Rotation Optimization on the Unit Quaternion Manifold and its Application for Robotic Grasping

By Philipp Zech, Hanchen Xiong, and Justus Piater

Institute for Computer Science, University of Innsbruck, Tyrol, Austria

Abstract

In this paper we consider black-box optimization of objects' grasp density functions relative to a gripper's orientation. For this, we introduce Monte-Carlo Tree Search on the unit quaternion manifold. Our experimental evaluation shows that our method is feasible and allows finding grasps of arbitrary quality with regard to a fixed budget of objective function evaluations.

1. Introduction

Optimization, either stochastic or numerical, plays a key role in robotics research and engineering. Usually, the objective function f is known, thus application of numerical methods, e.g., gradient search or newtonian methods to find the optimizer x^*, is feasible. The situation however changes if the objective function is not known. In such a case of black-box function optimization, numerical methods are not feasible anymore, hence motivating the application of stochastic methods like sampling or stochastic search.

A typical case of black-box function optimization arises in searching for robotic precision grasps by optimizing the unknown grasp density function $f \colon \mathcal{X} \to \mathbb{R}$ of an object,

$$x^* = \operatorname*{argmax}_{x \in \mathcal{X}} f(x)$$

where x^* denotes the optimal gripper pose and $\mathcal{X}$ the search space that contains all possible gripper poses. Typically, this implies that $\mathcal{X} \subset \mathbb{R}^{12}$ with $x \in \mathcal{X}$ describing both a 3D position in Euclidean space $\mathbb{R}^3$ and a 3D rotation in SO(3) by a 9D column vector, resulting in a 12D column vector. Obviously, optimization in 12 dimensions is not all that efficient.

For searching of robotic precision grasps the number of dimensions can be safely reduced by (i) fixing the gripper's position, and (ii) interpreting rotations in SO(3) as unit quaternions. The latter is the result of mapping SO(3) onto the unit quaternion manifold $\mathcal{M}_{\mathbb{H}}$, where $\mathcal{M}_{\mathbb{H}}$ is the unit 4D hypersphere $\mathcal{S}^3$ embedded in $\mathbb{R}^4$. Thence the dimensionality of $\mathcal{X}$ finally can be reduced to 3 dimensions. Consequently, rotations are treated as points on the unit hypersphere $\mathcal{S}^3$ determined by their hyperspherical coordinates ϕ, ψ, and θ, with ϕ ranging over $[0, 2\pi)$, and ψ and θ over $[0, \pi]$. Fixing the gripper's position is motivated by the intuition that optimal precision grasps usually can be found close to the rims of objects, which is where we fix the gripper's position.

In this paper we address stochastic black-box optimization of an unknown function on the 4D unit hypersphere $\mathcal{S}^3$, i.e., $f \colon \mathcal{S}^3 \to \mathbb{R}$. We apply Monte-Carlo Tree Search (MCTS) to efficiently search for the point $x^* \in \mathcal{X}$ that maximizes the unknown objective function f. To succesfully apply MCTS we introduce a hierarchical and equal decomposition of $\mathcal{S}^3$ along its axis-angles ϕ, ψ, and θ. We finally show how our method can be applied in optimizing robotic precision grasps with the gripper position fixed to some point close to rim of an object.

2. Monte-Carlo Tree Search

Monte-Carlo Tree Search (MCTS) is a recent stochastic optimization method for finding optimal solutions in a given domain by iteratively sampling from the search space $\mathcal{X}$ (Browne *et al.* 2012). During the search, MCTS builds an asymmetric k-ary search tree $\mathcal{T}$ that embodies a hierarchical decomposition of the search space $\mathcal{X}$. At each iteration, the most promising leave $\ell \in \mathcal{T}$ (a distinct subinterval of the search space $\mathcal{X}$) is selected and expanded, that is, ℓ is equally decomposed into a set of leaves $\mathcal{L}$ whose elements are added to the search tree. Upon reaching some computational budget (time, memory, or iteration constraint), MCTS returns with the optimal solution x^*. Algorithm 1 shows the general MCTS algorithm.

