**Proceedings of the ASME 2011 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2011
August 29-31, 2011, Washington, DC, USA**

DETC2011/MECH-47818

DESIGN OF NON-ANTHROPOMORPHIC ROBOTIC HANDS FOR ANTHROPOMORPHIC TASKS

Edgar Simo-Serra
Institut de Robòtica i Informàtica Ind.
(CSIC-UPC)
Llorens i Artigas, 4-6
08028 Barcelona, Spain
Email: esimo@iri.upc.edu

Francesc Moreno-Noguer
Institut de Robòtica i Informàtica Ind.
(CSIC-UPC)
Llorens i Artigas, 4-6
08028 Barcelona, Spain
Email: fmoreno@iri.upc.edu

Alba Perez-Gracia*
Institut de Robòtica i Informàtica Ind.
(CSIC-UPC)
Llorens i Artigas, 4-6
08028 Barcelona, Spain
And: College of Science and Engineering
Idaho State University
Pocatello, Idaho 83209
Email: aperez@iri.upc.edu

ABSTRACT

In this paper, we explore the idea of designing non-anthropomorphic, multi-fingered robotic hands for tasks that replicate the motion of the human hand. Taking as input data rigid-body trajectories for the five fingertips, we develop a method to perform dimensional synthesis for a kinematic chain with a tree structure, with three common joints and five branches.

We state the forward kinematics equations of relative displacements for each serial chain expressed as dual quaternions, and solve for up to five chains simultaneously to reach a number of poses along the hand trajectory using a hybrid global numerical solver that integrates a genetic algorithm and a Levenberg-Marquardt local optimizer.

Although the number of candidate solutions in this problem is very high, the use of the genetic algorithm lets us to perform an exhaustive exploration of the solution space and retain a subset of them. We then can choose some of the solutions based on the specific task to perform. Note that these designs could match the task exactly while having a finger design radically different from that of the human hand.

*Address all correspondence to this author.

NOMENCLATURE

r Number of revolute joints.
b Number of kinematic chains.
n Number of joints.
m Number of task positions.
ε Dual unit such that $\varepsilon^2 = 0$.
$\mathbf{v}$ A vector.
$\hat{Q}$ A dual quaternion.

INTRODUCTION

There are many applications for which a robotic system is needed to work in human environments and to perform tasks that are designed for the human hand. In most cases, the solution adopted for grasping and manipulation consists of anthropomorphic robotic hands, which imitate to a certain extent the topology and joint location of the human hand, see [1] for a review of applications and concept definition.

It is difficult to match the complexity of the human hand, commonly accepted to have 26 degrees of freedom when counting the motion at the wrist and the pronation/supination of the forearm. The anthropomorphic design must include a complex mechanical system, actuation and sensing in a small space [2]. In order to reduce complexity, actual designs limit the active degrees of freedom in many cases, through simplification of the mechanical structure or by designing underactuated hands. The design of simplified grippers limits the tasks that the robotic end-effector can perform to some grasping and manipulation actions. Pairing these designs with some degree of underactuation and compliance, they aim to perform robust grasping of objects of unknown shape. Dollar and Howe [3] present a simplified, underactuated design for reliable grasping. Ciocarlie and Allen [4] optimize an underactuated, non-anthropomorphic gripper for performing a series of grasps from a database. A more thorough review on underactuated hands can be found in [5].

In order to perform some of the more complex functions of the human hand (not only grasping and manipulation, but also perception through surface exploration), it seems that an end-effector with several independently-actuated fingers may be needed. There are many examples of design of anthropomorphic robotic hands, see for instance [6] for a relatively recent review. However, for complex multi-fingered designs, it may not be necessary that the robotic fingers mimic those of the human hand in order to perform human-like tasks. We want to explore these alternative designs with the use of kinematic synthesis.

Kinematic design of robotic hands has focused on the design of individual motion of fingers or parts of the hand. Dai and Wang [7] use kinematic synthesis to design a spherical mechanism to act as a the palm of a metamorphic hand. Van Varseveld and Bone [8] designed a finger mechanism for a non-anthropomorphic dexterous hand. Walker et al. [9] also design planar linkages for the fingers of a non-anthropomorphic, dexterous hand. Schafer and Dillman [10] present the kinematic design of a humanoid robotic wrist.

