Bimanual Grasp Planning

Nikolaus Vahrenkamp, Markus Przybylski, Tamim Asfour and Rüdiger Dillmann
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT)
Adenauerring 2, 76131 Karlsruhe, Germany
Email: {vahrenkamp,markus.przybylski,asfour,dillmann}@kit.edu

***Abstract*—The ability to grasp large objects with both hands enables bimanual robot systems to fully employ their capabilities in human-centered environments. Hence, algorithms are needed to precompute bimanual grasping configurations that can be used online to efficiently create whole body grasps. In this work we present a bimanual grasp planner that can be used to build a set of grasps together with manipulability information for a given object. For efficient grasp planning precomputed reachability information and a beneficial object representation, based on medial axis descriptions, are used. Since bimanual grasps may suffer from low manipulability, caused by a closed kinematic chain, we show how the manipulability of a bimanual grasp can be used as a quality measure. Therefore, manipulability clusters are introduced as an efficient way to approximatively describe the manipulability of a given bimanual grasp. The proposed approach is evaluated with a reference implementation, based on Simox [1], for the humanoid robot ARMAR-III [2]. Since the presented algorithms are robot-independent, there are no limitations for using this planner on other robot systems.**

I. Introduction

Thanks to progress in medicine and health care, people in our society today can expect to live longer than in the past. In order to allow people to lead independent lives despite declining physical strength, humanoid service robots capable of grasping and manipulating all sorts of objects need to be developed. Easily accomplished by humans, grasping is a quite difficult task for robots, as it requires consideration of a robot's kinematics, an object's geometric and physical properties, and also obstacles and forces.

Recently, many humanoid robot platforms have been introduced to the community, for example HRP2 [3], ARMAR [4], Dexter [5], Justin [6], or NASA Robonaut [7]. Research conducted on these platforms includes grasping and manipulation, dealing with daily objects, furniture, doors and also preparatory manipulation to achieve easier grasping [8].

When it comes to grasp planning, research in the recent years has mainly focused on approaches based on popular simulation environments such as GraspIt! [9], OpenRAVE [10] and Simox [1]. A whole family of approaches uses different heuristics to generate grasp candidates based on the object's known geometry. The first approach in this context was presented by Miller et al. [11] who manually decomposed objects into boxes, spheres, cylinders and cones in order to plan grasps on the individual primitives. Goldfeder et al. [12] used trees of superquadrics to represent objects. Huebner et al. [13] performed shape approximation using only minimum volume bounding boxes. Guided by the same idea, but aiming at improved shape approximation accuracy, we proposed a method using the medial axis for grasp planning [14], which we recently optimized in [15]. Also recently, Aleotti et al. [16] suggested to use the Reeb Graph for grasp planning. While the majority of the references cited above focused on the robot's hand and the object, some approaches also considered arm kinematics. Berenson et al. presented a method based on a grasp scoring function combining grasp quality, kinematic reachability and environmental obstacles (see [17], [18]). In [19] an approach is presented where the planner determines single-handed grasps that can be used to find grasps for a second hand. In our previous work [20] we investigated an approach integrating the search for a feasible grasp and the search for collision free grasping motions. The only work in the area of bimanual grasp planning we are aware of was presented in [18], where two-handed grasping was regarded as a special case of single handed grasping. The two open hands were considered as the fingers of one virtual big hand and the kinematic chain connecting the two hands was treated as a virtual wrist. Inspired from that work, we present an approach for planning bimanual grasps, where the manipulability of the configuration is considered as a quality measure of the grasp. Therefore we introduce the concept of manipulability clusters, which are volumes, in which the connection of both arms can maneuver while both hands remain at their grasping positions.

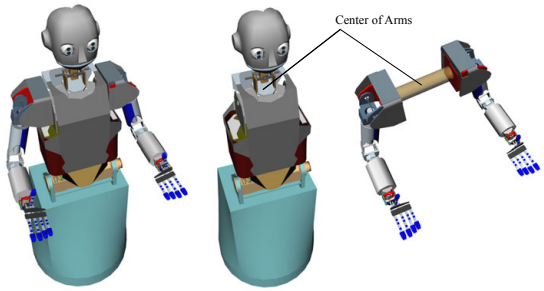

Fig. 1. The humanoid robot ARMAR-III (left) and the decomposition into two main kinematic structures (center and right).

II. Kinematic Structure of Two-Armed Mobile Manipulators

When considering bimanual grasping tasks, the kinematic structure of a two-armed mobile manipulator (e.g. a humanoid

robot) can be divided into two main parts. One part covers all arm and hand joints, representing the manipulation capabilities of the robot. Positioning and pose control of the robot are considered by the remaining joints. In Fig. 1 the proposed division of the kinematic structure is shown for the humanoid robot ARMAR-III.

