“© 2014 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.”

Task Oriented Area Partitioning and Allocation for Optimal Operation of Multiple Industrial Robots in Unstructured Environments

Mahdi Hassan, Dikai Liu, Shoudong Huang, Gamini Dissanayake
Centre for Autonomous Systems
University of Technology, Sydney
Australia
{Mahdi.Hassan, Dikai.Liu, Shoudong.Huang, Gamini.Dissanayake}@uts.edu.au

***Abstract*—When multiple industrial robots are applied in applications requiring complete coverage such as grit blasting and spray painting of steel bridges, coordination of multiple robots for maximizing productivity by means of area partitioning and allocation is crucial. This paper presents a novel approach to area partitioning and allocation by utilizing multiobjective optimization and voronoi partitioning. Multiobjective optimization is used to minimize: (1) completion time, (2) proximity of the allocated area relative to the robot, and (3) the torque experienced by each joint of the robot during the execution of a task. The seed points of the voronoi graph for voronoi partitioning are designed to be the design variables of the multiobjective optimization algorithm used. Simulation results of three different scenarios are presented to prove the effectiveness of the proposed approach and the advantage of incorporating torque capacity of the robots.**

***Keywords*—multiple robot arms, task allocation, workspace division, area partition, complete coverage**

I. INTRODUCTION

Effectively partitioning a surface area of a workpiece and appropriately allocating the partitioned sections amongst the industrial robots is necessary to maximize the productivity of the robots for tasks such as grit-blasting and spray painting in field environment. Partitioning and allocation is only carried out on the overlapped areas where the robots share their workspace. That is, surface areas that can be reached by two or more of the deployed industrial robots. The aim is to partition and allocate the overlapped surface areas in a manner that will minimize the completion time of the task and in turn increase productivity. Obviously, individual robot capabilities such as speed and the properties of the tool attached to the end-effector must be considered. It is also preferred to carry out partitioning and allocation in a manner that will improve the torque condition of the joints during the execution process.

Researchers have developed numerous algorithms that enable collaboration of multiple industrial robots to carry out a task. Collaboration between the robots should incorporate aspects such as collision avoidance, coordinated motion and path planning. The use of bounding volumes such as bounding boxes [1], spheres [2], and ellipsoids [3] can be used to represent the industrial robots for collision checking process utilized in collision-free motion planning methods [4]. Coverage path planning (CPP) ensures complete coverage by creating a path that passes through all points representing an area or volume. Readers are advised to refer to [5] for a comprehensive survey on the recent CPP methods.

Effective area partitioning and allocation can play an important role in maximizing the effectiveness of the aforementioned collaboration tasks. Numerous research work related to partitioning a given area into sections is available. Many researchers have studied convex decomposition to break down a complex area into number of sections. Lien and Amato [6] proposed a method for decomposing a polygon which contains holes. Some researchers have used polygon decomposition in multiple-robot workspace division. For example, a cooperative search method by multiple UAVs in an unknown environment which incorporates convex polygon decomposition is presented in [7]. Hert and Lumelsky [8] study the terrain-covering problem by multiple robots. In their paper, terrain-covering capabilities of the robots are considered when dividing the area into *n* polygonal sections. Each polygonal section is visited once by one of the *n* robots available. Another method for partitioning an area is grid graph bisection method which iteratively divides the solid graph into sections [9]. A partitioning method called Virtual Door algorithm is presented in [10], which considers partitioning a large area into numerous small rectangular shaped sections. This Virtual Door algorithm is shown to achieve time efficient complete coverage by multiple cleaning robots. These methods partition the areas of interest mainly to carry out the task in a time efficient manner by mobile robots.

In some applications, area partitioning and allocation method should take additional requirements and constraints of the industrial robots into consideration. These added requirements might not be present when multiple mobile robots are carrying out complete coverage tasks such as floor cleaning. One requirement is the torque needed by each joint of the industrial robot which depends on the base location of the robot and the pose generated to achieve acceptable orientation and position of the end-effector. Another requirement is to have the allocated sections from the overlapped surface areas to be close to their corresponding robots to improve maneuverability and reduce the potential collision between robots during execution of a task. Minimal completion time of the task is also a vital requirement which necessitates considering robot capacities such as speed and properties of the end-effector tool.