In this paper, we explore the idea of designing the full non-anthropomorphic, multi-fingered robotic hand for tasks that replicate the motion of the human hand at the fingertips. Taking as input data rigid-body trajectories for the human fingertips, we develop a method to perform dimensional synthesis for a kinematic chain with a tree structure, with three common joints and five branches. As a whole, the process entails the simultaneous solution of five serial chains, two of them having four independent joints and three of them with five independent joints, plus three common joints for all of them. The total degrees of freedom of the non-anthropomorphic hand design is 26, similar to the human hand.

We state the forward kinematics equation of relative displacements for each serial chain expressed as dual quaternions, and solve for all five chains simultaneously to reach a number of positions along the hand trajectory. The synthesis of spatial serial chains for up to five degrees of freedom was developed in [11]. We use a similar methodology together with a hybrid global numeric solver, composed of a genetic algorithm paired with a Levenberg-Marquardt local optimizer. For the tree-like kinematic structure, a high number of positions can be defined to perform exact synthesis, obtaining a good approximation for the desired trajectory.

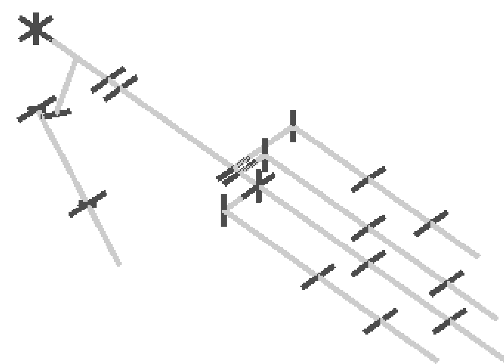


FIGURE 1: HAND SKELETON USED TO GENERATE POSITIONS.

Finding the complete solution set for the synthesis of complex kinematic chains is an unsolved problem. Only dyads, such as the RR kinematic chain [12], [13], and some triads with particular characteristics [14], have been fully studied. Even the 3R kinematic chain has not been completely solved with a closed algebraic expression [15]. The complexity of a tree-like kinematic chain with a total of 26 revolute joints, and our numerical results, lead to believe that there will be a very large amount of solutions. This is an issue that has been found before even for simpler kinematic chains, see [14]. In order to deal with this, additional constraints may be added to help in the selection of the final design, depending on the task. We present one of the solutions for a hand task that was synthetically generated. The designs could exactly match the task while having a finger design radically different from that of the human hand.

KINEMATIC HAND MODEL

The human hand has 5 fingers formed by 14 joints. These joints are created at the intersection of the 27 major bones forming the hand. The joints are not all simple joints and can be formed by multiple simple kinematic joints. However, they can be modelled using only simple revolute kinematic joints. The axes of these revolute joints do not necessarily have to intersect within the same joint.

The full hand model can be modelled by using a total of 26 joints if we consider the wrist and the pronation/supination of the forearm to be formed by 3 revolute joints. The index and middle finger have 4 revolute joints, while the third, fourth and thumb have 5 revolute joints. The dimensions are taken from the literature [16, 17]. The skeleton can be represented by drawing the common normals of the joint axes as shown in Fig.1.

TABLE 1: JOINT ANGLES USED TO GENERATE THE TRAJECTORY.

Chain	Hand task revolute joint limits [θ_{min}-θ_{max}]
Common	[-90°,90°],[-90°,90°],[-90°,90°]
Index	[-10°,90°],[-10°,10°],[0°,100°],[0°,90°]
Middle	[-10°,90°],[-10°,10°],[0°,100°],[0°,90°]
Third	[-10°,10°],[0°,20°],[-10°,10°],[0°,100°],[0°,90°]
Fourth	[-10°,10°],[0°,20°],[-10°,10°],[0°,100°],[0°,90°]
Thumb	[-25°,25°],[-25°,25°],[-10°,10°],[0°,70°], [-10°,85°]

Hand Task Generation

The task is defined as a series of finite positions and orientations for each finger tip, created by assigning a set of joint variables to the kinematic model. The range of motion of each joint varies immensely among population and it is therefore impossible to assign a precise range of motion to each joint. The trajectories are generated from random positions within the joint angle ranges defined in Tab.1 to simplify the convergence of the solver.

KINEMATIC SYNTHESIS

The goal of the dimensional kinematic synthesis is to find the location and orientation of a set of joint axes able to perform a given motion, where the number and type of joints are pre-defined. This is also known as the motion-to-form problem in which we are given a motion as an input and must calculate the form as a set of joints and angles that can perform the motion. In this paper, we follow the original idea of [18] of using the forward kinematics equations of the kinematic chain, but formulated as relative displacements and expressed as dual quaternions, see [11] for a complete description of this approach.

