Fall 9-13-2021

Precision Grasp using an Arm-Hand System as a Hybrid Parallel-Serial System: A Novel Inverse Kinematics Solution

Shuwei Qiu

Shuwei Qiu Ph.D., P.Eng.
Western University, mkermani@eng.uwo.ca

Precision Grasp using an Arm-Hand System as a Hybrid Parallel-Serial System: A Novel Inverse Kinematics Solution

Shuwei Qiu[1] and Mehrdad R. Kermani[1]

Abstract—In this paper, we present a novel inverse kinematics (IK) solution for a robotic arm-hand system to achieve precision grasp. This problem is kinematically over-constrained and to address the issue and to solve the problem, we propose a new approach with three key insights. First, we propose a human-inspired thumb-first strategy and consider one finger of the robotic hand as the "thumb" to narrow down the search space and increase the success rate of our algorithm. Second, we formulate the arm-thumb serial chain as a closed chain such that the entire arm-hand system is controlled as a hybrid parallel-serial system. The closed-chain formulation simplifies the task hierarchy of the entire arm-hand system. Third, we attach a virtual revolute joint to the thumb's tip with its rotation axis aligning with the thumb's contact normal The virtual joint will embody the thumb's functional redundancy. By selecting the thumb's joints including the added virtual revolute joint as active joints of the arm-thumb closed chain, the arm-thumb system's self-motion (i.e., the palm pose) and the thumb's functional redundancy can be directly controlled without using the null space projection. This provides a new possibility to control the self-motion of arm-hand systems. Simulation results will demonstrate the advantages and superior performance of the proposed approach for achieving precision grasp compared to other classical approaches.

Index Terms—Inverse kinamtics; Arm-hand system; Precision grasp; Thumb-first strategy; Closed-chain system; Virtual revolute joint; Functional redundancy; Self-motion control

I. INTRODUCTION

ROBOTS with multi-finger hands have attracted significant attention in recent years. These systems have a large number of degrees of freedom (DOFs) and ultra-high kinematic redundancies [1]. The high redundancies enable the system to perform multiple simultaneous tasks at the expense of a complex kinematic structure. Depending on the number of tasks, the inverse kinematics (IK) problem for such systems can be either under- or over-constrained.

The bulk of the literature focuses on the under-constrained IK problems with redundant systems. The challenge is uti-
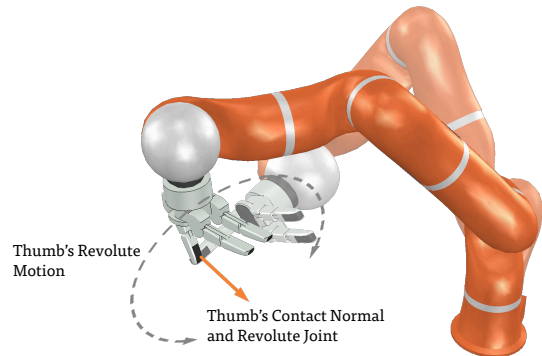
Fig. 1. Virtual revolute joint of the thumb on arm-thumb serial chain

lizing redundancies to achieve multiple tasks while considering the singularity/stability and solvability/reachability. In general, simultaneous execution of multiple tasks can be achieved using a weighting matrix or constructing a task hierarchy [2]. The weighting matrix can be imposed either in the Cartesian or joint space. Sugihara [3] proposed a Levenberg-Marquardt based IK solution using the error information with a small bias as the damping factor. In this method, a weighting matrix was imposed on the task errors to reflect the task priorities and absorb the physical metric differences between different tasks. Chan and Dubey [4] proposed an influential solution for joint limit avoidance by imposing a weighting matrix in the joint space. Flacco and Luca [5] proposed a reverse priority method to achieve multi-task control of a redundant robot, in that the solution for lower-priority tasks was computed followed by the one for higher-priority tasks. Jarquin *et al.* [6] utilized a weighting method to smoothly modify the task priorities within a hierarchical IK problem. Lee *et al.* [7] and Hu *et al.* [1] utilized the relative Jacobian formulation to design a multi-task control framework for a dual-arm robotic system, in that the conflicting tasks were handled using different priorities.

To achieve a precision grasp, with specific contact points and contact normals, the hand's IK is often solved first to find a 6D pose for the end-effector [8][9]. The IK solution for the robot arm is subsequently solved. Liu *et al.* [10] proposed a mixed-integer conic programming formulation for the hand's IK problem and solved it using a branch-and-bound algorithm. Gori *et al.* [11] formulated the hand's IK problem as an optimization problem which was solved using a nonlinear optimization algorithm [12]. Rosales *et al.* [13] proposed a

method to identify all possible hand configurations for a given set of grasp points based on position analysis of the linkages. The authors extended this work by formulating the IK problem of a robotic hand as a system of polynomials solved using a linear relaxation-based technique [14].