How to effectively partition and allocate overlapped areas of a surface which can be covered by multiple robots while satisfying all the aforementioned requirements and constraints? This question is the main focus of this paper. The problem is mainly applicable to unstructured environments for one-off tasks. The problem can be modeled as a single objective or multiobjective optimization. Multiobjective optimization approach is beneficial since multiple objectives can be minimized and the Pareto solutions can be found. The goal is therefore to partition the area in a manner that will result in minimal cost or fitness value.

The approach presented in this paper makes use of voronoi partitioning to effectively partition areas of a planar surface. Voronoi partitioning has been used in path planning [11, 12] and exploration of unknown environment by multiple coordinating robots [13]. The proposed approach in this paper makes use of voronoi partitioning to create voronoi cells on the surface area of the workpiece. The cells need to be created on certain surface areas which two or more of the robots can reach with acceptable orientation and position of the end-effector. For *n* number of robots, *n* numbers of voronoi cells are created. The size and the shape of each cell is iteratively modified by making the seed points of the voronoi graph to be the design variables in the multiobjective optimization. The aim is to create cells which will result in minimal cost or fitness values.

This paper is structured as follows. First, the problem definition which consists of detailed explanation of the problem is given in Section II. Next, the mathematical model is outlined in Section III. Simulation results of three different scenarios starting from a simple scenario to a more complicated scenario where torque and obstacle is taken into account are presented in Section IV. Finally, conclusions and future work are discussed in Section V.

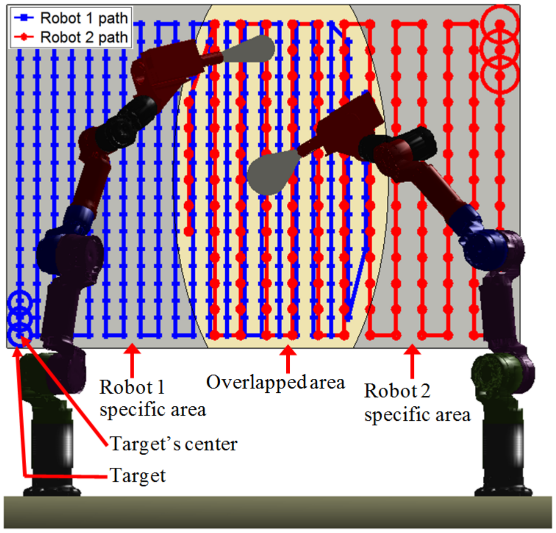


Figure 1. Example of an application scenario

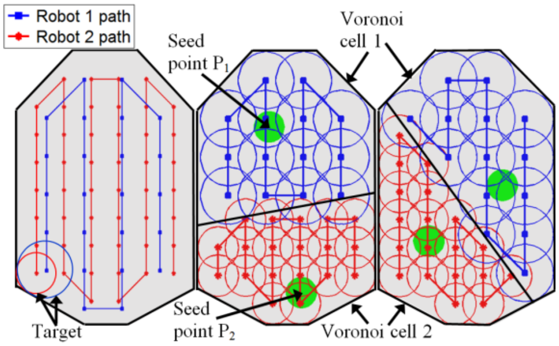
II. PROBLEM DEFINITION

Fig. 1 shows an example application scenario where a flat rectangular surface will be grit-blasted or cleaned by two robots. Two sections of the surface are highlighted in Fig 1. The first section consists of the *specific areas* which only one of the robots' end-effector can reach with a valid orientation and position. The second section is the *overlapped area* that can be covered by the two robots. Obviously, partitioning and allocation approach is concerned with the overlapped areas.