A. Reachability analysis

By dividing the kinematic structure as described above, the dual-arm manipulability can be independently analyzed in an offline step in order to support IK-queries in the online phase. Therefore, the reachability of both hands with respect to each other is approximated by a discretized 6D (position and orientation) reachability volume. This representation consists of a set of 6D voxels, which hold a value, that is related to the probability that an IK-solution exists [21], [22]. In [23] we showed how hand-over motions can be efficiently planned supported by precomputed reachability information. Reachability information for dual-arm tasks is also considered in [24] to analyze the task structure and map this data onto subordinate planners.

The buildup of the reachability representation can be realized by two approaches: Either the task space is sampled and IK-solutions are queried for each 6D voxel or joint values are sampled randomly and by computing the corresponding tool center point (TCP), the related 6D voxel can be determined. The latter method was used for building the reachability representations for both kinematic chains that were used in the experiments.

1) Arm Movements: In this work, the kinematic chain covering the arm joints is considered, whereas one hand is regarded as the base joint and the kinematic chain to the other hand is used for reachability analysis. To achieve this, the joint and link definitions of the first arm have to be inverted, so that a valid kinematic chain can be constructed. In Fig. 2 (left) a cut through the 3D visualization of the reachability data for the kinematic chain from left to right TCP is shown. The color intensity is proportional to the probability that there exists an IK-solution for the right hand with the left hand at the current position.

2) Robot Pose: Further, a reachability representation of the kinematic chain going from the robot's base position to the center of the arms is generated in order to quickly decide if a bimanual grasp is reachable (see Fig. 2 (right)).

B. Manipulability

The manipulability of a configuration q can be expressed in different ways. A widely used manipulability measure (also called manipulability index) was described by Yoshikawa in [25]. The index $m(J)$ depends on the Jacobi Matrix J and it's transposed:

$$m(J) = \sqrt{det(JJ^T)} \tag{1}$$

Equation 1 can be generalized (see e.g. [26]) to:

$$m(J) = (\lambda_1, \ldots, \lambda_n)^{\frac{1}{2}}, \tag{2}$$

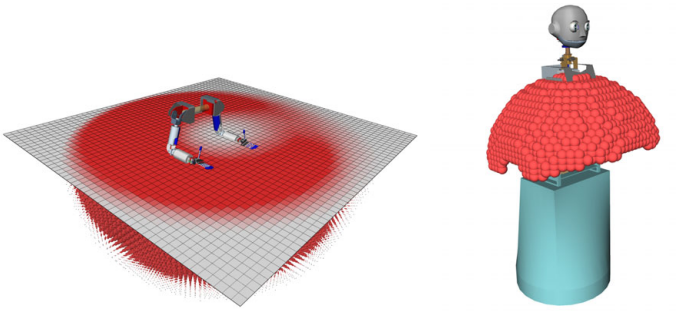

Fig. 2. A cut through the reachability representation of the 14 DoF kinematic chain going from the left to the right hand of ARMAR-III. (left). The reachability of the 4 DoF kinematic chain that goes from the platform yaw joint over the three hip joints and ends in the fixed coordinate system *Center of Arms* (right).

where λ_i denotes the i-th eigenvalue of JJ^T. Another possibility to measure the manipulability is to consider the smallest eigenvalue λ_n, that describes the distance from a singular configuration in the direction of the corresponding eigenvector. The condition number

$$c(J) = \frac{\lambda_1}{\lambda_n} \tag{3}$$

is defined as the ratio of the smallest to the largest eigenvalue. Further details can also be found in [27].

The manipulability of a closed kinematic chain, that is formed when a two arm system is grasping an object with both end-effectors, is addressed in [26]. In section IV-E we introduce the concept of the manipulability clusters, which are approximated with 6D ellipsoids in which the *Center Of Arms* (CoA) joint can maneuver while holding the object with both end-effectors. This measure gives a good hint from which positions the object can be grasped for a given bimanual grasp and further the allowed movements of the closed kinematic chain are described.

III. Object Representation

In this section we describe the object representation and the method we use to generate grasps. Both object representation and grasp planning algorithm are motivated by our assumption that grasps for objects can be found more easily if the object representation used by the grasp planner reflects local symmetry properties of the object's geometry.

