Electronic version of an article published as
International Journal of Humanoid Robotics, Vol. 9(4), 2012
DOI: 10.1142/S0219843612500351

Efficient Inverse Kinematics Computation based on Reachability Analysis

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

Received 18 Nov. 2011
Accepted 05 Sep. 2012
Published 15 Nov. 2012

In this work we show how precomputed reachability information can be used to efficiently solve complex inverse kinematics (IK) problems such as bimanual grasping or re-grasping for humanoid robots. We present an integrated approach which generates collision-free IK solutions in cluttered environments while handling multiple potential grasping configurations for an object. Therefore, the spatial reachability of the robot's workspace is efficiently encoded by discretized data structures and sampling-based techniques are used to handle arbitrary kinematic chains. The algorithms are employed for single-handed and bimanual grasping tasks with fixed robot base position and methods are developed that allow to efficiently incorporate the search for suitable robot locations. The approach is evaluated in different scenarios with the humanoid robot ARMAR-III.

Keywords: Inverse Kinematics; Humanoid Robots; Reachability Analysis

1. Introduction

Mobile robots operating in cluttered environments must be able to find suitable grasping configurations for manifold tasks. Therefore setups for bimanual grasping or re-grasping as well as for positioning the robot have to be found efficiently. While industrial robot systems operate in well-defined work cells, where collision-free grasping configurations can be easily precomputed, service or humanoid robots have to deal with a changing human-centered environment. Hence, robots must be able to find suitable grasps in the presence of obstacles and for varying object positions. Further, it must be possible to efficiently determine suitable robot base poses that place the robot in a collision-free position relatively to the target object so that a grasp can be applied successfully.

The classical inverse kinematics (IK) problem is discussed extensively in literature. When considering a kinematic chain K of n joints, the problem is formulated as follows: For a given Cartesian pose $p \in SE(3)$, a joint configuration $q \in \mathbb{R}^n$ is

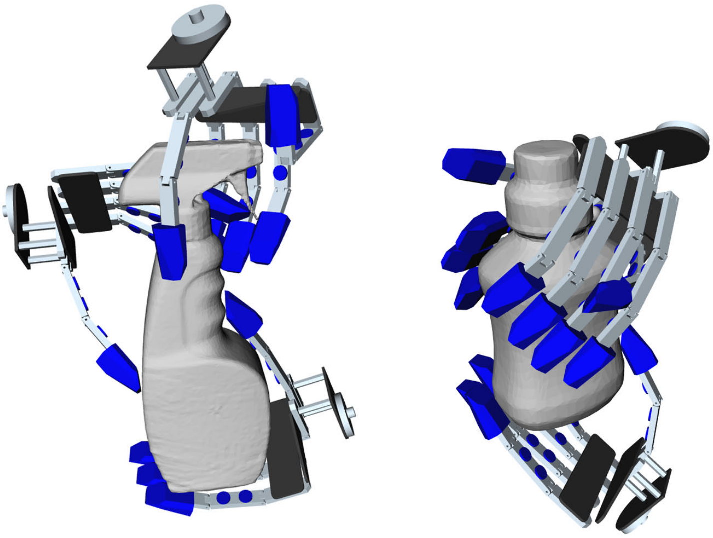

Fig. 1. Two objects, each with three automatically generated grasps. The grasp wrench space approach [9] was used to build the grasping configurations.

searched that, when applied to K, brings the tool center point (TCP) to the pose p. When the number of joints is greater than six, redundancy is introduced, resulting in a solution space in which an optimal solution can be searched. Such redundancy can be exploited, e.g. when a target is partially blocked by an obstacle or when multiple tasks have to be fulfilled.

When the IK problem is addressed in the context of Cartesian control of a robot's end effector, Jacobian-based approaches are widely used, since usually the steps in workspace are small and gradient descent methods are suitable for controlling the pose of the robot. Several approaches are known in literature, such as pseudoinverse methods [1] or Jacobian transpose approaches [2]. Problems at singularities or oscillating behavior can be reduced with damped least squares methods [3,4] and multiple end effectors can be handled with the selectively damped least squares approach [5]. Full body control for a humanoid robot with a floating base is presented in by Mistry et al. [6]. Constraints such as respecting joint limits or center of mass computations can be considered by performing null space projections [7]. Null space methods are also used by Maciejewski et al. for obstacle avoidance [8].

Kanoun et al. present a method for prioritizing linear equality and inequality systems with applications to humanoid motion planning [10]. With the approach local solutions can be planned for reaching motions while satisfying constraints such as the stability of a legged robot. An extended approach was presented by Escande et al., where the hierarchized inverse kinematics problem is addressed with inequality constraints. Due to the so called Hierarchized Complete Orthogonal Decomposition, the algorithm is very efficient and can be used for reactive control of a humanoid robot [11].

Although such iterative gradient-descent methods perform very well for small Cartesian movements and several constraints can be considered, they are not suited to solve all IK-related problems. Problems may arise when gradient descent gets stuck in local minima or when it is difficult or even impossible to specify a set of inequalitities to describe the task. Further, collision avoidance may be challenging when cluttered scenes with many obstacles are considered [12]. Since usually the shortest distance between all obstacles and the robot is used for generating a gradient in joint space, only local avoidance strategies can be considered. Hence, the global task of finding a collision-free configuration that is not affected by local minima is still challenging.

In contrast to iterative methods, closed-form solutions can be computed by analytic approaches. Such algorithms are usually robot-specific, which means that for each manipulator or kinematic chain, the formula have to be determined manually [13,14]. Kallmann presents a work where postures of a virtual human character are generated for the hip and both arms. The analytic method is efficient and collisions can be considered, but the approach cannot be generalized since several properties of the kinematic model are exploited for determining the closed-form approach [15]. Konietschke et al. show how whole body IK-solutions covering several parts of the robot can be efficiently computed by exploiting analytic IK-solvers that serve solutions for sub-chains which are combined to solve the whole body IK-problem [16]. The IKFast method, presented by Diankov, allows to automatically determine a set of equations for closed-form IK-solving [17]. The algorithm performs well for kinematic chains covering up to six DoF and hence, this approach may be used to generate closed-form IK-solvers to be used within an hybrid IK-solver (see Section 3.1). The IKFast algorithm can be combined with a discretized sampling strategy in case redundancy has to be considered [18]. Cortes and Simeon present an IK-approach that dynamically changes the number of active joints while trying to solve a loop-closure constrained IK task. A sampling-based algorithm is combined with a spherical workspace approximation in order to increase the efficiency of complex IK-queries [19].

In the context of mobile manipulation, usually not one target pose $p \in SE(3)$ has to be considered, but multiple potential targets, encoded as a set of object specific grasping configurations, are available for grasping. A grasping configuration g is defined by the hand to object transformation and the joint configuration of the fingers when applying the grasp. A set of potential grasps can be generated manually or by grasp planners [20,21,22] that can be found in software tools such as GraspIt! [23], OpenRave [17] or Simox [24]. An exemplary set of grasps that have been automatically generated for two objects can be seen in Fig. 1.

When considering cluttered environments that can be found in human-centered surroundings, the search for suitable IK-solutions includes the selection of a feasible grasp and the avoidance of collisions. Berenson et al. present a planning approach, where a collision-free grasping motion is generated while considering a set of precomputed grasping configurations[25]. The algorithm is based on a scoring function that is used to rank potential grasps with respect to the environment and their spatial position. This general framework allows including custom scoring functions, but several parameters are introduced that have to be manually adjusted to retrieve optimal results.

