (5) shows that the load distributed on robot $r \in A$ is proportional to its stiffness $\boldsymbol{C}_r^{-1}$. Furthermore, the joint torques $\boldsymbol{\tau}_r$ of robot r, dependent on its Jacobian $\boldsymbol{J}_r$, can be expressed as follows:

$$\boldsymbol{\tau}_r = \boldsymbol{J}_r^\intercal\boldsymbol{w}_r \tag{6}$$

2.2 Stiffness-Oriented Performance Index

As a multi-robot extension to the task compatibility index (Chiu, 1988), the objective function $\mathcal{J}_{\mathrm{proc}}$ for stiffness in the direction of unit process wrench $\boldsymbol{\xi} = \boldsymbol{w}^{\mathrm{TCP}}/|\boldsymbol{w}^{\mathrm{TCP}}|$ is described as follows:

$$\mathcal{J}_{\mathrm{proc}} = \boldsymbol{\xi}^\intercal\boldsymbol{C}^{\mathrm{TCP}}\boldsymbol{\xi} \tag{7}$$

The process wrench may deviate from the predicted $\boldsymbol{w}^{\mathrm{TCP}}$ by disturbance due to inaccurate process modeling, vibration etc. Therefore, the objective function also considers the weakest direction of robot stiffness, i.e. the longest major axis length of the compliance matrix:

$$\mathcal{J}_{\mathrm{major}} = \max\left\{\mathrm{eig}\left(\boldsymbol{C}^{\mathrm{TCP}}\right)\right\} \tag{8}$$

The overall stiffness-oriented objective function is expressed as a weighted sum as follows:

$$\mathcal{J}_{\mathrm{step}} = w_{\mathrm{proc}}\mathcal{J}_{\mathrm{proc}} + w_{\mathrm{major}}\mathcal{J}_{\mathrm{major}} \tag{9}$$

Further terms such as a penalty of singularities can be added, since stiff configurations tend to be close to singularities.

3. PROBLEM FORMULATION

The planning of cooperative manufacturing behavior can be described as a path-constrained trajectory planning problem as in Kabir et al. (2021). The optimization variables for such a problem are chosen to be joint configurations $\boldsymbol{q}$ along the manufacturing path and placement $\boldsymbol{y}$, which are combined as behavioral variables $\boldsymbol{\theta} = [\boldsymbol{q}, \boldsymbol{y}]$. Such a description is specified enough to consider coupling and path constraints as well as stiffness criteria while it remains light-weight without involving complex dynamics and control of robots. The optimized behavior can be used as a reference in the trajectory generation and tracking afterwards. In case of a manufacturing with n_{r} coupled

robots, behavioral variables with subscripts r for robots, wp for workpiece are as follows:

$$\boldsymbol{\theta} = [\boldsymbol{q}_1^\intercal, ..., \boldsymbol{q}_{n_\mathrm{r}}^\intercal, \boldsymbol{y}_\mathrm{r}^\intercal, \boldsymbol{y}_\mathrm{wp}^\intercal]^\intercal \tag{10}$$

The vector $\boldsymbol{\theta}$ at a certain moment in the manufacturing process spans the space V_θ with the dimension:

$$\dim(V_\theta) = n_\mathrm{j} n_\mathrm{r} + n_\mathrm{wp} + n_\mathrm{pr} \tag{11}$$

Hereby, the axis number of a robot is n_j, the adjustable dimension of workpiece placement is n_wp and the dimension of placement of all robots $\boldsymbol{y}_\mathrm{r}$ is n_pr.

To deal with both static and non-static behavioral variables in one optimization problem, we use B-splines (Piegl and Tiller, 2012) with different orders. For example, a third-order B-spline guarantees the continuity of the angular acceleration of joints. A zero-order B-spline gives a constant value, which is suitable for placement variables. B-splines are completely defined by an array of knots $\boldsymbol{U}$ and several control points $\boldsymbol{x} \in \mathbb{R}^n$. Therefore behavioral variables are expressed as $\boldsymbol{\theta}(u, \boldsymbol{U}, \boldsymbol{x})$, where $u \in [0, 1]$ represents the progression of the B-spline. As a simplification, the array of knots $\boldsymbol{U}$ is evenly distributed between zero and one. In this way, the planning problem is converted to a parameter optimization problem of $\boldsymbol{x}$.

The cooperative behavior should be planned according to the predefined process variables including tool path and process forces in task space, which are denoted as $\boldsymbol{s}(u, \boldsymbol{x})$. They vary in progress $u \in [0, 1]$ of the process and their position and orientation are dependent on the placement variables included in $\boldsymbol{x}$.