Algorithm 1 General MCTS Algorithm

1: **function** MCTS($\mathcal{X}$)
2:      create initial tree $\mathcal{T}_0$ representing the search space $\mathcal{X}$
3:      **while** within computational budget **do**
4:          $\ell = \underset{\ell \in \mathcal{T}_n}{\operatorname{argmax}} f(\ell)$
5:          decompose ℓ into $\mathcal{L} = \{\ell_{h+1, kj+i}\}$
6:          evaluate the elements of $\mathcal{L}$ and add them to $\mathcal{T}_n$
7:      **end while**
8:      **return** $x^* = \underset{x \in \mathcal{T}_n}{\operatorname{argmax}} f(x)$
9: **end function**

---

3. Black-box Rotation Optimization on the Unit Quaternion Manifold

Typically, in black-box function optimization, the objective function f can only be evaluated pointwise. Consequently, it is crucial to choose good points $x \in \mathcal{X}$ during the optimization process, as these points ultimately guide the search towards the optimizer $x^* \in \mathcal{X}$ of f. A common approach at this is to hierarchically decompose the search space $\mathcal{X}$ into partitions p of equal size, i.e., $\mathcal{X} = \{p_1, \ldots, p_n : p_i \cap p_j = \emptyset, i \neq j\}$ to then sample a representative point $x_i \in p_i$, e.g., by taking the mean of a partition p_i, evaluate the objective function at this point and subsequently decide where to continue the search for the optimizer x^* relative to the outcome of $f(x_i)$.

A crucial requirement for that such a decomposition of $\mathcal{X}$ is feasible is that $\mathcal{X}$ and its partitions are equipped with the notion of a "center point". In the special case of SO(3), however, such a notion of a "center point" does not exist. Consequently, in order to perform rotation optimization, an alternative interpretation of a rotation is necessary.

We suggest to map SO(3) onto the unit quaternion manifold $\mathcal{M}_{\mathbb{H}}$ and treat it as the 4D unit hypersphere $\mathcal{S}^3$ embedded in $\mathbb{R}^4$ to render rotation optimization feasible. By decomposing $\mathcal{S}^3$ into partitions of equal size, sampling of representative points x_i that guide the search towards to optimizer x^* is viable.

3.1. *Mapping SO(3) onto the Unit Quaternion Manifold $\mathcal{M}_{\mathbb{H}}$*

The unit quaternion manifold $\mathcal{M}_{\mathbb{H}}$ is defined as a Riemannian manifold where each point $q \in \mathcal{M}_{\mathbb{H}}$ describes a three-dimensional rotation around the origin of the Euclidean space $\mathbb{R}^3$. Thus, there exists a natural mapping between rotations in SO(3) to quaternions $q \in \mathcal{M}_{\mathbb{H}}$,

$$\xi : SO(3) \to \mathcal{M}_{\mathbb{H}}$$

with ξ defined as

$$
\begin{aligned}
q_w &= \frac{\sqrt{1+tr(R)}}{2} \\
q_x &= \frac{R_{2,1} - R_{1,2}}{4 * q_w} \\
q_y &= \frac{R_{0,2} - R_{2,0}}{4 * q_w} \\
q_z &= \frac{R_{1,0} - R_{0,1}}{4 * q_w}
\end{aligned}
$$

where R is a 3×3 rotation matrix and $tr(\cdot)$ denotes the trace of a matrix. By applying ξ on any R one finally arrives at a rotation in quaternion notation,

$$
q = q_w + q_x i + q_y j + q_z k
$$

where i, j, and k are the imaginary parts with the identity

$$
i^2 = j^2 = k^2 = ijk = -1.
$$

As of the one-to-one correspondence between elements of SO(3) and $\mathcal{M}_{\mathbb{H}}$, there also exist an inverse mappings $\xi^{-1} \colon \mathcal{M}_{\mathbb{H}} \to SO(3)$ with ξ^{-1} defined as

$$
\begin{bmatrix}
1 - 2q_y^2 - 2q_z^2 & 2q_x q_y - 2q_z q_w & 2q_x q_z + 2q_y q_w \\
2q_x q_y + 2q_z q_w & 1 - 2q_x^2 - 2q_z^2 & 2q_y q_z - 2q_x q_w \\
2q_x q_z - 2q_y q_w & 2q_y q_z + 2q_x q_w & 1 - 2q_x^2 - 2q_y^2
\end{bmatrix}.
$$

Thus, ξ defines an isomorphic map between SO(3) and $\mathcal{M}_{\mathbb{H}}$.

