New Kinetostatic Criterion for Robot Parametric Optimization

Martin Švejda
Department of Cybernetics
Faculty of Applied Sciences, University of West Bohemia
Pilsen, Czech Republic
e-mail: msvejda@kky.zcu.cz

***Abstract*—The paper deals with a practical approach of parametric optimization of robots. The main idea is to introduce a new criterion which makes possible to evaluate maximum demanded 2-norm of forces/torques of the robot actuators in the case that the end-effector of the robot is to move inside given workspace (specified by the desired positions) with required maximum acceleration in any direction. Therefore, dynamic behaviour of the robot can naturally be taken into account without the need to specify a particular motion trajectory (e.g. moving along the curve with demanded velocity/acceleration profile). Such a criterion is particularly useful in the cases where customer requirements on a new robotic architecture design are too vague and do not include a specific motion of the end-effector of the robot. The new proposed criterion is further used in a minimax discrete optimization problem. Computational effective culling algorithm is used for finding a global optimum. Finally, an illustrative example describes the optimization of the kinematic parameters of the planar parallel robot in order to minimize actuators torques.**

***Keywords*—discrete optimization; parametric optimization; robotics; objective function; robot dynamics**

I. Introduction

Current development in the field of robotics is often devoted primarily to new advanced control algorithms of complex robotic systems. There we can found many algorithms and procedures for the control of robots which are based on advanced centralized actuator control, optimization of standard cascade actuators control, predicitive control, vibration damping, smooth motion trajectory planning, redundancy control for robot motion optimization, etc. Such approaches certainly play a key role in modern robotics and make possible to design powerful control systems which lead to an optimal behaviour of the robot mechanical structure according to the desired motion. On the other hand there still exit elementary open problems which have a major impact on the functioning of the entire robotic systems, regardless of the control system itself. These are the problems of initial mechanical design and its optimization known as the structural and parametric design of the robot architectures. In other words, how to construct robotic systems such a way to meet all requirements and at the same time to minimize complications in associated problems (inverse kinematics computation, control algorithm design, actuator consumption and force/torque and/or velocity load, etc.). Unfortunately, the field of structural and parametric optimization is often neglected which results in crucial problems which can be difficult to solve or may not be solvable at all. For example, structural optimization or synthesis is directly related with the primary problem of robot mechanical construction regarding joints, links and actuators (for parallel robots) deployment. For common industrial robots and their prototyping these problems are often solved based on intuition and experience of the specialists because of high complexity and limited possibility to cope with them through appropriate mathematical formulation and the following solution. The problem of parametric optimization is reduced ”only” on design of convenient kinematic parameters, e.g. Denavit-Hartenberg [1] or Khalil-Kleinfinger [2], in the case where robot joints and links deployment are given. In general, it can be easy to shown that improper design kinematic parameters of the robot can result in poor property of the whole system despite the fact that its structural design is suitable.

The proposed paper deals with a new approach for global parametric robot optimization based on a practical optimization criterion. The paper is divided into the following parts: A brief overview of the parametric optimization algorithms and optimization criteria are summarized in Section II. The basic idea and a formulation of the parametric optimization problem, a global discrete optimization algorithm and a new optimization criterion definition are presented in Section III. Illustrative example of parametric optimization of the parallel planar robot is discussed in Section IV.

II. State of the Art

Parametric optimization of robot architectures can be divided into two key problems. Firstly, the optimal design problem based on an appropriate optimization criterion has to be defined. Secondly, an effective and robust mathematical apparatus should be used for solving the complex optimization problem.

A. Optimization problem definition

A general parametric optimization problem based on maximization of the objective function is given as follows:

$$\begin{aligned} \boldsymbol{\xi}^\star &= \underset{\boldsymbol{\xi}\in\boldsymbol{\Xi}}{\texttt{argmax}}\left(J(\boldsymbol{X}_{\rm opt},\boldsymbol{\xi})\right) \\ \text{w.r.t.:}\quad & \mathbf{Eq}(\boldsymbol{X},\boldsymbol{\xi})=0,\quad \mathbf{Ineq}(\boldsymbol{X},\boldsymbol{\xi})\geq 0 \end{aligned} \tag{1}$$

where J is an objective function, $\boldsymbol{\xi}$ is the vector of optimized kinematic parameters from feasible set Ξ, $\boldsymbol{X}$ is the state of the robot (e.g. position and/or velocity, acceleration of the end-effector) from a required robot workspace $\boldsymbol{X}_{\text{opt}}$, $\mathbf{Eq}$ resp. $\mathbf{Ineq}$ are functions of given equality resp. inequality constraints.