The constrained parameter optimization problem of behavior planning is as follows:

$$\min_{\boldsymbol{x}} \mathcal{J}_\mathrm{total}(\boldsymbol{x}) = \sum_{u_i \in [0,1]} \mathcal{J}_\mathrm{step}(\boldsymbol{\theta}(u_i, \boldsymbol{x}), \boldsymbol{s}(u_i, \boldsymbol{x})) \tag{12a}$$

$$\text{s.t.} \quad \boldsymbol{e}_\mathrm{pose}(\boldsymbol{\theta}(u_i, \boldsymbol{x}), \boldsymbol{s}(u_i, \boldsymbol{x})) = \boldsymbol{0} \tag{12b}$$

$$\boldsymbol{\theta}_\mathrm{min} \le \boldsymbol{\theta}(u_i, \boldsymbol{x}) \le \boldsymbol{\theta}_\mathrm{max}, \quad u_i \in [0, 1] \tag{12c}$$

The objective function $\mathcal{J}_\mathrm{step}$ is given in (9). The constraints $\boldsymbol{e}_\mathrm{pose} = \boldsymbol{0}$ include elimination of coupling error and path tracking. The joint limits (12c) can be replaced by sufficient condition (13), because the interpolated configurations at any point $u \in [0, 1]$ do not exceed the values of minimal and maximal control points:

$$\boldsymbol{\theta}_\mathrm{min} \le \boldsymbol{x} \le \boldsymbol{\theta}_\mathrm{max} \tag{13}$$

A manufacturing task for n_r robots with a path containing $l \in \mathbb{N}$ control points for each joint constitutes a problem in a new vector space V_x, extended from (11), spanned by the behavioral variables of the dimension:

$$\dim(V_\mathrm{x}) = n = n_\mathrm{j} n_\mathrm{r} l + n_\mathrm{wp} + n_\mathrm{pr} \tag{14}$$

At each moment on the path, rigid coupling and path tracking constraints are imposed. Coupling each additional robot limits three rotational and three translational DOF, resulting in $6(n_\mathrm{r} - 1)$ DOF losts. For path tracking, TCP must be at the required 3-dimensional way-point on track, possibly with further 1-3 rotational constraints, so the DOF lost by tracking is $d_\mathrm{tr} \le 6$. To obtain a behavior without constraint violation, The optimization at this moment has to be carried out in a redundant subspace $V_{\theta,\mathrm{red}}$ of dimension:

$$\begin{aligned} \dim(V_{\theta,\mathrm{red}}) &= \dim(V_\theta) - 6(n_\mathrm{r} - 1) - d_\mathrm{tr} \\ &= n_\mathrm{j} n_\mathrm{r} + n_\mathrm{wp} + n_\mathrm{pr} - 6(n_\mathrm{r} - 1) - d_\mathrm{tr} \end{aligned} \tag{15}$$

Throughout the whole path, the configuration of redundant joints is determined by l control points. So the joint redundancy $n_\mathrm{j} n_\mathrm{r} - 6(n_\mathrm{r} - 1) - d_\mathrm{tr}$ is multiplied by l. The space of overall redundancy $V_{x,\mathrm{red}}$ has the dimension:

$$\dim(V_{x,\mathrm{red}}) = (n_\mathrm{j} n_\mathrm{r} - 6(n_\mathrm{r} - 1) - d_\mathrm{tr}) l + n_\mathrm{wp} + n_\mathrm{pr} \tag{16}$$

4. SOLUTION APPROACH

This section illustrates the procedure of an iteration of the reduced Hessian method with one descending step under constraint satisfaction and one step to eliminate constraint violations. The former step requires novel methods to obtain a null-space projector (Subsection 4.3) and a reduced gradient vector (Subsection 4.4) in context of coupled kinematics of robots and B-spline joint paths.

At the k-th iteration of solving the problem (12), a SQP method should generate a descending step $\boldsymbol{d} \in \mathbb{R}^n$ from the current iterate point $\boldsymbol{x}_k \in \mathbb{R}^n$ by solving the following subproblem as stated in Schmid and Biegler (1994):

$$\min_{\boldsymbol{d} \in \mathbb{R}^n} \nabla \mathcal{J}_{\mathrm{total},k}(\boldsymbol{x}_k)^\intercal \boldsymbol{d} + \frac{1}{2} \boldsymbol{d}^\intercal \boldsymbol{B}_k \boldsymbol{d} \tag{17a}$$

$$\text{s.t.} \ \boldsymbol{e}_\mathrm{pose}(\boldsymbol{x}_k) + \nabla \boldsymbol{e}_\mathrm{pose}^\intercal(\boldsymbol{x}_k) \boldsymbol{d} = \boldsymbol{0} \tag{17b}$$

$$\boldsymbol{\theta}_\mathrm{min} \le \boldsymbol{x}_k + \boldsymbol{d} \le \boldsymbol{\theta}_\mathrm{max} \tag{17c}$$

$\boldsymbol{B}_k$ is an approximation towards the Hessian of the original objective function $\mathcal{J}_\mathrm{total}$ at $\boldsymbol{x}_k$. The solution is decomposed into two components in null space and range-space of pose errors respectively:

$$\boldsymbol{d} = \boldsymbol{h}_k + \boldsymbol{v}_k \tag{18}$$

The null-space component $\boldsymbol{h}_k$, also known as horizontal step, is a descending step towards higher stiffness without violating the linearized coupling and tracking constraints. The range-space component $\boldsymbol{v}_k$, also known as vertical step, corrects residual violations of constraints caused by linearization errors.