A. Medial-axis analysis

In this work we use the *grid of medial spheres* object representation we previously introduced in [15] and which is based on the medial axis transform (MAT) [28]. The MAT is a complete shape descriptor, i.e. it is capable of representing objects of arbitrary shapes. Construction of an object's MAT is achieved by inscribing spheres of maximum diameter into the object's shape. Each of these maximally inscribed spheres has to touch the original object's surface at two or more different points. As a result, the object can be described by a collection of inscribed spheres, where each sphere has - apart from its

center's position - two parameters: the sphere radius and the object angle.

- The sphere radius r describes the thickness of the object at this specific point.
- The object angle α_o [29] denotes the maximum angle included by two vectors from the sphere's center to two different nearest surface points of the original shape touched by this sphere.

Sphere radius and object angle of a inscribed sphere determine how much the respective sphere contributes to the object's shape. Spheres with small object angles and small radii describe edges and corners and other surface details of the object, while spheres with big object angles and big radii rather describe the volume occupied by the object (see Fig. 3). We sort the inscribed spheres into a three dimensional grid structure with respect to the Cartesian coordinates of their sphere centers and thereby obtain the *grid of medial spheres*. The advantage of this grid structure is that we can efficiently access the spheres in a local neighborhood around a given query sphere, simply by determining the grid cell containing the query sphere and considering the spheres in neighbor grid cells. We will exploit this benefit with our grasp planning algorithm which we describe in the next section.

B. Grasp Candidates for bimanual manipulation

In the following, we explain the core of our grasp planning method we introduced in [15]. A grasp candidate consists of an approach point P_g, an approach vector P_d and a hand orientation vector P_o. The approach point is the point the robot's hand approaches during grasping. The approach vector specifies the direction from which to approach the object. Finally, the hand orientation vector describes an imaginary axis the hand wraps around during closing, on the assumption that the closing process starts from a parallel hand preshape. Using our grid of medial spheres object representation described in the previous section, we are able to extract geometrically meaningful grasp candidates directly from the object's geometry. We consider only spheres with object angles $\alpha_{o_i} \geq 120°$ for grasp planning, as this removes surface details of the object but preserves the object's basic local symmetry properties such as local symmetry planes and local symmetry axes. For all the remaining spheres we determine whether their centers are located on a local symmetry axis, on the rim of a local symmetry plane or inside a local symmetry plane. For each single query sphere, we achieve this by analyzing the sphere centers in a local neighborhood with search radius r_s around the query sphere. Using our grid representation, we can efficiently determine these sphere centers. In order to extract grasp candidates, we perform principal components analysis (PCA) on the sphere centers. Let λ_1, λ_2 be the first two eigenvalues and $\vec{e}_1, \vec{e}_2$ the first two eigenvectors from the PCA. In order to decide whether the object's shape in the neighborhood of the query sphere is rather oblate or rather elongate, we compute the ratio ρ_{ev} of λ_1 and λ_2:

$$\rho_{ev} = \frac{\lambda_2}{\lambda_1} \tag{4}$$

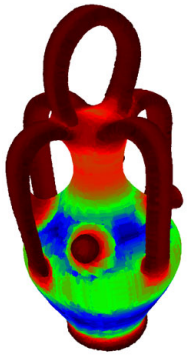

Fig. 3. A MAT-based representation of an object. The color of the inscribing spheres is proportional to the radius (small values are shown in red, large radii are encoded in blue).

For very small values ($\rho_{ev} \leq \rho_{axis}$) we assume the query sphere to be located on a local symmetry axis. For moderate values ($\rho_{axis} \leq \rho_{ev} \leq \rho_{plane}$) we assume the query sphere to be located at the rim of a local symmetry plane. For big values ($\rho_{ev} > \rho_{plane}$) we classify the sphere as located inside a local symmetry plane. For all experiments in this paper, we used the values $\rho_{axis} = 0.01$ and $\rho_{plane} = 0.4$. We generate grasp candidates depending on the results of the PCA. In case of a local symmetry axis, we choose $P_o = \vec{e}_1$. Possible approach directions P_{d_i} are all vectors perpendicular to the direction of the symmetry axis $\vec{e}_1$. In case of a local symmetry plane, we choose $P_o = \vec{e}_1$. We compute the direction P_d of the hand from $\vec{e}_2$. In order to ensure that the hand's approach vector is directed toward the rim from the outside of the local symmetry plane, we evaluate the vector $\vec{v}_{COG}$ from the query sphere's center to its local neighborhood's center of gravity COG_N. If $\vec{e}_2$ and $\vec{v}_{COG}$ include an angle $\beta \geq 90°$, then we set $P_d = -\vec{e}_2$, otherwise $P_d = \vec{e}_2$. For all grasp candidates, we use the query sphere's center as the approach point P_g. There is one further condition for the approach point. We state that, especially for big objects, it is necessary to carefully select the places where to grasp the object, as for successful grasping the hand must be able to close around the object. This is not the case if the object's shape is too thick at the respective point. Yet, the radii of the inscribed spheres in our grid of medial spheres object representation indicate whether the hand can wrap around the object at the respective point. Spheres with radii to big for the hand are not considered for grasp candidate generation.