1.1. *Contribution*

In this work we show how precomputed reachability information can be used to efficiently solve complex inverse kinematics problems such as bimanual grasping or re-grasping for humanoid robots. Therefore we present a framework to address the following IK-related tasks:

- **Multiple grasps** can be handled and feasible grasps are selected autonomously.
- **Collision-detection** is performed to avoid self-collisions and to generate valid solutions in cluttered scenes.
- **Single arm and bimanual IK-queries** can be processed in order to grasp an object with both hands or to hand-over an object.
- **Robot placement** can be incorporated in order to search feasible robot base poses for grasping.
- **Efficiency** is achieved by using precomputed reachability data.

Although some parts of the presented IK-algorithm are based on existing approaches, we present a unifying framework that allows to solve IK-queries for various problems in challenging setups. Additionally an advanced technique for generating inverse reachability information is described, which can be used to position the robot for grasping.

We will motivate the use of discretized reachability distributions (RD) in the next section. It is shown how a reachability analysis can be efficiently performed for arbitrary kinematic chains of a robot. Further, an encoding for promising robot base poses for grasping based on inverse reachability distributions (IRD) will be presented. In Section 3 several IK-algorithms are developed which are evaluated in Section 4. Finally a conclusion is given in Section 5.

2. Reachability Analysis

The structure of the reachable workspace of a mobile manipulator holds information about which areas are reachable for single or dual-handed tasks. This information can be used to support IK-queries, i.e. to quickly report that no solution exists since the target pose is located outside the reachable workspace. Further, reachability analysis helps to filter feasible grasps from a set of potential grasps that may be stored with an object representation. Another application of reachability data is presented in Section 2.2, where inverse reachability data is created in order to find potential robot base poses for grasping.

To represent the reachability, a 6D voxelization of the workspace is employed, similar to existing approaches[18,26,27]. This data structure offers an efficient way to query the reachability of a pose, since the voxel entries can be accessed quickly via look-up tables. It has been shown that reachability information can also be used in the context of bimanual manipulation in order to improve the efficiency of IK related tasks[28,29]. Compared to the work of Zacharias et al., where a 3D grid is used to represent the position in workspace and the orientation is encoded by shape primitives[27,29], we encode full 6D poses by a voxelized data grid. This allows us to encode positions and orientations uniformly and alleviates any following processing steps, e.g. the generation of inverse reachability data. Note, that the capability representation of Zacharias et al. can be transformed to a binary 6D grid allowing to use the presented IK-solvers.

2.1. *Reachability Distribution (RD)*

A discretized representation of the reachability can be determined by solving a large number of IK requests and counting the number of successful queries for each voxel in workspace. An analytic approach of generating a representation of the 3D-reachability is presented by Kee and Karwowski[30]. Another way of generating the reachability data is to randomly sample the joint values while using the forward kinematics to determine the pose of the end effector (EEF) and thus the corresponding 6D voxel[31]. This approach creates an approximated representation of the volume in C-space that maps to a specific voxel. Throughout this work we use this algorithm to create the reachability data, although it introduces a preference of singular configurations since at singular configurations large displacements in C-space are mapped to identical voxels. In order to counteract this issue, the manipulability measure can be taken into account, as a pose quality index that penalizes singular configurations. Therefore, Yokishawa's manipulability index[32] can be used, which is related to the size of the manipulability ellipsoid.

To build up the reachability data, a large number of joint configurations are randomly sampled and the resulting pose is checked for self-collisions. If a collision-free configuration has been sampled, the entry of the corresponding RD voxel is increased either by 1 or, in case the manipulability is taken into account, by the pose's manipulability index. Finally, the resulting reachability representation is normalized by dividing all entries by the maximum value.

Note, that the presented IK-approaches do not directly rely on the algorithm that is used to create the reachability data. Even a binary representation, indicating that at least one IK-solution exists within the extends of a 6D voxel, will be sufficient for the following IK-algorithms. The use of more sophisticated representations increases the performance, since the number of falsely assumed valid grasps can be lowered and hence, unnecessary calls to the IK-solver can be avoided.

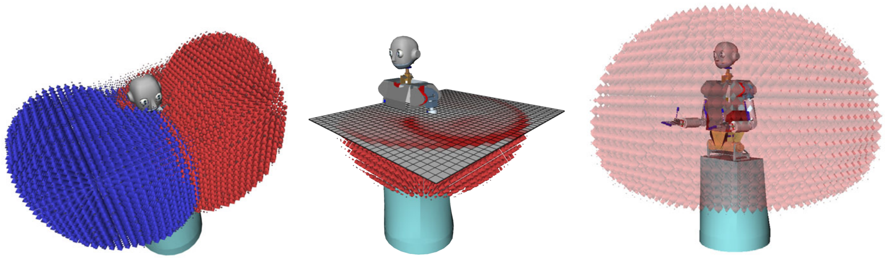

Fig. 2. Left: A 3D representation of the reachability of the left and the right arm of the humanoid robot ARMAR-III (7 DoF each). Center: The reachability of the left end effector considering the 8 DoF-kinematic chain covering the hip-yaw joint and the left arm. A cut through the 3D visualization of the reachability data is shown, where the intensity is proportional to the RD entry for the corresponding pose. Right: The reachability of the kinematic chain from platform base to the left end effector. The joint selection covers the orientation of the platform, three hip and seven arm joints (11 DoF).

The reachability data can be computed for several kinematic chains, such as an arm (see Fig. 2 (left)), one hip joint and an arm (see Fig. 2 (center)) or by considering the complete chain from the robot's base to an end effector (see Fig. 2 (right)).

The reachability of a given pose $p \in SE(3)$, related to a kinematic chain K with corresponding end effector, can be true or false. When p is reachable, there must exist a configuration q of the given kinematic chain, so that the corresponding pose of the end effector is equal to p. When considering a discretized workspace representation, the reachability of a voxel v cannot be reliably expressed as a binary value, since there might be poses $p_r \in v$ that are reachable while other poses $p'_r \in v$ are not. Hence, the reachability of a voxel can just give a hint (or a probability) that a pose inside that voxel is reachable. When considering the reachability of a voxel as a probability that a pose is reachable, the discretized reachability data can be interpreted as a frequency distribution known from descriptive statistics[33] and therefore we use the term reachability distribution in this work to name the discretized reachability data that is represented by entries of 6D workspace voxels. In Fig. 2, the RD of different kinematic chains of the humanoid robot ARMAR-III are depicted as a 3D visualization. The entry of each 3D voxel (x, y, z) is built by accumulating the entries of all 6D voxels $(x, y, z, \alpha, \beta, \gamma)$ with arbitrary rotation values (α, β, γ). The size and the color intensity of the shown voxels correspond to the magnitude of this value.

2.2. *Inverse Reachability Distribution (IRD)*

The RDs can be used to decide if a grasping pose is reachable for a given robot position. In the context of mobile manipulation an extended problem formulation, where a suitable base pose for grasping is searched, is often of interest. Here, the base pose of the robot is not assumed to be fixed in the world and additionally to the joint configuration that solves the inverse kinematics problem, a base pose of

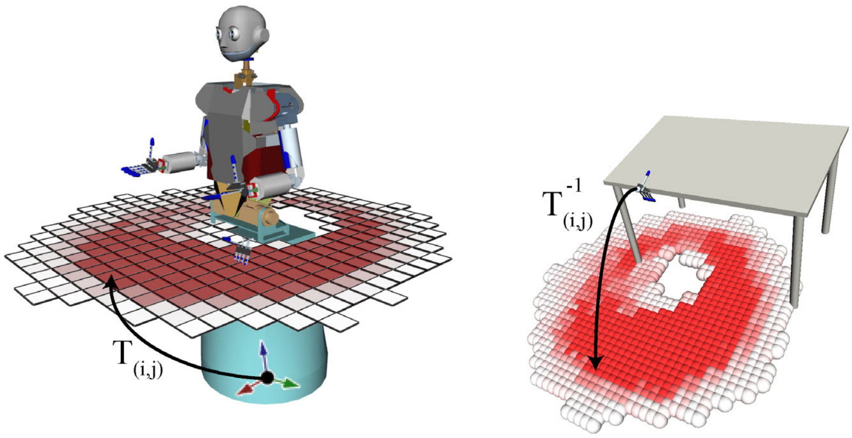

Fig. 3. Left: A 2D reachability distribution is defined by the height and the orientation of the grasping pose. The intensity is proportional to the probability that a pose with the height and the orientation of the depicted grasp is reachable when using 11 joints of platform, hip and left arm. Right: The IRD is constructed by inverting the base-to-target transformations $T(i, j)$.