4.1 Range-Space Step and Constraint Dimensions

As in Schmid and Biegler (1994), the range-space step is calculated by $\boldsymbol{v}_k = -\left(\nabla \boldsymbol{e}_\mathrm{pose}^\intercal(\boldsymbol{x}_k)\right)^\dagger \boldsymbol{e}_\mathrm{pose}(\boldsymbol{x}_k)$. The pseudoinverse designated by $\dagger$ can only be carried out when $\nabla \boldsymbol{e}_\mathrm{pose}(\boldsymbol{x}_k) \in \mathbb{R}^{n \cdot \dim(V_\mathrm{e})}$ has full column rank, where V_e is the space spanned by the pose errors. Therefore, V_e has the dimension:

$$\dim(V_\mathrm{e}) \le \dim(V_\mathrm{x}) - \dim(V_{x,\mathrm{red}}) = (6(n_\mathrm{r} - 1) + d_\mathrm{tr}) l \tag{19}$$

This means that the equality constraints (12b) can only be imposed on l points on the path if the null-space descent is utilized. The tracking precision on the stretches between the selected l points cannot be guaranteed. This is adequate for a behavioral planning to obtain stiff poses for a given manufacturing process optimally, but the trajectory needs to be refined in the post-processing.

4.2 Null-Space Step and Reduction of Dimension

The null-space step is obtained from a null-space projection $\boldsymbol{h}_k = \boldsymbol{Z}_k \bar{\boldsymbol{d}}$, where $\bar{\boldsymbol{d}} \in \mathbb{R}^{\dim(V_{x,\mathrm{red}})}$ is the descent in subspace of redundancy. The null-space projector $\boldsymbol{Z}_k \in \mathbb{R}^{n \cdot \dim(V_{x,\mathrm{red}})}$ projects the subspace descent into the space

spanned by all decision variables without affecting the pose errors, which demands:

$$\nabla \boldsymbol{e}_{\text{pose}}^{\intercal}(\boldsymbol{x}_k)\boldsymbol{Z}_k = \boldsymbol{0} \tag{20}$$

By substituting (18) with the known range-space step and the unknown null-space step in (17a), the SQP subproblem becomes a quadratic programming with only boundary constraints (Schmid and Biegler, 1994):

$$\min_{\bar{\boldsymbol{d}}} \bar{\boldsymbol{g}}_k^{\intercal}\bar{\boldsymbol{d}} + \boldsymbol{v}_k^{\intercal}\boldsymbol{B}_k\boldsymbol{Z}_k\bar{\boldsymbol{d}} + \frac{1}{2}\bar{\boldsymbol{d}}^{\intercal}\bar{\boldsymbol{B}}_k\bar{\boldsymbol{d}} \tag{21}$$

$$\text{s.t. } \boldsymbol{\theta}_{\min} - \boldsymbol{x}_k - \boldsymbol{v}_k \le \boldsymbol{Z}_k\bar{\boldsymbol{d}} \le \boldsymbol{\theta}_{\max} - \boldsymbol{x}_k - \boldsymbol{v}_k \tag{22}$$

where $\bar{\boldsymbol{g}}_k = \boldsymbol{Z}_k^{\intercal}\nabla\mathcal{J}_{\text{total},k}$ is the reduced gradient and $\bar{\boldsymbol{B}}_k = \boldsymbol{Z}_k^{\intercal}\boldsymbol{B}_k\boldsymbol{Z}_k$ is the reduced Hessian. As such, the dimension of the quadratic programming subproblems in SQP will be reduced from $\dim(V_{\text{x}})$ to $\dim(V_{x,\text{red}})$. In the routine of solving the quadratic subproblem, $\bar{\boldsymbol{B}}_k$ is updated by BFGS formula (Nocedal and Wright, 1999) and the cross term $\boldsymbol{v}_k^{\intercal}\boldsymbol{B}_k\boldsymbol{Z}_k\bar{\boldsymbol{d}}$ is commonly ignored (Schmid and Biegler, 1994). Therefore, for solving the optimization problem, only the null-space projector $\boldsymbol{Z}_k$ and the reduced gradient $\bar{\boldsymbol{g}}_k$ are still to be determined.