The definition of the objective function J is very important for an efficient optimization process and unreasonable or unusable results can be easily obtained in the case of its incorrect choice. Most of objective functions are based on local properties of the robot [3] given by the kinetostatic duality (2) which can be parametrized by the minimum $\sigma_{\min}$ and/or maximum $\sigma_{\max}$ singular values of the kinematic Jacobian $\boldsymbol{J}(\boldsymbol{Q})$, $\sigma_{\min} \leq \sigma_{\max}$.

$$\dot{\boldsymbol{X}} = \boldsymbol{J}(\boldsymbol{Q}) \cdot \dot{\boldsymbol{Q}}, \quad \boldsymbol{F} = \left(\boldsymbol{J}^T(\boldsymbol{Q})\right)^{-1} \cdot \boldsymbol{\tau} \tag{2}$$

where $\dot{\boldsymbol{X}}$ resp. $\boldsymbol{F}$ is the velocity resp. the force/torque of the end-effector and $\dot{\boldsymbol{Q}}$ resp. $\boldsymbol{\tau}$ are velocities resp. forces/torques of the actuators.

Standard definitions of the objective function J based on kinetostatic duality have the following meaning: Minimization of actuator velocities: $J = \sigma_{\min}$. Minimization of actuator forces/torques: $J = \frac{1}{\sigma_{\max}}$. Dexterity index optimization [4], [5] (compromise between the above criteria): $J = \frac{\sigma_{\min}}{\sigma_{\max}}$ (the most isotropic robot). Considering the most popular dexterity index optimization there are some drawbacks which encourage to use them with care. Inconsistency of the end-effector and joints physical units (e.g. mm, m, deg, rad, etc.) can often lead to irrelevant optimal results. Therefore the normalization/homogenization of the kinematic Jacobian is necessary [6]–[8]. Robustness of the dexterity index is discussed in [9] where the most isotropic point and its variance is taken into account. An alternative approach where the objective function is based on the robot power transmission from the actuators to the end-effector is presented in [10], [11].The proposed objective function gives a relevant measurement independent of the physical units, scale, etc.

Next, the objective function based on kinetostatic performance is generally suitable mainly for comparison of the performance of robots. But it is not always convenient for robots to be used to perform a given task in real engineering processes (especially for moving along a given path or for moving inside a given workspace subject to specific dynamic properties). It is clear that only static properties (dependent on the robot position) can be included in the dexterity index (none requirements on robot dynamic behaviour can be taken into account).

Although many methods for integration of equality and inequality constraints to the optimization process can be found in [12], [13], one of the simply and efficient methods which transforms optimization with constraints to an unconstrained optimization problem is the method of penalty functions which makes possible to add the weighted value of the penalty directly to the objective function [14].

B. *Solving optimization problem*

Assume that all constraints are integrated in the objective function through the penalty method and hence the unconstrained optimization problem has to be solved. There are many methods for unconstrained optimization which are based on gradient methods where the objective function is usually approximated by a suitable (quadratic) model: the steepest descent and the Newton method, the conjugate gradient method, the quasi-Newton method [12]–[14]. The main drawback of the gradient methods is a need to compute the gradient resp. the Hessian of non-trivial (and often non-smooth or discontinuous) objective functions. Therefore, the non-gradient methods play an important role in robot optimization. The most well-known non-gradient methods are the direct search method [15] (e.g. pattern search algorithm, the controlled random search, Mote Carlo methods) or heuristic methods [16] (e.g. genetic algorithms, simulated annealing, particle swarm optimization, gravitational search).