3.2. *Decomposition of $\mathcal{M}_{\mathbb{H}}$ by Hyperspherical Coordinates*

By virtue of the equality of $\mathcal{M}_{\mathbb{H}}$ and the 4D unit hypersphere $\mathcal{S}^3$ embedded $\mathbb{R}^4$, quaternions $q \in \mathcal{M}_{\mathbb{H}}$ can either be specified by their real and imaginary parts q_w, q_x, q_y and q_z, or by using their corresponding hyperspherical coordinates ϕ, ψ, and θ on the unit hypersphere $\mathcal{S}^3$. Thence, there exists a transformation between these two coordinate systems,

$$
\zeta \colon \mathcal{S}^3 \to \mathcal{M}_{\mathbb{H}}
$$

with ζ defined as

$$
\begin{aligned}
q_w &= \cos(\phi) \\
q_x &= \sin(\phi)\cos(\psi) \\
q_y &= \sin(\phi)\sin(\psi)\cos(\theta) \\
q_z &= \sin(\phi)\sin(\psi)\sin(\theta).
\end{aligned}
$$

The inverse transformation $\zeta^{-1} \colon \mathcal{M}_{\mathbb{H}} \to \mathcal{S}^3$ also exists with ζ^{-1} defined as

$$
\begin{aligned}
\phi &= \frac{q_w}{\sqrt{q_w^2 + q_x^2 + q_y^2 + q_z^2}} \\
\psi &= \frac{q_x}{\sqrt{q_x^2 + q_y^2 + q_z^2}} \\
\theta &= \frac{q_y}{\sqrt{q_y^2 + q_z^2}}.
\end{aligned}
$$

Hence, the search space $\mathcal{X}$ finally is reduced to the 3D space of hyperspherical coordinates ϕ, ψ, and θ of $\mathcal{S}^3$ which can be easily decomposed in a hierarchical manner into an 8-ary tree by recursively splitting at each axis-angle interval's mean. Thus, if $\mathcal{X} = \{\phi, \psi, \theta\}$ with $\phi = [0, 2\pi), \psi = [0, \pi], \theta = [0, \pi]$, after one decomposition of $\mathcal{X}$, we arrive at

$$p_1 = \{[0, \pi), [0, \frac{\pi}{2}], [0, \frac{\pi}{2}]\}, \qquad p_2 = \{[0, \pi), [0, \frac{\pi}{2}], [\frac{\pi}{2}, \pi]\},$$

$$p_3 = \{[0, \pi), [\pi, \frac{\pi}{2}], [0, \frac{\pi}{2}]\}, \qquad p_4 = \{[0, \pi), [\pi, \frac{\pi}{2}], [\frac{\pi}{2}, \pi]\},$$

$$p_5 = \{[\pi, 2\pi), [0, \frac{\pi}{2}], [0, \frac{\pi}{2}]\}, \qquad p_6 = \{[\pi, 2\pi), [0, \frac{\pi}{2}], [\frac{\pi}{2}, \pi]\},$$

$$p_7 = \{[\pi, 2\pi), [\frac{\pi}{2}, \pi], [0, \frac{\pi}{2}]\}, \qquad p_8 = \{[\pi, 2\pi), [\frac{\pi}{2}, \pi], [\frac{\pi}{2}, \pi]\}.$$

4. Application for Robotic Grasping

In this section we apply black-box optimization on $\mathcal{S}^3$ for finding robotic precision grasps of arbitrary quality, where the quality of a grasp ultimately depends on the budget of the MCTS algorithm. As a variant of MCTS we use Bayesian Multi-Scale Optimistic Optimization (Wang *et al.* 2014) which introduces prior knowledge to decide whether to consider a branch of the tree for searching for the optimizer x^* or not. We use the Grasp Wrench Space (GWS) measure as a quality indicator for a grasp (Miller *et al.* 1999).