Targets are used to represent the surface. Each robot is assigned a set of targets that represent all the overlapped and specific areas of the surface which can be covered by that particular robot. Hence, for *n* robots, *n* sets of targets are available. Fig. 1 shows the targets and their centers associated with each of the two robots. The diameter of each set of targets is determined by considering the properties of the tool attached to the end-effector of the robot such as the nozzle size. Density of the targets depends largely on the application. The work in [14] explains how the geometric data obtained from sensors during the exploration process can be used to create a triangle mesh of the surface and in turn Scale-Like Disc (SLD), which is called target in this paper. As shown in Fig. 1, coverage path planning must ensure that the generated paths pass through all targets. At the completion of the partitioning and allocation task, each robot is assigned a subset of targets from the original sets of targets which represent the overlapped areas.

The following should be taken into account while partitioning and allocating the overlapped areas: (1) completion time of the task should be minimized to enhance productivity, (2) allocated areas should be close to their corresponding robots to reduce the potential collision between robots and improve maneuverability of the robots during the execution of the task, and (3) the sum of torques experienced by the robots during the execution of the task should be minimized to improve the lifespan of the robots.

The following are assumed to be the input to the model: (1) targets associated with each of the robots which represent the overlapped areas and the specific areas, (2) fixed base location of each robot, (3) robot poses generated by a planner for each target in the overlapped areas, and (4) capacities of the robots such as the end-effector speed and the properties of the tool attached to the end-effector which play a role in determining the diameter of the target.



(a) (b) (c)

Figure 2. a) Overlapped area of two robots; b) Voronoi partitioning example 1; c) Voronoi partitioning example 2

III. MATHEMATICAL MODEL

A. *Design Variables*

The design variables are designed to be the position coordinates of the seed points corresponding to the voronoi cells. Fig. 2 shows how an overlapped area can be partitioned in different ways using voronoi partitioning. Voronoi partitioning divides an area into *n* number of sections based on the location of the *n* seed points. The generated sections are called voronoi cells. The targets in each cell are closest to the seed point of that particular cell. Hence, distance query between the seed points and each target in the overlapped areas determines the cell for which a target belongs to. Targets which fall on the boundary of two adjacent cells have equal distance to the seed points of the two cells. It is clear in Fig. 2 that the size of the cells and the targets they consist of can be altered by changing the location of the seed points. Hence, in the optimization, the seed points are designed to be the design variables to iteratively partition the overlapped areas and ultimately create cells that result in lowest cost or fitness values. In this paper, *Allocated cell* refers to the section of the overlapped areas which is allocated to a particular robot.

For *n* robots, 2 × *n* design variables specify the position coordinates of the seed points on a planar surface. That is, for *n* robots, the design variables are

$$X = (P_1, P_2, \dots, P_n), \tag{1}$$

where seed points $P_i = (x_i\,, y_i)$ for *i* = 1, 2, … , *n*, and $(x_i\,, y_i)$ is the position coordinates of a seed point P_i.

B. *Design Objectives*

1) *Objective 1*

The first objective is to achieve minimal completion time and therefore enhance productivity. Efficiency in terms of time is optimal when all robots simultaneously finish their allocated portions of the surface. Simultaneous finish means that all the robots have been active throughout the execution process and hence minimal completion time is obtained. Obviously it is not always possible to have the robots finish at the same time; however the difference between the completion times of the robots can be minimized. The objective can be to minimize the variance, σ^2 of the completion time of the robots. Therefore, for *n* robots, the first objective is to minimize

$$f_1 = \frac{1}{n}\sum_{i=1}^{n}\left(t_i - \bar{t}\,\right)^2, \tag{2}$$

where i is the robot index, t_i is the completion time of robot i, and $\bar{t}$ is the mean completion time of all the robots. The completion time, t_i can be calculated as

$$t_i = \left(l_i^{sh} + l_i^{sp}\right) v_i^{-1}, \tag{3}$$

where v_i is the end-effector speed of robot i, l_i^{sh} is the path length of robot i to cover the allocated part of the overlapped area, and l_i^{sp} is the path length of robot i to cover the specific area that only robot i can cover. In applications such as grit-blasting and spray painting, the end-effector needs to move at a constant speed relative to the surface. Path lengths l_i^{sh} and l_i^{sp} can be calculated as

$$l_i^{sh} = d_i \times \left(S_i^{sh} - 1\right), \tag{4}$$

and

$$l_i^{sp} = d_i \times \left(S_i^{sp} - 1\right), \tag{5}$$

respectively, where d_i is the Euclidian distance between the centers of two adjacent target points along the path, S_i^{sh} is the number of targets in the allocated cell, and S_i^{sp} is the number of targets in the specific area associated with robots i.