4.3 Null-Space Projector

Before dealing with $\boldsymbol{Z}_k$, we set up the projector $\boldsymbol{Z}_{\boldsymbol{\theta}} \in \mathbb{R}^{\dim(V_\theta)\cdot\dim(V_{\theta,\text{red}})}$ that projects the adjustment of redundant configurations in subspace $V_{\theta,\text{red}}$ into V_θ without affecting pose errors. Each column of $\boldsymbol{Z}_{\boldsymbol{\theta}}$ should represent a mode of null-space motion when one redundant dimension in $V_{\theta,\text{red}}$ is adjusted. Elements in this column represent the reactive adjustment of each configuration in response so that pose errors are kept unchanged. It is recommended in Biegler et al. (1995) to construct null-space projectors to be orthonormal in sake of numerical stability and robustness, but in robotic application, movement simplicity and interpretability in the column assignments is of significance. For example, when a redundant joint of a robot is adjusted, only the joint configurations of this single robot is allowed to react, but the placements do not. Such a projector column is generated as follows.

Firstly, among the configurations in V_θ, redundant ones in $V_{\theta,\text{red}}$ are assigned by an assignment matrix $\boldsymbol{A} \in \mathbb{R}^{\dim(V_{\theta,\text{red}})\cdot\dim(V_\theta)}$ with its elements either 0 or 1. Such assignment is described by $\dot{\boldsymbol{\theta}}_{\text{red}} = \boldsymbol{A}\dot{\boldsymbol{\theta}}$. For example, the third joint of 7-axis robots, which is absent in 6-axis-robots, and all $n_{\text{wp}} + n_{\text{pr}}$ placement variables are assigned.

Secondly, a Jacobian $\boldsymbol{J}_{\text{e}}$ mapping the adjustment rate of configurations $\dot{\boldsymbol{\theta}}$ towards the changing rate of pose error $\dot{\boldsymbol{e}}_{\text{pose}}$ is obtained from the model of robot-workpiece ensemble. Such mapping is described as follows:

$$\dot{\boldsymbol{e}}_{\text{pose}} = \boldsymbol{J}_{\text{e}}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} \tag{23}$$

Afterwards, null-space projectors from the space spanned by vector $\dot{\boldsymbol{\theta}}$ toward itself are generated using pseudoinverse which is expressed as follows (Slotine and Siciliano, 1991):

$$\boldsymbol{N}(\boldsymbol{\theta}) = \boldsymbol{I} - \boldsymbol{J}_{\text{e}}(\boldsymbol{\theta})^{\dagger}\boldsymbol{J}_{\text{e}}(\boldsymbol{\theta}) \tag{24}$$

A projected adjustment rate $\dot{\boldsymbol{\theta}}_{\boldsymbol{N}} = \boldsymbol{N}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$ has no effect on pose errors at this configuration. An adjustment rate in the redundant configurations can be projected similarly: $\dot{\boldsymbol{\theta}}_{\boldsymbol{N}} = \boldsymbol{N}(\boldsymbol{\theta})\boldsymbol{A}^{\dagger}\dot{\boldsymbol{\theta}}_{\text{red}}$. Therefore the projector is derived:

$$\boldsymbol{Z}_{\boldsymbol{\theta}}(u,\boldsymbol{x}) = \boldsymbol{Z}_{\boldsymbol{\theta}}(\boldsymbol{\theta}(u,\boldsymbol{x})) = \boldsymbol{N}(\boldsymbol{\theta}(u,\boldsymbol{x}))\boldsymbol{A}^{\dagger} \tag{25}$$

The required null-space projector $\boldsymbol{Z}_k$ is derived from $\boldsymbol{Z}_{\boldsymbol{\theta}}$ using the base functions of B-spline:

$$\frac{\partial q_{r,j}(u)}{\partial \boldsymbol{x}_{r,j}} = \boldsymbol{R}_k(u), \quad 1 \le r \le n_{\text{r}}, 1 \le j \le n_{\text{j}} \tag{26}$$

where $\boldsymbol{R}_k(u) = [R_{1,k}(u), ..., R_{l,k}(u)]^{\intercal}$ has a length equal to the number of control points l, and $k \in \mathbb{N}$ is the spline order. r stands for robot index and j joint index. For static workpiece placement, $k = 0$ and $l = 1$. So the vector of base functions becomes a scalar value $\frac{\partial y_{\text{wp}}}{\partial x_{\text{wp}}} = 1$. The placement of robots is similar to that of the workpiece. Thus, a matrix is constructed, mapping control point adjustment to configuration changes at any given $u \in [0,1]$:

$$\frac{\partial \boldsymbol{\theta}\,(u,\boldsymbol{x})}{\partial \boldsymbol{x}} = \begin{bmatrix} \boldsymbol{R}_k(u) & & \\ & \ddots & \\ & & \boldsymbol{R}_k(u) \\ & & & \boldsymbol{I} \end{bmatrix} \in \mathbb{R}^{n\cdot\dim(V_\theta)} \tag{27}$$