There are two inherent limitations in the separate consideration of the robotic arm and the hand's IK problem. First, the end-effector pose obtained from the hand's IK problem is not guaranteed to be feasible for the arm. Second, the error of the hand's IK solution is often exacerbated by the error of the arm's IK solution, considering that the realization of the fingers' contact points and contact normals relies on a shared arm. To overcome these two disadvantages, we consider the robotic arm and the hand together as an integrated system in this work. The IK problem of an integrated arm-hand system is over-constrained. As an example, let us consider the KUKA-Barrett system shown in Fig. 1, which has 14 degrees of freedom (DOFs) in total assuming all joints are controllable (see Fig. 2). There are 15 degrees of constraints (DOCs) imposed by the desired contact points and contact normals (9 DOCs for the desired contact points and 6 DOCs for the desired contact normals). For over-constrained IK problems, the number of solutions is quite limited. This condition dramatically increases the difficulty of obtaining the IK solution for such high-DOF systems. Task conflicts almost always exist in the IK solution of an arm-hand system. To this effect, a well-known strategy is constructing a task hierarchy with the null space projection technique. The highest-priority task is achieved by exploiting all DOFs in the system while lower-priority tasks are executed within the null space of the higher-priority tasks. Consequently, all tasks except those at the top of the hierarchy can be achieved sub-optimally to not interfering with higher-level tasks. To improve the accuracy of lower-priority tasks, task transition [6][15] and null space shaping techniques [16] have been applied to hierarchical IK problems. These techniques provide a partial solution since they are essentially a trade-off between different tasks' accuracy and a compromised treatment of the problem. An extended Jacobian formulation [17] and relative Jacobian formulation [1][7] have also been applied to handle task conflicts. But the extended Jacobian formulation only suits under-constrained IK problems and the relative Jacobian formulation is more suitable for handling task conflicts of systems composed of multiple redundant manipulators such as dual-arm systems. Closed-form solutions [18] have also been applied to complex robotic systems for parts of the kinematic chain. A closed-form solution is usually not available for a general complex robotic system. In our previous work, the solution of an over-constrained IK problem for an arm-hand system using a hierarchical approach was studied [19]. In this work, the task hierarchy was constructed for the position and orientation of all fingers and the orientation of the palm. Within this hierarchy, thumb-related tasks had higher priorities and the tasks related to other fingers were solved within the null space of the thumb and the palm using the null solution space projection technique. Consequently, sub-optimal solutions with less than

ideal success rate were achieved. In this paper, the IK problem of an arm-hand system for achieving precision grasps with a focus on handling conflicts among different fingers is studied. We propose a human-inspired thumb-first strategy to narrow down the search space and use a hybrid parallel-serial formulation, instead of task hierarchy, to address task conflicts among different fingers without compromising their accuracy.

We consider one finger in the robot hand as the opposing "thumb" and locate the thumb at its desired pose first. This approach enforces meaningful constraints on other fingers and narrows down the IK search space significantly. Our approach is inspired by human grasping where the thumb is often used to counterbalance other fingers. The use of the thumb is unique to humans and some primates and it is infrequent in nature [20][21]. The significance of the thumb in grasping has been highlighted in other studies [22], [23].

Following the thumb's desired pose, we then consider the arm-thumb serial chain as a closed chain (see Fig. 1). The entire arm-hand system is controlled as a hybrid parallel-serial system. In the arm-thumb closed chain, the arm and the thumb are regarded as the supporting "legs" whereas the hand palm is regarded as the end-effector. In this approach, the thumb's tip is fixated and the palm's pose is controlled in the thumb's null space without the use of null space projection, which simplifies the task hierarchy for the entire system.

In addition to kinematics redundancies due to extra DOFs, the redundancies due to task definitions also exist. Some tasks do not require all six DOFs in the Cartesian space to be controlled. As a result, some extra degrees of redundancy appear in the robotic system, which is often referred to as functional redundancy [24]. In the case of precision grasps, while there are requirements for contact points and contact normals, the rotation around the contact normals is redundant. This rotation is an example of the fingers' functional redundancy. The functional redundancy is commonly exploited in null space shaping and projection [24]. This is an indirect and passive use of functional redundancy and it may affect the performance of the tasks whose execution relies on the functional redundancy. In this paper, we also propose a new method to directly and actively utilize the thumb's functional redundancy. To this end, we attach a virtual revolute joint to the thumb's tip with its rotation axis aligning with the thumb's contact normal (see Fig. 1). This virtual joint embodies the thumb's functional redundancy. We can then directly control the thumb's functional redundancy by selecting this virtual joint as one of the active joints in the arm-thumb closed chain.

The contributions of this work include 1) A human-inspired thumb-first strategy that dramatically narrows down the IK search space is implemented. 2) The use of a virtual revolute joint at the thumb's tip to incorporate the thumb's functional redundancy is proposed. 3) The arm-hand system is formulated as a parallel-serial hybrid system in that the arm-thumb is formulated as a closed chain. 4) Our approach provides new possibilities for explicitly utilizing the self-motion and the functional redundancy of the arm-thumb
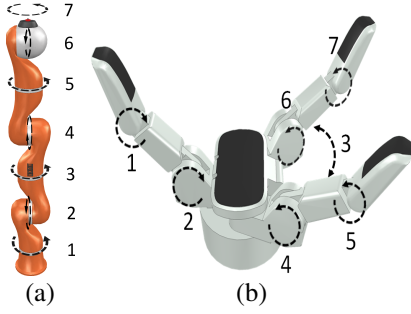
(a) (b)

Fig. 2. 14 DOFs (a) KUKA LWR 4+ manipulator and (b) Barrett hand

system by including and treating the thumb's virtual revolute joint as an active joint of the arm-thumb closed chain.

The rest of this paper is structured as follows. Section II provides the details of the approach. Section III evaluates the performance of the proposed approach and validates its effectiveness through comprehensive simulation results. Finally, Section IV concludes the paper.