2) *Objective 2*

Allocated cells should be close to their corresponding robots so that the possibility of collision between the robots or negative effect on the maneuverability of the robots during the execution of the task is minimized. Inappropriate allocation might also result in undesirable large change in joint angles due to aspects such as restricted reachability and avoiding collision between the robots during the motion.

The yellow filled circles in Fig. 3 represent the center of mass of the specific areas. Let $C_i \in \mathbb{R}^2$ be the center of mass position of the specific area associated with robot i. Thus, the objective is to minimize the sum of distances between C_i and each target, $T_{ij} \in \mathbb{R}^2$ in the allocated cell corresponding to robot i. Fixed robot base location can be considered as C_i in situation where specific areas don't exist. Therefore, for *n* robots, the second objective is to minimize

$$f_2 = \sum_{i=1}^{n}\sum_{j=1}^{S_i^{sh}}\left\|C_i - T_{ij}\right\|. \tag{6}$$

3) *Objective 3*

The sum of torques experienced by the joint of each robot during the execution of a task should be minimized. Robot's joints might be able to provide the necessary torque to operate and complete a task, however allocating areas of the surface that will reduce the amount of torque experienced by the robot will help maintain the good condition of the joints. Therefore, the third objective is designed to minimize the sum of maximum torque ratios during the execution of a task. Torque ratio corresponding to each joint is calculated based on the torque experienced by the joint divided by its torque capacity. Hence, for each robot pose $Q_j = [\theta_1, \theta_2, \dots, \theta_b]$ corresponding to a target j, maximum torque ratio is the largest value from all the torque ratios of the joints (actuators) of the robot. The third objective is to minimize

$$f_3 = \sum_{i=1}^{n}\sum_{j=1}^{S_i^{sh}} \max_k \left|\frac{T_{ik}(Q_j, F_i)}{T_{ik}^{cap}}\right|, \tag{7}$$

where T_{ik} is the torque experienced by joint k (k = 1, 2, … , b) of robot i due to end-effector force, F_i and robot pose Q_j. b is the total number of joints of robot i, θ_1 to θ_b are the joint angles, and T_{ik}^{cap} is the torque capacity of joint k of robot i. The torque experienced by each joint can be mainly due to the end-effector force and the weight of the joints and links if the motion of the robots is slow which is the case in grit-blasting and spray painting applications. Poses generated by a planner which are part of the input to the model are assumed to have considered all the relevant constraints such as joint angle limits.

IV. SIMULATION AND RESULTS

The Schunk industrial robots simulated in Fig. 1 that carry out grit-blasting are used for the simulation study in this paper. As shown, the end-effector of the robots is equipped with a nozzle. The blasting stream coming out of the nozzle aims at and follows the path. The blasting spot of the blasting stream is required to hit each of the targets within a small allowable deviation. For each target, there exists a collision-free robot pose that can correctly position and orientate the nozzle to aim at that particular target.

The work presented in this paper uses NSGA-II algorithm to carry out the optimization. NSGA-II is a multiobjective optimization algorithm. Readers are advised to refer to [15] for detailed explanation of this algorithm. Multiobjective optimization allows multiple objectives to be minimized and as a result a set of Pareto solutions can be obtained. These Pareto solutions are equally optimal and the selection of the most suitable solution from the Pareto-optimal solutions is mainly dependent on the application. For example, minimal completion time (objective 1) in grit-blasting application is most important. Hence, a small subset of solutions which have better results in terms of completion time is first chosen from the Pareto-optimal solutions. Then, from the subset of Pareto-optimal solutions, a final solution based on minimal torque (objective 3) is selected. Although objective 2 is considered in the optimization; it is not part of the selection strategy of the final solution.

