

“© 2015 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

A Framework for Task-Based Evaluation of Robotic Coworkers

Marc G. Carmichael, Bryan Moutrie, and Dikai Liu
Centre for Autonomous Systems
University of Technology, Sydney, Australia
Email: marc.carmichael@uts.edu.au

Abstract—Compared to a robotic system that performs a task alone, a robot coworker performing tasks in collaboration with a human operator is subject to additional constraints which can limit the ability of the system to perform the task as required. This work presents a framework for analyzing the ability of a robotic coworker to perform specific tasks in collaboration with a human. The framework allows systematic evaluation of robotic systems based on traditional robot performance measures such as reachable workspace and payload capacity, as well as considering additional factors which arise due to the task being performed collaboratively with a human; such as the reach and strength of the human, human-robot collision, and satisfying desired assistance paradigms. Application of the framework is demonstrated in a case study analyzing a robot designed to assist a human during a materials handling task.

I. INTRODUCTION

It is anticipated that in the near future robotic systems will become a part of everyday life, working alongside and collaboratively with humans in a variety of industrial and domestic tasks [1]. It is easy to imagine a scenario in which a human wishes to select a robot coworker to assist them in a specific task, but choosing the best suited robot is a challenge due to the range of robots available. Alternatively, when there are several tasks that need to be performed, it is expected that the task best suited for the robot be chosen. What is needed is a means for evaluating the robot's ability to perform the desired tasks.

Traditional metrics used for evaluating robots such as reachable workspace or payload capacity are good measures of how suited a robotic system is when performing a task in isolation. However when operating as a coworker, providing assistance to a human operator, the system is subject to additional constraints that limit the ability of the robot to perform a desired task. Consider the scenario shown in Fig. 1 where a human operator controls a tool attached to the end-effector of the robot, whilst the robot provides physical assistance by supporting some proportion of the load as the task is performed. With the tool coupled to both the human and the robot coworker, the task is limited not only by the robot's reachable workspace but by the human's reachable workspace as well. Tool positions which are reachable by both the human and robot individually may not be reachable collectively, as doing so requires them to adopt configurations occupying the same physical space. The requirement of avoiding human-robot collision can significantly limit how a task can be performed [2]. Another limitation may be due to an application requiring

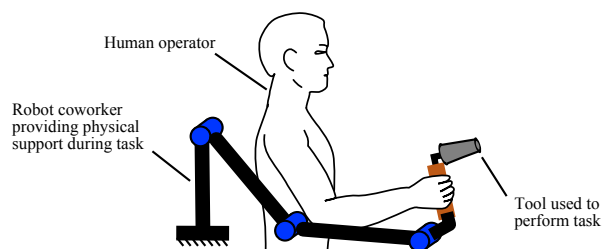


Fig. 1. Example of a task being performed collaboratively by a human and robot coworker. The human controls a tool attached to the robot's end-effector, while the robot assists by supporting a portion of the task load.

the robot to provide assistance to its operator governed by a specific paradigm developed for that application. An example is the Assistance-As-Needed paradigm which is often used in robotic rehabilitation applications [3], [4], [5]. Implementing such assistance paradigms often means that the amount of assistance required from the robot (i.e. its required payload capacity) depends not only on the task being executed, but also the capability of the human to contribute to the task as well. Overall, additional constraints which arise from a robot being utilized as a coworker, such as the human's reachable workspace and task contribution, human-robot collisions, and satisfying a specific assistance paradigm are all factors that need to be considered when evaluating the suitability of the robot for specific tasks; and considering these factors requires new analytical frameworks for evaluating robotic coworker systems to be developed.

This work presents an analytical framework for task-based evaluation of robotic coworker systems that work collaboratively with human operators. The framework allows the ability of a robot to be quantitatively evaluated in the context of the desired task, and identify the factors limiting the extent to which the task can be performed as desired. A crucial component in the framework is a component representing the human operator. This component is used to check for collisions with the robot, and to calculate the physical strength the worker has to contribute towards the task; required for analyzing the robot's ability to satisfy certain assistance paradigms. Also presented are proposed methods of handling redundancies which occur when the robot and/or human have more degrees-of-freedom (DOF) than are required to perform the task.