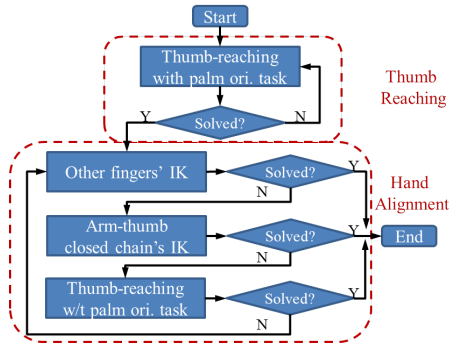## II. PROPOSED APPROACH

A. Overview

Fig. 3. The pipeline of the proposed algorithm

To explain the working principle of the proposed approach, let us consider the KUKA LWR 4+ manipulator and the Barrett hand as an example of an arm-hand system. We assume all joints are controllable and the KUKA-Barrett system has 14 DOFs in total (see Fig. 2). The fingers are labelled as Finger 1, 2 and 3 as depicted in Fig. 4 and their contact normal directions on the fingertips are denoted as $\vec{n}_i$, $(i = 1,2,3)$. In this work, Finger 1 is regarded as the thumb. The pipeline of the proposed algorithm is shown in Fig. 3. The goal is to obtain a solution for a precision grasp with desired contact points and contact normals for all fingers. The proposed algorithm consists of two phases: the Thumb Reaching phase and the Hand Alignment phase. In the following, the details of these two phases are presented.

B. Thumb Reaching Phase

To implement the thumb-first strategy, a conventional hierarchical inverse kinematics (HIK) procedure with successive null space projection is conducted. The Thumb Reaching phase contains three tasks including (1) palm orientation task to achieve the palm's desired orientation (highest priority), (2) thumb position task to achieve the thumb's desired contact
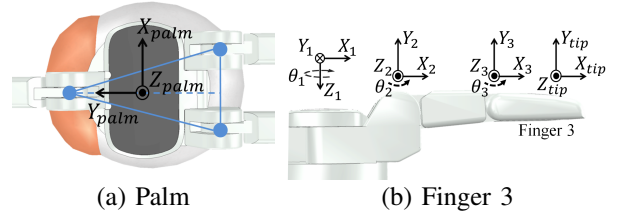

(a) Palm (b) Finger 3

Fig. 4. Coordinate frame assignment of the Barrett hand

point, and (3) thumb orientation task to achieve the thumb's desired contact normal (lowest priority). The inclusion of the palm orientation task in this phase will facilitate obtaining a solution. Since we formulate the problem as a serial-parallel chain, one should note that the workspace of a parallel robot is quite limited. Without the palm orientation task, Finger 2 and 3 will be unlikely close to their desired poses when thumb-related tasks are achieved. Thus, a big motion of the arm-thumb closed-chain would be required to minimize the fingers' errors, and it will most likely move beyond its workspace boundary. To avoid this situation, the palm orientation task is included in the Thumb Reaching phase to facilitate obtaining a solution in the following phase.

1) Desired palm orientation: The desired palm orientation is not a task requirement. and is estimated based on the hand structure. For the Barrett hand with the coordinate frames shown in Fig. 4(a), the palm's frame is assigned such that its *z*-axis, Z_{palm}, aligns with the palm's outward normal, its *x*-axis, X_{palm}, is parallel to the base of the isosceles triangle formed by three fingers' base points, and its *y*-axis, Y_{palm}, points toward the thumb's base point along the median of the isosceles triangle. Using these definitions, the axes of the palm coordinate frame can be represented as,

$$X_{palm} = p_{2_b} - p_{3_b}, \ Y_{palm} = p_{1_b} - \frac{p_{2_b} + p_{3_b}}{2} \quad (1)$$

$$Z_{palm} = X_{palm} \times Y_{palm}, \ X_{palm} = Y_{palm} \times Z_{palm} \quad (2)$$

where p_{i_b} is the base point of Finger i ($i = 1,2,3$), and the conditions in (2) are to guarantee the axes' orthogonality. Since, the fingers' base points are also not a task requirement, their values can be substituted in (1) and (2) with the fingers' desired contact points. Although the desired palm orientation may be ill-estimated by this simple method, it would still significantly reduce the motion of the arm-thumb closed chain. The palm orientation can be corrected in the Hand Alignment phase. Next, we describe how to solve the HIK in the Thumb Reaching phase.

2) Thumb Reaching phase HIK solution: The arm-thumb system is considered as a serial chain and its Jacobian is obtained using the Denavit-Hartenberg (D-H) method. The joint movements for achieving the three tasks (i.e., "palm orientation", "thumb position", and "thumb orientation") are solved independently using damped least-squares methods. The resulting joint movements are combined to obtain the total joint movements using successive null space projection.

For the "palm orientation" task, the error (denoted by $\vec{e}_{hp_o}$) is defined as the difference between the desired and the current palm orientations expressed in Euler angles, i.e., $\vec{e}_{hp_o} \triangleq [\alpha_d, \beta_d, \gamma_d]^\top - [\alpha_c, \beta_c, \gamma_c]^\top$. The corresponding null

space projector is obtained as $N_{hp} = I - J_{hp}^{\dagger} J_{hp}$ where J_{hp} is the analytical Jacobian of the palm orientation.

As for the remaining tasks, the thumb's position and orientation error (denoted by $\vec{e}_{t_p}$ and $\vec{e}_{t_o}$, respectively) are defined as,

$$\vec{e}_{t_p} \triangleq \vec{p}_{t_d} - \vec{p}_{t_c}, \quad \vec{e}_{t_o} \triangleq \varphi_t \frac{\vec{n}_{t_c} \times \vec{n}_{t_d}}{\|\vec{n}_{t_c} \times \vec{n}_{t_d}\|} \quad (3)$$