Three test scenarios are demonstrated in the following subsections, starting from a simple scenario to a more complicated scenario where obstacle is introduced in the environment and the effect of torque is investigated. The designed simulations are carried out with end-effector speed of 0.1 meters per second for all robots. Notations used in the legend of the figures are, Ri (Robot i), t (completion time in seconds), L (total length of the path, in meters), and d (Euclidean distance between two adjacent targets along a path, in meters). The completion time, t is calculated based on (3), and the total length of the path, L is from (4) and (5).

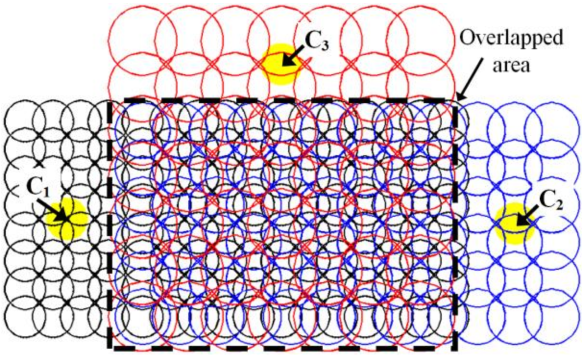


Figure 3. Three robots having common overlapped areas

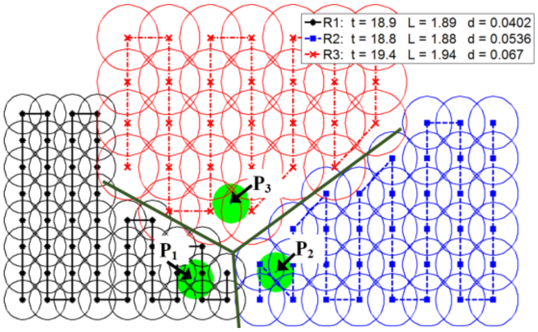


Figure 4. Result of simulation 1

A. Simulaion 1 – three robots with common overlapped areas

Assume that the specific and overlapped areas of a surface associated with three robots are as shown in Fig. 3. The first simulation is mainly to demonstrate how the proposed area partitioning and allocation approach can result in minimal completion time and appropriate allocation. Thus, the third objective (torque minimization) is discarded in this scenario. Robot 1 is equipped with the smallest nozzle size and robot 3 with the largest nozzle size and hence, the diameter of the targets associated with each robot is different.

It is clear in Fig. 4 that each allocated cell is closest to its corresponding specific area which is expected due to objective 2. The length of the paths for the three robots is approximately the same, however robot 3 is covering a greater portion of the surface since larger nozzle is used by this robot. Completion times are 18.9 (s), 18.8 (s) and 19.4 (s) for robots 1 to 3, respectively. It is clear that the difference in completion times are insignificant as would be expected.

B. Simulation 2 - four robots with different overlapped areas

This scenario is designed to demonstrate the condition at which voronoi partitioning is faced with a problem. To help with demonstrating the problem, only objective 1 (minimal completion time) is used, and assumptions are made in terms of coverage area for each robot and the size of the targets. Overlapped areas of the robots can be different as shown in Fig. 5. In this condition, voronoi partitioning might face a problem of incomplete coverage as shown in Fig. 6. Incomplete coverage is due to the fact that some sections of the allocated cells can't be covered (unreachable) by their corresponding robots. Although the result is appropriate in terms of the first objective which is minimal difference in completion time of the robots; clearly this result is unsuitable for the application.