4.1. *Bayesian Multi-Scale Optimistic Optimization*

Bayesian Multi-Scale Optimistic Optimization (BaMSOO) is a quite recent MCTS algorithm that uses prior knowledge, i.e., $\mathcal{D}_{n-1} = \{(x_1, f(x_1)), \ldots, (x_{n-1}, f(x_{n-1}))\}$, when evaluating the children $\ell_{h+1, kj+i}$ of a selected leaf ℓ (Wang *et al.* 2014), where n denotes the iteration count, h the tree height, and $kj + i$ the index of leaf ℓ at height $h + 1$. For this purpose, Gaussian Processes (GP) are applied to estimate an upper ($\mathcal{UCB}$) as well as a lower confidence bound ($\mathcal{LCB}$) for the children of leaf ℓ, where $\mathcal{UCB}(\ell_{h+1, kj+i} | \mathcal{D}_n) = \mu(\ell_{h+1, kj+i} | \mathcal{D}_n) + B_n \sigma(\ell_{h+1, kj+i} | \mathcal{D}_n)$ and $\mathcal{LCB}(\ell_{h+1, kj+i} | \mathcal{D}_n) = \mu(\ell_{h+1, kj+i} | \mathcal{D}_n) - B_n \sigma(\ell_{h+1, kj+i} | \mathcal{D}_n)$ with $B_n = \sqrt{2log(\pi^2 N^2)/6\eta}$ the confidence, $\eta \in (0, 1)$, and N the leaf counter. $\mu(\cdot)$ denotes the empirical mean, and $\sigma(\cdot)$ the standard deviation of children $\ell_{h+1, kj+i}$ as resulting from the GP after applying the Sherman-Morrison-Woodbury formula (Wang *et al.* 2014). Subsequently, a leaf $\ell_{h+1, kj+i}$ is only evaluated and considered in the next iteration if its $\mathcal{UCB}$ is greater or equal to the highest reward achieved so far. BaMSOO returns the point x^* which optimizes f.

For an in-depth discussion of BaMSOO the reader is referred to Wang *et al.* 2014.

4.2. *Optimization of Robotic Grasps*

We conducted a series of experiments to evaluate the proposed optimization scheme in robotic grasping. All experiments were done on the same machine running with an Intel i7-4550U CPU with 1.5 GHz per core and 8 GB of memory. Further, BaMSOO was implemented in Python. As a simulation environment for executing a grasp and calculating the GWS we used RobWork (Ellekilde *et al.* 2010). All grasps were executed with a Schunk SDH gripper. Figure 1 shows a screenshot from RobWork when simulating a successful grasp with the Schunk SDH gripper applied to a pan.

