This document is published in:

IEEE Transactions on Robotics, December 2011, Vol. 27, Issue, 6, pp. 1132-1137

DOI: http://dx.doi.org/10.1109/TRO.2011.2160668

The research leading to these results has received funding from the RoboCity2030-II-CM project (S2009/DPI-1559), funded by Programas de Actividades I+D en la Comunidad de Madrid and cofunded by Structural Funds of the UE.

Task-Oriented Kinematic Design of a Symmetric Assistive Climbing Robot

Alberto Jardón, Martin F. Stoelen, Fabio Bonsignorio, and Carlos Balaguer

***Abstract*—ASIBOT is an assistive climbing robot, capable of aiding in daily tasks from fixed docking stations in the environment. A task-oriented design process was applied to improve the robot kinematic structure, based on the Grid Method. Twelve different robot designs were optimized for typical kitchen scenarios, followed by a quantitative comparison.**

***Index Terms*—Assistive Robotics, Personal Robots, Kinematics, Animation and Simulation, Task-Oriented Design.**

I. Introduction

Assistive robots have the ability to aid disabled and elderly persons in daily life tasks and provide personalized assistance. The assistive robot ASIBOT has been developed at Universidad Carlos III de Madrid (UC3M). The unique feature of this assistive robot is its ability to attach itself to the environment by specially designed, low-cost Docking Stations (DS) [1]. These are placed to empower the robot, allowing it to move and work throughout the entire environment. This includes surfaces that are typically under-utilized, like ceilings and walls, as well as the user's wheelchair. See Fig. 1.

Fig. 1. The ASIBOT robot attached to the wheelchair DS in the kitchen test-bed at UC3M. Two fixed DS can be seen mounted on the walls.

The robot has been used for several years now, and one of the current lines of research is focused on how to improve the kinematic structure. ASIBOT's current design has five Degrees Of Freedom (DOF), is symmetric about the central joint, and is divided in two parts: the articulated arm structure, and the tips. The latter have docking mechanisms to connect the robot to the DS (with 24V power connectors) and also a gripper. The robot symmetry is one of the main requirements for this study, as it enables the robot to perform the same set of tasks regardless of which end is currently docked and used as a base. The Denavit-Hartenberg (DH) parameters for the current design is shown in Table I. See Fig. 5a for a visual representation.

Park, Chang and Yang [2] attempted a Task-Oriented Design (TOD) approach for an assistive robot that would guarantee that at least a set of tasks with high priority would be achievable in a given environment. The end-result should then also be capable of performing tasks that are similar. The approach begins with an investigation into potential tasks, including robot base location, environment obstacles and task points (TP). The latter is used to denote the position and orientation requirement of the robot end-effector to achieve a given task. This is followed by an optimization of the kinematic structure using the information from the task analysis. For this a grid-based method was developed, the Grid Method. Several other methods for optimizing kinematics exist. Kim and Khosla [3] proposed a comparatively complete algorithm for optimizing the design of general manipulators based on a genetic algorithm. Chocron and Bidaud [4] developed a method based on combining modular segments of a fixed number of types into a modular robotic system. However, the grid-based approaches are typically more efficient, as they have a fixed set of design variables and do not require the calculation of the inverse kinematics of the complete manipulator during optimization.

TABLE I
DH PARAMETERS OF THE CURRENT ASIBOT ROBOT

Joint	θ	l	d	α	*Range*
1	$\theta 1$	0	268	90	360°
2	$\theta 2$	400	0	0	270°
3	$\theta 3$	400	0	0	270°
4	$\theta 4$	0	0	90	270°
5	$\theta 5$	0	268	0	360°

The study presented here attempts to apply a task-oriented design approach to an assistive climbing robot like ASIBOT. The Grid Method for kinematic optimization was used and adapted to allow for the inclusion of robot symmetry. A general design methodology was then developed to allow for the consideration of a large number of tasks in different environments when selecting the final design.