the robot is requested. Such IK-problems can be tackled in several ways. Stulp et al. use experience-based learning approaches to learn a generalized success model which discerns between poses from which grasping or manipulation succeeds or fails [34]. Online, this model is used to compute a so-called *ARPLACE*, a probability distribution that maps poses to a predicted probability of successful manipulation. Reachability information can be used to determine promising robot poses in workspace for 3D trajectory execution of an arm [35]. Therefore Zacharias et al. compute the reachability pattern of a given trajectory and correlated with the so-called 3D reachability sphere map. In contrast to this work, we consider the orientation of the robot as part of the kinematic chain when building the reachability data. This allows us to encode the robot's orientation directly within the RD and hence no additional step for creating reference rotations of the reachability data is needed. Diankov proposes a method, where reachability distributions of an arm are used to build equivalent classes that represent the rotated poses of a grasp in the plane [18]. These equivalent classes are used to build discretized 2D maps encoding the reachability of the given grasp from the corresponding base pose of the robot. This 2D map can be used to find base poses of a mobile manipulator for which a high probability exist that the given grasping pose is reachable, i.e. an IK-solution exists.

Instead of building equivalent classes, we propose an approach for building 2D reachability data by inverting a reachability distribution that was built with a kinematic chain covering the orientation of the robot [36]. Such an inverse reachability distribution can be seen in Fig. 2 on the right, where the platform yaw, three hip and seven arm joints were used to build the RD for the left end effector. Since the orientation of the robot is encoded in the reachability distribution, the relationship

between target pose and the robot's orientation is already represented in the data and no equivalent classes have to be built and further discretization steps can be avoided. An IRD can be efficiently created as follows:
Let $p \in SE(3)$ be a target pose in workspace, (r_x, r_y, r_α) the robot's base position and orientation and RD_K the reachability distribution for the kinematic chain K that covers the orientation of the robot's base and ends with the tool center point (TCP). The target pose can be expressed with the translational component (p_x, p_y, p_z) and three angles $(p_\alpha, p_\beta, p_\gamma)$ representing the orientation.

A 2D distribution can be built by fixing the orientational components and the height of the pose. This distribution, visualized in Fig. 3(left), describes the reachability of a target position at height p_z and with the orientation $(p_\alpha, p_\beta, p_\gamma)$ in the robot's base coordinate system. The base coordinate system could be defined in the platform for a wheeled mobile robot or in the torso when considering a legged humanoid robot. When changing the point of view from the robot's base coordinate system to a specific grasping pose, the IRD can be constructed by applying the inverse transformations to the grasping pose. Thus, instead of defining the reachability for a TCP pose in the robot's base coordinate system, we are now defining the distribution of the 2D base positions describing the probability that a specific target pose p is reachable. Due to the discretized reachability structure, the entries in the IRD can be efficiently calculated by transforming each grid point to the (x, y)-plane in the robot's base coordinate system by applying the inverse base-to-EEF transformation for the actual target pose.

In Fig. 3, the 2D reachability distribution together with the corresponding target pose is visualized on the left. The transformations $T(i, j)$, describing the position of the 2D grid cells in the robot's base coordinate system, are inverted and applied to the grasping pose to compute the IRD. A visualization is shown in Fig. 3 on the right.

2.3. *IRD for Multiple Grasps*

A common approach for dealing with feasible grasps is to run a grasp planning algorithm in an offline step to build a set of feasible grasps which can be applied to the target object. This grasp set is object and EEF specific and thus it has to be computed once per object/EEF combination. When a set of grasps is defined for an object, the united IRD can be used to build a representation of feasible robot poses for grasping. Entries of the united IRD will then not only describe the probability of finding an IK solution, furthermore a link to all reachable grasps is stored in the 2D grid.

If a set of k grasping configurations $g = (g_1, \ldots, g_k)$ is defined for an end effector and an object, the resulting united inverse reachability distribution IRD_g is defined for each position (x, y) as the maximum value that exists in any IRD_{g_i} at (x, y), where IRD_{g_i} is the inverse reachability distribution that is related to grasp g_i.

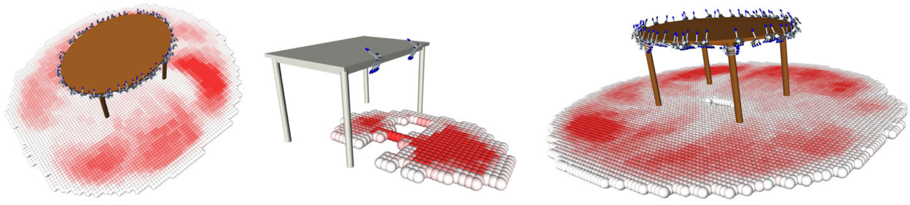

Fig. 4. Left: The united IRD for a set of grasps, defined for the left end effector. Center: The bimanual IRD for a bimanual grasping setup. Right: The united bimanual IRD for 60 predefined grasps for both EEFs.

$$IRD_g(x, y) = \max_{i \in \{1,\ldots,k\}} IRD_{g_i}(x, y). \tag{1}$$

The united IRD for a set of 50 grasping configurations of the left hand can be seen in Fig. 4 (left). Note that the grasps have been generated by a sampling-based approach and hence, they are not uniformly placed and the IRD is therefore not uniformly distributed around the table.

2.4. *Bimanual IRD for Dual-Handed Grasps*

If there are two sets of grasps defined, G_l for the left and G_r for the right hand of a humanoid robot, it is possible to define dual-arm grasping combinations by testing all $|G_l| \cdot |G_r|$ possible combinations in advance. All solution to the dual-arm IK problem are then stored in a dual-arm grasp set. Since this approach would introduce high computational costs, we propose a different way of finding dual-arm grasping combinations which can be done online. By computing the minimum of the two IRDs for the left and the right hand, the unified bimanual inverse reachability distribution IRD_{bi} gives a good hint where potential robot poses for applying dual-arm grasps are located (see Eq. 2). Here, the minimum of both IRDs is used, since the resulting value should represent the probability of finding a dual arm IK solution and when the search for one arm fails, the whole IK-algorithm fails. Additionally to the probability of reaching the grasping poses, a link to all reachable grasps is stored in each cell of IRD_{bi}, so that the set of potential reachable grasps can be retrieved quickly later on. In Fig. 4(center) and (right) IRD_{bi} is depicted for two and for 60 grasps.

$$IRD_{bi}(x, y) = \min(IRD_{left}(x, y), IRD_{right}(x, y)) \tag{2}$$

3. Inverse Kinematics

In this section, efficient approaches are presented to solve the inverse kinematics problem for mobile manipulators. Therefore, precomputed reachability information

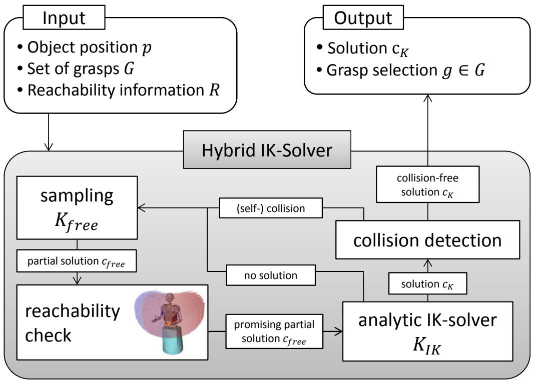

Fig. 5. An overview of the hybrid IK-approach.

is used to speed up IK-requests for single- and dual-armed tasks. Therefore RD and IRD data is generated in an offline processing phase with the methods described in Section 2. This precomputing step has to be done once for each kinematic chain that is considered for IK solving. The resulting IK solvers are able to handle the following setups:

- **Single- and dual-handed:** IK-solutions are generated for grasping tasks considering one or both hands.
- **Handover:** Handover configurations can be computed for re-grasping tasks.
- **Robot pose:** The IK-solvers can be used to search suitable robot poses for grasping.