In Section II a generalized framework for analyzing robotic systems is presented. Next in Section III a case study demon-

strates the framework applied to a robotic system designed to assist in a materials handling task. Section IV discusses the presented analytical framework.

II. FRAMEWORK

The framework comprises of three parts used to represent and evaluate; 1) the robot coworker, its reachable workspace and payload capacity, 2) the human operator, their reachable workspace and strength, and 3) the task to be performed.

A. Robot reachable workspace and payload capacity

The kinematic configuration of the robot coworker is defined using vector $\mathbf{q}_R = [q_{R1}, q_{R2}, \dots, q_{Rv}]^T$ of generalized coordinates. With the robot mounted on a fixed base, \mathbf{q}_R may simply define the joint angles of the robot. Alternatively if the robot is mobile, coordinates in \mathbf{q}_R may represent joint angles as well as its location in the environment.

Certain task-related variables have a kinematic dependency on the robot's generalized coordinates, for example the position of the end-effector. Using forward kinematics this relationship can be calculated. This is represented as $\mathbf{x} = G(\mathbf{q}_R)$ with vector \mathbf{x} representing the kinematically dependent task-variables. The inverse $\mathbf{q}_R = G^{-1}(\mathbf{x})$ is required for calculating a robot configuration corresponding to a particular vector \mathbf{x} . If an inverse solution for a vector \mathbf{x} does not exist, then \mathbf{x} is said to be outside the reachable workspace of the robot.

The robot's payload capacity is the amount of force it can support at its end-effector. When acting as a coworker to help a human operator support a load, its payload capacity determines how much assistance the robot can provide, and is necessary for determining if an assistance paradigm can be satisfied. We generalize the robot's payload capacity as f_R , calculated as a function of its coordinates (1).

$$f_R = F_R(\mathbf{q}_R) \quad (1)$$

B. Human reachable workspace and strength capacity

The manner in which the human operator is represented in the framework has many similarities to that of the robot. The configuration of the human is defined using a vector of generalized coordinates $\mathbf{q}_H = [q_{H1}, q_{H2}, \dots, q_{Hw}]^T$. As is the case with the robot, certain task-related variables have a kinematic dependency on the human's configuration. A kinematic model of the human allows the relationship $\mathbf{x} = H(\mathbf{q}_H)$ to be calculated. The inverse $\mathbf{q}_H = H^{-1}(\mathbf{x})$ is required for calculating the human configuration corresponding to a particular vector \mathbf{x} . If an inverse solution does not exist, then \mathbf{x} is said to be outside the reachable workspace of the human operator.

The amount of strength that the human has to contribute to the task is analogous to the payload that the robot has to contribute. In this work it is assumed the human performs the task by controlling a tool using their hand, and hence the human's capacity to contribute to the task is their strength at their hand. The human's strength, which we represent as f_H (2) is calculated from a musculoskeletal model of the upper limb [6] using a previously developed optimization model [7].

$$f_H = F_H(\mathbf{q}_H) \quad (2)$$

C. Desired task-space versus performable task-space

To represent the desired task to be performed, we define the following;

- A desired task-space (represented by set \mathbf{T}_D) with dimensions corresponding to task-related variables
- An assistance paradigm the robot is required to satisfy
- Any remaining task-related parameters required to perform the task-based analysis

The task-space contains dimensions corresponding to task-based variables that need to be fulfilled during operation. This space is segmented into two distinct subspaces. The first subspace consists of m dimensions corresponding to variables that have a kinematic dependency on the human or robot, for example the desired positions of a tool. Points within this first subspace are represented using the vector \mathbf{x} . The second subspace consists of n dimensions corresponding to variables kinematically independent from the human and robot. An example of such a task-variable would be the mass of a load being carried, if this load was known to vary. Points within this second subspace are represented using the vector \mathbf{s} .