As such, an adjustment of redundant configurations at a certain u in space $V_{\theta,\text{red}}$ is projected into the adjustment of control points by $\frac{\partial \boldsymbol{\theta}(u,\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{Z}_{\boldsymbol{\theta}}$. Throughout the B-spline with $u \in [0,1]$, redundant configurations are adjusted at l different values of u. The corresponding adjustment of control points is accumulated. A null-space projector is obtained by:

$$\boldsymbol{Z}_k^{'} = \sum_{i=1}^{l} \frac{\partial \boldsymbol{\theta}\,(u_i,\boldsymbol{x})}{\partial \boldsymbol{x}}\boldsymbol{Z}_{\boldsymbol{\theta}}(u_i)\bar{\boldsymbol{A}}(u_i) \tag{28}$$

where $\bar{\boldsymbol{A}}(u_i) \in \mathbb{R}^{\dim(V_{\theta,\text{red}})\cdot\dim(V_{x,\text{red}})}$ determines the indices of redundant control points corresponding to the redundant configurations. However, there remains a problem to $\boldsymbol{Z}_k^{'} \in \mathbb{R}^{\dim(V_{\text{x}})\cdot\dim(V_{\text{x,red}})}$ that the elements projecting static placement are added to the same entry of $\boldsymbol{Z}_k^{'}$ for l times while the elements projecting non-static joint configurations are added to different entries which correspond to l control points. To compensate this, a weighting matrix is constructed $\boldsymbol{W}_k = \sum_{i=1}^{l}\frac{\partial \boldsymbol{\theta}(u_i,\boldsymbol{x})}{\partial \boldsymbol{x}}\cdot\boldsymbol{1}\cdot\bar{\boldsymbol{A}}(u_i)$, where $\boldsymbol{1}$ is a matrix of the same dimension as $\boldsymbol{Z}_{\boldsymbol{\theta}}(u_i)$ with all elements set to one. The final null-space projector is thus obtained by an element-wise division $\oslash$ at entries for placement variables: $\boldsymbol{Z}_k = \boldsymbol{Z}_k^{'} \oslash \boldsymbol{W}_k$.

4.4 Reduced Gradient

A direct calculation of the reduced gradient $\bar{\boldsymbol{g}}_k = \boldsymbol{Z}_k^{\intercal}\nabla\mathcal{J}_{\text{total},k}$ without obtaining the gradient $\nabla\mathcal{J}_{\text{total},k}$ with respect to each control point is desirable, because the number of cost function evaluations in finite differentiation is reduced from $\dim(V_{\text{x}})$ to $\dim(V_{\text{x,red}})$. For each $u_i \in [0,1], i = 1, ..., l$ in the B-spline, the $\dim(V_{\theta,\text{red}})$ variations of redundant configurations $\delta\boldsymbol{\theta}_{\text{red}}$ are created, each represented by $\varepsilon\boldsymbol{e}_j, j = 1, ..., \dim(V_{\theta,\text{red}})$. Hereby, ε is a small number and $\boldsymbol{e}_j \in \mathbb{R}^{\dim(V_{\theta,\text{red}})}$ is a vector with 1 in the j-th element and 0 in others. Then $\dim(V_{\theta,\text{red}})$ variations of all configurations are obtained by a null-space projection $\delta\boldsymbol{\theta} = \boldsymbol{Z}_{\boldsymbol{\theta}}\delta\boldsymbol{\theta}_{\text{red}}$ so that the pose errors do not change when the configuration is varied by $\delta\boldsymbol{\theta}$. The gradient $\nabla_{\text{red}}\mathcal{J}_{\text{step}}(u_i) \in \mathbb{R}^{\dim(V_{\theta,\text{red}})}$ is computed:

$$\nabla_{\mathrm{red}}\mathcal{J}_{\mathrm{step}}(u_i) = \frac{\mathcal{J}_{\mathrm{step}}(\boldsymbol{\theta} + \boldsymbol{Z}_{\boldsymbol{\theta}}\delta\boldsymbol{\theta}_{\mathrm{red}}, \boldsymbol{s}) - \mathcal{J}_{\mathrm{step}}(\boldsymbol{\theta}, \boldsymbol{s})}{\boldsymbol{Z}_{\boldsymbol{\theta}}\delta\boldsymbol{\theta}_{\mathrm{red}}}$$