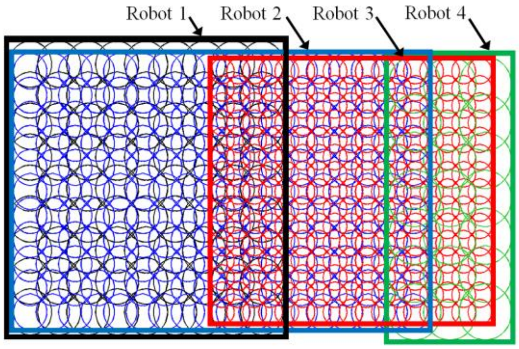


Figure 5. Four robots with different overlapped areas

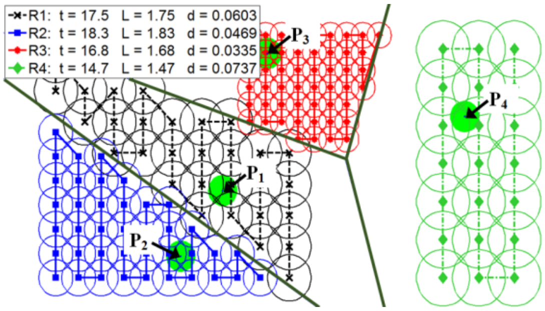


Figure 6. Unsuitable result due to unreachable sections in the allocated cells

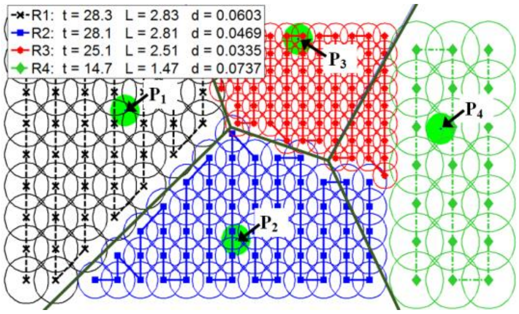


Figure 7. Acceptable result by further allocation of missed out sections

This problem can be fixed by further re-allocating the missed out sections to other robots that can reach those sections. That is, each target in the missed out sections is appropriately allocated to the cell with a seed point closest to that particular target. In summary, after performing voronoi partitioning, missed out sections are appropriately allocated to other cells and hence the size of the cells are modified prior to evaluating the cost or fitness values. This solution is found to be effective as shown in Fig. 7. Completion times are 28.3 (s), 28.1 (s), 25.1 (s) and 14.7 (s) for robots 1 to 4, respectively. Since robot 4 can reach much less area of the surface, it is preferred that it covers all of its reachable area and in turn minimize the difference in completion times of the robots. As expected, simulation result shows that robot 4 is allocated all of its reachable area.

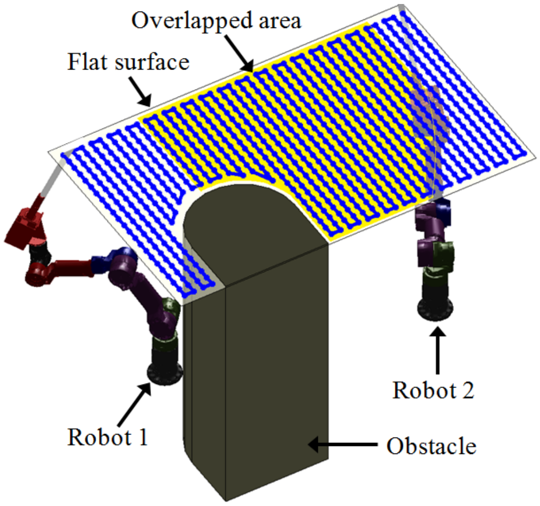


Figure 8. Two robots with an obstacle in the environment

C. *Simulation 3 – two robots with obstacle in the environment*

The purpose of this simulation is to demonstrate how area partitioning and allocation can help with minimizing the sum of torques required by the joints of the robot during the execution of a task. All three objectives are used in this scenario. Fig. 8 shows a flat surface which is 2 meters in length and 1.2 meters in width. It is positioned 1.5 meters above the base of the robots. Both robots are equipped with the same size nozzle and hence, the diameter of the targets is the same.

The result of the simulation is shown in Fig. 9. The completion time for robot 1 and robot 2 is 203 (s) and 201 (s), respectively. It is clear that the difference in completion time of the two robots is insignificant. For comparison, the simulation was repeated without objective 3. That is, Torque was not considered when generating the result shown in Fig. 10. The completion time of both results is the same; however the first simulation result shown in Fig. 9 will reduce the torque required by the joints of the robots during the execution of the task.