A point defined in all $m+n$ dimensions of the task-space is a combination of vectors \mathbf{x} and \mathbf{s} , and is represented by task vector \mathbf{t} (3). Although it is expected that there will always be some task variables with kinematic dependency on the human and/or robot, it is possible that some tasks may contain no kinematically independent variables, in which case $n = 0$ and the task vector is simply $\mathbf{t} = \mathbf{x}$. With the task-space defined, each of its dimensions are bounded and quantized to produce a finite set of task vectors represented as \mathbf{T}_D (3), with the subscript denoting that this is the *desired* task-space.

$$\mathbf{t} \in \mathbf{T}_D, \text{ where } \mathbf{t} = \begin{bmatrix} \mathbf{x} \\ \mathbf{s} \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} \text{ and } \mathbf{s} = \begin{bmatrix} s_1 \\ \vdots \\ s_n \end{bmatrix} \quad (3)$$

As an example, imagine a task that requires a tool to achieve a set of predefined positions and orientations in three dimensional space; with the tool used to carry an object of mass M which may vary between 0 and 10kg. Since the tool's position and orientation are kinematically related to the robot and human we would make the first $m = 6$ dimensions of \mathbf{T}_D correspond to the x - y - z position and roll-pitch-yaw orientation of the tool. The last $n = 1$ dimension of the task-space would then correspond to the variable mass M bounded between 0 and 10kg. The result is a 7-dimensional task-space which fully defines the desired task to be performed. Any remaining task-related parameters which are not represented in the task-space but are still necessary to analyze the robot's ability to perform the task are predefined and remain constant during the analysis; if not they would be included as additional dimensions in the task-space.

If the robot is required to assist the human operator under a specific assistance paradigm, then this paradigm is also

defined. This high-level objective defines how the robot should assist its human operator, and is formulated as an expression dependent on the capabilities of the human, the robot, and the task load. For example the following statement is a paradigm which could be desired for a specific application: *The robot should support 50% of the task load at all times, unless the human is unable to perform the task unassisted, in which case the robot should support 100%*. The ability of a robot to satisfy an assistance paradigm depends on the robot's payload capacity f_R and the human operator's strength f_H . With the capability of the human and robot to contribute to the task calculated, a function P (4) is created to determine whether or not the paradigm is satisfied.

$$P(f_R, f_H) = \begin{cases} 0, & \text{if the paradigm is satisfied} \\ 1, & \text{if the paradigm is not satisfied} \end{cases} \quad (4)$$

Analyzing the desired task-space \mathbf{T}_D to determine how much of this space can actually be performed forms the basis of evaluating the robotic coworker. Task-space \mathbf{T}_D is first quantized into discrete points with a resolution suitable for the analysis. Each point is then analyzed individually based on the following three criteria:

- 1) The point is mutually reachable by both the human operator and the robot coworker (i.e. solutions for $\mathbf{q}_H = H^{-1}(\mathbf{x})$ and $\mathbf{q}_R = G^{-1}(\mathbf{x})$ exist)
- 2) The point can be reached without the human and robot colliding (i.e. $C(\mathbf{q}_H, \mathbf{q}_R) = 0$, see Section II-E)
- 3) The robot is able to assist the human so as to satisfy the assistance paradigm (i.e. $P(f_R, f_H) = 0$)

Once the entire desired task-space \mathbf{T}_D has been analyzed, all points found to satisfy the above criteria are collected to produce set \mathbf{T}_P called the performable task-space. This subset, formally expressed as shown in (5), represents the task-space that can be successfully performed by the robot.

$$\mathbf{T}_P = \left\{ \mathbf{t} \mid \mathbf{t} \in \mathbf{T}_D, \begin{array}{l} \mathbf{q}_R = G^{-1}(\mathbf{x}), C(\mathbf{q}_H, \mathbf{q}_R) = 0, \\ \mathbf{q}_H = H^{-1}(\mathbf{x}), P(f_R, f_H) = 0 \end{array} \right\} \quad (5)$$