The gradient of redundant configurations is then mapped into the space of control points V_{x}:

$$\nabla\mathcal{J}'_{\mathrm{total},k} = \sum_{i=1}^{l}\left(\frac{\partial\boldsymbol{\theta}(u_i, \boldsymbol{x})}{\partial\boldsymbol{x}}\right)^{\mathsf{T}}\boldsymbol{A}^{\dagger}\nabla_{\mathrm{red}}\mathcal{J}_{\mathrm{step}}(u_i) \quad (29)$$

Unlike the gradient w.r.t. all control points $\nabla\mathcal{J}_{\mathrm{total},k} \in \mathbb{R}^{\dim(V_{\mathrm{x}})}$, the obtained gradient $\nabla\mathcal{J}'_{\mathrm{total},k} \in \mathbb{R}^{\dim(V_{\mathrm{x}})}$ is a vector that can be completely contained in space $V_{\mathrm{x,red}}$ and results in a zero vector if projected in space $V_{\mathrm{x}} \setminus V_{\mathrm{x,red}}$. The reduced gradient is calculated by $\bar{\boldsymbol{g}}_k = \boldsymbol{Z}_k^{\mathsf{T}}\nabla\mathcal{J}'_{\mathrm{total},k}$.

4.5 Pre- and Post-Processing

The optimization problem (12) needs the initial guess $\boldsymbol{x}_0$. Redundant configurations have to be given. The rest of joint configurations can be obtained using inverse kinematics on sampled points of the tool path. The joint configurations make up a matrix $\boldsymbol{Q}_{1:l_{\mathrm{smp}}} \in \mathbb{R}^{l_{\mathrm{smp}} \times n_{\mathrm{j}}}$ with l_{smp} being the number of sampled points on tool path. Control points of the k-th ordered B-spline $\boldsymbol{X}_{1:l} \in \mathbb{R}^{l \times n_{\mathrm{j}}}$ are to be generated by solving the system of equations:

$$\boldsymbol{Q}_{1:l_{\mathrm{smp}}} = \boldsymbol{\Phi}\boldsymbol{X}_{1:l} \quad (30)$$

$$\begin{bmatrix}\boldsymbol{Q}_1\\ \vdots\\ \boldsymbol{Q}_{l_{\mathrm{smp}}}\end{bmatrix} = \begin{bmatrix}R_{1,k}(u_1) & \dots & R_{l,k}(u_1)\\ \vdots & \ddots & \vdots\\ R_{1,k}(u_{l_{\mathrm{smp}}}) & \dots & R_{l,k}(u_{l_{\mathrm{smp}}})\end{bmatrix}\begin{bmatrix}\boldsymbol{X}_1\\ \vdots\\ \boldsymbol{X}_l\end{bmatrix} \quad (31)$$

Afterwards, $\boldsymbol{X}_{1:l}$ is reorganized in a vector and combined with the initial values of placement variables to get the initial guess of $\boldsymbol{x}_0$.

After solving (12), it returns a local minimizer $\boldsymbol{x}$ with residue pose errors due to the limited points for error elimination as indicated in the inequality (19). A refinement by imposing cost function for pose errors on more sampling points is conducted. To generate smooth motion trajectories, a velocity profile along $u \in [0, 1]$ is obtained by B-spline interpolation using Runge Kutta algorithm (Cheng et al., 2002).

5. RESULTS

The method proposed in Section 4 can be validated in manufacturing processes which are imitated by the scenario of two physically coupled robots pulling an elastic band. Fig. 1 illustrates the experimental setup. Fig. 2 visualizes the imitated manufacturing path and the process force. The path is a parallelogram on x-y plane with base 0.05 m, height 0.18 m, and corners rounded with the radius 0.02 m. The process force reaches its maximum of 70 N when the TCP is moved to the top right corner of the path (the yellow point in the zoomed view), 0.23 m from the workpiece origin (labeled as WP in the zoomed view). To imitate a manufacturing process with adjustable workpiece placement, adjustments of the path are allowed in three translational and one rotational (z) directions. The amount and direction of the process force w.r.t. the path (workpiece coordinate) are not changed by these adjustments, because the mounting point of the elastic band on the plattform is changed accordingly.

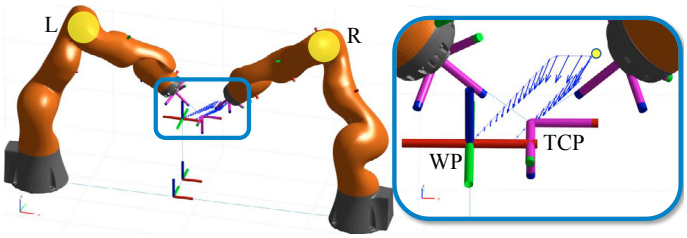

Fig. 2. Visualization of physically coupled robots following a path with process force shown in quivers

5.1 Numerical Experiments

The computational performance of the proposed method (redHess) is compared to some available algorithms in the Matlab function fmincon (interior point and active set), as shown in Table 1. Experiments are conducted both for 7-axis KUKA LBR iiwa 14 R820 robots (KUKA) and 6-axis Comau NJ-290-3.0 robots (Comau).