The input data for the synthesis are the $m-1$ relative transformations $\hat{P}_{1j} = \cos\frac{\Delta\hat{\phi}_{1j}}{2} + \sin\frac{\Delta\hat{\phi}_{1j}}{2}\mathsf{P}_{1j},\ j = 2,\dots,m,$ defining the task; the output are the Plücker coordinates $\mathsf{S}_i = \mathbf{s}_i + \varepsilon\mathbf{s}_i^0 = \mathbf{s}_i + \varepsilon\mathbf{c}_i \times \mathbf{s}_i$, $i = 1,\dots,n$, of the n joints that define the kinematic chain, and also the joint variables $\boldsymbol{\Delta\hat{\theta}}_{ij} = \boldsymbol{\theta}_{ij} - \boldsymbol{\theta}_{i0} + (\mathbf{d}_{ij} - \mathbf{d}_{i0})\varepsilon$, $i = 1,\dots,n,\ j = 2,\dots,m$, used to reach the task positions, measured from the reference configuration.

Forward Kinematics

Given a kinematic serial chain with n joints, we can write the kinematics equations using the product of exponentials of the screws corresponding to the joint axes, as described in [19]. In this paper, instead of calculating the exponentials using matrix algebra, we do the exponentials for the Clifford even subalgebra of the projective space, in which the unit elements, also known as dual quaternions, express spatial displacements. The exponential of a screw represented by the Clifford algebra element $\mathsf{J} = (1 + \mu\varepsilon)\mathsf{S}$, where μ is the pitch relating the slide d and the rotation θ along and about the screw, and S is the screw axis, yields a finite displacement,

$$e^{\frac{\theta}{2}\mathsf{J}} = (\cos\frac{\theta}{2} - \frac{d}{2}\sin\frac{\theta}{2}\varepsilon) + (\sin\frac{\theta}{2} + \frac{d}{2}\cos\frac{\theta}{2}\varepsilon)\mathsf{S}$$
$$= \cos\frac{\hat{\theta}}{2} + \sin\frac{\hat{\theta}}{2}\mathsf{S}. \quad (1)$$

For a serial chain with n joints, in which each joint can rotate an angle θ_i and slide a distance d_i, around and along the axis S_i, $i = 1,\dots,n$, we calculate the forward kinematics of relative displacements (with respect to an arbitrary reference configuration),

$$\hat{Q}(\boldsymbol{\Delta\hat{\theta}}) = e^{\frac{\Delta\hat{\theta}_1}{2}\mathsf{S}_1} e^{\frac{\Delta\hat{\theta}_2}{2}\mathsf{S}_2} \cdots e^{\frac{\Delta\hat{\theta}_n}{2}\mathsf{S}_n}$$
$$= (\cos\frac{\Delta\hat{\theta}_1}{2} + \sin\frac{\Delta\hat{\theta}_1}{2}\mathsf{S}_1)\cdots(\cos\frac{\Delta\hat{\theta}_n}{2} + \sin\frac{\Delta\hat{\theta}_n}{2}\mathsf{S}_n), \quad (2)$$

where $\boldsymbol{\Delta\hat{\theta}}_i = (\boldsymbol{\theta}_i - \boldsymbol{\theta}_0 + (\mathbf{d}_i - \mathbf{d}_0)\varepsilon)$ contains the joint variables, as relative values with respect to the joint parameters of the chain $\boldsymbol{\theta}_0$ and $\mathbf{d}_0$ when in the reference configuration.

Synthesis Design Equations

The dimensioning of the articulated system has to be done so that the forward kinematics equations in Eqn.(2) can reach all the desired task positions $\hat{P}_{1j}$,

$$\hat{P}_{1j} = e^{\frac{\Delta\hat{\theta}_{1j}}{2}\mathsf{S}_1} e^{\frac{\Delta\hat{\theta}_{2j}}{2}\mathsf{S}_2} \cdots e^{\frac{\Delta\hat{\theta}_{nj}}{2}\mathsf{S}_n}, \quad j = 2,\dots,m \quad (3)$$

This results in $8(m-1)$ design equations. The design variables that determine the dimensions of the chain are the n joint axes S_i, $i = 1,\dots,n$, in the reference configuration. In addition, the equations contain the $n(m-1)$ pairs of joint parameters $\Delta\hat{\theta}_{ij} = \Delta\theta_{ij} + \Delta d_{ij}\varepsilon$, which are also unknown.