III. New Practical Optimization Method

The new proposed parametric optimization algorithm is based on discretization of an admissible set of kinematic parameters Ξ and a required set of the end-effector positions $\boldsymbol{X}_{\text{opt}}$ (workspace) which results in a discrete optimization problem as follows. Note, that possible constraints are directly integrated into the objective function J through penalization.

$$J^\star(\boldsymbol{X}, \boldsymbol{\xi}^\star) = \max_{i=1\dots N}\left(\min_{j=1\dots M} \boldsymbol{J}_{\text{val}}(j,i)\right)$$
$$i^\star = \operatorname*{argmax}_{i=1\dots N}\left(\min_{j=1\dots M} \boldsymbol{J}_{\text{val}}(j,i)\right), \; \boldsymbol{\xi}^\star = \Xi\{i^\star\} \tag{3}$$

where

$$\boldsymbol{J}_{\text{val}} = \begin{bmatrix} J(\boldsymbol{X}_{\text{opt}}\{1\}, \Xi\{1\}) & J(\boldsymbol{X}_{\text{opt}}\{1\}, \Xi\{2\}) & \dots \\ J(\boldsymbol{X}_{\text{opt}}\{2\}, \Xi\{1\}) & J(\boldsymbol{X}_{\text{opt}}\{2\}, \Xi\{2\}) & \dots \\ \vdots & \vdots & \end{bmatrix}$$

$\boldsymbol{X}_{\text{opt}} = \{\boldsymbol{X}_1 \dots \boldsymbol{X}_M\}$, $\boldsymbol{X}_j \in \mathbb{R}^m$, $\Xi = \{\boldsymbol{\xi}_1 \dots \boldsymbol{\xi}_N\}$, $\boldsymbol{\xi}_i \in \mathbb{R}^p$ where $\boldsymbol{X}_j$ resp. $\boldsymbol{\xi}_i$ represents a discrete point of the workspace (robot positions) resp. one realization (vector) of the kinematic parameters and $J(\boldsymbol{X}, \boldsymbol{\xi})$ is the objective function which minimal value over the workspace is supposed to be maximized (minimax problem in discrete optimization).

A. *Objective function definition*

Suppose that a practically reasonable requirement on an optimization process is to minimize forces/torques in the robot actuators which can be derived from a dynamic model:

$$\boldsymbol{M}(\boldsymbol{Q}) \cdot \ddot{\boldsymbol{Q}} + \boldsymbol{C}(\boldsymbol{Q}, \dot{\boldsymbol{Q}}) \cdot \dot{\boldsymbol{Q}} + \boldsymbol{G}(\boldsymbol{Q}) = \boldsymbol{\tau} - \boldsymbol{J}^T(\boldsymbol{Q}) \cdot \boldsymbol{F} \tag{4}$$

where $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{G}$ are appropriate dynamic matrices/vectors, $\boldsymbol{Q}$ are joints positions, $\boldsymbol{\tau}$ resp. $\boldsymbol{F}$ are joints resp. end-effector forces/torques and $\boldsymbol{J}$ is the kinematic Jacobian. It is clear, see right side of (4) and (2), that only external forces/torques acting on the end-effector can be taken into account when using maximal singular values of the Jacobian (or its dexterity

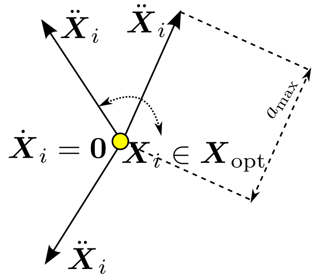

Fig. 1. Robot workspace with the end-effector acceleration definition

index) as the objective function. Assuming that the objective function is given as (penalty function J_{pen} can be selected arbitrarily)

$$J(\boldsymbol{X}, \boldsymbol{\xi}) = \frac{1}{J_{\text{pen}} + \|\boldsymbol{\tau}\|} \tag{5}$$

it can be computed according to (4) only if the complete end-effector trajectory (resp. corresponding joint velocity $\dot{\boldsymbol{Q}}$ and acceleration $\ddot{\boldsymbol{Q}}$) is given. In the case that only corresponding positions $\boldsymbol{Q}$ are known (e.g. only a required workspace is given without a particular trajectory of motion), $\boldsymbol{\tau} = \boldsymbol{G}(\boldsymbol{Q}) + \boldsymbol{J}^T(\boldsymbol{Q}) \cdot \boldsymbol{F}$, and the objective function includes only the effect of the end-effector external forces/torques and gravity acting on the robot links (only static force/torque optimization is assumed).