Figure 2 shows the results of our experiments where we applied the proposed black-box optimization scheme to various objects. For each of these objects, we randomly selected five points (to fix the gripper's spatial location) close to the rims of the objects to find an applicable grasp relative to the gripper's orientation. As can be seen from Figure 2, for each of the chosen points, feasible grasps were found within 300 iterations (the budget for the MCTS algorithm). Despite
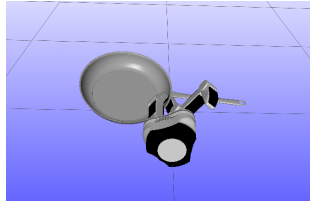
Figure 1: Screen shot from RobWork when simulating a successful grasp with a Schunk SDH gripper applied to a pan.
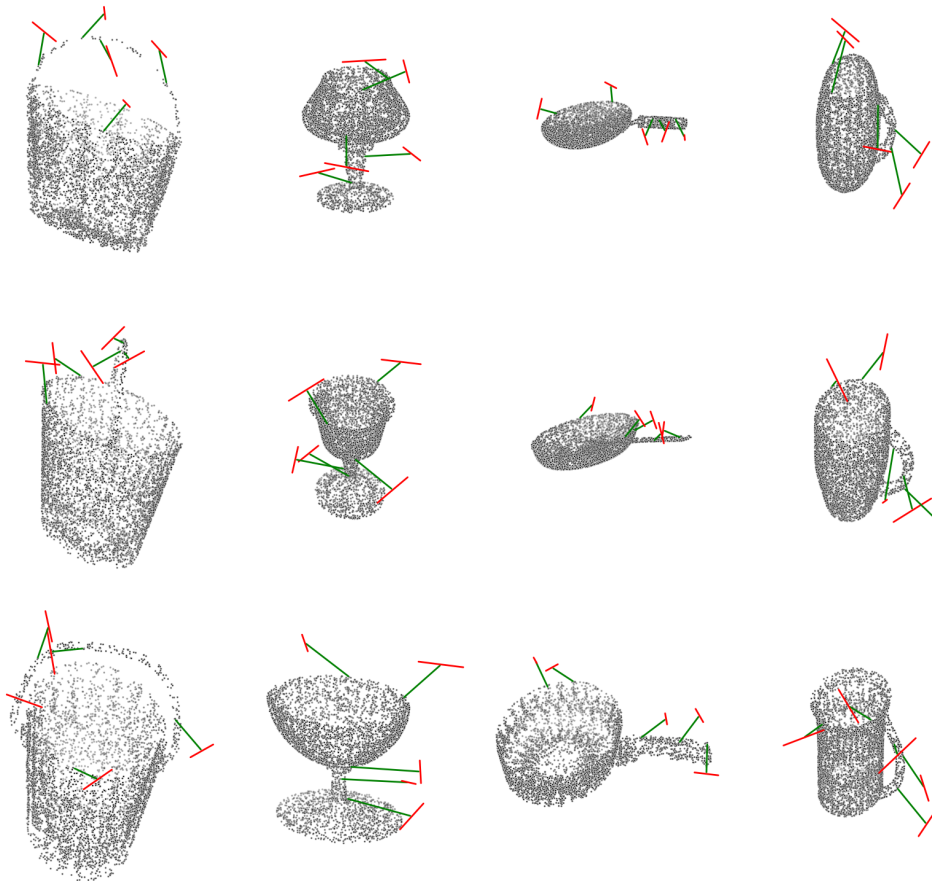

Figure 2: Grasps that resulted from our experiments. Green indicates the spatial orientation of the gripper in the *xyz* plane and red the orientation of the gripper relative to the wrist, where the fingers' span is along the red line (best viewed in color).

which point and object combination we consider, all found grasps seem intuitive from a human's point of view. Especially if we consider the grasps found at the sterns of the various goblets, their orientation is quite reasonable in a sense that a human would grasp a goblet similarly at its stern. Another interesting grasp worth mentioning was found for the third pan (bottom row in Figure 2) at the contact point located on the handle close to the pan itself. From a human's point of view, grasping a pan there is quite intuitive.

One could argue, with respect to efficiency, that MCTS may not be optimized towards the purpose of rotation optimization for robotic grasping and thus maybe surpassed by other, more tailored algorithms. However, we clearly have to point out that the goal of this work was not to invent another grasp optimization algorithm. In contrast, our intention was to show that the use of a general-purpose optimization algorithm already allows us to achieve good results given proper parameterization of the optimization algorithm.

We want to conclude our application case study with two final remarks. Firstly, for our experiments we used a parallel jaw gripper with one actuated degree of freedom. Consequently, our method smoothly transforms to different grippers, given that such grippers are actuated by one degree of freedom. The use of grippers with more degrees of freedom however would raise open questions regarding optimization of joint and torque values, which we do not address in this paper. Secondly, we have only considered optimization of a gripper's rotation in $\mathbb{R}^3$, but not its position which is simply due to that positions and orientations lie in different groups ($E(3)$ and $SO(3)$) which are not trivially unifiable.