II. Method

A. Modified Grid Method for Symmetric Robots

1) Original Approach: The algorithm described here is based on the Grid Method for optimizing robot kinematics using the Very Fast Simulated Annealing (VFSA) optimization method, as presented in Park, Chang and Kim [5]. See Park, Chang and Yang [2] for a more detailed introduction of the Grid Method itself. Grid-based methods are commonly used in for example modeling heat transfer and fluid flow. Common for the problems for which these methods are applied is that the problem boundary conditions are known, and the interior conditions are unknown. The process then involves splitting the area to be solved for into smaller unit grids, and permeate the boundary conditions into inner regions by applying governing equations to these unit grids successively. Each unit grid uses information from its own local boundaries.

When applied to optimizing robot kinematic designs in a TOD process, each unit grid represents the properties of one joint for one specific task point. The 4 design variables used for each joint are the Cartesian x, y, and z position and twist angle α, together denoted as $\mathbf{x}$. Each joint uses information from the previous and next joint in the kinematic chain and the same joint for the previous and next task point. Thus each joint is treated separately, with no need for calculating the inverse kinematics of the full kinematic chain. This also means that these 4 properties are the only design variables for the optimization process. For each joint a grid operation is performed. This includes first converting the design variables to DH parameters for the current joint, then evaluating the DH parameters with a weighted unit grid cost function. The grid operation is performed successively on all the joints in the kinematic chain and for all the task points used. A global convergence criteria based on the sum

Alberto Jardón, Martin F. Stoelen, and Carlos Balaguer are members of the RoboticsLab research group within the Department of System Engineering and Automation, Universidad Carlos III de Madrid, (ajardon, mstoelen, balaguer)@ing.uc3m.es

Fabio Bonsignorio is a member of the RoboticsLab as Santander Chair of Excellence in Robotics at Universidad Carlos III de Madrid and CEO/founder of Heron Robots of Genova, Italy, fabio.bonsignorio@uc3m.es

DOI: 10.1109/TRO.2011.2160668

of the cost functions for each unit grid (joint) determines when a sufficiently optimized solution has been found.

The Grid Method approach also has its limitations, however. The optimization process is only performed on one joint at a time, as this is the definition of the unit grid used. This makes the symmetry of two joints difficult to enforce. A modified version of the Grid Method was developed to overcome this, which is described below.

2) Modified Approach: The most important modification made to the Grid Method was to expand the unit grid to a pair of symmetric joints. This allows for enforcing the desired symmetry within each grid operation. Thus symmetric grid operations in general have 8 design variables, independent of the number of DOF and task points used. The grid operation is performed over each symmetric joint pair *i* of *c* total, for all *j* of *m* task points, where *c* is the center joint. The second change was to solve the problem as two manipulators, each with one of the global end-effectors as base. These *left* and *right* manipulators share the global center joint, and the local task point for each is the second to last joint for the other. In the original Grid Method the end-effector orientation required an additional constraint, the Desired Orientation Constraint (DOC). Here the Center (CEN) constraint is used to enforce the orientation of the center joint. A final change in the original approach was needed to ensure that the first and last links are symmetrical. Thus the base frame is no longer collocated with the zero frame (at global joint 1). Instead, the base frame is assumed to be at the end-effector of the first link (global link 0).

3) Symmetric Unit Grid Cost Function: The symmetric unit grid cost function can be seen in Equation (1). The equalization constraint (EC) ensures that the DH parameters of a joint are as close as possible to that of the same joint for the next and the previous task point (except the joint angles).

$$\begin{aligned} F_{SUG}(\mathbf{x}_{i,j}^{left}, \mathbf{x}_{i,j}^{right}) = & \\ & w_{SYM} \cdot f_{SYM((i,j)^{left},(i,j)^{right})} \\ + \quad & w_{TLC} \cdot f_{TLC(j)} \\ + \quad & w_{CEN} \cdot f_{CEN(j)} \\ + \quad & w_{EC} \cdot (f_{EC(i,j)^{left}} + f_{EC(i,j)^{right}}) \\ + \quad & w_{LC} \cdot (f_{LC(i,j)^{left}} + f_{LC(i,j)^{right}}) \\ + \quad & w_{OA} \cdot (f_{OA(i,j)^{left}} + f_{OA(i,j)^{right}}). \end{aligned} \tag{1}$$

The limit constraint (LC) allows for constraining one or more of the DH parameters for a joint to within a given range. The obstacle avoidance constraint (OA) penalizes the depth of penetration of a joint or link with a spherical obstacle. These three constraints are unchanged from the original Grid Method and are presented in detail in Park, Chang and Yang [2]. The symmetry constraint (SYM), total length constraint (TLC) and center constraint (CEN) are described below. Note that a measure to optimize performance once the constraints are met is not used for the work presented here, but can be included if needed. For example an equivalent of the Dexterity Measure (DM) of the original Grid Method.