The developed IK-approaches have the following characteristics:

- **Collision-free solutions:** Self-collisions as well as collisions with the environment are avoided.
- **Grasp set:** Predefined sets of potential grasps can be handled without the need of choosing a suitable grasping configuration in advance, since a suitable grasp is implicitly determined by the IK-algorithms.
- **Sampling:** Due to the randomized design of the IK-algorithms, the set of possible IK solutions can be sampled with the presented approaches. This allows to integrate the IK-algorithm to IK-based motion planning algorithms, where goal regions that are induced by target locations have to be sampled while planning collision-free motions[28,37,38].

3.1. *Hybrid IK-Solver*

When an analytic IK-solver is present for a kinematic chain of a robot system (e.g. one can think of an 6 DoF arm), it can be included in a randomized IK-solver in or-

Algorithm 1 $GraspIK(K_{free}, K_{IK}, G, p)$

Input: Two kinematic chains (K_{free}, K_{IK}), a set of precomputed grasps G and the workspace pose of the object $p \in SE(3)$.

Output: Solution configuration c or $NULL$.

$K \leftarrow K_{free} \bigcup K_{IK}$
$G_{reachbale} \leftarrow ReachableGrasps(K, G, p)$
while $(!TimeOut())$ **do**
  $p_{grasp} \leftarrow SampleRandom(G_{reachbale}) \cdot p$
  $c_{free} \leftarrow SampleFreeParameters(K_{free})$
  $SetRobotConfiguration(K_{free}, c_{free})$
  if $(Reachability(K_{IK}, p_{grasp}) > 0)$ **then**
    $c_{IK} \leftarrow AnalyticIK(K_{IK}, p_{grasp})$
    $c \leftarrow \{c_{free}, c_{IK}\}$
    if $(c_{IK}$ & $!Collision(c))$ **then**
      return c
    end if
  end if
end while
return $NULL$

der to handle more complex setups efficiently. The resulting hybrid IK-solver covers the kinematic chain K, consisting of the kinematic chain K_{IK} that is handled by the analytic IK-solver and the kinematic chain K_{free} handled by sampling-based techniques. By using precomputed reachability information, complex IK-queries can be answered efficiently by only process those samples of K_{free} for which the probability that an IK solution for K_{IK} exists is greater zero. An overview of the proposed method can be found in Fig. 5.

3.2. *Single-Handed Grasping*

In Alg. 1 an IK-solution is searched for the kinematic chain K, consisting of a free part K_{free} and a part K_{IK}, that is covered by the analytic IK solver. The algorithm can handle a set of grasps G together with an object pose p. Here, we assume that a grasp g is given as a TCP-to-object transformation and hence $g \in SE(3)$. At first, the reachable subset $G_{reachable} \subset G$ of all grasps w.r.t. the current object pose p is determined. Note, that for this step the reachability of the complete kinematic chain K has to be considered. This can be realized efficiently, since internally only calls to a look-up table have to be processed. Afterwards, a randomized loop for searching IK-solutions is performed. The grasps and the joint values of K_{free} are sampled randomly and the reachability of the resulting grasping pose $p_{grasp} \in SE(3)$ is determined. In case the partial solution c_{free} results in a robot configuration that allows to reach the current target pose, the analytic IK solver is called. Note, that

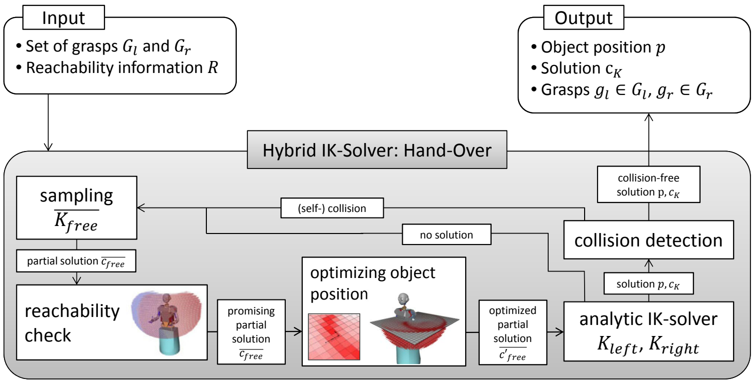

Fig. 6. Hybrid IK-solver for Hand-Over configurations.

the reachability information used here does only cover the kinematic chain K_{IK}. Since the reachability data is linked to the start of K_{IK} and this joint may move when changing the robot's configuration to c_{free}, the transformation of the reachability data has to be re-computed in every loop. Finally, the solution is checked for collisions.

3.3. *Bimanual Grasping*

The hybrid IK-approach can also be used to solve bimanual grasping tasks. Therefore two kinematic chains, one for each arm (K_{left} and K_{right}), are considered and an optional kinematic chain K_{free} can be used as described before. The algorithm for bimanual IK solving is analogous to the single arm approach shown in Alg. 1. Instead of considering one grasp set, two sets of grasps are used (G_{left} and G_{right}). Further, two IK solutions for the sub-chains K_{left} and K_{right} have to be searched and the final solution is composed of c_{free}, c_{left} and c_{right}. Additionally the result is checked against self-collisions and collisions with the environment.

3.4. *Re-Grasping*

If the robot should re-grasp or hand over an object, the search for a valid re-grasping configuration includes a collision-free object pose and a valid and collision-free IK-solution for both arms.

This leads to an IK problem, where the free parameters do also cover the spatial 6D-pose of the object which is denoted by $\overline{K_{free}}$. Hence, a sample $\overline{c_{free}}$ consists of the 6D object pose in workspace p_{free} and a configuration c_{free} of K_{free}.

To find a object pose in the reachable workspace of the robot, the 6D pose of the object and the configuration of K_{free} can be sampled randomly until a call of the IK solver is successful for one of the poses. Therefore, the Cartesian position of the object is limited to the extent of the reachable space and the rotation component does not have any restrictions. As shown in Fig. 6, sampled partial solutions $\overline{c_{free}} = \{p_{free}, c_{free}\}$ are checked whether the configuration c_{free} and the sampled object pose p_{free} result in a setup which is reachable. If so, the object pose p_{free} is optimized, so that the reachability is locally maximized in order to increase the probability to find an bimanual IK-solution in the next step. For the resulting optimized partial solution $\overline{c'_{free}} = \{p'_{free}, c_{free}\}$ the analytic IK-solvers are called and when a solution for both arms can be generated the result is checked against collisions.

In Alg. 2 potential object poses and the remaining free parameters are sampled and the reachability of the resulting setup is analyzed. If the sample results in a configuration that has a reachability greater zero for the left and the right end-effector, the object pose p is locally optimized in order to achieve the maximum reachability. Therefore, the neighboring voxels for the left and the right RD are evaluated and in case a direction with higher accumulated reachability is present, p is moved accordingly. This procedure is performed iteratively until no better pose can be obtained and the local maximum is reached. Afterwards, the resulting grasping poses are generated by applying the grasps g_{left} and g_{right} to the optimized object pose p' and the IK-solvers are queried. In case a collision-free result can be determined, the final solution consists of the object pose p' and the robot configuration c.