In order to integrate dynamic behaviour of the robot over the required workspace (without defining a specific trajectory of motion) a new criterion is proposed. Let's assume that the robot is supposed to move from the steady state ($\dot{\boldsymbol{X}} = \boldsymbol{0} \Rightarrow \dot{\boldsymbol{Q}} = \boldsymbol{0}$) in any direction in the required workspace with maximum required acceleration, see Fig. 1.

$$\max_{\boldsymbol{X} \in \boldsymbol{X}_{\mathbf{opt}}} \|\ddot{\boldsymbol{X}}\| = a_{\max}$$

It can be shown by time derivative of velocity dependencies in (2) and assuming $\dot{\boldsymbol{X}} = \boldsymbol{0}$ that

$$\ddot{\boldsymbol{Q}} = \boldsymbol{J}^{-1}(\boldsymbol{Q}) \cdot \ddot{\boldsymbol{X}} \tag{6}$$

and therefore the dynamic equation (4) results in

$$\boldsymbol{M}(\boldsymbol{Q}) \cdot \boldsymbol{J}^{-1}(\boldsymbol{Q}) \cdot \ddot{\boldsymbol{X}} + \boldsymbol{G}(\boldsymbol{Q}) = \boldsymbol{\tau} \tag{7}$$

The restriction on a 2-norm of the joints forces/torques is derived from (7) as

$$\|\boldsymbol{\tau}\| \leq \|\boldsymbol{M}(\boldsymbol{Q}) \cdot \boldsymbol{J}^{-1}(\boldsymbol{Q}) \cdot \ddot{\boldsymbol{X}}\| + \|\boldsymbol{G}(\boldsymbol{Q})\| \tag{8}$$

and from the linear algebra of induced matrix norms it can be easily shown that the maximum 2-norm of joints forces/torques $\tau_{\max}$ for the end-effector to be at the position $\boldsymbol{X}$ (corresponding joint position $\boldsymbol{Q}$) and moving from the steady state in any direction with maximum acceleration $a_{\max}$ is given as:

$$\begin{aligned} \|\boldsymbol{\tau}\| &\leq \sigma_{max}\left(\boldsymbol{M}(\boldsymbol{Q}) \cdot \boldsymbol{J}^{-1}(\boldsymbol{Q})\right) \cdot \|\ddot{\boldsymbol{X}}\| + \|\boldsymbol{G}(\boldsymbol{Q})\| \\ \|\boldsymbol{\tau}\|_{max} &= \sigma_{max}\left(\boldsymbol{M}(\boldsymbol{Q}) \cdot \boldsymbol{J}^{-1}(\boldsymbol{Q})\right) \cdot a_{max} + \|\boldsymbol{G}(\boldsymbol{Q})\| \end{aligned} \tag{9}$$

where $\sigma_{\max}(\star)$ is the maximal singular value of the matrix. Note, that the constant $a_{\max}$ represents a weighting factor between static optimization ($a_{\max} = 0$) and dynamic optimization ($a_{\max} >> 0$, the influence of the gravity is almost neglected, e.g. high speed applications). The new objective function is derived by substituting $\|\boldsymbol{\tau}\|_{\max}$ for $\|\boldsymbol{\tau}\|$ in (5).

B. Optimization algorithm

It is very important to chose an appropriate global solver for the discrete optimization problem (3) especially in the case of early robot design when only a poor idea about optimal kinematic parameters is known. Despite of the computational cost of the discrete optimization problem grows rapidly with an increasing number of discretization points (workspace and admissible set of kinematic parameters) and therefore often it is not possible to obtain highly accurate results, globality of these results (e.g. global optimum of the kinematic parameters in the sense of their discretization) allows to approach to the optimal robot design. The (inaccurate) global results can be further used as initial condition for some local optimization algorithms.