4) Symmetry Constraint: The symmetry constraint is used to ensure that the two symmetric joints in the unit grid are similar. As can be seen from Equation (2), the constraint applies to both the previous and following link with respect to the joint in question. A special weight ω_{ang} is applied to the angular values to equalize the order of magnitude of angles with those of distances. In addition the link twist angles are either added or subtracted depending on the definition of the DH parameters for the symmetric robot. This is governed by the *n* values of K_r, which are set to 1 or minus 1 respectively.

$$\begin{aligned} f_{SYM((i,j)^{left},(i,j)^{right})} &= f_{LS((i-1,j)^{left},(i-1,j)^{right})} \\ &\quad + f_{LS((i,j)^{left},(i,j)^{right})} \\ &\quad + w_{ang} \cdot f_{\alpha S((i-1,j)^{left},(i-1,j)^{right})}, \\ where: & \\ f_{LS((k,j)^{left},(k,j)^{right})} &= (l_{k,j}^{left} - l_{k,j}^{right})^2 + (d_{k,j}^{left} - d_{k,j}^{right})^2, \\ and: & \\ f_{\alpha S((r,j)^{left},(r,j)^{right})} &= (\alpha_{r,j}^{left} + K_r \cdot \alpha_{r,j}^{right})^2. \end{aligned} \tag{2}$$

5) Total Length Constraint: For an assistive climbing robot the total length is important, as it influences among other the portability of the robot and the structural requirements of the DS. The limit constraint (LC) [2] only applies to the DH parameters of the one joint being optimized, therefore a new total length constraint (TLC) was introduced. The total length of the robot, L_{total}, was simply defined as the sum of all the link lengths and offsets for the complete robot. The constraint penalizes a total length larger than a set maximum or smaller than a set minimum, as can be seen in Equation (3).

$$f_{TLC(j)} = \begin{cases} (L_{total} - L_{max})^2 & for\ L_{total} > L_{max} \\ (L_{total} - L_{min})^2 & for\ L_{total} < L_{min} \\ 0 & otherwise. \end{cases} \tag{3}$$

6) Center Constraint: The two symmetric manipulators share the last joint (the global center joint) which thus have the same *x*, *y*, *z* position and twist angle α. However the *Z* axes for each must also coincide. This was ensured with an additional center joint constraint, which can be seen in Equation (4). Here $\mathbf{z}$ represents the *Z* unit vector of the final joint of the respective manipulator. The constraint was applied to all the symmetric joint pairs in the kinematic chain to take into account their possible effect on the orientation of the global center joint. Forward kinematics was used to propagate these effects from a given joint to the global center joint.

$$f_{CEN(j)} = ||\mathbf{z}_{c,j}^{left} - \mathbf{z}_{c,j}^{right}||^2. \tag{4}$$

7) Obstacle Definition: The obstacle avoidance constraint used here is the same as that used in [2]. To be able to model a relatively realistic environment, the obstacles used were initially defined as simple rectangular prism primitives. See Fig. 3. These obstacles were then grown by the radius of the robot, here 62.5 mm was used, and approximated by spheres. See Fig. 4a. If the distance from a link or joint origin to the center of a sphere is less than the radius of the sphere a collision is defined. The cost of the collision is then proportional to the depth of the penetration.