D. Handling redundancy within the framework

To analyze each point in the task-space, the corresponding configuration of the human operator and robot coworker needs to be calculated. An example of transforming from the task-space into the coordinate spaces of the human and robot is shown in Fig. 2. A challenge arises when the human or robotic system contains more degrees of freedom than dependent task-space variables, i.e. $\dim(\mathbf{q}_R) > \dim(\mathbf{x})$ or $\dim(\mathbf{q}_H) > \dim(\mathbf{x})$. This redundancy means that the task may be performed by the human or robot in many, possibly infinite ways. For example there may be infinite combinations of robot coordinates that place the tool at the desired position. We propose several methods of handling this redundancy within the scope of the presented framework.

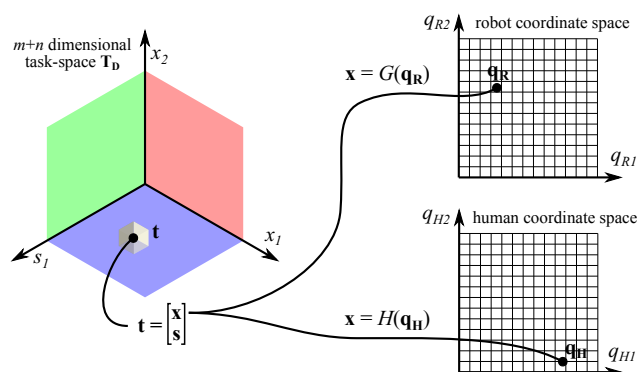


Fig. 2. Relationship between the task-space \mathbf{T}_D , and the robot and human's coordinate spaces. The example illustrated shows a task-space with three dimensions (x_1 , x_2 , s_1), a human coordinate space with two dimensions, and a robot coordinate space with two dimensions. From our definition of the task-space, only the first two dimensions (x_1 and x_2) have kinematic relationships with the human and robot in the example shown.

1) Introduce constraints which resolve the redundancy:

One approach is to introduce constraints into the system so as to resolve the redundancy. For example, treating the human upper limb as a kinematic chain with a 3-DOF shoulder and 1-DOF elbow, and fixing the position of the wrist in 3D space there exists a 1-DOF redundancy which is often referred to as the swivel angle [8]. A method of resolving this redundancy is to specify the swivel angle using a function dependent on the position of the wrist. Enforcing this as a constraint allows the inverse kinematic transformation from task-space into the robot or human coordinate spaces to be resolved.

2) *Parameterize the redundancy and augment the task-space:* How a redundancy is resolved may affect factors such as human-robot collision, or how much the human and robot can contribute to the task. It may therefore be beneficial to examine this as part of the analysis. For a given point \mathbf{t} in the task-space, the redundant DOFs in the human and robot systems create a null space which can be explored. A method of handling the redundancies is to parameterize, quantize, and then add each null space dimension to the task-space. This augmented task-space can then be analyzed as if no redundancy exists, since each point has all redundancies resolved. Once the analysis on the augmented task-space has been performed, the results may be projected into the original task-space using suitable heuristics.

3) *Utilize optimization methods:* It may be desired to not only find a solution which satisfies the task, but also resolve the redundancy so as to achieve a secondary objective. For example the task may be performed whilst minimizing the human discomfort, or maximizing robot manipulability. In this approach, for each point in the task-space, an optimization explores the null space to maximize/minimize the objective. Due to the non-linearity of the human and robot systems, and the constraints that need to be satisfied, significant computational power may be required to implement this approach.

E. Human-robot collision checking

An important component of the framework is checking for collisions between the human and robotic system. Collisions can significantly reduce the effective workspace that the robot can operate in, and have motivated the development of analytical frameworks that evaluate this [2]. In general terms, with the robot and human configurations defined by vectors \mathbf{q}_R and \mathbf{q}_H , collision is evaluated by calculating if they occupy the same physical space. This notion is represented by (6).

$$C(\mathbf{q}_H, \mathbf{q}_R) = \begin{cases} 0, & \text{if there is no collision} \\ 1, & \text{if there is a collision} \end{cases} \quad (6)$$

III. CASE STUDY

In this case study the ability of a robotic system to assist a human during a materials handling task is evaluated. The purpose of the case study is to clarify the steps involved in using the framework to evaluate robotic systems, and to demonstrate how the framework can be used to gain significant insights about factors limiting a robot's ability to perform desired tasks.