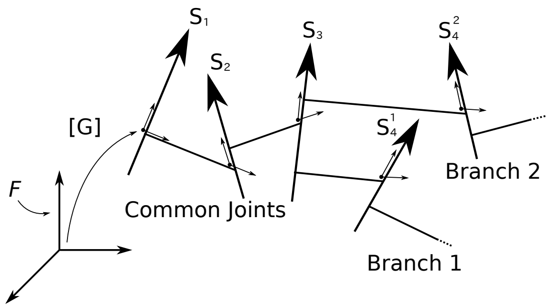


FIGURE 2: TOPOLOGY OF THE KINEMATIC CHAIN.

NON-ANTHROPOMORPHIC HAND SYNTHESIS

In this paper, we are interested in the synthesis of robotic grippers for human-hand tasks. We don't impose any limitation on the link dimensions or placement. The resulting design should be able to perform a given human-hand task while having a non-anthropomorphic aspect.

System Topology

As input topology, we define a tree-like kinematic chain with three common joints in series at the base, connected to five serial chains arranged in parallel, two of them with four degrees of freedom and three of them with five degrees of freedom. This structure follows the generally accepted joint arrangement of the human hand plus the wrist motion. Figure 2 shows the topology of the kinematic chain to be synthesized.

Tree topologies can also be used to represent pure serial topology, when there is only one branch, and loop topologies, when end effectors of multiple branches are placed at the same position.

Design Equations

We state the design equations adapting Eqn.(3) to our particular topology. For each one of the serial chains in the parallel arrangement we state a set of design equations, to obtain the total system of equations

$$\begin{aligned}\hat{P}^k_{1j} &= e^{\frac{\Delta\hat{\theta}_{1j}}{2}\mathsf{S}_1} e^{\frac{\Delta\hat{\theta}_{2j}}{2}\mathsf{S}_2} e^{\frac{\Delta\hat{\theta}_{3j}}{2}\mathsf{S}_3} e^{\frac{\Delta\hat{\theta}^k_{4j}}{2}\mathsf{S}^k_4}\cdots e^{\frac{\Delta\hat{\theta}^k_{7j}}{2}\mathsf{S}^k_7}, k\in\{1,2\},\\ \hat{P}^k_{1j} &= e^{\frac{\Delta\hat{\theta}_{1j}}{2}\mathsf{S}_1} e^{\frac{\Delta\hat{\theta}_{2j}}{2}\mathsf{S}_2} e^{\frac{\Delta\hat{\theta}_{3j}}{2}\mathsf{S}_3} e^{\frac{\Delta\hat{\theta}^k_{4j}}{2}\mathsf{S}^k_4}\cdots e^{\frac{\Delta\hat{\theta}^k_{8j}}{2}\mathsf{S}^k_8}, k\in\{3,4,5\},\\ & \qquad j=2,\ldots,m. \end{aligned} \tag{4}$$

Here k identifies the kinematic chain and $j = 2,\ldots,m$ is the index of the task position. The fingers are denoted by: index, 1; middle, 2; third, 3; fourth, 4 and thumb, 5.

Each of the five serial chains, taken individually, has a total of seven or eight degrees of freedom, depending on the finger it corresponds to. Even though the exact dimensional synthesis does not apply to serial chains with six or more degrees of freedom, by having three common degrees of freedom in this case allows us to state the design equations of the system as a whole.

Notice that, even though each individual finger have more than six degrees of freedom, the tree-like architecture is not kinematically redundant when defining the motion of several branches at the same time.

KINEMATIC SOLVER

In order to deal with the large system of equations presented in the previous section, the selected solver tries to minimize the error in Eqn.(4) for each kinematic chain. In principle, problems with selection of the metric does not apply to this case, as we are targeting exact synthesis.

The objective of the solver is to perform general inverse kinematics, that is adjust both joint angles and joint axes to follow a motion. However, by making the joint axes constant it can also perform inverse kinematics.

Objective Functions

Given a vector $d = \{4,4,5,5,5\}$ with the number of independent joints for each finger, the design equations from Eqn.(4) can be written as a set of unconstrained functions,

$$\hat{F}^k_j(\mathbf{S}^k, \boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^k_j) = \underbrace{\prod_{i=1}^{3} e^{\frac{\Delta\hat{\theta}_{ij}}{2}\mathsf{S}_i}}_{\text{Common}} \underbrace{\prod_{i=4}^{3+d_k} e^{\frac{\Delta\hat{\theta}^k_{ij}}{2}\mathsf{S}^k_i}}_{\text{Individual}} - \hat{P}^k_{1j}, \quad \begin{array}{l} j=2,\ldots,m \\ k=1,\ldots,5 \end{array} \tag{5}$$

where we can clearly see the common joints and the individual joints that belong to each branch.