8) Global Convergence: The Grid Method performs an optimization for each unit grid (here a symmetric joint pair), starting from the extremes, for each task point. The global convergence criterion used was the sum of the cost functions for all the symmetric unit grids, across *m* task points and *c* symmetric joint pairs. The criterion can be seen in Equation (5).

$$F_{total} = \sum_{j=1}^{m} \left(\sum_{i=1}^{c} F_{SUG}(\mathbf{x}_{i,j}^{left}, \mathbf{x}_{i,j}^{right}) \right) < u. \tag{5}$$

9) Special Formulation for Spherical Center: Some robot configurations, like the 7 DOF configuration used here, have joints with several intersecting axes. This also means that a large number of DH parameters need to be forced to zero, restricting the search for solutions. As was done in [2] with a spherical wrist, the three central joints for the 7 DOF configuration used here were replaced with

DOI: 10.1109/TRO.2011.2160668

a spherical joint located in the center joint location. These joints could then be represented as a single joint with a single set of design variables $(x, y, z, \alpha)_c$. The three joint angles were then solved for using inverse kinematics during optimization of this joint.

B. Practical Considerations

1) Setting the Cost Function Weights: All the constraints used here have zero as the optimum. The angles and distances for the kinematic description all have magnitudes on the order of 10^0 as radians and meters are used as units. This simplifies the process of setting the weights manually somewhat. A strategy used here was to set the obstacle avoidance weight several orders of magnitude higher than the remaining weights, to reduce the chance of the robot getting trapped in a local minima with a collision. For example in the overlap between two spheres representing an obstacle. The remaining weights were then manually tuned to attempt to balance the respective terms in the cost function. In [5] an adaptive algorithm was used to alter the weights during execution. This can help reduce the time and effort to find an optimal solution, and is particularly important if measures are included. For example the dexterity measure used in the original Grid Method, that should be minimized, but not necessarily to zero. Including a measure and adapting the weights in a similar manner for the modified Grid Method is beyond the scope of the work presented here, but is probably feasible given the similarity of the cost function and the constraints used.

2) Efficiency of the Modified Grid Method: A simple comparative study was performed to assess the effect of the symmetry-specific modifications made to the Grid Method. The modified method (*ModGrid*) was compared with the original Grid Method (*Grid*) and the General Formulation Method (*GFM*). The latter was also used as the basis for comparison in [2], and performs the optimization over all the joints and task points at the same time. It is thus a non-grid approach and has $4 * m * n$ design variables, as compared with 4 and 8 for the original and modified Grid Method, respectively. The total computation time in Matlab on a 3 MHz Intel Core 2 Duo was compared for 5 and 7 DOF robots optimized for 2, 3 and 4 task points. The constraints used were EC and C (with weights 1 and 2, respectively). For *Grid* and *GFM* the center constraint C was replaced with an equivalent of the Desired Orientation Constraint (DOC) used in the original Grid Method (with weight 1). The convergence criteria used was increased with both the number of task points and DOF used, specifically $u = 0.0005 * m * n$. As can be seen in Fig. 2, the results indicate that while not as efficient as the original, the modifications made to enable optimization of symmetric robots still makes it more than an order of magnitude faster than the *GFM* as the number of DOF and task points increases.

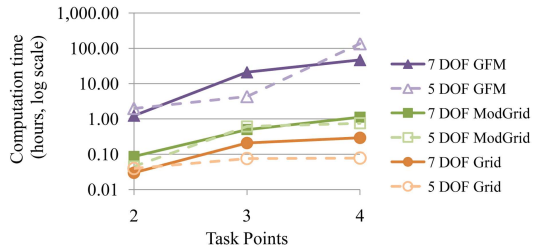
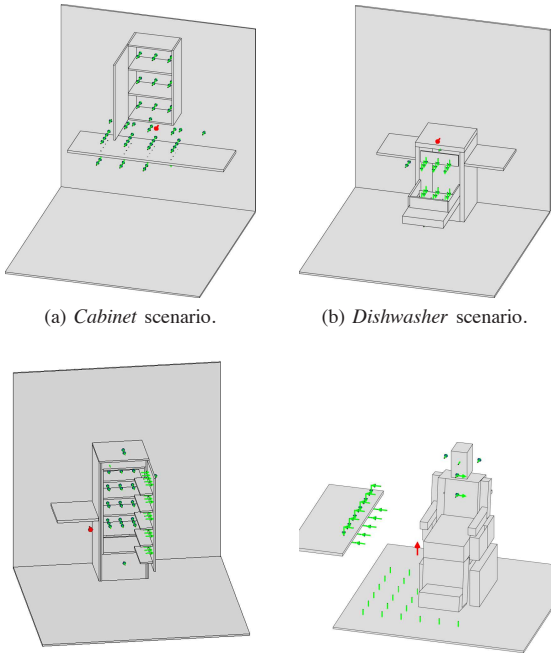


Fig. 2. Total computation time for the original (*Grid*) and modified Grid Method (*ModGrid*), as well as the General Formulation Method (*GFM*).