where $\vec{p}_{t_c}$ and $\vec{p}_{t_d}$ are the thumb current and desired contact points, $\vec{n}_{t_c}$ and $\vec{n}_{t_d}$ are the thumb current and desired contact normasl, and $\varphi_t = 2\arctan2(\|U\|,\|V\|)$ is the angle between $\vec{n}_{t_c}$ and $\vec{n}_{t_d}$, in that $U = \|\vec{n}_{t_d}\|\vec{n}_{t_c} - \|\vec{n}_{t_c}\|\vec{n}_{t_d}$ and $V = \|\vec{n}_{t_d}\|\vec{n}_{t_c} + \|\vec{n}_{t_c}\|\vec{n}_{t_d}$ [24][25]. The Jacobian matrices related to the "thumb position" and "thumb orientation" tasks are denoted by J_{t_p} and J_{t_o}, respectively. The total joint movement is obtained by $\Delta\vec{q} = \Delta\vec{q}_{hp} + N_{hp}\Delta\vec{q}_{t_p} + N_{hp}N_{t_p}\Delta\vec{q}_{t_o}$, where N_{t_p} is the null space of the "thumb position" task and $\Delta\vec{q}_{hp}$, $\vec{q}_{t_p}$, and $\vec{q}_{t_o}$ are the independent joint movements to achieve the three tasks.

C. Hand Alignment Phase

With the termination of the Thumb Reaching phase, the Hand Alignment phase begins obtaining a solution for tasks related to other fingers (i.e., Finger 2 and 3) while maintaining the thumb's pose. In this phase, the arm-thumb serial chain is formulated and controlled as a closed chain. Meanwhile, the remaining fingers are controlled separately as serial chains. Hence, the arm-hand system is controlled as a hybrid parallel-serial mechanism.

As shown in Fig. 3, three steps are executed in sequence in each iteration of this phase which includes (1) solving IK for remaining fingers', (2) solving the arm-thumb closed chain IK, and (3) solving Thumb Reaching without the "palm orientation" task. Step (1) is to minimize the remaining fingers' errors under the current palm pose. If the fingers' errors are still unacceptable after Step (1), Step (2) is conducted to adjust the palm's pose to reduce the remaining fingers' errors. Step (3) is repeats of the Thumb Reaching phase without the "palm orientation" task to keep the thumb at its achieved pose. The details of Step (1) and (2) are explained next.

1) IK of remaining fingers: Let us consider Finger 3 as an example, with the coordinate frame shown in Fig. 4(b). The direction of its contact normal (i.e., Y_{tip}) with respect to the hand palm can be calculated using the forward kinematics as $Y_{tip} = \begin{bmatrix} -s_1 s_{23}, & c_1 s_{23}, & c_{23} \end{bmatrix}^{\top}$ where $s_1 = \sin(\theta_1)$, $s_{23} = \sin(\theta_2 + \theta_3)$, $c_1 = \cos(\theta_1)$, and $c_{23} = \cos(\theta_2 + \theta_3)$ in that θ_i ($i = 1,2,3$) is the joint variable of Finger 3. The value of θ_1 and $\theta_2 + \theta_3$ can be obtained from the desired contact normal expressed in the current palm's frame. The IK for the position of this finger expressed in the palm's frame can be then expressed as a constrained nonlinear minimization problem with the constraints being the values of θ_1 and $\theta_2 + \theta_3$ (denoted as θ_{1_d} and θ_{23_d}, respectively) as well as

their joint limits, i.e.,

$$\min_{\theta_2,\theta_3} \| {}^{hp}\vec{p}_d - {}^{hp}\vec{p}_c\|^2 \quad (4)$$

$$\text{s.t. } \theta_1 = \theta_{1_d}, \;\; \theta_2 + \theta_3 = \theta_{23_d}$$

$$\theta_2, \theta_3 \in [\theta_{i_{min}}, \theta_{i_{max}}] \qquad i = 2,3$$

where ${}^{hp}\vec{p}_d$ and ${}^{hp}\vec{p}_c$ are the desired and the current contact point of the finger expressed in the current palm's frame. The "fmincon" function provided by MATLAB was used to obtain a solution for (4).

2) The arm-thumb closed chain: If the errors for the remaining fingers are unacceptable after the previous step, it is an indication of an ill-estimated palm orientation. To adjust the palm's pose without interfering with the previous thumb-related tasks, we formulate the arm-thumb system as a closed chain (see Fig. 1) with the palm being the end-effector and the thumb and the arm being the two supporting "legs". A virtual revolute joint is attached to the thumb's tip with the rotation axis aligning with the thumb's contact normal to realize the thumb's functional redundancy. In this closed-chain formulation, the thumb must be controlled reversely (i.e., the thumb's tip and the virtual revolute joint must be the base and the first joint of the "leg", respectively), otherwise, the thumb's joints including the virtual joint would not affect the palm's pose. Using the thumb's virtual joint and its other joints as active joints, the thumb's self-motion and its functional redundancy can be exploited without using null space projection.

3) Jacobian formulation of the arm-thumb closed chain: Following the procedure presented in [26], we formulate the Jacobian of the arm-thumb closed chain based on screw theory as follows. The kinematic constraint in this closed chain is,

$$J_{t_r}\vec{q}_t = J_{arm}\vec{q}_{arm} \Leftrightarrow [J_{t_r} \; -J_{arm}] \begin{bmatrix} \vec{q}_t^{\top} & \vec{q}_{arm}^{\top} \end{bmatrix}^{\top} = \vec{0} \quad (5)$$

where J_{t_r} and J_{arm} are the screw-based body Jacobian of the reversed thumb and the arm, respectively, and $\vec{q}_t$ and $\vec{q}_{arm}$ are the corresponding joint variables. For the KUKA-Barrett system, since the mobility of the arm-thumb closed chain is four, based on Grübler's formula [27], each "leg" is required to have at least four DOFs otherwise the end-effector singularity would appear. But the thumb has only three DOFs (including the virtual joint). To prevent potential end-effector singularities and maintain the constraint (5), we mathematically consider the 7th joint of KUKA (denoted as q_{arm_7}) as part of $\vec{q}_t$ rather than $\vec{q}_{arm}$.