The global objective of the solver is to solve the design equations defined in Eqn.(4). This can be written as,

$$\sum_{j=2}^{m}\sum_{k=1}^{5} |\hat{F}^k_j(\mathbf{S}^k, \boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^k_j)| = 0 \tag{6}$$

Genetic algorithms however behave better with defined positive maximization objective functions. We can convert Eqn.(6) to a minimization function and invert it to obtain,

$$\max \left\{ \frac{1}{\displaystyle\sum_{j=2}^{m}\sum_{k=1}^{5} |\hat{F}^k_j(\mathbf{S}^k, \boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^k_j)|} \right\} \tag{7}$$

which we can see is defined positive if the domain of $\hat{F}_j^k(\mathbf{S}^k, \boldsymbol{\Delta}\hat{\boldsymbol{\theta}}_j^k)$ is finite. This is a single function that is also known as the fitness function.

For the Levenberg-Marquadt local optimizer we use the minimizing least squares set of objective functions,

$$\min \left\{ \sum_{j=2}^{m} \sum_{k=1}^{5} \hat{F}_j^k(\mathbf{S}^k, \boldsymbol{\Delta}\hat{\boldsymbol{\theta}}_j^k)^2 \right\} \tag{8}$$

Dimension of the Equation Set

The equations and variables in the objective functions Eqn.(5) are not all linearly independent. The equations are formed by unit dual quaternions $\hat{Q} = \hat{q} + \varepsilon q^0$ which have a dimension of eight, but are subject the two implicit constraints $\hat{q}\hat{q}^* = 1$ and $\hat{q} \cdot \hat{q}^0 = 0$ which reduce the independent dimension of the dual quaternion to 6. Similarly only four of the six Plücker components of each line are independent as they are subject to the two implicit constraints $\|\mathbf{s}\| = 1$ and $\mathbf{s} \cdot \mathbf{s}^0 = 0$. The number of variables x and independent variables x^0 can be written as,

$$\begin{aligned} x &= r(6 + (m-1)) \\ x^0 &= r(4 + (m-1)) \end{aligned} \tag{9}$$

The number of equations f and independent equations f^0 can be written as,

$$\begin{aligned} f &= 8b(m-1) \\ f^0 &= 6b(m-1) \end{aligned} \tag{10}$$

An important question remaining is how many task positions m must be imposed for the system to have a finite number of solutions, see [20] for details. This can be obtained by comparing imposing $f^0 = x^0$ and solving to obtain,

$$m = \frac{4r}{6b - r} + 1 \tag{11}$$

The number of task positions obtained is the minimum to have a finite solution. However, if more task positions are provided the system becomes overdetermined. This can lead to problems of convergence and much slower performance.

Applying Eqn.(11) to our model with $b = 5$ branches and $r = 26$ revolute joints, three of them being common to all the branches, we obtain that we can solve exactly for a task defined by $m = 27$ finite positions for each finger. This gives a total of 156 structural parameters and 676 joint variables for our set of equation.

Solver Implementation

Genetic algorithms have already been used in many kinematic problems [21, 22], however the complexity of the problem that we are tying to solve is especially challenging. Through the use of Clifford algebra the number of equations and variables can be reduced in comparison with the matrix algebra approach. This allows the entire equation system to be represented with only 832 input variables between both structural and joint variables.

The genetic algorithm performs an exhaustive exploration of the solution space in order find a solution. However, due to complexity of the system a pure genetic algorithm has trouble with convergence past a certain fitness. This problem was overcome by reducing the solution space used by genetic algorithm to only the local minima of the entire solution space. This converts the solution space from a continuous domain to a discrete finite domain that improves the behaviour of the genetic algorithm. The local minima are found by a Levenberg-Marquardt optimizer.

The genetic algorithm chromosomes consist of sets of variables that belong to the full solution space. After being generated they are then converged on a local minima. This local convergence is also done when chromosomes are crossed between each other or mutated to ensure each chromosome always represents a local minima of the search space.