C. Design Methodology

1) Overview: This section describes the application of the modified Grid Method for symmetric robots to the design of an assistive climbing robot. The design methodology followed was to define 4 scenarios (including tasks and obstacles) in a specific assistive robotics environment (kitchen). 3 different robot configurations (5, 6 and 7 DOF) were then optimized for a subset of task points in each scenario, followed by a quantitative comparison of each robot design on the complete set of tasks in all scenarios.

2) Task Scenarios: A kitchen environment was used, based on an exact model of a real kitchen test-bed in the RoboticsLab at UC3M, see Fig. 1. The 4 scenarios selected in this environment can be seen in Fig. 3. Each had 25 (*dishwasher*) to 72 (*cabinet*) task points, consisting of two distinct types. The first type required all 6 DOF (position and rotations) of the end-effector to match the desired task. For example DS locations. Other task points had a relaxed yaw requirement (in the end-effector frame), enabling the 5 DOF robot to perform tasks where the yaw rotation was not essential. The different tools used were also taken into account in the task point placement. For example the task points representing the user eating from a spoon. These were placed 100 mm in front of the user's mouth to simulate the length of the spoon.



(a) *Cabinet* scenario. (b) *Dishwasher* scenario.

(c) *Refrigerator* scenario. (d) *Wheelchair* scenario.

Fig. 3. Task scenarios defined. Small green arrows indicate the origin and direction of the *Z* axis of the end-effector for a given task point, while large red arrows indicate the DS location used as the base.

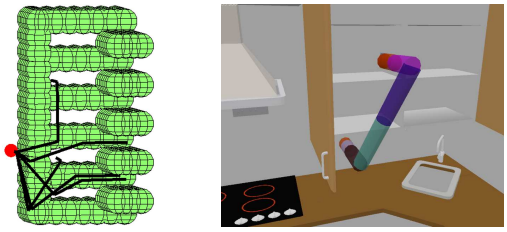
3) Robot Configurations: By robot configuration is here meant the definition of a set of basic properties that were not to be optimized. This included the number of DOF and the twist angle of each joint. Three main considerations were used. First, the robot should have 5, 6 or 7 revolute DOF. The current design has 5, limiting it in some tasks where 6 DOF of the end-effector is required. Second, the robot should have the ability to roll the end-effectors about the local z-axis. This may be needed to perform the docking procedure and to interchange end-effectors. Third, the robot should be symmetric. The definition of robot symmetry used here is that the robot must be able

DOI: 10.1109/TRO.2011.2160668

to perform the same set of tasks when docked with the left and when docked with the right end-effector in the same DS.

With these considerations 3 robot configurations were defined. The 5 DOF version was based on the current ASIBOT configuration (*RPPPR*). Symmetry is across the central joint (joint 3). The 6 DOF robot was made symmetric about the center link. The end joints had to be roll joints according to the constraints, and the remaining joints were chosen to give a *RPYYPR* configuration. The 7 DOF configuration was also based on the ASIBOT design, but with roll joints added to the two links adjoining the central pitch joint (*RPRPRPR*). The properties to optimize for each configuration were the two pairs of symmetric link lengths (for the 6 DOF configuration also the center link). In addition the joint limits and any required joint offsets needed to accommodate the range of movement could be defined for each design.