5. Related Work

In face of related work to our research as discussed in the following we focus on both, black-box rotation optimization (Section 5.1), and optimization of robotic grasps (Section 5.2). However, we will not focus on research that addresses grasp learning, as this clearly differs from our research in that our work focuses on optimizing grasps but not learning them from scratch.

5.1. *Black-box Rotation Optimization*

Rotation optimization has been studied in many disciplines, however, most of existing work only addresses the case where the analytic form of an objective function is given and thus numerical methods can be employed by exploiting its properties, e.g., convexity or local convexity. To the best of our knowledge, rotation optimization in the face of black-box functions was rarely touched. For general black-box function optimization, there exist several popular strategies, for instance, tree-based search, response surface methods, evolutionary algorithms and simulation-based search (Rios and Sahinidis 2013). However, applying these methods on rotation optimization is not straightforward, as the feasible search space is constrained within a manifold. Therefore, in this sense, our work is also related to constrained optimization (Gardner *et al.* 2014). Our stochastic method, as a matter of fact, is a combination of tree-based search, response surface methods and simulation-based search on the unit quaternion manifold.

A closely related work was presented by Papazov and Burschka (2011), where an optimal rotation is searched for the purpose of point cloud registration. Their work however differs from ours in several aspects. Firstly, the search space is SO(3) instead of the unit quaternion manifold. Secondly, their decomposotion of SO(3) results in spherical boxes and thus requires uniform sampling in these boxes. In contrast, our hierarchical partitioning based on the hyperspherical coordinates of $\mathcal{S}^3$ is much easier. Thirdly, their method is a stochastic method that combines simulated annealing with tree-based sampling while our approach utilizes more heuristics (the principle of optimism in the face of uncertainty (Munos 2011) and Bayesian optimization) to exploit the smoothness of the objective function to keep the iteration budget of MCTS low.

5.2. *Robotic Grasp Optimization*

Cordella *et al.* (2012) discuss a bio-inspired method for optimization of power grasps. Their method is based on analyzing experimental evidence as a result from observation of human behavior to subsequently define a suitable objective function relative to finger trajectories for the optimization of a power grasp. Jameson and Leifer (1987) introduce a method for locating stable grasps in accordance to a grasp goal function. Finding a most stable grasp subsequently is

treated as an optimization problem, where the goal function incorporates physical constraints like torque and motion limitations. Yu *et al.* (1998) investigate the optimization of power grasps. Their method considers satisfying a required external force set, i.e., the set of external forces which the power grasp must contain. The work of Yoshikawa *et al.* (2001) is in the same direction as in case of Yu *et al.* (1998), however, Yoshikawa *et al.* (2001) consider grasping of multiple instead of a single object. Again, the optimization objective is to find the optimal power grasps that satisfies the set of required external forces. Youshen *et al.* (2004) consider grasping-force optimization as a problem of minimizing and objective function subject to form-closure and balance constraints of external forces. Their optimization procedure searches for a set of contact forces that allow to hold an object stable and at the same time compensate external forces. For this, they use a recurrent neural network with global convergent properties which finds optimal grasping forces in real time. Yun-Hui (1999) also addresses the optimization of form-closure grasps, however, in his work, the optimization problem is interpreted as linear programming problem. The underlying intuition of his method is to query whether the wrench space lies within the convex hull of the primitive contact wrenches. If so, an optimal grasp has been found.

It is worth mentioning that in our discussion of related work for robotic grasp optimization existing work only deals with the problem of numeric optimization, that is, the objective function f is always known. Our work thus is innovative in the sense that it considers stochastic optimization of grasps where the objective function f is unknown.