The grasp candidates generated by our method so far are only grasps for one hand. Bimanual grasps can be constructed by combining two of these grasp candidates, where additional requirements have to be met. The construction of bimanual grasps is described in the following.

IV. Bimanual Grasp Planning

Performing bimanual grasp planning requires to consider multiple tasks, such as finding valid TCP configurations for both end-effectors, determining the multi-hand grasp quality and ensure that the kinematic structure of the arms allows to reach both grasping positions. In this work we propose a

planner that uses an object representation based on its medial axis together with reachability analysis to achieve efficient planning as shown in Algorithm 1. The resulting bimanual solutions cover the TCP positions of both hands together with an exemplary IK-solution for the kinematic chain of the arms. This guarantees that at least one IK-solution exists and that both TCP positions are reachable. The IK-solution is locally optimized, in a way that the manipulability of the arm joints is maximized. Additionally to this exemplary IK-solution, the size of an manipulability cluster is computed, that approximatively describes the spatial positions of the *Center of Arms* joint for which an IK-solution exists. This information can be used during an online query to efficiently compute whole body IK-solutions.

Algorithm 1 $PlanBimanualGrasp(GC, RI)$

Input: A set of grasp candidates GC and reachability information RI.

Output: Bimanual grasping information with IK-solution for arm joints and manipulability cluster.

1: $gc_1 \leftarrow GC.RandomCandidate()$
2: $p_1 \leftarrow ComputeEEFPose(gc_1)$
3: $GC_2 \leftarrow ReachableCandidates(p_1, RI, GC)$
4: **while** $(|GC_2| > 0)$ **do**
5: $gc_2 \leftarrow GC_2.RandomCandidate()$
6: $GC_2.RemoveCandiate(gc_2)$
7: $p_2 \leftarrow ComputeEEFPose(gc_2)$
8: $IK \leftarrow SearchIKSolution(p_1, p_2)$
9: **if** $(IK$ && !$Collision(IK))$ **then**
10: $cp_1 \leftarrow ContactPoints(CloseHand_1(p1))$
11: $cp_2 \leftarrow ContactPoints(CloseHand_2(p2))$
12: **if** $(ForceClosure(cp_1 \bigcup cp_2))$ **then**
13: $IK \leftarrow OptimizeManipulability(IK, p_1, p_2)$
14: $rc \leftarrow ManipulabilityCluster(IK, p_1, p_2)$
15: **if** $(rc.volume > 0)$ **then**
16: **return** (gc_1, gc_2, IK, rc)
17: **end if**
18: **end if**
19: **end if**
20: **end while**
21: **return** NULL

A. Finding a grasp candidate for one end-effector

As described in section III-B, grasp candidates for one end-effector can be easily extracted from the medial axis representation of the object (see Fig. 4 (left)). The position of the first end-effector is determined by randomly choosing a grasp candidate that does not result in a collision with the hand when it is in an opened shape (see Algorithm 1 line 1).

B. Finding grasp candidates for the other end-effector

In line 3 of Algorithm 1 all grasps candidates are filtered regarding the reachability information (see section II-A1), so that only the reachable grasp hypotheses remain for further investigation (see Fig. 4 (right)). Due to the discretized structure of the reachability information, we just can sort out candidates that are most likely not reachable and we have to check the reachability for the second end-effector by determining an IK-solution. Since this IK-solution is also needed in following steps, no computational overhead is produced by this IK-search. In order to realize a robot independent IK-solver, we use a gradient descent method based on the Jacobian's Pseudo-Inverse, that does not rely on any analytic analysis of the kinematic structure. Since this approach may get stuck in local minima, there might be IK-solutions that are overseen and thus potential bimanual grasps are not found by this method. By producing lots of samples and randomly varying the starting configuration of the IK-search, the effects of local minimas can be reduced. When an IK-solution for the arms can be determined (Algorithm 1 line 8), the result is checked for collisions and the bimanual grasp is analyzed as described in the next section.