Algorithm 2 $ReGraspIK(K_{free}, K_{left}, K_{right}, G_{left}, G_{right})$

Input: Three kinematic chains (K_{free}, K_{left} and K_{right}) and two sets of predefined grasps (G_{left}, G_{right}).

Output: A solution vector with configuration c and object pose p or $NULL$.

while $(!TimeOut())$ **do**
    $p \leftarrow SampleRandomObjectPose()$
    $\{g_{left}, g_{right}\} \leftarrow \{RandomGrasp(G_{left}), RandomGrasp(G_{right})\}$
    $\{p_{left}, p_{right}\} \leftarrow \{g_{left} \cdot p, g_{right} \cdot p\}$
    $c_{free} \leftarrow SampleFreeParameters(K_{free})$
    $SetRobotConfiguration(K_{free}, c_{free})$
    if $(Reachability(K_{left}, p_{left}) > 0 \;\&\; Reachability(K_{right}, p_{right}) > 0)$ **then**
        $p' \leftarrow OptimizeGraspingPose(K_{left}, K_{right}, g_{left}, g_{right}, p)$
        $c_{left} \leftarrow AnalyticIK(K_{left}, g_{left} \cdot p')$
        $c_{right} \leftarrow AnalyticIK(K_{right}, g_{right} \cdot p')$
        $c \leftarrow \{c_{free}, c_{left}, c_{right}\}$
        if $(c_{left} \;\&\; c_{right} \;\&\; !Collision(c))$ **then**
            return (c, p')
        end if
    end if
end while
return NULL

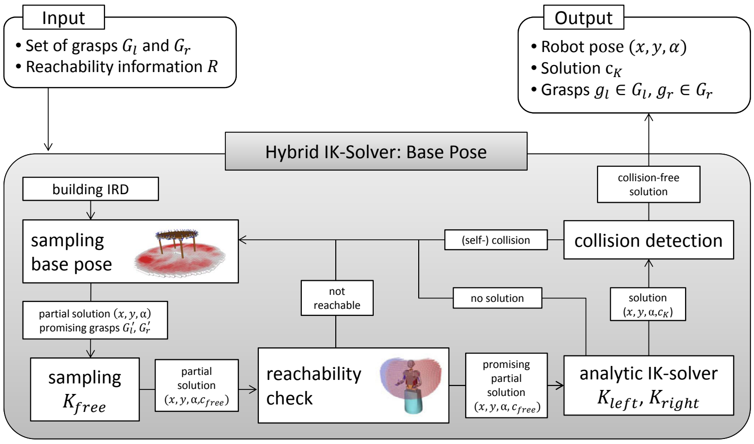

Fig. 7. Bimanual IK-approach including the search for valid robot poses.

3.5. *Considering the robot's base pose*

When a mobile robot is supposed to grasp an object that is located outside the reachable workspace, a suitable base pose for grasping has to be searched. A common approach of searching IK-solutions for such tasks is to independently handle the three tasks that need to be solved:

- **Robot base pose:** A suitable pose of the robot in the world has to be found. This pose must ensure that the object can be grasped without any collisions.
- **Grasp selection:** A grasp from a predefined set of potential grasps has to be chosen. The grasp must be reachable so that a collision-free IK-solution can be found in the next step.
- **IK-solver:** The inverse kinematics problem has to be solved for the chosen robot pose and the selected grasp while collisions have to be avoided.

Due to the stepwise processing of the three tasks, problems can arise such as the inappropriate selection of robot base poses or grasps. This can lead to situations where no IK-solutions can be found in the third step and a proper handling must be implemented. The difficulties with such stepwise approaches are not surprising, since the set of collision-free reachable IK-solutions in cluttered scenes cannot be easily determined. Hence, no exact knowledge of the collision-free reachability is present in the first two steps. Heuristics can be used to guide the search to promising base poses or grasps, but usually no guarantees can be achieved until the IK-solver is queried.

Because of the described challenges we propose an integrated IK-approach, where random sampling strategies are used offensively to deal with the uncertainties that arise from selecting base poses and grasps (see Fig. 7).

In the following we will consider the robot's base pose in a flat 2D world, so that the position of the robot can be described with three values: the 2D position (x, y) and the orientation α. By incorporating these values into K_{free} the algorithms of the last sections can be used to solve the IK-problem covering the base pose.

Since the extends of the translational dimensions of the configuration space tend to be very large, a non-guided sampling will result in a long computation time, due to the large number of unsuccessful pose samples. Hence, heuristics are often used to limit the sampling of the base position around the target object. Further improvements can be achieved when IRDs (see Section 2.2) are used, since reachable base poses are encoded in such data. The resulting sampling strategy that we propose is twofold: In the first step the IRDs are queried in order to sample promising robot base poses, for which a subset $G' \subseteq G$ of potential reachable grasps is determined. In the second step, samples of K_{free} are generated and the resulting configuration is checked for reachability. Depending on the task (one handed or bimanual) at least one grasp for the hand or two grasps, one for each hand, must be reachable. In Fig. 7 the bimanual case is shown, since single-handed tasks can be solved analogous. When a set of reachable grasps is found, the resulting steps are similar to the bimanual IK-approach, where the analytic IK-solvers are queried and the complete solution is checked against collisions.

4. Evaluation

In this section the proposed algorithms are evaluated with the humanoid robot ARMAR-III. Therefore, several simulation setups are defined and an implementation based on the robot simulation environment Simox[24] is used to measure the performance of the different IK-queries. Since randomized algorithms are used, we present averaged results of 100 test runs carried out on a 3 GHz Linux PC.

4.1. *Fixed Robot Base Pose*

In this setup, the humanoid robot ARMAR-III is located in front of an object. Several grasps are pre-defined for the left and the right hand and the proposed IK-approaches are used to find collision-free grasping configurations.

4.1.1. *Single Handed Grasping Tasks (10 DoF)*

As shown in Fig. 8 the robot is located in front of the fridge and a bottle with 150 predefined grasps is placed inside the fridge. In Table 1 an evaluation of the performance is given as averaged results of 100 test runs. In order to serve meaningful results, each IK query was performed with a varied object pose. Therefore, the position of the bottle and its upright orientation was randomly set to varying values. For each test run, the IK-approach served a collision-free solution for three hip and seven arm joints of the robot. As shown in Table 1 the average query time was measured with 54.3 milliseconds and internally 13.5 calls to the analytic IK-solver

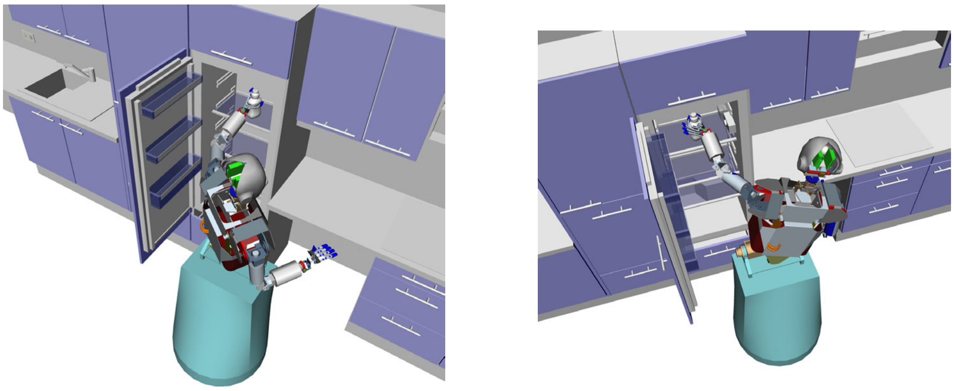

Fig. 8. Two exemplary IK-solutions of the single handed IK task. The 10 DoF solution covers three hip and seven arm joints of ARMAR-III.

for the arm have been performed. There are two reasons why a call to the IK-solver could fail (see Fig.5): Either no IK solution for the current hip configuration exists (this may happen due to the discretized structure of the reachability distribution) or the IK solution is in collision with the environment.