Therefore the branch and bound algorithms for solving the minimax optimization problem based on the culling algorithm [17], [18] was modified and used. It identifies a non-optimal subspace of the admissible set of kinematic parameters Ξ and culls them until only the global optimum remains. The algorithm can be briefly written as follows:

Formulation:

$$\begin{aligned} &\boldsymbol{\xi}^\star = \underset{\boldsymbol{\xi} \in \Xi}{\operatorname{argmax}} \left(\min_{\boldsymbol{X} \in \boldsymbol{X}_{\text{opt}}} J(\boldsymbol{X}, \boldsymbol{\xi}) \right) \text{ or alternatively:} \\ &\boldsymbol{\xi}^\star = \underset{\boldsymbol{\xi} \in \Xi}{\operatorname{argmax}}\, \boldsymbol{\Psi}(\boldsymbol{\xi}),\ \boldsymbol{\Psi}(\boldsymbol{\xi}) = \min_{\boldsymbol{X} \in \boldsymbol{X}_{\text{opt}}} J(\boldsymbol{X}, \boldsymbol{\xi}) \end{aligned} \tag{10}$$

Initialization:

- $\boldsymbol{P}_0 = \Xi$, $\boldsymbol{S}_0(\boldsymbol{\xi}) = 1$, $\boldsymbol{\xi} = \Xi\{i\}$, $\forall i = 1 \dots N$, $\boldsymbol{S}_0$ represents the lowest known value of the objective function $J(\boldsymbol{X}, \boldsymbol{\xi})$ for all admissible set of $\boldsymbol{\xi}$.
- Choose $\boldsymbol{\xi}^0 = \hat{\boldsymbol{\xi}}^0 \in \Xi$, $\hat{\boldsymbol{\xi}}^0$ is the last, so far the best, known candidate for an optimal set of parameters.

Iteration ($i = i + 1$):

1) Finding $\boldsymbol{X}_i \in \boldsymbol{X}_{\text{opt}}$ with the minimum value of the objective function for a given set of parameters:

$$\boldsymbol{X}_i = \underset{\boldsymbol{X} \in \boldsymbol{X}_{\text{opt}}}{\operatorname{argmin}} \left(J(\boldsymbol{X}, \boldsymbol{\xi}^i)\right)$$

2) Updating the candidate for an optimal set of parameters:

$$\hat{\boldsymbol{\xi}}^{i+1} = \begin{cases} \boldsymbol{\xi}^i & \text{pokud } \boldsymbol{\Psi}(\boldsymbol{\xi}^i) > \boldsymbol{\Psi}(\hat{\boldsymbol{\xi}}^i) \\ \hat{\boldsymbol{\xi}}^i & \text{jinak} \end{cases}$$

3) Updating the lowest known value of the objective function for parameter sets (in the point $\boldsymbol{X}_i$):

$$\boldsymbol{S}_{i+1} = \Big[\min_{\boldsymbol{\xi} \in \boldsymbol{P}_i\{1\}} \{\boldsymbol{S}_i(\boldsymbol{\xi}), J(\boldsymbol{X}_i, \boldsymbol{\xi})\}, \\ \min_{\boldsymbol{\xi} \in \boldsymbol{P}_i\{2\}} \{\boldsymbol{S}_i(\boldsymbol{\xi}), J(\boldsymbol{X}_i, \boldsymbol{\xi})\}, \dots \Big]$$

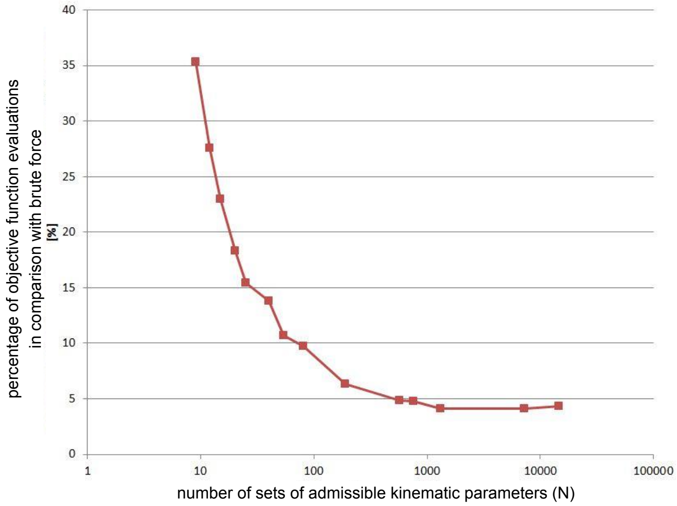

Fig. 2. Computational cost of the culling algorithm ($M = 45$)