The task to be analyzed requires the human and robot to cooperatively manipulate an object of weight $L = 100N$ throughout a workspace in front of the human. For convenience this workspace is shaped as a cube with sides 0.4m in length as shown in Fig. 3. The tool carrying the object is defined as being coincident with the human's wrist, and is required to reach across the entire cubic workspace; specific tool orientations at each position are not required. It is assumed the human is in a seated position and hence their torso is considered as being in a fixed location. The base of the robot is also fixed in a position such that its shoulder is offset 0.1m to the right, and 0.1m upwards relative to the human's shoulder. The offset is defined to simulate kinematic misalignment for the purpose of demonstration.

During the task the robot is required to provide assistance such that the carried load is fully supported, however the amount of physical effort the human is required to contribute (relative to the maximum they are capable of) is a task variable which may be changed in response to external factors. For example as the human becomes fatigued their expected contribution to the task might be reduced. This forms the basis of the assistance paradigm that the robot is required to satisfy.

A. Defining the desired task-space and assistance paradigm

To represent the desired task, a task-space with four dimensions is created. The first three dimensions $\mathbf{x} = [x_1, x_2, x_3]^T$ represent the x - y - z position of the tool in the cubic workspace since these task variables have a kinematic dependency on the human and robot. These first dimensions are bounded to the size of the cubic workspace and quantized into discrete points spaced 0.05m apart. The fourth task-space dimension, $s = s_1$, is used to represent how much effort the human operator is expected to contribute towards supporting the load, relative to their maximum contribution. This dimension is bounded from

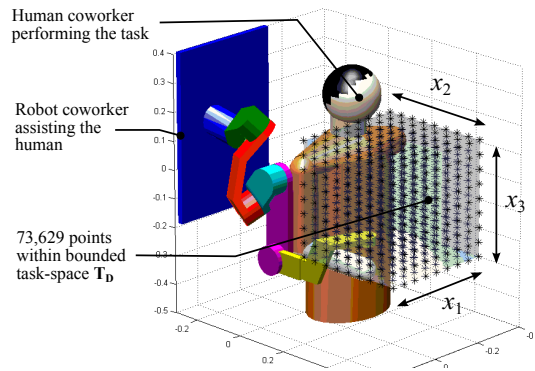


Fig. 3. The task requires the load to be supported throughout a cubic workspace located in front of the human and robot. The location of the load in the workspace is parameterized by $\mathbf{x} = [x_1, x_2, x_3]^T$.

0 to 1, quantized in steps of 0.01. The resulting desired task-space \mathbf{T}_D contains 73,629 discrete points, each represented by vector $\mathbf{t} = [x_1, x_2, x_3, s_1]^T$.

To check if the assistance paradigm is satisfied the maximum force contributions that the human operator and robot coworker can produce to support the load against gravity, represented as f_H and f_R , are calculated using the appropriate methods. Whether or not the assistance paradigm can be satisfied is then expressed as (7).

$$P(f_H, f_R) = \begin{cases} 0, & \text{if } s_1 \cdot f_H + f_R \geq L \\ 1, & \text{if } s_1 \cdot f_H + f_R < L \end{cases} \quad (7)$$

B. Human reachable workspace and strength

With the human operator's torso assumed fixed, only the upper limb needs to be represented. To model the upper limb a publicly available musculoskeletal model is utilized [6]. The model contains degrees of freedom representing the shoulder, elbow and wrist; however since wrist orientation is not important, only the shoulder and elbow are considered. This reduces the model to consisting of a 3-DOF shoulder and a 1-DOF elbow.