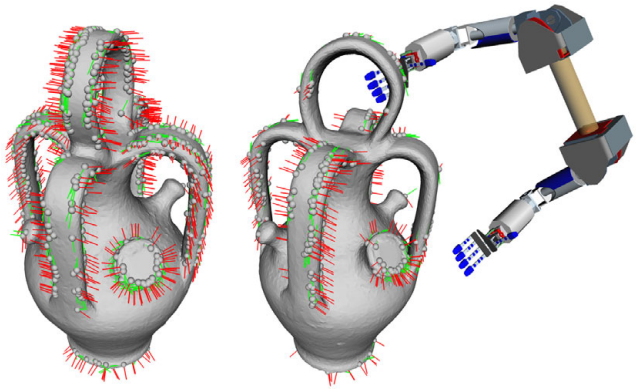

Fig. 4. A set of 300 grasp candidates with corresponding approach direction depicted as red lines (left). Reachability information is used to determine the set of reachable candidates for the left hand (right).

C. Grasp Quality

Both end-effectors are closed in line 10 and 11 and the contact points on the object's surface are determined. The quality of the resulting set of contacts is then computed by performing Grasp Wrench Space analysis. We use the standard ϵ measure for force-closure as described in [30], where the six dimensional convex hull of the contact points and normals is computed and the distance from the origin of the Grasp Wench Space to the nearest facet of the convex hull indicates how well the grasp can resist disturbances by externally applied wrenches. In case a force closure grasping configuration can be verified, the solution is computed as described in the following sections.

D. Maximizing the Manipulability

To represent the manipulability of a two-arm system we independently consider the manipulability index for the left and the right arm, since we are interested in flexible configurations that allow to move both TCPs as good as possible. This measure is maximized as shown in Algorithm 2: The Jacobian for the joints of the arm that belongs to the first

end-effector is denoted with J_1 and the Jacobian J_2 covers the other side of the bimanual kinematic chain. In line 7 all possible combinations Δ_x of discrete steps for moving the *Center of Arms* joint are built. Note, that the step size δ is the same for translational and rotational components of the movements in order to preserve the readability. For each of those Δ_x Jacobian-based movements are performed. First the kinematic chain from the first end-effector to the CoA Joint is considered (line 9). When the joint-limits allow a movement in the given direction, it is tried to move the second end-effector back to it's grasping pose p_2. When the kinematic structure allows such movements and the manipulability of both arms could be increased, a better IK solution was found (line 14-17). This procedure is repeated until the local maximum is reached and the corresponding joint values are returned (line 23). An exemplary result of this optimization process can be seen in Fig. 5.

Algorithm 2 $OptimizeManipulability(IK, p_1, p_2)$

Input: A configuration IK for both arms and the Cartesian positions p_1, p_2 of both end-effectors.

Output: The local maximum of the dual arm manipulability.

1: $p_{CoA} \leftarrow ForwardKinematics(IK)$
2: $m_1 \leftarrow \sqrt{det(J_1 J_1^T)}$
3: $m_2 \leftarrow \sqrt{det(J_2 J_2^T)}$
4: $IK_{best} \leftarrow IK$
5: **while** $(!TimeOut())$ **do**
6: $\quad \Delta_{best} \leftarrow \{0,0,0,0,0,0\}^T$
7: $\quad$ **for all** $(\Delta_x \in \{-\delta, 0, \delta\}^6)$ **do**
8: $\quad\quad p'_{CoA} \leftarrow MoveCartesianPose(p_{CoA}, \Delta_x)$
9: $\quad\quad$ **if** $(MoveJointToPose(J_1, p'_{CoA}))$ **then**
10: $\quad\quad\quad$ **if** $(MoveJointToPose(J_2, p_2))$ **then**
11: $\quad\quad\quad\quad m'_1 \leftarrow \sqrt{det(J_1 J_1^T)}$
12: $\quad\quad\quad\quad m'_2 \leftarrow \sqrt{det(J_2 J_2^T)}$
13: $\quad\quad\quad\quad$ **if** $((m'_1 \geq m_1)\ \&\ (m'_2 \geq m_2)\ \&\ (m'_1 + m'_2 > m_1 + m_2))$ **then**
14: $\quad\quad\quad\quad\quad IK_{best} \leftarrow GetJointValues()$
15: $\quad\quad\quad\quad\quad m_1 \leftarrow m'_1$
16: $\quad\quad\quad\quad\quad m_2 \leftarrow m'_2$
17: $\quad\quad\quad\quad\quad \Delta_{best} \leftarrow \Delta_x$
18: $\quad\quad\quad\quad$ **end if**
19: $\quad\quad\quad$ **end if**
20: $\quad\quad$ **end if**
21: $\quad$ **end for**
22: $\quad$ **if** $(\Delta_{best} == \{0,0,0,0,0,0\}^T)$ **then**
23: $\quad\quad$ **return** IK_{best}
24: $\quad$ **end if**
25: **end while**
26: **return** IK_{best}