For comparison, the second row of Table 1 shows the results of the same IK query, but without using reachability information within the hybrid IK-solver. Due to the missing reachability information, the offered grasps have to be selected randomly for IK search, resulting in a large number of calls to the analytic IK-solver. The last row shows the results of a Jacobian-based approach, where a grasp is randomly selected and the TCP is moved via the Pseudoinverse Jacobian towards the resulting pose. Additionally collision detection is performed in order to discard configurations that are in collision.

GraspIK 10 DoF	Query time (total)	Discarded due to collisions	Calls to analytic IK solver
with reachability	54.3 ms	81.5 %	13.5
without reachability	511.2 ms	88.1 %	536.3
Jacobain-based	125.1 ms	79.0 %	-

Table 1. 10 DoF: Average results of 100 test runs. The object's position and orientation has been slightly varied for every IK query.

4.1.2. *Bimanual Grasping Tasks (17 DoF)*

Two exemplary results of the bimanual setup can be seen in Fig. 9. The robot is placed in front of the oven and a wok has to be grasped with both hands. A

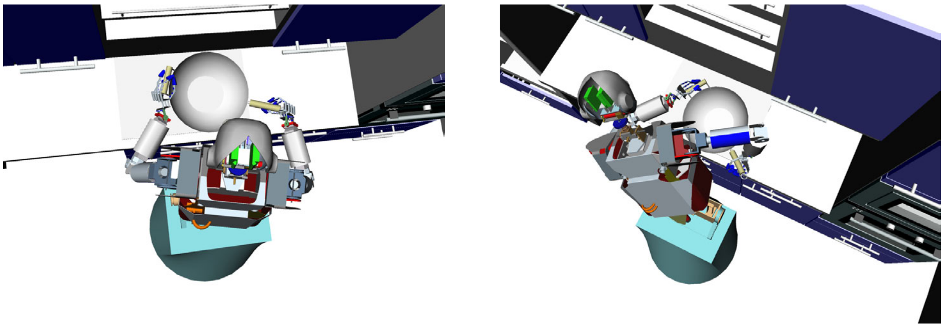

Fig. 9. Two exemplary IK-solutions of the bimanual IK task. The 17 DoF solution covers three hip and 14 arm joints of ARMAR-III.

grasp planner has generated 300 grasps for each end effector in a preprocessing step and IK solutions are queried with the *BimanualGraspIK* algorithm, described in Section 3.3. The 17 DoF joint set considered for this task consists of three hip joints and seven joints for each arm. Again, the object is randomly re-positioned for each IK-query in order to generate differing tasks for the IK-solver. All 100 test runs resulted in a collision-free bimanual IK solution. The average runtime was measured with 237.7 ms and as shown in Table 2, a large number of results were discarded due to collisions. This artifact is caused by the setup of the scene, where the workspace below the target object is blocked by the oven and therefore a large number of grasping configurations resulted in collision. Due to these collisions, 15.4 bimanual robot configurations were generated until an collision-free IK-solution was found. Hence, about 30 successful calls to the analytic IK solver[a] have to be performed for IK results that are in collision. The remaining calls to the analytic IK solver were made for workspace poses that are not reachable. The second row of Table 2 shows the results when no reachability information is present in order to select the set of potential grasps for IK-solving. Here, all grasps are taken into account and random pairs for the left and the right hand are selected until the bimanual IK-query succeeds.

Bimanual GraspIK 17 DoF	Query time (total)	Discarded due to collisions	Calls to analytic IK solver
with reachability	237.7 ms	93.5 %	56.5
without reachability	1839.2 ms	96.4 %	2270.3

Table 2. 17 DoF: Average results of 100 test runs. The top row shows the results of the proposed *BimanualGraspIK* algorithm. The second row shows the results when no reachability information was taken into account for selecting potential pairs of grasps.

[a]Since both arms are considered, at least two calls have to be performed for a bimanual solution.

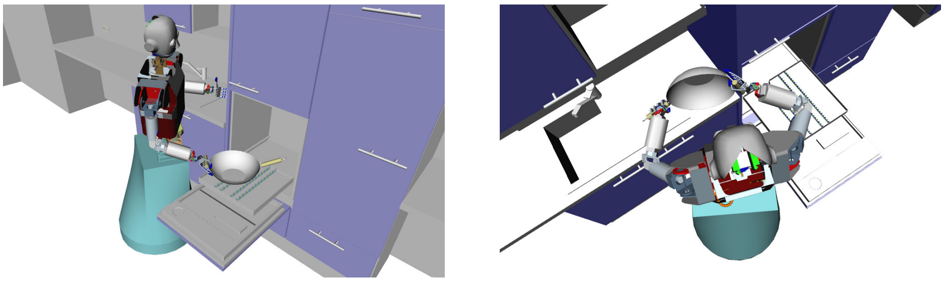

Fig. 10. Re-grasping: The wok is initially grasped with the right hand (left) and an exemplary IK-solution of the bimanual re-grasping task is shown on the right. The 17 DoF solution covers three hip and 14 arm joints of ARMAR-III.

4.1.3. *Re-Grasping Tasks (17 DoF)*

In this setup, a wok should be handed over from the right to the left hand. Therefore, the object is already grasped with the right hand as shown in Fig. 10 on the left. The IK-solver of Section 3.4 is used to determine collision-free IK-solutions for re-grasping considering 15 pre-defined grasps. Since the IK-solver implicitly selects a feasible grasp and a suitable object pose, these values do not have to be specified in advance. The extends for sampling the object's position is restricted to locations in front of the robot within a radius of one arm length, while the sampling of the orientation is not restricted.

Again, all 100 test runs succeeded and the results are shown in Table 3. The average runtime was measured with 111.6 ms.

ReGraspIK 17 DoF	Query time (total)	Discarded due to collisions	Calls to analytic IK solver
with reachability	111.6 ms	72.2 %	7.5

Table 3. Re-grasping with 17 DoF: Average results of 100 test runs.

4.2. *Variable Robot Base Pose*

When the base pose of the robot is not predefined, the IK-algorithms of Section 3.5 can be used in order to find a suitable base poses for grasping. The additional three DoF covering the platform's position and orientation are handled by using IRDs (see Section 2.2).

4.2.1. *Single Handed Grasping Tasks (13 DoF)*

The IK-solvers are used to generate collision-free grasping configurations covering 13 DoF of ARMAR-III, considering the position and the orientation of the plat-