Fig. 11 and Fig. 12 illustrate the torque heatmaps of the results shown in Fig. 9 and Fig. 10, respectively. The torque due to robots' poses is used to generate the torque heatmaps. For each robot pose corresponding to a target, only the maximum torque ratio which is the largest value from all the torque ratios of the joints (actuators) of the robot is taken into account when producing the heatmaps. Darker areas in Fig. 11 and Fig. 12 illustrate large torque regions. It is clear that Fig. 11 shows better results in terms of torque since the selection of large torque regions from the overlapped areas is significantly reduced for robot 1. Robot 2 will experience minor increase in torque by this allocation, however this increase is insignificant.

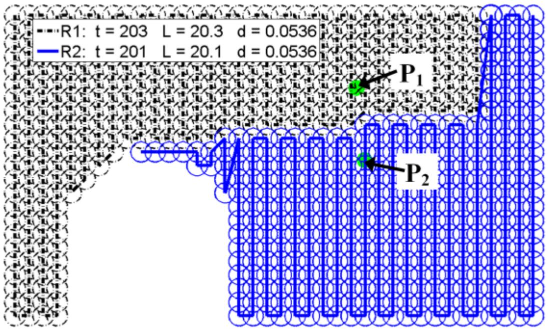


Figure 9. First simulation result with all objectives considered

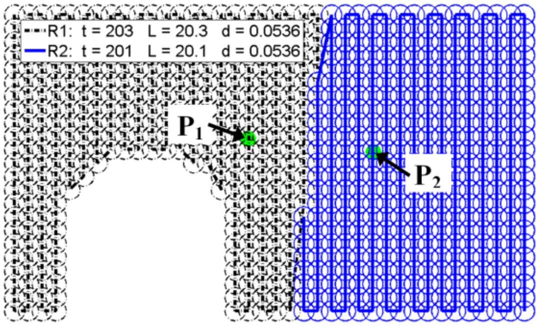


Figure 10. Second simulation result without considering objective 3

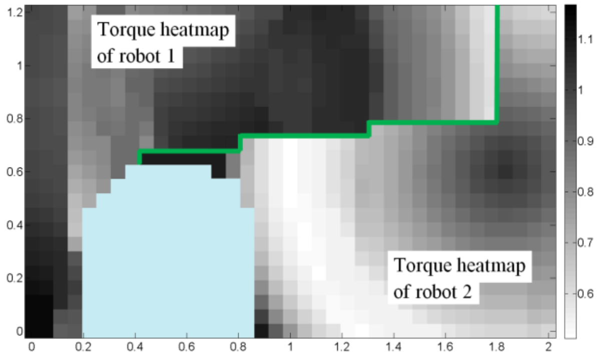


Figure 11. Torque heatmap of the first simulation

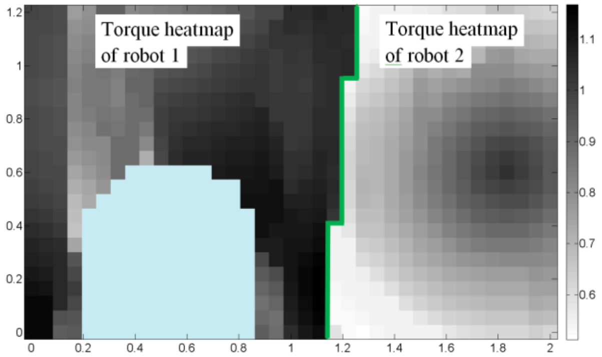


Figure 12. Torque heatmap of the second simulation

In field environment, occurrence of large torque regions can be due to various conditions. When a robot is working in close proximity to an obstacle such as the one shown in Fig. 8, large torques could be experienced by the joint of the robot. The cause of the large torque is mainly due to the configuration of the joints of the robot to generate a pose that will avoid collision with the obstacle while simultaneously achieving acceptable end-effector position and orientation. Another cause for large torque regions is the poses generated by the robot to correctly position and orientate the end-effector considering the limited space between the base of the robots and the surface to be blasted. Large torque regions might also be present at the boundaries of the robot's workspace. Objective 3 will help with reducing the allocation of these large torque regions to each of the robots.