Since the mobility of the arm-thumb closed chain is four, four active joints are required. Although any joints can be chosen as active joints in general, it is suggested to select the joints whose rotation axes are not co-planar to avoid possible singularities caused by the selection of active joints [26]. We chose the virtual revolute joint (q_v), the thumb's 2nd joint (q_{t_2}), and the arm's 7th and 6th joint (q_{arm_7} and q_{arm_6}) as the active joints. After rearranging the joints into the groups of active joints and passive joints, (5) can be re-written as,

$$[H_a \; H_p] \begin{bmatrix} \vec{q}_a^{\top} & \vec{q}_p^{\top} \end{bmatrix}^{\top} = \vec{0} \Leftrightarrow \vec{q}_p = -H_p^{-1}H_a\vec{q}_a = H\vec{q}_a \quad (6)$$

where $\vec{q}_a = [q_v, q_{t_2}, q_{arm_7}, q_{arm_6}]^{\top}$ and $\vec{q}_p =$

$[q_{t_1}, q_{arm_1}, \ldots, q_{arm_5}]^\top$ are the active and passive joint variables of the arm-thumb closed chain, respectively, and q_{t_i} ($i = 1, 2$) and q_{arm_i} ($i = 1, \ldots, 7$) are the joint variables of the thumb and the arm, respectively. The arm-thumb closed chain's Jacobian (J_{cc}) is then obtained from the arm's Jacobian (excluding the 7th joint) as,

$$J_{cc} = J_{arm} \left[\vec{h}_1^\top, \ldots, \vec{h}_5^\top, \vec{e}_6^\top \right]^\top \tag{7}$$

where $\vec{h}_i$ ($i = 1, \ldots, 5$) is the i-th row of the H matrix in (6) and $\vec{e}_6 = [0, 0, 0, 1]$. After constructing the arm-thumb closed chain's Jacobian matrix, we need to find the palm's desired motion to minimize other fingers' errors. We describe the palm's desired motion by the difference between its desired and current position and orientation (denoted as $\vec{e}_{hp_p}$ and $\vec{e}_{hp_o}$, respectively). $\vec{e}_{hp_p}$ and $\vec{e}_{hp_o}$ are computed from other fingers' errors.

4) Calculation of the palm's rotational motion $\vec{e}_{hp_o}$: Let us denote $_d^c R_{hp}$ as the rotation matrix from the palm's current orientation to its desired orientation. Suppose other fingers' orientation errors can be eliminated with the palm's desired orientation, we have the following equations,

$$_{hp}^b R_c \cdot _d^c R_{hp} \cdot _{F_2}^{hp} R_c = _{hp}^b R_c \cdot _{F_2}^{hp} R_c \cdot _d^c R_{F_2} \tag{8}$$

$$_{hp}^b R_c \cdot _d^c R_{hp} \cdot _{F_3}^{hp} R_c = _{hp}^b R_c \cdot _{F_3}^{hp} R_c \cdot _d^c R_{F_3} \tag{9}$$

where $_{hp}^b R_c$ and $_{F_i}^{hp} R_c$ are the current rotation matrix from the robot base to the hand palm and that from the palm to the tip of Finger i ($i = 2, 3$), respectively. $_d^c R_{F_i}$ is the rotation matrix from the current to the desired contact normal of Finger i (denoted as $\vec{n}_{i_c}$ and $\vec{n}_{i_d}$, respectively) which is calculated as,

$$_d^c R_{F_i} = I_3 + \hat{w} \sin(\zeta) + \hat{w}^2 (1 - \cos(\zeta)) \tag{10}$$

where $\vec{w} = \vec{n}_{i_c} \times \vec{n}_{i_d}$, $\hat{w} = \text{skew}\left(\frac{\vec{w}}{\|\vec{w}\|}\right)$ in that skew($\cdot$) is the operator for calculating the skew-symmetric matrix from a 3×1 vector, and $\zeta = \arccos\left(\frac{\vec{n}_{i_c} \cdot \vec{n}_{i_d}}{\|\vec{n}_{i_c}\| \|\vec{n}_{i_d}\|}\right)$, ($i = 2, 3$). From (8) and (9), we can obtain,

$$\underbrace{\begin{bmatrix} I_3 \\ I_3 \end{bmatrix}}_{A} {}_d^c R_{hp} = \underbrace{\begin{bmatrix} _{F_2}^{hp} R_c \cdot _d^c R_{F_2} \cdot _{F_2}^{hp} R_c^\top \\ _{F_3}^{hp} R_c \cdot _d^c R_{F_3} \cdot _{F_3}^{hp} R_c^\top \end{bmatrix}}_{B} \tag{11}$$

The least squares solution of (11) is given by $_d^c R_{hp} = (A^\top A)^{-1} A^\top B$. To guarantee $_d^c R_{hp} \in SO(3)$, singular value decomposition (SVD) is applied. The palm's rotational motion ($\vec{e}_{hp_o}$) is obtained as $\vec{e}_{hp_o} = \text{unskew}\left(\log {}_d^c R_{hp}\right)$, where unskew($\cdot$) is the operator for attaining the corresponding 3×1 vector from a skew-symmetric matrix.