The fitness is calculated as the inverse of the sum of the error as seen in Eqn.(7). This makes the fitness a continuous positive function, which allows the usage of roulette-wheel selection when choosing pairs from the genetic algorithm population to crossover. It also converts the genetic algorithm to a maximization problem. The crossover rate is kept low to encourage diversity in the population, since the strong convergence is provided by the Levenberg-Marquardt optimizer.

To avoid explicit constraints the chromosomes are generated in the proximity of the ideal kinematic solution. This acts as a soft limit for the possible shapes of the robotic hand. It is not unusual for the Levenberg-Marquardt minimizer to move far from the generation space and find extremely non-anthropomorphic solutions.

The solver is executed until the fitness surpasses the value of 10^{10} at which it is considered to have arrived to a solution. The error is attributed to the imprecision in the computer representation of real numbers.

Kinematic Subsystems

It is interesting to note is that it is possible to solve a subsystem of the full kinematic chain tree. These subsystems are independently solvable as long as they have a finite number of

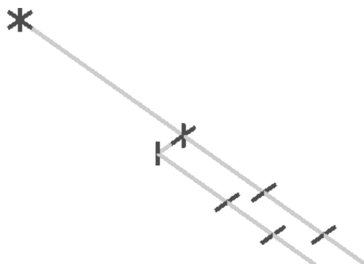


FIGURE 3: TOPOLOGY OF THE SOLVABLE (4R, 4R) FINGERS SUBSYSTEM.

task positions m needed for a finite number of solutions. To have $m \in \mathbb{R}^+$ the subsystem must comply with the following inequation,

$$6b - r > 0 \quad (12)$$

which is obtained from imposing $m > 0$ in Eqn.(11). Table 2 shows all the possible variations of the tree-like kinematic chain that can be solved.

The most interesting configurations to solve are the 5 kinematic chains at once, to minimize the needed positions, and the configuration with only two 4R branches to reduce the amount of variables needed. The two 4R branches are the index and middle fingers and can be seen in Fig.3. Afterwards the remaining fingers can be solved individually as the common revolute joints are now identified. There are also other systems that can be solved that are a compromise between number of positions and the number of variables needed.

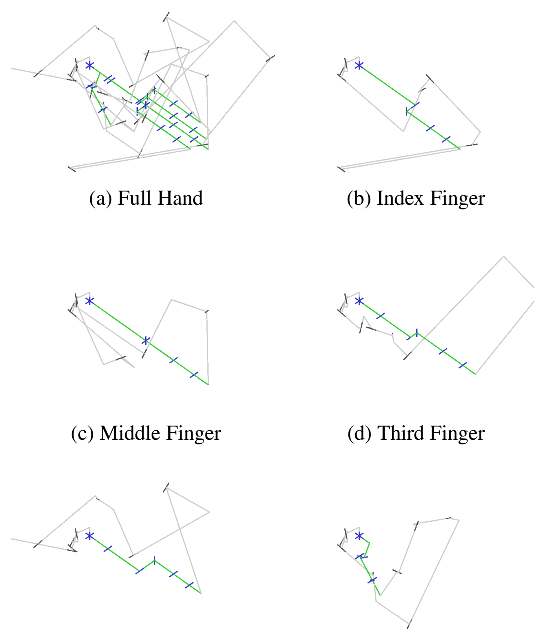
RESULTS

The algorithm has a parallel nature and can be adjusted to run on supercomputers or other distributed computing systems to increase the calculation speed. Table 3 contains runtime information of example solutions. These results were obtained on an Intel®Core™i7-870 CPU at 2.93GHz. Examples of solutions are shown in Fig.4 and Fig.5 where the green lines represent the anthropomorphic skeleton used to generate the data and the gray lines represent the non-anthropomorphic solution found. An example of the solver's converge can be seen in Fig.6.

Best results have been obtained with a population size of 100 and using very different positions in the task. The computational time and generations needed to find a solution vary greatly with different sets of positions. To minimize the computational time it is important to use very different positions as it simplifies the convergence of the algorithm.