V. CONCLUSION AND FUTURE WORK

An approach for effectively partitioning an area of a workpiece and appropriately allocating each of the partitioned sections amongst industrial robots was proposed in this paper. Importance of partitioning based on factors such as completion time, torque experienced by robot's joints, and proximity of the allocated section relative to the robot was demonstrated. These factors were shown to maximize productivity and improve robot's operating condition. A multiobjective optimization algorithm (NSGA-II) was used to carry out the optimization and minimize the cost values associated with the aforementioned factors. Voronoi partitioning was implemented by creating voronoi cells in the overlapped areas. The design variables are designed to be the position coordinates of the seed points corresponding to the voronoi cells.

Future work includes area partitioning and allocation of three dimensional and nonplanar workpiece. The effectiveness of the presented approach when incorporating manipulability measure of the industrial robots will need to be examined. Experiments are also required to examine the reliability of the approach on real life scenarios.

ACKNOWLEDGMENT

This work is supported by SABRE Autonomous Solutions Pty Ltd, the Centre for Autonomous Systems (CAS) and the University of Technology Sydney.

REFERENCES

[1] U. Assarsson and T. Moller, "Optimized view frustum culling algorithms for bounding boxes," Journal of Graphics Tools, vol. 5, pp. 9-22, 2000.

[2] J. Xu, D. K. Liu, and G. Fang, "An efficient method for collision detection and distance queries in a robotic bridge maintenance system," in Robotic Welding, Intelligence and Automation. vol. 362, T.-J. Tarn, S.-B. Chen, and C. Zhou, Eds. Springer Berlin Heidelberg, 2007, pp. 71-82.

[3] T. Larsson, "An efficient Ellipsoid-OBB intersection test," Journal of Graphics Tools, vol. 13, pp. 31-43, 2008.

[4] P. Chotiprayanakul, D. Liu, D. Wang, and G. Dissanayake, "A 3-dimensional force field method for robot collision avoidance in complex environments," 24th International Symposium on Automation and Robotics in Construction, pp. 139-145, 2007.

[5] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," Robotics and Autonomous Systems, vol. 61, pp. 1258-1276, 2013.

[6] J.-M. Lien and N. M. Amato, "Approximate convex decomposition of polygons," 20th Annual Symposium on Computational Geometry, USA, pp. 17-26, 2004.

[7] I. Maza and A. Ollero, "Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms," in Distributed Autonomous Robotic Systems 6, Eds. Springer, 2007, pp. 221-230.

[8] S. Hert and V. Lumelsky, "Polygon area decomposition for multiple-robot workspace division," International Journal of Computational Geometry & Applications, vol. 8, pp. 437-466, 1998.

[9] C. H. Papadimitriou and M. Sideri, "The bisection width of grid graphs," 1st Annual ACM-SIAM Symposium on Discrete Algorithms, USA, pp. 97-110, 1990.

[10] J. H. Lee, J. S. Choi, B. H. Lee, and K. W. Lee, "Complete coverage path planning for cleaning task using multiple robots," Systems, Man and Cybernetics, USA, pp. 3618-3622, 2009.

[11] S. Garrido, L. Moreno, M. Abderrahim, and F. Martin, "Path planning for mobile robot navigation using voronoi diagram and fast marching," Intelligent Robots and Systems, China, pp. 2376-2381, 2006.

[12] A. Breitenmoser, M. Schwager, J.-C. Metzger, R. Siegwart, and D. Rus, "Voronoi coverage of non-convex environments with a group of networked robots," International Conference on Robotics and Automation, Alaska, pp. 4982-4989, 2010.

[13] L. Wu, M. Á. García García, D. Puig Valls, and A. Solé Ribalta, "Voronoi-based space partitioning for coordinated multi-robot exploration," Journal of Physical Agents, vol. 1, pp. 37–44, September 2007.

[14] G. Paul, N. Kwok, and D. Liu, "A novel surface segmentation approach for robotic manipulator-based maintenance operation planning," Automation in Construction, vol. 29, pp. 136-147, 2013.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," Evolutionary Computation, IEEE Transactions on, vol. 6, pp. 182-197, 2002.