Table 1: Performance of algorithms

Robots	Algorithm	Computing time [s]	Iteration count	Performance index
KUKA x2	interiorPt	65.99	300	-4.87%
	activeSet	21.73	17	-5.06%
	redHess	65.89	100	-36.82%
Comau x2	interiorPt	50.91	78	-8.27%
	activeSet	38.30	75	-5.11%
	redHess	36.72	75	-59.49%

After around five iterations with less than 9% reduction of performance index, the existing algorithms get trapped in local optima enclosed by the constraints and the objective function. The proposed method helps find a "corridor" toward stiffer solutions with permanent constraint satisfaction. For 6-axis robots, lower dimension of redundant space as in (16) leads to shorter time per iteration.

5.2 Process Experiments

Experiments are conducted with KUKA for two catalogs of cooperating behavior: (I) generated from a given configuration by (31), and (II) after optimization of the first catalog only allowing adjustment of the work-piece placement without changing robot distance. For each catalog, two groups of experiments of coupled robots (a) under process force, and (b) without process force are conducted. We do a subtraction of joint torques between the two goups (a)-(b) to obtain joint torques that are induced by the process force without effects of control errors. TCP displacement and stiffness can be calculated from the identified values in Busson et al. (2017) using equations (1), (5), and (6). The diagrams in Fig. 3 show the joint loads before and after optimization. The loads on the left and right elbows $\tau_{\mathrm{L,elb}}, \tau_{\mathrm{R,elb}}$ are representative to be plotted. Elbows are the fourth joint of each robot, marked in yellow in Fig. 2. The measured (meas.) torques of both elbows are significantly reduced by the optimization. Moreover, the predictions (pred.) of load distribution according to the stiffness model in (6) fit well with the measured data, indicating the applicability of the stiffness model. The 1-norm of measured loads of all 14 joints $\|\boldsymbol{\tau}\|_1$ is plotted in black. The optimization reduces the peak value of $\|\boldsymbol{\tau}\|_1$ by 33% and, by calculation, the peak TCP-deviation by 43%. The diagrams in Fig. 4 show the process induced internal force $\boldsymbol{F}_{\mathrm{int}}$ in the coupler, which is taken from the three

translational entries of $\boldsymbol{w}_{\text{int}} = \boldsymbol{w}_{\text{L}} - \boldsymbol{w}_{\text{R}}$, calculated according to (6). It implies that the reduction of the internal force leads to the overall reduction of $\|\boldsymbol{\tau}\|_1$.