4) Kinematic Optimization of Robot Designs: The modified Grid Method described in section II-A was used to obtain one kinematic design for each configuration for each scenario. See Fig. 4a. A subset of around 5 task points was used for each optimization to limit the computational burden. The subset was chosen to represent the different classes of tasks possible in each scenario, while being achievable with the design constraints used. For example moving from DS to DS, manipulating objects in the cabinet and on the kitchen desktop in the *cabinet* scenario. In this way, the probability that similar tasks of all classes could be performed with the final design was increased. The process was performed iteratively until a suitable subset was achieved. A common set of constraints was used for all. The first was that the maximum length should not be longer than 1500 mm, to make sure the robot would remain portable. The second constraint used was to limit the first real link (from the end-effector to the first non-roll joint) to between 200 and 400 mm. The remaining length of the robot was constrained to be within 400 mm and 1000 mm. A virtual 7th joint was added to the middle of the central link of the 6 DOF configuration with all properties except link length forced to zero. For the joint limits, ASIBOT was used as a reference. The roll joints were given a 360° range. The pitch joints in ASIBOT all have ±135° joint limits, mainly due to lack of joint offsets. The weights of the unit grid cost function were set manually. The goal was to achieve a smallest possible design that could reach the set of task points given without exceeding joint limits or colliding with the obstacles used. A satisfactory design was assumed when this could be achieved with a reasonably coherent design for all the task points used (typically ±10 mm for each link length).



(a) Optimization. (b) Virtual testing.

Fig. 4. Examples of optimization (7 DOF robot in *refrigerator* scenario shown) and virtual testing (5 DOF robot in *cabinet* scenario shown).

5) Virtual Testing of Robot Designs: Virtual testing was used to compare the 12 designs on the full set of tasks in all 4 scenarios to obtain a global "optimal" kinematic design. The criteria used was the number of task points achievable in each case. The virtual environments included realistic models of the obstacles in all scenarios and simplified robot models representing the designs obtained. See Fig. 4b. Collisions between the robot and the environment were simulated and joint limits used to limit the workspace of the robots. Approximately the same DS location as used for the optimization was used, but the testing was performed on the complete set of task points for each scenario. For each task point, the robot was commanded to the desired Cartesian position and orientation of the task. If allowed for the specific task, the yaw rotation requirement (in end-effector frame) for the task was ignored in the differential inverse kinematics solver used. If the robot successfully reached the task point a success was recorded.

6) Mass Estimation of Robot Designs: Another aspect considered when comparing the robot designs found was the expected total mass of the robot. The main assumption made was that the robot designs would have a similar distribution of mass across subsystems as that of ASIBOT. This meant that the total mass could be split into two parts. The mass of the motors, reductors and motor drivers vary with the number of DOF and actuators. This mass represents about 40% of the total mass in ASIBOT and was assumed to represent the same in the robot designs obtained here. The remaining 60% then represents the mass of the structure, common electronics and the end-effector docking mechanisms. Both masses were assumed to vary linearly with the total length of the robot. This allowed for a crude estimation of the additional structural and motor torque requirements for lifting the same payload with a longer moment arm. The ASIBOT data used was a mass of 12 kg and total length of 1336 mm.

III. Results and Discussion

A. Optimized Robot Kinematic Designs

The results of the optimization process can be seen in Table II. The robots are named with a single letter (*C-cabinet*, *D-dishwasher*, *R-refrigerator* and *W-wheelchair*) followed by the number of DOF. $L_{extreme}$ is the length of the symmetric link pair from the end-effector to the first non-roll joint, L_{middle} is for the other symmetric link pair, while L_6 is the total length of the center link unique to the 6 DOF configuration. The total length, L_{total} and the total estimated mass, M_{total} are also shown for each robot. As can be seen from the table, the robots optimized for *cabinet* has the longest $L_{extreme}$ in comparison with those optimized for other scenarios. From the DS used this makes sense, as the robot is required to avoid the underside of the cabinet to access any of the shelfs. The total length for the robots optimized ranges from 990 mm (*D5*) to the maximum 1500 mm (*C5*). The robots optimized for the *cabinet* scenario are the longest, while the robots optimized for the *refrigerator* and *dishwasher* scenarios are the shortest. The average length of the robot designs found was 1223 mm and the average estimated mass was 11.9 kg (ranging from 8.9 kg to 14.3 kg).