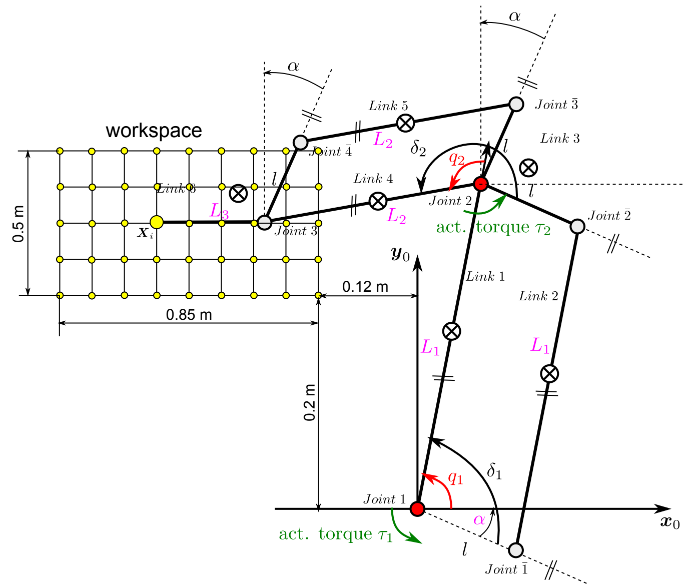

Fig. 3. Optimized robot and its workspace

4) Culling apriori non-optimal sets of parameters:

$$\boldsymbol{P}_{i+1} = \{\boldsymbol{\xi} \in \boldsymbol{P}_i,\ \boldsymbol{S}_{i+1}(\boldsymbol{\xi}) \geq \boldsymbol{\Psi}(\hat{\boldsymbol{\xi}}^{i+1})\}$$

5) Selecting the potentially most suitable set of parameters:

$$\boldsymbol{\xi}^{i+1} = \underset{\boldsymbol{\xi} \in \boldsymbol{P}_{i+1}}{\operatorname{argmax}}(\boldsymbol{S}_{i+1}(\boldsymbol{\xi}))$$

$i = i + 1$ and go to 1).

Finally, all non-optimal parameter sets are culled out and only the global optimal parameter set $\boldsymbol{\xi}^\star$ remains in $\boldsymbol{P}$.

Computational cost of the culling algorithm is shown in Fig. 2 in comparison with a brute force (evaluation of the objective function for all admissible sets of the parameters for all discrete points in the workspace). It can be seen that only about 5 % of evaluations of the objective function is needed for solving the optimization problem with 45 discrete points of the workspace and for more than 1000 elements of admissible kinematic parameter sets.

IV. Illustrative Example

The new practical optimization method is illustrated on kinematic optimization of a 2 DoF parallel planar robot, see Fig. 3.

The kinematic parameters to be optimized (link lengths L_i [m] and parallelogram tilt α [rad]) are:

$$\boldsymbol{\xi}_i = [L_1, L_2, L_3, \alpha]^T \in \boldsymbol{\Xi} = \{\boldsymbol{L}_1 \times \boldsymbol{L}_2 \times \boldsymbol{L}_3 \times \boldsymbol{\alpha}\} \quad (11)$$

$$\boldsymbol{L}_1 = [0.5, 0.52, ..., 0.7]\,,\ \boldsymbol{L}_2 = [0.4, 0.42, ..., 0.6]\,,$$
$$\boldsymbol{L}_3 = [0.1, 0.12, ..., 0.3]\,,\ \boldsymbol{\alpha}_1 = [-0.1, -0.08, ..., 0.1]$$

The workspace of the robot in xy plane is:

$$\boldsymbol{X}_i = [x, y]^T \in \boldsymbol{X}_{\text{opt}} = \{\boldsymbol{x} \times \boldsymbol{y}\} \quad (12)$$

$$\boldsymbol{x} = [-0.12, -0.22625, ..., -0.97]\,,\ \boldsymbol{y} = [0.2, 0.325..., 0.7]$$

The kinematic and dynamic model of the robot and the additional penalty function definition (limitation of parallelograms alignment) for the objective function computation (5, 9) can be found in [19].