E. Manipulability Clusters

Due to the optimization process of the last section, an exemplary IK-solution for both arms can be stored together with the bimanual grasping configuration, which allows a locally maximized manipulability. Further information, that might be of interest, can be obtained by determining the extend of allowed movements of the *Center of Arms* joint. We call the approximated 6D-volume of the allowed CoA-movements for which the end-effectors remain at their grasping positions manipulability cluster. A quite simple, but effective, method of determining the extends of a reachability cluster is given by moving the CoA-joint in all translational and rotational directions until the joint limits inhibit the correct re-positioning of the TCPs according to the fixed grasping positions. The three-dimensional visualization of a manipulability cluster is given in Fig. 6. It can be seen, that the reachability in the three main axis of the local CoA coordinate system is approximatively represented by the ellipsoid.

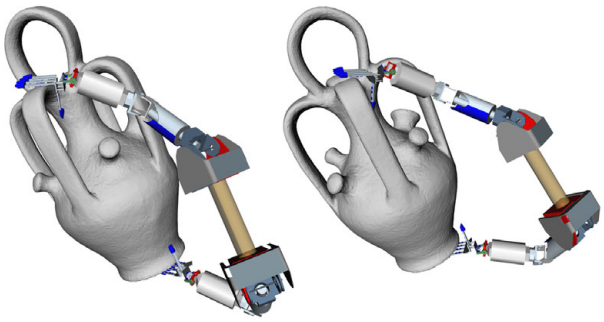

Fig. 5. A bimanual grasping configuration (left) and the configuration with optimized manipulability.

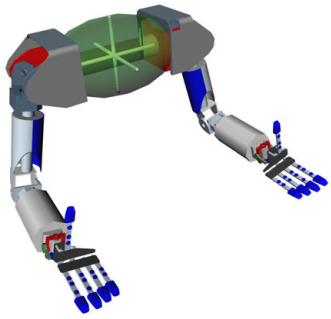

Fig. 6. The 3D visualization of the approximated 6D manipulability cluster for the *Center of Arms* joint. The lines depict the allowed movements in the three main axis of the local coordinate system.

By analyzing the volume of the manipulability cluster, one can derive the opportunity to move the CoA-joint around while holding the TCPs at the grasping position. A volume near to zero indicates, that the manipulability of the grasping configuration is constricted and thus the quality of the grasp is limited. As shown in Alg. 1 line 15, manipulability clusters with a non-existent volume are discarded by the planner.

V. Results

The presented approach is evaluated with two large objects that should be grasped with both hands.

TABLE I
PERFORMANCE EVALUATION

Average of 100 runs	**Vase**	**Bicycle**
Total:		
Planning one bimanual grasp	821 ms	5321 ms
Bi-Grasp candidates evaluated	1399	10232
Bi-Grasps until valid result	1.9	3.7
Per hypothesis:		
Manipulability cluster computation	79 ms	52 ms
Wrench space computations	22 ms	17 ms
Collision detection	2.5 ms	0.9 ms

A. *Vase*

In this experiment bimanual grasping configurations are generated for a large vase. In Fig. 7 (left) a 3D-visualization of the 6D clusters is given for 250 generated grasping configurations. The manipulability clusters can be used to support an whole body IK-solver. The IK-solving is done by first selecting a reachable bimanual grasping configuration (regarding the reachability information of the *robot pose* kinematic chain, see Fig. 2 (right)) and then moving the *Center of Arms* joint of both kinematic parts (arms and body) towards each other An exemplary IK result can be seen in Fig. 7 on the right.

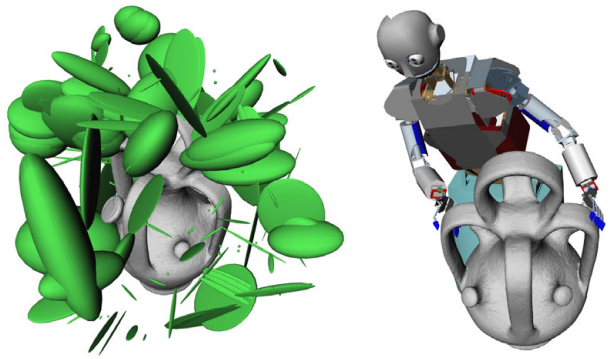

Fig. 7. The reachability clusters for 250 planned bimanual grasps (left). An exemplary whole body IK-solution for the humanoid robot ARMAR-III (right).