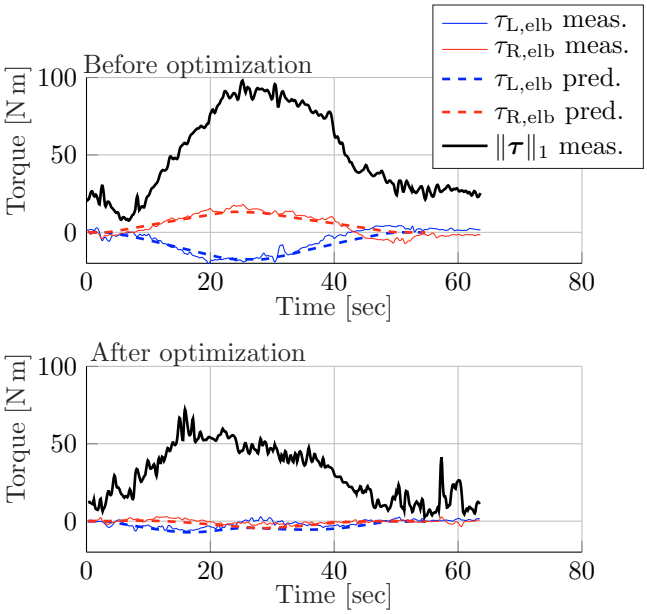

Fig. 3: Joint loads

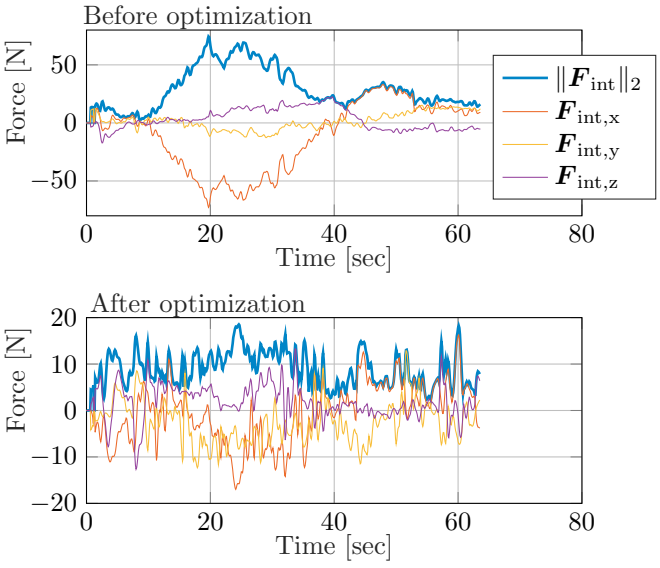

Fig. 4: Internal force $\boldsymbol{F}_{\text{int}}$

6. CONCLUSION AND OUTLOOK

A reduced Hessian method is proposed to generate stiffness-optimized behavior for coupled cooperating robots in manufacturing with the performance surpassing existing algorithms. The optimization results in cooperative behaviors with reduced joint load, reduced process-induced internal force, as well as enhanced stiffness.

An outlook is to control the overall internal force in the coupler caused by the control error when two coupled robots run their individual position controllers. We are developing a hybrid motion-force controller so that the coupled robots can maintain a desired force interaction while tracking the manufacturing path. With an optimized internal force, robots can support each other in a complementary way to enhance precision and manufacturing quality. Moreover, the presented optimization problem is non-convex and exhibits multiple local optima of stiffness. Global search methods with multiple instances of the reduced Hessian algorithm can thus be applied.

REFERENCES

Abele, E., Weigold, M., and Rothenbücher, S. (2007). Modeling and identification of an industrial robot for machining applications. *CIRP Annals*, 56(1), 387–390. doi:https://doi.org/10.1016/j.cirp.2007.05.090.

Biegler, L.T., Nocedal, J., and Schmid, C. (1995). A reduced hessian method for large-scale constrained optimization. *SIAM Journal on Optimization*, 5(2), 314–347.

Busson, D., Bearee, R., and Olabi, A. (2017). Task-oriented rigidity optimization for 7 dof redundant manipulators. *IFAC-PapersOnLine*, 50(1), 14588–14593.

Cheng, M.Y., Tsai, M.C., and Kuo, J.C. (2002). Real-time nurbs command generators for cnc servo controllers. *International Journal of Machine Tools and Manufacture*, 42(7), 801–813.

Chiu, S.L. (1988). Task compatibility of manipulator postures. *The International Journal of Robotics Research*, 7(5), 13–21.

Erhart, S. and Hirche, S. (2015). Internal force analysis and load distribution for cooperative multi-robot manipulation. *IEEE Transactions on Robotics*, 31(5), 1238–1243.

Gabay, D. (1982). Reduced quasi-newton methods with feasibility improvement for nonlinearly constrained optimization. In *Algorithms for Constrained Minimization of Smooth Nonlinear Functions*, 18–44. Springer.

Kabir, A.M., Thakar, S., Malhan, R.K., Shembekar, A.V., Shah, B.C., and Gupta, S.K. (2021). Generation of synchronized configuration space trajectories with workspace path constraints for an ensemble of robots. *The International Journal of Robotics Research*, 40(2-3), 651–678.

Mühlbeier, E., Gönnheimer, P., Hausmann, L., and Fleischer, J. (2020). Value stream kinematics. In *Congress of the German Academic Association for Production Technology*, 409–418. Springer.

Nicola, G., Pedrocchi, N., Mutti, S., Magnoni, P., and Beschi, M. (2018). Optimal task positioning in multi-robot cells, using nested meta-heuristic swarm algorithms. *Procedia CIRP*, 72, 386–391.

Nocedal, J. and Wright, S.J. (1999). *Numerical optimization*. Springer.

Perzylo, A., Rickert, M., Kahl, B., Somani, N., Lehmann, C., Kuss, A., Profanter, S., Beck, A.B., Haage, M., Hansen, M.R., et al. (2019). Smerobotics: Smart robots for flexible manufacturing. *IEEE Robotics & Automation Magazine*, 26(1), 78–90.

Piegl, L. and Tiller, W. (2012). *The NURBS book*. Springer Science & Business Media.

Schmid, C. and Biegler, L.T. (1994). Quadratic programming methods for reduced hessian sqp. *Computers & chemical engineering*, 18(9), 817–832.

Schulz, V.H. (1998). Solving discretized optimization problems by partially reduced sqp methods. *Computing and visualization in science*, 1(2), 83–96.

Slotine, S.B. and Siciliano, B. (1991). A general framework for managing multiple tasks in highly redundant robotic systems. In *proceeding of 5th International Conference on Advanced Robotics*, volume 2, 1211–1216.