Given that the human model is 4-DOF and there are three kinematically dependent task-space dimensions, there exists a 1-DOF redundancy that needs to be handled. To represent the non-redundant DOFs, coordinates defining the position of the wrist are used, i.e. $\mathbf{x} = [x_1, x_2, x_3]^T$. The remaining redundant DOF is parameterized using coordinate ϕ to represent the swivel angle, referring to the angle made by the elbow rotating about an axis passing through the human's shoulder center of rotation and their wrist [8]. The configuration of the arm is hence fully defined by vector $\mathbf{q}_H = [x_1, x_2, x_3, \phi]^T$. To resolve the redundancy the method described in Section II-D1 is utilized. This method requires a constraint to be added such that solutions for the inverse relationship $\mathbf{q}_H = H^{-1}(\mathbf{x})$ can be found. The constraint added in this case study makes ϕ dependent on x_1 , x_2 , and x_3 such that an arm posture consistent with natural arm reaching is produced [8].

The capacity of the human to contribute towards the task is calculated by estimating the worker's strength at their hand

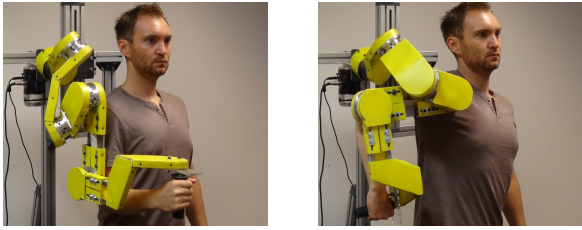


Fig. 4. The physically assistive robot analyzed in the case study.

to support the load being carried against gravity, represented as f_H . As previously mentioned this is calculated directly from the upper limb musculoskeletal model using a previously developed optimization model [7].

C. Robot reachable workspace and payload capacity

The robot coworker that assists the human performing the task is the arm-type robot shown in Fig. 4. This robot is specially developed for research on physical assistance applications, and has kinematics consisting of a 3DOF shoulder mechanism and a 1-DOF elbow roughly matching that of the human arm. The operator controls the robot by applying forces to a handle held in their hand, causing the robot's end-effector to follow their wrist by means of an admittance control scheme [9]. It is assumed the operator's wrist and the robot's end-effector remain coincident as the task is performed.

The configuration of the 4-DOF robot is defined by generalized coordinates $\mathbf{q}_R = [q_{R1}, q_{R2}, q_{R3}, q_{R4}]^T$ relating to the angles in the joints. Similarly to the case with the human arm model there exists a 1-DOF redundancy. This is resolved by aligning the swivel angle of the robot arm with that of the human arm using a projection method described in [10]. The payload capacity of the robot is calculated as a function of \mathbf{q}_R using traditional methods.

D. Calculation of the performable task-space

Each point \mathbf{t} in task-space \mathbf{T}_D is analyzed individually in a series of stages. The first stage involves finding the inverse kinematic solutions for both the human and the robot models, i.e. $\mathbf{q}_R = G^{-1}(\mathbf{x})$ and $\mathbf{q}_H = H^{-1}(\mathbf{x})$. If either solution cannot be found then the point in the task-space cannot be kinematically reached by the human and/or robot, and analysis starts over at the next point in the task-space. If the point is reachable, the next stage analyzes potential collisions between the human and robot by using their configurations along with geometric models to test if they occupy the same physical space. If it is deemed that no collision is occurring, analysis continues to the third stage. The forces that the robot and human are able to contribute to support the carried load are calculated, and then used in (4) to check whether or not the assistance paradigm is satisfied. This procedure is repeated for each point in the task-space, with points satisfying all the required criteria combined to create subset \mathbf{T}_P .

E. Results

To measure how capable the robot is at performing the task, the number of points in the performable task-space \mathbf{T}_P can

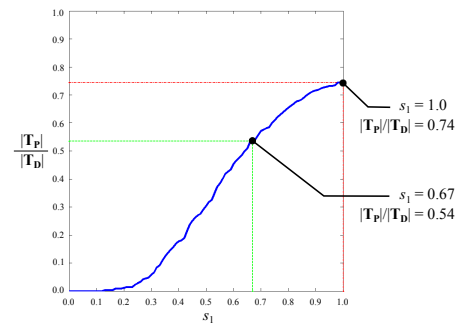


Fig. 5. The ratio of points in the performance task-space \mathbf{T}_P , compared to the desired task-space \mathbf{T}_D , as a function of task-variable s_1 which sets how much strength the human operator is required to contribute towards the task.