The performance of the presented approach was measured by planning 100 bimanual grasps while analyzing the runtime of the different steps of the algorithm. As shown in Table I, the time, that was needed to plan one 14 DoF bimanual grasping configuration, including all manipulability information, was measured with less than a seconds on average. This value includes the time needed for performing the wrench space computations to ensure a force closure grasp (22ms on average) as well as the reachability optimization and the generation of manipulability clusters. The time for building up a manipulability cluster was measured with 79 ms on average. Note, that potentially more than one manipulability cluster has to be built until a valid result can be generated, since zero-volume clusters are discarded. On average 2.5 ms were needed to check a configuration for collisions and 1.9 bimanual grasps have been generated until a valid result was found. The number of two-hand candidates (consisting of two single-hand candidates) that were considered until a result could have been generated, was counted with 1399. The majority of them were quickly discarded due to collisions or not allowed joint movements during the bimanual IK-search phase.

B. *Bicycle*

In this experiment the bimanual grasp planner is used to create a set of grasps for a bicycle. The challenging structure of the 3D-model results in longer computations compared to the last setup (see Table I). The average time needed to plan one bimanual grasping configuration was measured with 5.3 seconds. The large number of created created bimanual hypotheses (>10K) is caused by a large number of grasping configurations that result in collisions. Further we observed that the force closure test often failed, since many generated grasps are caging configurations around the bicycle frame that do not have sufficient contact information. In Fig. 8 the manipulability clusters of 250 planned grasps are visualized on the left and an exemplary IK-result is depicted on the right.

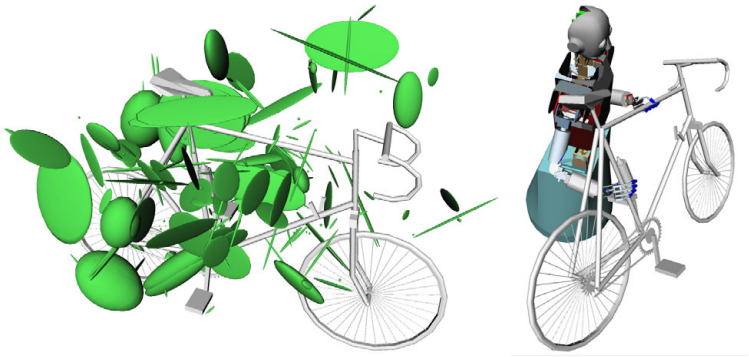

Fig. 8. The bimanual reachability of the bicycle is represented by 250 manipulability clusters generated by the proposed grasp planner (left). The humanoid robot ARMAR-III uses a planned grasp to hold the bicycle with both hands (right).

VI. CONCLUSION

In this work a bimanual grasp planner was presented, that can be used to efficiently determine dual-arm configurations for grasping large objects. The search for grasp candidates is efficiently realized by using precomputed reachability information and an object representation, based on medial axis descriptions.

Since bimanual grasping configurations often suffer from strictly constrained movements caused by the closed kinematic chain that is formed when applying the grasp, we introduced a grasp quality measurement based on manipulability information. This allows to focus the search on bimanual grasps that have a high manipulability, so that the configuration can be better adjusted to the needs of an online IK-query. Further, those grasps with good manipulability reduce the constraints in post-grasping actions, which otherwise may prohibit a planned execution.

To evaluate the performance and the usability of the approach, we generated bimanual grasp sets with two challenging objects for the humanoid robot ARMAR-III. It was shown, that suitable grasping configurations can efficiently be generated for a high-DoF system.

ACKNOWLEDGMENT

The work described in this paper was partially conducted within the German Humanoid Research project SFB588 funded by the German Research Foundation (DFG: Deutsche Forschungsgemeinschaft) and the EU Cognitive Systems project GRASP (IST-FP7-IP-215821) funded by the European Commission.

REFERENCES

[1] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Simox: A Simulation and Motion Planning Toolbox for C++," Karlsruhe Institute of Technology (KIT), Tech. Rep., 2010.

[2] T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, "Armar-III: An integrated humanoid platform for sensory-motor control." in *IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006)*, December 2006, pp. 169–175.

[3] K. Okada, T. Ogura, A. Haneda, J. Fujimoto, F. Gravot, and M. Inaba, "Humanoid motion generation system on hrp2-jsk for daily life environment," in *Mechatronics and Automation, 2005 IEEE International Conference*, vol. 4, july-1 aug. 2005, pp. 1772 – 1777 Vol. 4.

[4] T. Asfour, P. Azad, N. Vahrenkamp, K. Regenstein, A. Bierbaum, K. Welke, J. Schröder, and R. Dillmann, "Toward Humanoid Manipulation in Human-Centred Environments," *Robotics and Autonomous Systems*, vol. 56, pp. 54–65, January 2008.