TABLE 2: SOLVABLE SYSTEMS OF EQUATIONS FOR DIFFERENT COMBINATIONS OF CHAINS.

b	r	x	f	m	Notes
5	26	832	1092	27	Full model
4	22	1100	1452	45	(5R, 5R, 5R, 4R) fingers
4	21	714	938	29	(5R, 5R, 4R, 4R) fingers
3	17	1258	1666	69	(5R, 5R, 4R) fingers
3	16	608	800	33	(5R, 4R, 4R) fingers
2	11	550	726	45	(4R, 4R) fingers
1	5	130	170	21	5R finger, common solved
1	4	56	72	9	4R finger, common solved

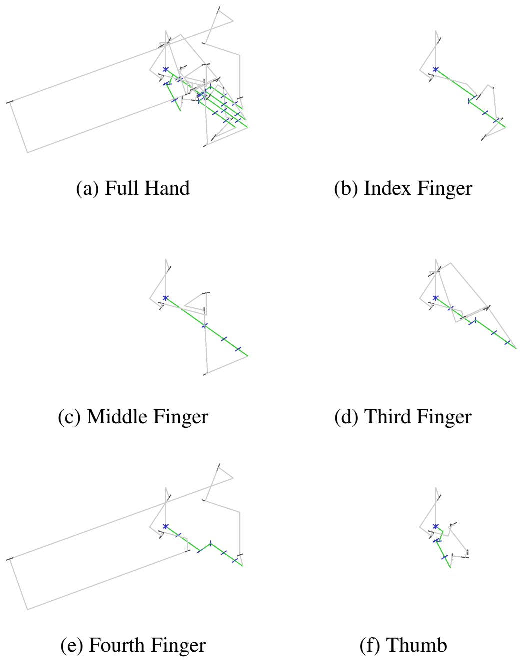


(a) Full Hand
(b) Index Finger
(c) Middle Finger
(d) Third Finger
(e) Fourth Finger
(f) Thumb

FIGURE 4: OVERVIEW OF A SOLUTION FOUND.

CONCLUSIONS

In this paper we present a method for the kinematic synthesis of tree-like articulated systems, with an application in the design of a robot to perform human-hand tasks. This methodology allows obtaining non-anthropomorphic designs that can perform an



(a) Full Hand (b) Index Finger

(c) Middle Finger (d) Third Finger

(e) Fourth Finger (f) Thumb

FIGURE 5: OVERVIEW OF A SECOND SOLUTION FOUND.

TABLE 3: SOLVER EXECUTION INFORMATION.

Solution	Generations	Time (hours)
1	13	20.8
2	5	20.5
3	10	16.5
4	12	85.0

anthropomorphic finite-position task exactly, while having very different dynamics. These designs have applications in exoskeleton design, as they can be chosen to mount on the hand without interfering with the morphology of the hand.

The methodology presented can also be applied to serial, loop and generic tree topologies of articulated systems. This flexibility allows for its application in other topologies not only limited hand models, making the solver a powerful tool.

From the dimensional synthesis point of view, the interesting result is that, due to the tree structure, we can synthesize general serial chains with more than five degrees of freedom. We may conclude from this that, despite common believe that the human/wrist hand is a redundant mechanical system, it may not be so when we consider a task in which several fingers must act.

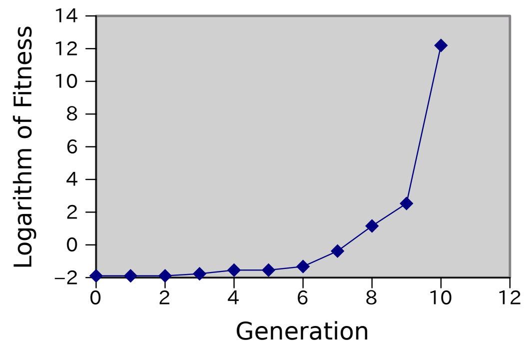


FIGURE 6: CONVERGENCE OF A SOLVER EXECUTION.

Dimensional synthesis for articulated systems like the one presented here, with a high number of joints, yield many solutions. A good selection process is required in order to choose a solution from the pool of candidates. For instance, for the application presented in this paper, additional constraints could be imposed either in the solving process or in the post-processing phase in order to find a suitable design. Among others, we can cite size or location restrictions for exoskeletons mounted on the human hand, or dexterity conditions at given configurations for manipulation in human environments.

In order for the designs to be able to perform realistic human tasks, not only on the fingertip position but also fingertip forces and infinitesimal motion need to be considered. Future work will focus on including task velocities and accelerations in order to define grasping actions.

ACKNOWLEDGMENT

This work is partially supported by the Spanish Ministry of Science and Innovation under projects DPI2010-18449, DPI2008-06022, Consolider Ingenio 2010 CSD2007-00018, and a Ramon y Cajal Research Fellowship. Also by the EU project GARNICS FP7-247947.