B. Quantitative Comparison of Robot Designs

Table III shows the results of the virtual testing of each robot design for each of the four scenarios. For each case a number from 0 to 1 is shown, representing the ratio of the task points achieved to the task points available in the given scenario. The result for a robot both optimized and tested for a given scenario (for example *C5* for the *cabinet* scenario) is highlighted in bold. As can be seen from the table, 10 out of 12 of the robots optimized for a given scenario also had the best results for this scenario and number of DOF. For the two that did not, the scores were not far from the average of the other designs with the same DOF for that scenario. This was the *C6* robot with a score of 0.26 for the *cabinet* scenario (average 0.28 for 6 DOF designs) and the *R5* robot with a score of 0.46 in the *refrigerator* scenario (average 0.47 for 5 DOF designs). An unpaired t-test found no significant difference between the mean score of the 12 robots optimized and

DOI: 10.1109/TRO.2011.2160668

TABLE II
OPTIMIZED ROBOT DESIGNS WITH LENGTHS AND ESTIMATED MASS (IN MM AND KG RESPECTIVELY)

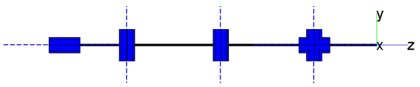
DOF	Robot	$L_{extreme}$	L_{middle}	L_6	L_{total}	M_{total}
5	*C5*	385	365	n/a	1500	13.5
	D5	200	295	n/a	990	8.9
	R5	195	315	n/a	1020	9.2
	W5	260	400	n/a	1320	11.9
6	*C6*	260	235	340	1330	12.9
	D6	215	220	300	1170	11.3
	R6	200	220	230	1070	10.4
	W6	250	215	380	1310	12.7
7	*C7*	270	415	n/a	1370	14.3
	D7	210	320	n/a	1060	11.0
	R7	255	340	n/a	1190	12.4
	W7	205	465	n/a	1340	14.0

TABLE III
TASKS ACHIEVABLE WITH EACH ROBOT DESIGN IN EACH SCENARIO (C-CABINET, D-DISHWASHER, R-REFRIGERATOR, W-WHEELCHAIR)

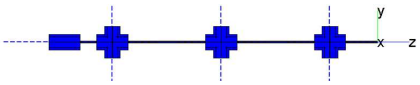
DOF	Robot	*C*	*D*	*R*	*W*	Total
5	*C5*	**0.57**	0.56	0.51	0.38	2.01
	D5	0.39	**0.56**	0.37	0.28	1.60
	R5	0.44	0.56	**0.46**	0.34	1.81
	W5	0.54	0.56	0.55	**0.61**	2.26
6	*C6*	**0.26**	0.80	0.52	0.44	2.03
	D6	0.31	**0.80**	0.40	0.43	1.93
	R6	0.28	0.80	**0.52**	0.30	1.90
	W6	0.26	0.72	0.43	**0.49**	1.91
7	*C7*	**0.74**	0.56	0.54	0.79	2.62
	D7	0.63	**0.56**	0.48	0.46	2.12
	R7	0.65	0.56	**0.58**	0.57	2.37
	W7	0.64	0.56	0.54	**0.90**	2.64

tested in the same scenario (0.59) and the remaining robots (0.50), $t(46) = 1.66$, $p = 0.103$. However, this is partially due to the very uniform results of the *dishwasher* scenario in general. For the remaining three scenarios a weakly statistically significant increase of 23% was found, $t(34) = 1.96$, $p = 0.059$. Paired t-tests were used to explore the level of generalization provided by the robots optimized for given scenarios. It was found that the mean score for the robots optimized for the *cabinet* scenario (0.56) was significantly higher than for those optimized for the *dishwasher* (0.47) and *refrigerator* (0.51) scenarios, with $t(11) = 2.76$, $p = 0.019$ and $t(11) = 2.24$, $p = 0.047$, respectively. There was a weakly statistically significant increase for the robots optimized for the *wheelchair* (0.57) with respect to the *dishwasher* scenario, $t(11) = 2.09$, $p = 0.061$.