6. Conclusion

In this paper we have introduced MCTS on the unit quaternion manifold $\mathcal{M}_{\mathbb{H}}$ for black-box rotation optimization in SO(3). We have established a hierarchical decomposition of a search space $\mathcal{X}$ by the 4D unit hypersphere $\mathcal{S}^3$ embedded in $\mathbb{R}^4$. We further have successfully applied our black-box optimization method in the context of optimizing robotic precision grasps. We believe that the research presented in this paper offers an efficient method for black-box rotation optimization in SO(3) that can be applied to different problems in robotics like camera pose estimation, as well as pose estimation in general, but also in different fields of engineering where one is faced with an unknown objective function to be optimized that can only be evaluated point-wise.

Acknowledgments

REFERENCES

BROWNE, C. B. & POWLEY, E. & WHITEHOUSE, D. & LUCAS, S. M & COWLING, P. I. & ROHLF-SHAGEN, P. & TAVENER, S. & PEREZ, D. & SAMOTHRAKIS, S. & COLTON, S. 2012 A Survey of Monte-Carlo Tree Search Methods. In *IEEE Transactions on Computational Intelligence and AI in Games* **4**, 1–43.

CORDELLA, F.. & ZOLLO, L. & GUGLIELMELLI, E. & SICILIANO, B. 2012 A Bio-inspired Grasp Optimization Algorithm for an Anthropomorphic Robotic Hand. In *International Journal on Interactive Design and Manufacturing (IJIDeM)* **6**, 113–122.

ELLEKILDE, L. P. & JORGENSEN, J. A. 2010 RobWork: A Flexible Toolbox for Robotics Research and Education. In *41st International Symposium on Robotics (ISR) and 6th German Conference on Robotics (ROBOTIK)*, 1–7.

Rotation Optimization on the Unit Quaternion Manifold and its Application for Robotic Grasping 8

GARDNER, J. & KUSNER, M. & WEINBERGER, Q.K. & CUNNINGHAM, J. & XU, Z. 2014 Bayesian Optimization with Inequality Constraints. In *Proceedings of the 31st International Conference on Machine Learning* , 937–945.

JAMESON, J. W. & LEIFER, L. J. 1987 Automatic Grasping: An Optimization Approach. In *IEEE Transactions on Systems, Man and Cybernetics* **17**, 806–814.

MILLER, A T. & ALLEN, P. K. 1999 Examples of Grasp Quality Computations. In *IEEE Proceedings of the International Conference on Robotics and Automation* **2**, 1240–1246.

MUNOS, R. 2011 Optimistic Optimization of Deterministic Functions Without the Knowledge of its Smoothness. In *Advances in Neural Information Processing Systems* **24**, 783–791.

PAPAZOV, C. & BURSCHKA, D. 2011 Stochastic Global Optimization for Robust Point Set Registration. In *Computer Vision and Image Understanding* **12**, 1598-1609.

RIOS, L. M. & SAHINIDIS, N. V. 2013 Derivative-free Optimization: A Review of Algorithms and Comparison of Software Implementations *Journal of Global Optimization* **3**, 1247-1293.

WANG, Z. & SHAKIBI, B. & JIN, L. & DE FREITAS, N. 2014 Bayesian Multi-Scale Optimistic Optimization. In *Artificial Intelligence and Statistics (AISTATS)*, 1005–1014.

YOSHIKAWA, T. & WATANABE, T. & DAITO, M. 2001 Optimization of Power Grasps for Multiple Objects. In *Proceedings of the IEEE International Conference of Robotics and Automation* **2**, 1786–1791.

YOUSHEN, X. & JUN, W. & LO-MING, F. 2004 Grasping-force Optimization for Multifingered Robotic Hands Using a Recurrent Neural Network. In *IEEE Transactions on Robotics and Automation* **20**, 549–554.

YUN-HUI, L. 1999 Qualitative Test and Force Optimization of 3-D Frictional Form-closure Grasps Using Linear Programming. In *Proceedings of the IEEE International Conference of Robotics and Automation* **15**, 163–173.

YU, Y. & TAKEUCHI, K. & YOSHIKAWA, T. 1998 Optimization of Robot Hand Power Grasps. In *Proceedings of the IEEE International Conference of Robotics and Automation* **4**, 3341–3347.