REFERENCES

[1] Siciliano, B., and Khatib, O., 2008. *Handbook of Robotics*. Springer, Berlin.
[2] Soto Martell, J., and Gini, G., 2007. "Robotic hands: Design review and proposal of new design process". *World Academy of Science, Engineering and Technology*, ***26***.
[3] Dollar, A., and Howe, R., 2007. "Simple, robust autonomous grasping in unstructured environments". In Proc. of the 2007 International Conference on Robotics and Automation, pp. 4693–4700.

[4] Ciocarlie, A., and Allen, P., 2010. "Data-driven optimization for underactuated robotic hands". In Proc. of the 2010 International Conference on Robotics and Automation, pp. 1292–1299.
[5] Briglen, L., Laliberte, T., and Gosselin, C., 2008. *Underactuated Robotic Hands*. Springer.
[6] Biagiotti, L., Lotti, F., Melchiorri, C., and Vassura, G., 2004. How far is the human hand? a review of anthropomorphic robotic end-effectors. internal report. Tech. rep., Universita di Bologna.
[7] Dai, J., and Wang, D., 2007. "Geometric analysis and synthesis of the metamorphic robotic hand". *ASME Journal of Mechanical Design,* ***129***, pp. 1191–1197.
[8] van Varseveld, R., and Bone, G., 1999. "Design and implementation of a lightweight, large workspace non-anthropomorphic dexterous hand". *ASME Journal of Mechanical Design,* ***121***.
[9] Ramos, A., Gravagne, I., and Walker, I., 1999. "Goldfinger: A non-anthropomorphic, dextrous robot hand". In Proceedings of the 1999 IEEE International Conference on Robotics and Automation.
[10] Schaffer, C., and Dillmann, R., 2001. "Kinematic design of a humanoid robot wrist". *Journal of Robotic Systems,* ***18(12)***, pp. 747–754.
[11] Perez Gracia, A., and McCarthy, J. M., 2006. "The kinematic synthesis of spatial serial chains using clifford algebra exponentials". *Proceedings of the Institution of Mechanical Engineers, Part C, Journal of Mechanical Engineering Science,* ***220(7)***, pp. 953–968.
[12] Perez, A., and McCarthy, J. M., 2003. "Dimensional synthesis of bennett linkages". *ASME Journal of Mechanical Design,* ***125(1)***, pp. 98–104.
[13] Brunnthaler, K., 2006. "Synthesis of 4r linkages using kinematic mapping". PhD thesis, Institute for Basic Sciences and Engineering, University of Innsbruck, Innsbruck, Austria.
[14] Su, H., McCarthy, J. M., and Watson, L. T., 2004. "Generalized linear product homotopy algorithms and the computation of reachable surfaces". *ASME Journal of Computers and Information Science and Engineering,* ***4(3)***, pp. 226–235.
[15] Lee, E., Mavroidis, C., and Merlet, J. P., 2004. "Five precision point synthesis of spatial rrr manipulators using interval analysis". *ASME Journal of Mechanical Design,* ***126(5)***, pp. 842–850.
[16] Fadi J. Beijjani, J. M. F. L., 1989. *Basic Biomechanics of the Musculoskeletal System*. Lea & Febiger, ch. Biomechanics of the hand, pp. 275–304.
[17] Chang, L., and Matsuoka, Y., 2006. "A kinematic thumb model for the act hand". In Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pp. 1000 –1005.
[18] Lee, E., and Mavroidis, C., 2002. "Solving the geometric design problem of spatial 3r robot manipulators using polynomial homotopy continuation". *ASME Journal of Mechanical Design,* ***124(4)***, pp. 652–661.
[19] Murray, R. M., Li, Z., and Sastry, S. S., 1994. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., Boca Raton, FL.
[20] Perez, A., and McCarthy, J. M., 2004. "Dual quaternion synthesis of constrained robotic systems". *ASME Journal of Mechanical Design,* ***126(3)***, pp. 425–435.
[21] Chocron, O., and Bidaud, P., 1997. "Evolutionary algorithms in kinematic design of robotic systems". In Intelligent Robots and Systems, 1997. IROS '97., Proceedings of the 1997 IEEE/RSJ International Conference on, Vol. 2, pp. 1111 –1117 vol.2.
[22] Yang, Y., Peng, G., Wang, Y., and Zhang, H., 2007. "A new solution for inverse kinematics of 7-dof manipulator based on genetic algorithm". In Automation and Logistics, 2007 IEEE International Conference on, pp. 1947 –1951.