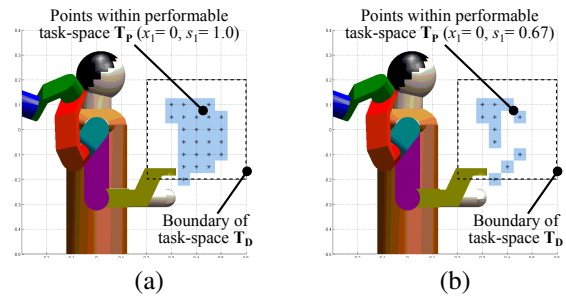


Fig. 6. Visual comparison of the points in the performable task-space with task-variable s_1 at two different values. Results shown are a slice of the cubic workspace with $x_1 = 0$. (a) Points with $s_1 = 1.0$. (b) Points with $s_1 = 0.67$.

be compared to the number in the desired task \mathbf{T}_D . In this case study the desired task-space contained 73,629 points, with 12,039 of these also existing in the performable task-space after the analysis was complete. This comparison alone does not provide much information, however further insight can be gained by comparing the points within certain subspaces, or with respect to different variables to observe their effect on the robot meeting the task's requirements.

To analyze the how much of the task can be kinematically reached it is convenient to analyze only the kinematically dependent first m -dimensions of the task-space. This subspace in \mathbf{T}_D contained 729 unique points. The first stage of the analysis found 560 (76.8%) of these points to be reachable by the human, and 526 (72.2%) reachable by the robot. This result suggests that the human is slightly more capable of reaching through the desired task-space compared to the robot when evaluated individually. When calculating the points reachable by both the human and the robot only 366 (50.2%) are found to be reachable. This significant reduction in task-space coverage can be explained by the offset present between the human and robot shoulders, causing the intersection of their reachable task-space to be significantly smaller than their individual reachable task-spaces. Further analysis taking human-robot collision into account found that the number of reachable positions reduced only slightly to 359 (49.2%). This interesting result indicates that collision between the human and robot is not a predominant factor limiting the regions of the task-space which can be kinematically reached. An explanation of this

is the fact that the robot coworker analyzed was designed to accommodate the human arm, and hence should inherently be suited to avoiding collisions when used as intended.

To analyze how changing the amount of effort the human is expected to contribute affects the extent to which the task can be performed, the number of points in T_P compared to T_D is calculated at set values of task-variable s_1 . The results shown in Fig. 5 indicate that as s_1 is increased (i.e. the strength contribution required by the human is increased) the size of the performable task-space also increases. This is not surprising, since having the human contribute more to the task means that less contribution is required from the robot, and hence more of the task can be performed. This behavior can be analyzed visually by plotting the points in performable task-space. Plotting the results also allows the task-space regions most affected by certain parameters to be observed. Figure 6 shows the performable task-space as a 2D slice ($x_1 = 0$) for clarity, comparing results with $s_1 = 0.67$ and $s_1 = 1.0$.

IV. DISCUSSION

A challenge when analyzing a robotic system working collaboratively with a human to perform a task is that the task, robot and human are independent systems, each operating in their own coordinate spaces. Calculations required for evaluation are relatively simple when performed in a system's native coordinate space, for example calculating the human strength as a function of its own generalized coordinates. To evaluate a robotic coworker based on calculations performed in the separate task, robot, and human coordinate spaces, the capability of transforming between these spaces is required. This is relatively simple in the case when the systems have the same degrees of freedom, and solutions for transforming back and forth between the coordinate systems are available. However due to the large redundancy of the human body it is anticipated that often the human and/or robot will have more DOFs than in the desired task. Typically the task will have equal or fewer DOF than the robot and human, and for this reason the analysis is performed predominantly in the task space.