The initial kinematic parameters and minimum value of the objective function (resp. norm of joints torques) over the workspace are:

$$\boldsymbol{\xi}_{\text{init}} = [0.6, 0.5, 0.2, 0]^T\,,\ \boldsymbol{\Psi}(\boldsymbol{\xi}_{\text{init}}) = 1.335 \cdot 10^{-3}$$
$$1/\boldsymbol{\Psi}(\boldsymbol{\xi}_{\text{init}}) = \sqrt{\tau_1^2 + \tau_2^2} = 749.1\ [Nm] \quad (13)$$

Note, that the penalty function can only be $J_{\text{pen}} = +\infty$ (for constraints violation) or $J_{\text{pen}} = 0$ (otherwise).

Optimization was performed through the presented algorithm with $a_{\max} = 1$ and it results in the following optimal kinematic robot parameters. The robot workspace and the objective function is shown in Fig. 4.

$$\boldsymbol{\xi}^\star = [0.62, 0.46, 0.28, -0, 1]^T\,,\ \boldsymbol{\Psi}(\boldsymbol{\xi}^\star) = 1.478 \cdot 10^{-3}$$
$$1/\boldsymbol{\Psi}(\boldsymbol{\xi}^\star) = \sqrt{\tau_1^2 + \tau_2^2} = 676.7\ [Nm] \quad (14)$$

It can be seen that the number of the objective function evaluations for a brute force algorithm is $N \cdot M = 11^4 \cdot 45 = 658\,845$ and in the case that the evaluation of the objective function takes $5\,ms$, the total computational time is $54.9\,min$. Only 27 754 evaluations (4.2 %) corresponding to 138 s is needed when the culling algorithm is used.

V. Conclusion

The proposed method for robot kinematic parameters optimization was based on a new type of the objective function (minimizing the norm of the joints forces/torques) which makes possible to integrate the dynamic behaviour of the robot in the case that only robot workspace (demanded end-effector

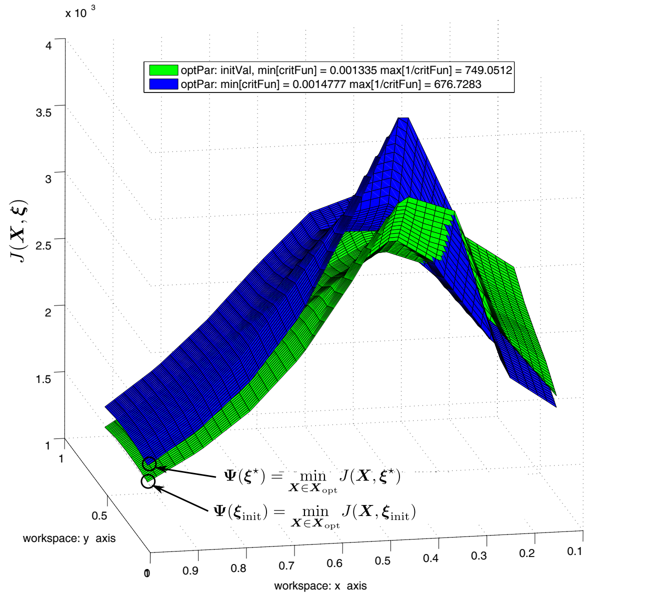

Fig. 4. The objective function values on the robot workspace for initial (green) and optimal (blue) robot kinematic parameters

positions) is known. Only one adjustable parameter (max. acceleration $a_{\max}$ of the end-effector in any direction in the workspace) is used for setting a compromise between static and dynamic optimization. A high effective culling discrete optimization algorithm was used for solving the minimax optimization problem. Further improvement of the algorithm can be found in [20] including finding a second, third, etc. global optimum which makes possible to start the following local optimization algorithm from different initial conditions. Future work will be devoted to the enhancement of the new criterion in such a way that the requirements on non-zero end-effector velocity in any direction inside the robot workspace can be included. As a result, such a criterion will be generally applicable for early robot design for industrial applications where the robot is to work in given workspace with requirement on maximal velocity and acceleration in any direction.

Acknowledgment

The work was supported by the project LO1506 of the Czech Ministry of Education, Youth and Sports and the Center for Intelligent Drives and Advanced Machine Control (CIDAM) TE02000103 from the Technology Agency of the Czech Republic.