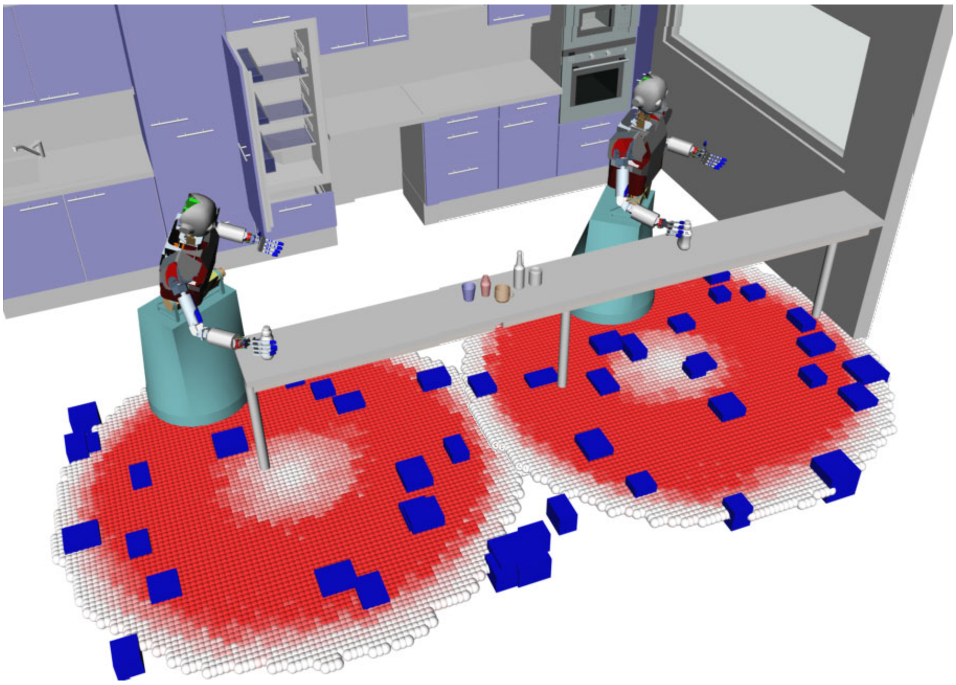

Fig. 11. Two exemplary IK-solutions of the single handed setup. The 13 DoF solution covers the platform position and orientation, three hip and seven arm joints of ARMAR-III. The IRD is shown for 300 automatically generated grasps.

form, three hip and seven arm joints. A grasp planner [24] was used to automatically generate 300 grasps for the target object in an offline step. These grasps are used during IK-search to build the according IRD as described in Section 3.5. For each of the 100 test cases the object was placed randomly on the sideboard. Two results are shown in Fig. 11, where the IK-solution and a visualization of the IRD are depicted. As shown in Table 4, the average runtime was measured with 37.7 ms when no obstacles were positioned on the ground and 71 % of the internally created configurations are discarded due to collisions. These collisions are caused either by an inappropriate positioning of the platform or a collision between the sideboard and the upper body, the arm or the hand was detected. The second line of Table 4 shows the result when 20 randomly placed obstacles are considered additionally to the environment (see Fig. 11). In this setup, the query time increased to 69.2 ms, mainly caused by the high number of configurations that had to be discarded due collisions. The third row shows the results when 30 obstacles were added to the scene. While for the first two setups the IK-solver reported a valid solution for all test runs, the IK-solver failed in 7 % of the tests in this setup (due to the randomized approach, we stopped the IK-query after a timeout of one minute). Since we did not consider any constraints when placing the obstacles, we assume that the obstacles blocked the complete area, causing the IK-search to fail.

GraspIK Base Pose 13 DoF	Query time (total)	Discarded due to collisions	Generated robot base poses
no obstacles	37.7 ms	70.9 %	4.7
20 obstacles	69.2 ms	99.0 %	101.9
30 obstacles	405.3 ms	99.9 %	4654.3

Table 4. Averaged results of the extended 13 DoF IK task that includes the search for a suitable robot base pose.

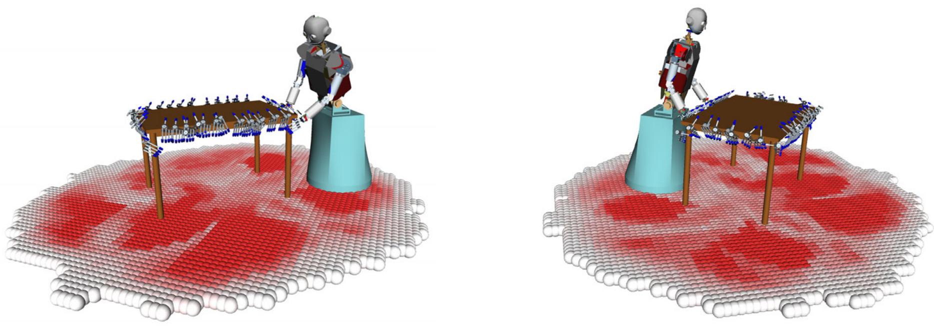

Fig. 12. Two exemplary IK-solutions of the bimanual setup. The 20 DoF solution covers the platform position and orientation, three hip and 14 arm joints of ARMAR-III.

4.2.2. *Bimanual Grasping Tasks (20 DoF)*

In this setup, ARMAR-III is supposed to grasp a table in order to cooperatively solve a transport task. We assume that the robot can choose it's base pose and either another robot or a human will assist in transporting the table. When a cooperative partner has already grasped the table, this can be modeled by selecting a subset of potential grasping configurations and/or performing collision checks w.r.t. the partner. For evaluation we did not consider such constraints in this setup, but we showed how cooperative multi-robot grasping tasks can be performed in earlier work[36]. The results of the evaluation of this setup can be seen in Table 5. Again all IK-queries succeeded and the results are averaged over 100 test runs. The average query time until a collision-free bimanual grasping configuration is reported was measured with 180.6 milliseconds. Internally the IK-algorithm rejects several IK hypothesis since a collision between the robot and the target object is detected (on average 63.2% hypotheses are in collision) and due to the sampling-based approach some of the generated robot base poses do not result in a valid grasping configuration (last column of Table 5).

	Query time (total)	Discarded due to collisions	Generated robot base poses
Bimanual GraspIK Base Pose (20 DoF)	180.6 ms	63.2 %	4.2

Table 5. Results of the 20 DoF bimanual IK task.

5. Conclusion

In this work, we presented efficient algorithms for solving the inverse kinematics problem for single-handed and bimanual grasping tasks in cluttered environments. It was shown, that reachability analysis in combination with sampling-based techniques lead to efficient algorithms even when highly redundant kinematics are considered. The algorithms take care of collisions and implicitly select a reachable grasp out of a set of potential grasping configurations that are offered by an object representation. It was further shown how inverse reachability distributions, representing suitable robot base poses with respect to an object, can be derived efficiently from precomputed reachability distributions. This allows incorporating the search for suitable base poses for grasping into the IK-algorithms due to the sampling-based structure of the approach. The algorithms have been evaluated in several setups, such as grasping, bimanual grasping and re-grasping. The performance evaluation showed that the approach is feasible for real-world applications and that it can be used on real robot systems.

Future work may address pose quality evaluations in order to guarantee natural looking poses. The presented solvers always report the first valid solution, but in case more time can be spent on searching IK-solutions, quality criteria can be taken into account in order to serve the best solution that could have been found within a period of time. The pose quality could also be improved by post-processing steps, where the IK solution is optimized to meet quality constraints. Further, we assume that the approach is probabilistically complete, which has to be proven in future work. This would give a guarantee, that the algorithms are able to report any potential IK-solution, which is not the case for Jacobian-based approaches due to the local minimum problem.

Acknowledgements

The work described in this article 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. D. E. Whitney, "Resolved motion rate control of manipulators and human prostheses," *IEEE Trans. Man Machine Systems*, vol. 10, no. 2, pp. 47–53, Jun 1969.
2. W. Wolovich and H. Elliott, "A computational technique for inverse kinematics," in *The 23rd IEEE Conference on Decision and Control*. IEEE, Dec. 1984, pp. 1359–1363.
3. Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *J. Dyn. Sys. Meas. Control*, vol. 108, no. 3, pp. 163–171, Sep 1986.
4. C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, pp. 93–101, January 1986.
5. S. R. Buss and J.-S. Kim, "Selectively damped least squares for inverse kinematics," *Journal of graphics, gpu, and game tools*, vol. 10, no. 3, pp. 37–49, 2005.
6. M. Mistry, G. Cheng, G. Cheng, and S. Schaal, "Inverse kinematics with floating base and constraints for full body humanoid robot control," *Science And Technology*, vol. pp, pp. 22–27, 2008.
7. A. Liégeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-7, no. 12, pp. 868–871, 1977.
8. A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *The International Journal of Robotics Research*, vol. 4, no. 3, pp. 109–117, 1985.
9. 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.
10. O. Kanoun, F. Lamiraux, P.-B. Wieber, F. Kanehiro, E. Yoshida, and J.-P. Laumond, "Prioritizing linear equality and inequality systems: Application to local motion planning for redundant robots," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 2939–2944.
11. A. Escande, N. Mansard, and P.-B. Wieber, "Fast resolution of hierarchized inverse kinematics with inequality constraints," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 3733–3738.
12. F. Kanehiro, F. Lamiraux, O. Kanoun, E. Yoshida, and J.-P. Laumond, "A local collision avoidance method for non-strictly convex polyhedra," in *Proceedings of Robotics: Science and Systems IV*, Zurich, Switzerland, June 2008, pp. 151–158.