The methods used to resolve redundant DOFs may significantly impact the result of the analysis, and therefore should be chosen with great consideration. Generally speaking, applying constraints to resolve redundancies will allow quicker calculations, but may skew results towards a smaller performable task-space as valid solutions that do not satisfy the constraints are overlooked. Augmenting the task space will give the best insight into the robot's ability to perform the task as redundant dimensions are fully evaluated; however this is at the cost of increased computational effort. Applying optimization methods will also take longer to compute compared to applying rigid constraints, but can ensure that a valid solution (if one exists) is found along the redundant DOFs. Therefore optimization methods may also tend to give the most optimistic results.

A feature of the presented framework is its flexibility. Not all steps of the framework need be implemented depending on its use. For example an assistance paradigm for the robot to

satisfy need not be defined, or human-robot collision checking ignored if the geometry of the robot is not yet known as it is still in its design stage. Flexibility of the framework also exists by accommodating other variables not presented in the case study. In this work we only ensure that the position between task, robot and human are consistent during the analysis. In some cases, dynamics may also need to be considered. This framework may be extended to include these as additional dimensions in the task space.

V. CONCLUSION

In this paper we presented an analytical framework for evaluating the ability of robotic coworkers to perform specific tasks in collaboration with a human operator. Factors taken into consideration include the reachable workspaces and force contributions of the human and robot, human-robot collisions, and the ability of the robot to satisfy defined assistance paradigms; all within a single flexible framework for the systematic and quantitative evaluation of robotic systems. Application of the framework was demonstrated in a case study, and insights into the factors limiting the robot's ability to perform a desired task were gained.

ACKNOWLEDGMENT

This work was supported by Burwell Technologies Pty Ltd and the Australian Government under a *Researchers in Business* research grant; and the Centre for Autonomous Systems at the University of Technology, Sydney (www.cas.uts.edu.au).

REFERENCES

- [1] A. De Santis, B. Siciliano, A. Deluca, and A. Bicchi, "An atlas of physical human-robot interaction," *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253–270, March 2008.
- [2] F. Zacharias, I. Howard, T. Hulin, and G. Hirzinger, "Workspace comparisons of setup configurations for human-robot interaction," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, Oct 2010, pp. 3117–3122.
- [3] H. Krebs, J. Palazzolo, L. Dipietro, M. Ferraro, J. Krol, K. Ranekleiv, B. Volpe, and N. Hogan, "Rehabilitation robotics: Performance-based progressive robot-assisted therapy," *Autonomous Robots*, vol. 15, pp. 7–20, 2003.
- [4] J. Emken, J. Bobrow, and D. Reinkensmeyer, "Robotic movement training as an optimization problem: designing a controller that assists only as needed," in *9th International Conference on Rehabilitation Robotics (ICORR), 2005*, June 2005, pp. 307 – 312.
- [5] M. Carmichael and D. Liu, "Experimental evaluation of a model-based assistance-as-needed paradigm using an assistive robot," in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, July 2013, pp. 866–869.
- [6] K. R. S. Holzbaur, W. M. Murray, and S. L. Delp, "A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control," *Annals of Biomedical Engineering*, vol. 33, no. 6, pp. 829–840, June 2005.
- [7] M. Carmichael and D. Liu, "Estimating physical assistance need using a musculoskeletal model," *Biomedical Engineering, IEEE Transactions on*, vol. 60, no. 7, pp. 1912–1919, 2013.
- [8] H. Kim, L. Miller, N. Byl, G. Abrams, and J. Rosen, "Redundancy resolution of the human arm and an upper limb exoskeleton," *Biomedical Engineering, IEEE Transactions on*, vol. 59, no. 6, pp. 1770–1779, June 2012.
- [9] M. G. Carmichael and D. Liu, "Admittance control scheme for implementing model-based assistance-as-needed on a robot," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2013, pp. 870–873, 2013.
- [10] D. Tolani, A. Goswami, and N. I. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graph Models*, vol. 62, no. 5, pp. 353–388, Sep 2000.