5) Calculation of the palm's translational motion $\vec{e}_{hp_p}$: $\vec{e}_{hp_p}$ is obtained by averaging the palm's translational motions for minimizing the errors of Finger i (denoted by $\vec{e}_{hp_{i_p}}$, $i = 2, 3$) as follows,

$$\vec{e}_{hp_p} = \frac{1}{2} \sum_{i=1}^{2} \vec{e}_{hp_{i_p}}, \; \vec{e}_{hp_{i_p}} = \vec{e}_{F_{i_p}} - \vec{e}_{F_{i_o}} \times {}_{F_i}^{hp}\vec{t} \tag{12}$$

where $\vec{e}_{F_{i_p}}$ and $\vec{e}_{F_{i_o}}$ are the position and orientation errors of Finger i ($i = 2, 3$), and $_{F_i}^{hp}\vec{t}$ ($i = 2, 3$) is the translation vector from the palm to the fingertip. $\vec{e}_{F_{i_p}}$ and $\vec{e}_{F_{i_o}}$ ($i = 2, 3$) are

calculated by

$$\begin{bmatrix} [\vec{e}_{F_{i_o}}] & \vec{e}_{F_{i_p}} \\ \underline{0} & 0 \end{bmatrix} = \log \begin{bmatrix} _d^c R_{F_i} & _{F_i}^{hp} R_c^\top {}_d^c\vec{t}_{F_i} \\ \underline{0} & 1 \end{bmatrix} \tag{13}$$

where $[\vec{e}_{F_{i_o}}]$ is the skew-symmetric matrix corresponding to $\vec{e}_{F_{i_o}}$ ($i = 2, 3$), $_d^c R_{F_i}$ is calculated from $\vec{n}_{i_c}$ and $\vec{n}_{i_d}$ as per (10), and $_d^c\vec{t}_{F_i} = \vec{p}_{i_d} - \vec{p}_{i_c}$ in that $\vec{p}_{i_d}$ and $\vec{p}_{i_c}$ are the desired and current contact point of Finger i ($i = 2, 3$). Note that $\vec{n}_{i_c}$, $\vec{n}_{i_d}$, $\vec{p}_{i_c}$ and $\vec{p}_{i_d}$ are expressed in the palm's frame since screw-based body Jacobian is used for the arm-thumb closed chain.

6) IK of the arm-thumb closed chain: After obtaining the palm's desired motions ($\vec{e}_{hp_p}$ and $\vec{e}_{hp_o}$), we now need to find the joint movements of the arm-thumb closed chain to conduct these motions. For the active joints' movement ($\Delta \vec{q}_a$), the weighted damped least-squares method [4] with null space projection is employed. The position and orientation IK of the arm-thumb closed chain are solved independently and the resulting joint movements are combined by using null space projection to obtain $\Delta \vec{q}_a$ as $\Delta \vec{q}_a = \Delta \vec{q}_{a_p} + N_{hp_p} \Delta \vec{q}_{a_o}$, where $N_{hp_p} = I - J_{cc_p}^\dagger J_{cc_p}$ and J_{cc_p} are the null space projector of the palm's position and the arm-thumb closed chain's position Jacobian, respectively, and $\Delta \vec{q}_{a_p}$ and $\Delta \vec{q}_{a_o}$ are the active joints' movement to minimize $\vec{e}_{hp_p}$ and $\vec{e}_{hp_o}$, respectively. After obtaining $\Delta \vec{q}_a$, we calculate the passive joints' movement ($\Delta \vec{q}_p$) as per (6). It is noteworthy that the calculated joint movements related to the thumb (including the virtual joint and the arm's 7th joint) in $\Delta \vec{q}_a$ and $\Delta \vec{q}_p$ have to take the opposite values of the calculated ones since the thumb is reversely controlled in the arm-thumb closed chain.

To prevent the arm-thumb closed-chain formulation from collapsing, we took two precautions in the weighted damped least-squares method [4]. First, we selected a considerably large damping factor to limit the joint motions of the arm-thumb closed chain. Second, a large weighting factor is assigned to the virtual revolute joint since we observed that this virtual joint affected the palm's motion more than other active joints.

III. SIMULATIONS

To test the performance of the proposed approach, a series of offline numerical tests were conducted. All tests were conducted in MATLAB and visualized in V-REP [28]. The kinematic model of the KUKA-Barrett system was used during these experiments. The unit of position and orientation error are mm and rad, respectively. The steps of the simulations are listed below,

1) The initial configuration of the KUKA-Barrett system for each run was set as shown in Fig. 5(a).
2) The hand configurations for grasping 6 different objects from the well-known YCB dataset [29] were found as shown in Fig. 5(b)-(g).
3) For each hand configuration, 10,000 arm configurations that were uniformly distributed within the arm's joint limits were randomly generated.
4) For each hand configuration with the corresponding 10,000 arm configurations, the forward kinematics (FK)
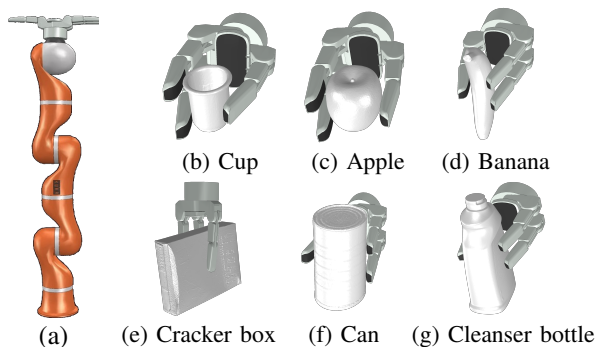
Fig. 5. (a) The initial arm-hand configuration (b)-(g) Barrett hand configurations for grasping different objects.

to generate 10,000 different sets of feasible contact points and contact normals were calculated.

5) The IK of the integrated KUKA-Barrett system to achieve the contact points and contact normals obtained in the previous step was solved.

6) The data was collected and analyzed.