Only a subset of the potential task points identified was used for the optimization. As a result not all the tasks in a given scenario were possible with the "optimal" robot. However, when including real-world tasks, complex obstacles and tight robot constraints in a TOD approach only a subset of task points that are achievable, with the design constraints given, will ensure that the optimization process can be successful. The design approach followed here is made less sensitive to the selection of this subset by the quantitative comparison of the different robot designs on the full set of tasks in all 4 scenarios. For designers this also means that quantitative trade-offs can be made when selecting the final design, for example based on user-preferences about the importance of one scenario versus another.



(a) Original ASIBOT design, length 1336 mm.



(b) Design with highest score (W7), length 1340 mm.

Fig. 5. Comparison of original and "best" design, drawn to scale.

The scores obtained for each design were summed up to get a total score. The 7 DOF robot *W7* achieved the highest total score, 2.64, and is visualized in Fig. 5b. The average for 7 DOF designs, 2.44, was also higher than the two other configurations. Interestingly the average for the 5 DOF configuration (1.92) was about the same as that for 6 DOF (1.94). The geometry of the 5 and 7 DOF configuration resembles that of the human arm. As the environments in assistive robotics are typically designed for the latter this may give the 5 and 7 DOF an advantage. The design most similar to the current ASIBOT robot, *W5*, gained the fourth highest score (2.26). This may indicate that the current design approaches the optimal for the DOF and the tasks used. It also indicates that moving to a 7 DOF design could increase the performance by about 17%, but that this would have to be traded off with a 17-18% increase in mass.

IV. CONCLUSION

A task-oriented design process was applied to the kinematic design of an symmetric assistive climbing robot, ASIBOT. A design methodology was proposed, beginning with the definition of 4 typical kitchen scenarios, including task points of interest and obstacles. Robot designs with 5, 6 and 7 DOF were then optimized on a subset of tasks for each scenario. The optimization method used was based on the Grid Method, modified to provide an efficient grid-based optimization for symmetric robots. A cost function with only simple constraints was used. Including more complex measures that further improve the design once the constraints have been met would likely require adaptive tuning of the weights to balance the different terms in the cost function during optimization. A quantitative comparison of the 12 designs on the full set of tasks in all 4 scenarios was used to decide upon a suitable robot kinematic structure. The results showed that a 7 DOF design could increase the number of tasks achievable by 17% in comparison to the best 5 DOF design, but that this would come at a cost of a 17-18% increase in total mass. Although a good design can also be reached without such an approach, the extensive comparison should increase the likelihood of finding one for a wide range of tasks and environments. It also seems suitable for other applications where there are large sets of well-known tasks to be performed. An interesting future consideration for climbing robots in general would be to also consider the grasp locations in the environment in the optimization process.

REFERENCES

[1] A. Jardón, A. Giménez, R. Correal, R. Cabas, S. Martínez, and C. Balaguer, "A portable light-weight climbing robot for personal assistance applications," *Industrial Robot: An International Journal*, vol. 33., no. 4. pp. 303-307, 2006.

[2] J. Y. Park, P. H. Chang, and J. Y. Yang, "Task-oriented design of robot kinematics using the Grid Method," *Advanced Robotics*, vol. 17, no. 9, pp. 879-907, 2003.

[3] J. O. Kim and P. K. Khosla, "A Formulation for Task Based Design of Robot Manipulators," in *Proc. IEEE/RSJ Int. Conf. in Intelligent Robots and Systems*, Yokohama, Japan, vol. 3, pp. 2310-2317, 1993.

[4] O. Chocron and P. Bidaud, "Evolutionary Algorithms in Kinematic Design of Robotic Systems," in *Int. Conf. on Intelligent Robots and Systems*, Grenoble, France, vol. 2, pp. 1111-1117, 1997.

[5] J. Y. Park, P. H. Chang, and J. O. Kim, "A Global Optimal Approach for Robot Kinematics Design using the Grid Method," *International Journal of Control, Automation, and Systems*, vol. 4, no. 5, pp. 575-591, 2006.

DOI: 10.1109/TRO.2011.2160668

The research leading to these results has received funding from the RoboCity2030-II-CM project (S2009/DPI-1559), funded by Programas de Actividades I+D en la Comunidad de Madrid and cofunded by Structural Funds of the EU.