13. C. Lee and M. Ziegler, "Geometric approach in solving inverse kinematics of puma robots," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-20, no. 6, pp. 695–706, nov. 1984.
14. T. Asfour and R. Dillmann, "Human-like motion of a humanoid robot arm based on a closed-form solution of the inverse kinematics problem," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, vol. 2, no. October, 2003, pp. 1407–1412.
15. M. Kallmann, "Analytical inverse kinematics with body posture control," *Comput. Animat. Virtual Worlds*, vol. 19, pp. 79–91, May 2008.
16. R. Konietschke and G. Hirzinger, "Inverse kinematics with closed form solutions for highly redundant robotic systems," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, may 2009, pp. 2945–2950.
17. R. Diankov and J. Kuffner, "OpenRAVE: A planning architecture for autonomous robotics," Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34, Jul. 2008.
18. R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, September 2010.
19. J. Cortes and T. Simeon, "Sampling-based motion planning under kinematic loop-closure constraints," in *In 6th International Workshop on Algorithmic Foundations of Robotics*. Springer-Verlag, 2004, pp. 59–74.
20. A. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, Sept. 2003, pp. 1824–1829 vol.2.
21. 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.
22. 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, pp. 1592–1599.
23. A. T. Miller, "Graspit!: a versatile simulator for robotic grasping." Ph.D. dissertation, Department of Computer Science, Columbia University, 2001.
24. N. Vahrenkamp, T. Asfour, and R. Dillmann, "Simox: A Simulation and Motion Planning Toolbox for C++," Karlsruhe Institute of Technology (KIT), Tech. Rep., 2010.
25. 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.
26. L. Guilamo, J. Kuffner, K. Nishiwaki, and S. Kagami, "Efficient prioritized inverse kinematic solutions for redundant manipulators," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, Aug. 2005, pp. 3921–3926.

27. F. Zacharias, C. Borst, and G. Hirzinger, “Capturing robot workspace structure: representing robot capabilities,” in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2007, pp. 3229–3236.
28. 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.
29. 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.
30. D. Kee and W. Karwowski, “Analytically derived three-dimensional reach volumes based on multijoint movements,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 44, pp. 530–544(15), 2002.
31. N. I. Badler, C. B. Phillips, and B. L. Webber, *Simulating Humans: Computer Graphics Animation and Control.* New York, Oxford: Oxford University Press, 1993.
32. T. Yoshikawa, “Manipulability of robotic mechanisms,” *The International Journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985.
33. S. T. Rachev, M. Höechstöetter, F. J. Fabozzi, and S. M. Focardi, *Probability and Statistics for Finance.* John Wiley & Sons, 2007.
34. F. Stulp, A. Fedrizzi, and M. Beetz, “Learning and performing place-based mobile manipulation,” in *Development and Learning, 2009. ICDL 2009. IEEE 8th International Conference on*, Jun. 2009, pp. 1–7.
35. F. Zacharias, W. Sepp, C. Borst, and G. Hirzinger, “Using a model of the reachable workspace to position mobile manipulators for 3-d trajectories,” in *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, dec. 2009, pp. 55 –61.
36. N. Vahrenkamp, E. Kuhn, T. Asfour, and R. Dillmann, “Planning multi-robot grasping motions,” in *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Nashville, USA, 2010, pp. 593–600.
37. E. Drumwright and V. Ng-Thow-Hing, “Toward interactive reaching in static environments for humanoid robots,” in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2006, pp. 846–851.
38. D. Berenson, S. Srinivasa, D. Ferguson, A. C. Romea, and J. Kuffner, “Manipulation planning with workspace goal regions,” in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2009, pp. 618–624.

Nikolaus Vahrenkamp received his Diploma and Ph.D. degrees from the Karlsruhe Institute of Technology (KIT), in 2005 and 2011, respectively. From 2011 to 2012, he was senior post-doc at the Cognitive Humanoids Lab of the Robotics, Brain and Cognitive Sciences Department, Italian Institute of Technology (IIT), where he worked on grasp and motion planning for the humanoid robot iCub. Currently, he is a postdoctoral researcher at the Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT) and works on software development, grasping and mobile manipulation for the humanoid robots of the ARMAR family.

Nikolaus Vahrenkamp is the author of over 30 technical publications, proceedings and book chapters. He was involved in the European Projects Xperience, GRASP and PACO-PLUS as well as in the collaborative research centre Humanoid Robots (SFB 588) funded by the German Research Foundation (DFG: Deutsche Forschungsgemeinschaft). His research interests include humanoid robots, motion planning, grasping and sensor-based motion execution.

Tamim Asfour received his Diploma and Ph.D. degrees from the Karlsruhe Institute of Technology (KIT), in 1994 and 2003, respectively. From 2003, he was senior research scientist and leader of the Humanoid Research Group at Humanoids and Intelligence Systems Lab, Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT). Since September 2010 he holds an Adjunct Professor position at the Georgia Institute of Technology (Georgia Tech), College of Computing, School of Interactive Computing. He is currently a Full Professor at the Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT).

Tamim Asfour is member of the Editorial Board of IEEE Transactions on Robotics and European Chair of the IEEE-RAS Technical Committee on Humanoid Robots. He is member the Executive Board of the German Association of Robotics (DGR: Deutsche Gesellschaft für Robotik). He is Principle Investigator in the Cognitive Systems projects Xperience, GRASP and PACO-PLUS, funded by the European Commission. His major research interest is humanoid robotics. In particular, his research topics include action learning from human observation, goal-directed imitation learning, dexterous grasping and manipulation, active vision and active touch, whole-body motion planning, cognitive control architectures, system integration, robot software and hardware control architecture, motor control and mechatronics.

Rüdiger Dillmann is full professor at the Karlsruhe Institute of Technology (KIT), Department of Informatics, and head of the Humanoids and Intelligence Systems Laboratory at the Institute for Anthropomatics. He is member of the Executive Board of the Technology Research Transfer Center (FZI). He received his PhD in Computer Science in 1980 and his habilitation on "Learning Robots" in 1986 from the University of Karlsruhe. In 1981-1986 he held an associate professor position at the Institute for Process Control and Robotics (IPR), Karlsruhe. Since 1987 he is full professor at the Department of Informatics at KIT. He is the scientific leader and coordinator of the German collaborative research centre Humanoid Robots (SFB 588), coordinator of the integrated European Cognitive Systems projects PACO-PLUS and Xperience funded by the European Commission.

Rüdiger Dillmann is member of numerous international and national associations such as IEEE/RAS, VDI-GMA. He was member advisory committee of the IEEE Robotics and Automation Society (IEEE-RAS) and general and program chair of several international robotics conferences and member of the technical program committees of numerous international conferences. His major research interests include humanoid robotics and human-centered robotics, learning from human observation and robot programming by demonstration, machine learning, active vision for human motion capture and scene understanding, cognitive cars, service robots, and medical applications of informatics.