In the proposed method, the classic damped least-squares (DLS) method [30] was employed to solve the IK of each independent task. The damping factor (λ) was set to be 1 and 150 for the arm-thumb serial and closed chain, respectively. The weighting factor for the virtual revolute joint was set to 10 while the factors for other active joints of the arm-thumb closed chain were set to 1. For comparison, we implemented the damped Levenberg-Marquardt method [3] (referred to as "EDLM" hereafter), as well as our previously proposed hierarchical IK solution for arm-hand systems [19] (referred to as "HIK-ArmHand" hereafter). For EDLM [3], the bias matrix in the damping factor ($\bar{W}_N$) was set to a diagonal matrix with all diagonal entries being 10. To reflect task priorities and absorb the physical metric differences, the error weighting matrix in EDLM (W_E) was set as a diagonal matrix, where the diagonal entries corresponding to the position tasks were set to 6×10^{-4}, 4×10^{-4}, and 2×10^{-4} for Finger 1, 2, and 3, respectively, and the diagonal entries corresponding to the orientation tasks were set as 5, 3, and 1 for Finger 1, 2, and 3, respectively. For HIK-ArmHand [19], the DLS method [30] was used to solve the IK of each task with λ being 1. As for other parameters, the maximum number of iterations was set to 10,000 for each run, and the error tolerance for position and orientation was set to 5 mm and 0.0524 rad ($3°$), respectively.

The results of the proposed, HIK-ArmHand and EDLM methods are summarized in Table I. All data were acquired using MATLAB r2019b on a personal computer powered by an i5-6400 CPU @ 2.70GHz and 16GB RAM. In this table, "Success Rate" is defined to be the ratio between the number of solved cases and the total number of cases (i.e., 10,000). For each run, one case is considered to be solved if the magnitudes of all the fingers' position and orientation errors were less than or equal to the error tolerances. The "Average Iterations" and "Average Time" are the average number of iterations and the average computation time consumed over all solved cases, respectively.

As observed in Table I, our proposed method significantly outperforms other implemented methods thanks to the thumb-first strategy and the arm-thumb closed-chain formulation. The thumb-first strategy contributes to the efficiency of our approach by dramatically narrowing down the search space. The arm-thumb closed-chain formulation allows our approach to actively utilize the null space of the thumb by controlling the palm as the end-effector of the arm-thumb closed chain, which contributes to the accurate achievement of other fingers' tasks. In addition, we observed that the virtual revolute joint rotated $0.67°$ in average with a standard deviation of $1.26°$ among the solved cases. This observation verifies that the exploitation of the thumb's functional redundancy and the adjustment of the estimated palm orientation contributed to the high success rate of the proposed approach. As a comparison, HIK-ArmHand [19] passively utilized the thumb's null space by the null space projection technique to conduct other fingers' tasks. This passive null space exploitation hampered the accomplishment of other fingers' tasks, which is reflected in the lower success rate of the HIK-ArmHand method [19]. During the algorithm execution, not considering the conflicting requirements would lead to system oscillation caused by algorithmic singularities which affected the convergence speed, or even non-convergence with non-negligible errors. These consequences were manifested in the unsatisfactory performance of EDLM.

Numerical tests also revealed the limitations of our proposed solution. The arm-thumb closed-chain formulation confronts three challenges including (1) a narrow workspace, (2) an unknown range of the virtual revolute joint, and (3) numerous singularities. In this work, the treatments for these challenges were relegated to the regularization with large damping and weighting factors and the compensation of extra thumb-related tasks, respectively. But these treatments cannot guarantee the stability of the arm-thumb closed chain. Furthermore, although large damping and weighting factors refrained the arm-thumb closed chain from unexpected big motions, it lowered the convergence speed. We believe the success rate and the accuracy of the proposed algorithm can be further improved after one will thoroughly study the workspace, the virtual revolute joint, and the singularities of the arm-thumb closed chain. In addition, collision detection is currently not involved in the proposed solution, which may cause hand-object penetration when dealing with objects with complex shapes. This problem can be resolved by employing collision detection techniques in the process of IK solving.

IV. CONCLUSIONS

In this paper, a novel inverse kinematics (IK) solution was proposed to achieve precision grasps using an integrated arm-hand system. A human-inspired thumb-first strategy was proposed to narrow down the search space. More importantly, we proposed to formulate the arm-thumb serial chain as a closed chain and attached a virtual revolute joint to the thumb's tip as the embodiment of the thumb's functional redundancy. By selecting the thumb's joints including the

TABLE I
COMPARISON BETWEEN IMPLEMENTED METHODS

Objects	Cup			Apple			Banana		
Methods	Proposed	HIK-ArmHand	EDLM	Proposed	HIK-ArmHand	EDLM	Proposed	HIK-ArmHand	EDLM
Success Rate (%)	**96.96**	70.77	42.37	**95.33**	80.58	21.76	**96.77**	77.48	44.68
Avg Time (s)	0.212	0.206	0.121	0.253	0.209	0.155	0.335	0.243	0.126
Ave Iterations	170.87	266.9	342.6	169.70	271.8	438.3	165.77	314.3	352.3
Objects	Cracker box			Can			Cleanser bottle		
Methods	Proposed	HIK-ArmHand	EDLM	Proposed	HIK-ArmHand	EDLM	Proposed	HIK-ArmHand	EDLM
Success Rate (%)	**94.49**	80.18	21.91	**94.72**	81.08	11.37	**96.90**	79.24	28.80
Avg Time (s)	0.320	0.239	0.163	0.612	0.270	0.199	0.329	0.206	0.149
Avg Iterations	184.45	311.4	464.1	215.26	350.7	569.7	178.82	268.6	425.0

added virtual revolute joint as active joints in the arm-thumb closed chain, we were able to directly control the arm-thumb system's self-motion and the thumb's functional redundancy without using the null space projection technique. Our approach provides new possibilities to control the self-motion of a robot manipulator. Simulation results demonstrated the superior performance of our proposed approach.