References

[1] J. Denavit, R. S. Hartenberg, “A kinematic notation for lower-pair mechanisms based on matrices,” Trans. of the ASME. Journal of Applied Mechanics, Vol. 22, 1955, pp. 215-221, 1955.

[2] W. Khalil, J. Kleinfinger, “A new geometric notation for open and closed-loop robots,” Proceedings. 1986 IEEE International Conference on Robotics and Automation, 1986, pp. 1174-1179.

[3] L. Sciavicco, B. Siciliano, “Modelling and Control of Robot Manipulators,” Springer, ISBN: 978-1-4471-0449-0, January 2000.

[4] S. Khatami and F. Sassani, “Isotropic design optimization of robotic manipulators using a genetic algorithm method,” Proceedings of the IEEE Internatinal Symposium on Intelligent Control, 2002, pp. 562-567.

[5] Jean-Pierre Merlet, “Jacobian, manipulability, condition number and accuracy of parallel robots,” ISRR, Oct 2005, San Francisco, 2005.

[6] K. L. Doty, C. Melchiorri, E. M. Schwartz and C. Bonivento, “Robot manipulability,” in IEEE Transactions on Robotics and Automation, vol. 11, no. 3, pp. 462-468, Jun 1995.

[7] S. Kucuk, Z. Bingul, “Comparative study of performance indices for fundamental robot manipulators,” Robotics and Autonomous Systems, Volume 54, Issue 7, pp. 567-573, 2006.

[8] L. Stocco, S. E. Salcudean and F. Sassani, “Matrix normalization for optimal robot design,” Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146), Leuven, pp. 1346-1351 vol.2, 1998.

[9] R. V. Mayorga, J. Carrera, M. M. Oritz, “A kinematics performance index based on the rate of change of a standard isotropy condition for robot design optimization,” Robotics and Autonomous Systems, Volume 53, Issue 3, Pages 153-163, 2005.

[10] I. Mansouri, M. Ouali, “A new homogeneous manipulability measure of robot manipulators, based on power concept,” Mechatronics, Volume 19, Issue 6, Pages 927-944, 2009.

[11] Imed Mansouri, Mohammed Ouali, “The power manipulability A new homogeneous performance index of robot manipulators,” Robotics and Computer-Integrated Manufacturing, Volume 27, Issue 2, Pages 434-449, 2011.

[12] J. Nocedal, S. Wright, “Numerical Optimization,” Springer-Verlag New York, ISBN: 978-0-387-40065-5, 2006.

[13] A. Ravindran, K. M. Ragsdell, G. Reklaitis, “Engineering Optimization: Methods and Applications, 2nd Edition,” Wiley, ISBN: 978-0-471-55814-9, June 2006.

[14] J. Snyman, “Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms,” Springer US, ISBN: 978-0-387-24349-8, 2005.

[15] T. G. Kolda, R. M. Lewis, V. Torczon, “Optimization by Direct Search: New Perspectives on Some Classical and Modern Methods,” Society for Industrial and Applied Mathematics, ISSN: 1095-7200, 2003.

[16] P. Erdogmus, M. Toz, “Heuristic Optimization Algorithms in Robotics,” chapter in “Serial and Parallel Robot Manipulators - Kinematics, Dynamics, Control and Optimization,” InTech, ISBN: 978-953-51-0437-7, March 2012.

[17] L. Stocco, S. Salcudean, F. Sassani “Fast constrained global minimax optimization of robot parameters,” Robotica, 16(6), pp. 595-605, 1998.

[18] L. Stocco, S. Salcudean, F. Sassani “Mechanism design for global isotropy with applications to haptic interfaces,” 6th Annual Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (The Winter Annual Meeting of the ASME), pp. 115-122, November 1997.

[19] M. Švejda “Kinematical, kinetostatical and dynamical analysis of 4DoF manipulator, parametric optimization of mechanical construction,” research report no. WP5-DV026, University of West Bohemia, Pilsen, Czech republic, 2016.

[20] M. Švejda “Optimization of robot architectures,” dissertation, Faculty of Applied Sciences, University of West Bohemia, Czech republic.