[5] R. Platt, "Learning and generalizing control based grasping and manipulation skills," Ph.D. dissertation, Department of Computer Science, University of Massachusetts Amherst, 2006.

[6] T. Wimbock, C. Ott, and G. Hirzinger, "Impedance behaviors for two-handed manipulation: Design and experiments," in *Robotics and Automation, 2007 IEEE International Conference on*, april 2007, pp. 4182 –4189.

[7] T. Martin, R. Ambrose, M. Diftler, J. Platt, R., and M. Butzer, "Tactile gloves for autonomous grasping with the nasa/darpa robonaut," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 2, 26-may 1, 2004, pp. 1713 – 1718 Vol.2.

[8] D. Kappler, L. Chang, M. Przybylski, N. Pollard, T. Asfour, and R. Dillmann, "Representation of Pre-Grasp Strategies for Object Manipulation," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Nashville, USA, December 2010.

[9] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *Robotics Automation Magazine, IEEE*, vol. 11, no. 4, pp. 110 – 122, Dec. 2004.

[10] R. Diankov and J. Kuffner, "Openrave: A planning architecture for autonomous robotics," Robotics Institute, Tech. Rep. CMU-RI-TR-08-34, July 2008. [Online]. Available: http://openrave.programmingvision.com

[11] A. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 2, Sept. 2003, pp. 1824 – 1829 vol.2.

[12] C. Goldfeder, C. Lackner, R. Pelossof, and P. K. Allen, "Grasp planning via decomposition trees," in *Robotics and Automation, 2007 IEEE International Conference on*, April 2007, pp. 4679 –4684.

[13] K. Huebner, S. Ruthotto, and D. Kragic, "Minimum volume bounding box decomposition for shape approximation in robot grasping," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2008, pp. 1628 –1633.

[14] M. Przybylski, T. Asfour, and R. Dillmann, "Unions of balls for shape approximation in robot grasping," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[15] M. Przybylski, T. Asfour, and R. Dillmann, "Planning grasps for robotic hands using a novel object representation based on the medial axis transform," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2011.

[16] J. Aleotti and S. Caselli, "Grasp synthesis by 3d shape segmentation using reeb graphs," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[17] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, "Grasp planning in complex scenes," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 29 2007-Dec. 1 2007, pp. 42 –48.

[18] D. Berenson and S. S. Srinivasa, "Grasp synthesis in cluttered environments for dexterous hands," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Dec. 2008, pp. 189 – 196.

[19] "Planning pick-and-place tasks with two-hand regrasping," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 4528–4533.

[20] N. Vahrenkamp, M. Do, T. Asfour, and R. Dillmann, "Integrated Grasp and Motion Planning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, USA, Mai 2010, pp. 2883–2888.

[21] N. I. Badler, C. B. Phillips, and B. L. Webber, *Simulating Humans: Computer Graphics Animation and Control*. New York, Oxford: Oxford University Press, 1993.

[22] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, September 2010.

[23] N. Vahrenkamp, D. Berenson, T. Asfour, J. Kuffner, and R. Dillmann, "Humanoid Motion Planning for Dual-Arm Manipulation and Re-Grasping Tasks," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, USA, October 2009, pp. 2464–2470.

[24] F. Zacharias, D. Leidner, F. Schmidt, C. Borst, and G. Hirzinger, "Exploiting structure in two-armed manipulation tasks for humanoid robots," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, oct. 2010, pp. 5446 –5452.

[25] T. Yoshikawa, "Manipulability of robotic mechanisms," *The International Journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985.

[26] F. C. Park and J. W. Kim, "Manipulability of closed kinematic chains," *Journal of Mechanical Design*, vol. 120, no. 4, pp. 542–548, 1998.

[27] J. Merlet, "Jacobian, manipulability, condition number and accuracy of parallel robots," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, S. Thrun, R. Brooks, and H. Durrant-Whyte, Eds. Springer Berlin / Heidelberg, 2007, vol. 28, pp. 175–184.

[28] H. Blum, *Models for the Perception of Speech and Visual Form*. Cambridge, Massachusetts: MIT Press, 1967, ch. A transformation for extracting new descriptors of shape, pp. 362–380.

[29] B. Miklos, J. Giesen, and M. Pauly, "Discrete scale axis representations for 3d geometry," in *ACM SIGGRAPH 2010 papers*, ser. SIGGRAPH '10. New York, NY, USA: ACM, 2010, pp. 101:1–101:10.

[30] C. Ferrari and J. Canny, "Planning optimal grasps," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, May 1992, pp. 2290 –2295 vol.3.