To further improve this approach, our future work will include the following aspects. The workspace and the singularities of the arm-thumb closed chain will be thoroughly studied. The range of the virtual revolute joint on the thumb's tip with different arm configurations will be investigated. A collision detection module will be included to avoid hand-object penetration.

REFERENCES

[1] Y. Hu, B. Huang, and G.-Z. Yang, "Task-priority redundancy resolution for co-operative control under task conflicts and joint constraints," in *Proc. - IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 2398–2405.
[2] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Soueres, and J.-Y. Fourquet, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 346–362, 2013.
[3] T. Sugihara, "Solvability-unconcerned inverse kinematics by the levenberg–marquardt method," *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 984–991, 2011.
[4] T. F. Chan and R. V. Dubey, "A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators," *IEEE Trans. Robot. Autom.*, vol. 11, no. 2, pp. 286–292, 1995.
[5] F. Flacco and A. De Luca, "A reverse priority approach to multi-task control of redundant robots," in *Proc. - IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 2421–2427.
[6] G. Jarquín, A. Escande, G. Arechavaleta, T. Moulard, E. Yoshida, and V. Parra-Vega, "Real-time smooth task transitions for hierarchical inverse kinematics," in *Proc.-IEEE-RAS Int. Conf. Humanoid Robots*, 2013, pp. 528–533.
[7] J. Lee, P. H. Chang, and R. S. Jamisola, "Relative task prioritization for dual-arm with multiple, conflicting tasks: Derivation and experiments," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2013, pp. 1928–1933.
[8] M. Abdeetedal and M. R. Kermani, "Grasp synthesis for purposeful fracturing of object," *Robot. Auton. Syst.*, vol. 105, pp. 47–58, 2018.
[9] ——, "Development and grasp analysis of a sensorized underactuated finger," in *Proc. - IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 6331–6336.
[10] M. Liu, Z. Pan, K. Xu, and D. Manocha, "New formulation of mixed-integer conic programming for globally optimal grasp planning," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 4663–4670, 2020.
[11] I. Gori, U. Pattacini, V. Tikhanoff, and G. Metta, "Three-finger precision grasp on incomplete 3d point clouds," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2014, pp. 5366–5373.
[12] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.
[13] C. Rosales, J. M. Porta, R. Suarez, and L. Ros, "Finding all valid hand configurations for a given precision grasp," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2008, pp. 1634–1640.
[14] C. Rosales, L. Ros, J. M. Porta, and R. Suárez, "Synthesizing grasp configurations with specified contact regions," *Int. J. Robot. Res.*, vol. 30, no. 4, pp. 431–443, 2011.
[15] S.-i. An and D. Lee, "Prioritized inverse kinematics with multiple task definitions," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2015, pp. 1423–1430.
[16] A. Dietrich, T. Wimbock, A. Albu-Schaffer, and G. Hirzinger, "Integration of reactive, torque-based self-collision avoidance into a task hierarchy," *IEEE Trans. Robot.*, vol. 28, no. 6, pp. 1278–1293, 2012.
[17] C. A. Klein, C. Chu-Jenq, and S. Ahmed, "A new formulation of the extended jacobian method and its use in mapping algorithmic singularities for kinematically redundant manipulators," *IEEE Trans. Robot. Autom.*, vol. 11, no. 1, pp. 50–55, 1995.
[18] R. Konietschke and G. Hirzinger, "Inverse kinematics with closed form solutions for highly redundant robotic systems," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2009, pp. 2945–2950.
[19] S. Qiu and M. R. Kermani, "Inverse kinematics of high dimensional robotic arm-hand systems for precision grasping," *J. Intell. Robot. Syst.*, vol. 101, no. 4, pp. 1–15, 2021.
[20] E. POUYDEBAT, M. LAURIN, P. GORCE, and V. BELS, "Evolution of grasping among anthropoids," *J. Evol. Biol.*, vol. 21, no. 6, pp. 1732–1743, 2008.
[21] R. W. Young, "Evolution of the human hand: the role of throwing and clubbing," *Journal of Anatomy*, vol. 202, no. 1, pp. 165–174, 2003.
[22] G. Cotugno, K. Althoefer, and T. Nanayakkara, "The role of the thumb: study of finger motion in grasping and reachability space in human and robotic hands," *IEEE Trans. Syst. Man Cybern.: Syst.*, vol. 47, no. 7, pp. 1061–1070, 2016.
[23] Y. Lin and Y. Sun, "Robot grasp planning based on demonstrated grasp strategies," *Int. J. Robot. Res.*, vol. 34, no. 1, pp. 26–42, 2015.
[24] L. Žlajpah, "On orientation control of functional redundant robots," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2017, pp. 2475–2482.
[25] W. Kahan, "How futile are mindless assessments of round-off in floating-point computation?" https://people.eecs.berkeley.edu/~wkahan/Mindless.pdf, 2006, [Online].
[26] K. M. Lynch and F. C. Park, *Modern Robotics*. Cambridge University Press, 2017.
[27] K. H. Hunt, *Kinematic geometry of mechanisms*. Oxford University Press, USA, 1978, vol. 7.
[28] E. Rohmer, S. P. Singh, and M. Freese, "V-rep: A versatile and scalable robot simulation framework," in *Proc. - IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1321–1326.
[29] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols," *arXiv preprint arXiv:1502.03143*, 2015.
[30] C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods," *IEEE Trans. Syst. Man Cybern.*, vol. 16, no. 1, pp. 93–101, 1986.