Modeling and Stochastic Optimization of Complete Coverage under Uncertainties in Multi-Robot Base Placements

Mahdi Hassan, Dikai Liu and Gavin Paul

Abstract— Uncertainties in base placements of mobile, autonomous industrial robots can cause incomplete coverage in tasks such as grit-blasting and spray painting. Sensing and localization errors can cause such uncertainties in robot base placements. This paper addresses the problem of collaborative complete coverage under uncertainties through appropriate base placements of multiple mobile and autonomous industrial robots while aiming to optimize the performance of the robot team. A mathematical model for complete coverage under uncertainties is proposed and then solved using a stochastic multi-objective optimization algorithm. The approach aims to concurrently find an optimal number and sequence of base placements for each robot such that the robot team's objectives are optimized whilst uncertainties are accounted for. Several case studies based on a real-world application using a real-world object and a complex simulated object are provided to demonstrate the effectiveness of the approach for different conditions and scenarios, e.g. various levels of uncertainties, different numbers of robots, and robots with different capabilities.

I. INTRODUCTION

Uncertainties, if not accounted for appropriately, can result in poor coverage and low efficiency when multiple mobile, Autonomous Industrial Robots (AIRs) are deployed to perform tasks where the robots must operate on all surface areas of a target object [1], e.g. in the grit-blasting and the spray painting applications. The mobility of an AIR introduces uncertainties due to sensing and localization errors, which in turn invalidates assumptions about base position accuracy. The magnitude of the errors depends on the quality of the sensors and the localization algorithm implemented. Nonetheless, the AIRs require the capacity to account for uncertainties to enable them to operate in complex and unstructured environments. The definition of AIR is the same as Autonomous Industrial Mobile Manipulator (AIMM) [2].

A challenge in the deployment of multiple AIRs for complete coverage tasks is to facilitate collaboration amongst the AIRs. Specifically, the problem addressed in this paper is to enable the AIRs to collaborate in a manner that will lead to complete coverage of the target objects while accounting for uncertainties in base placements of the AIRs.

The problem of automatic base placement is for each AIR to determine the number and visiting sequence of a set of base placements from which it will be operating on the object. In doing so, the robot team's objectives need to be optimized while considering the relevant uncertainties. The AIR team's objectives are a set of shared objective functions that the AIRs collectively aim to optimize and are task specific, e.g. for complete coverage tasks, maximizing coverage and minimizing makespan are vital team objectives.

For certain environments or applications, it may be possible to reduce uncertainties to a negligible level, e.g. by utilizing high quality sensors, and as such, negate the need to consider uncertainties in the mathematical model. However, this paper considers applications where the cost of utilizing high quality sensors is prohibitive or achieving precise localization is impractical due to the nature of the environment.

The research presented in this paper is an extension of the approach in [3], which determined a single base placement for each AIR assuming no uncertainties in base placements. However, this paper proposes a new mathematical model that considers large, complex objects where multiple base placements for each AIR are needed. It accounts for uncertainties in AIRs' base placements through Monte Carlo Simulations.

The problem of finding an optimal base placement for a single robot or a robotic manipulator has been addressed by researchers for a number of environments and applications, such as manufacturing environments [4], underwater environments [5], and applications where accurate operation of the robot is necessary [6]. However, there is a growing interest for multiple AIRs to operate in complex and unstructured environments, e.g. as the shift from mass production to custom manufacturing takes place [2]. Thus, the problem becomes increasingly complicated when AIRs need to each determine several base placements under uncertain conditions.

Literature exists that addresses the problem of complete coverage under uncertainty. For example, in the work by Bretl and Hutchinson [7] the uncertainty in mobile robot's position and velocity is taken into account and an approach for guaranteed coverage is proposed. However, due to over-coverage a longer path is generated. Galceran and Carreras [8], account for uncertainty in the position of an Autonomous Underwater Vehicle (AUV) that is surveying/inspecting the seabed using a two stage approach: (1) off-line planning where the complete coverage path of the AUV is calculated, and (2) periodic real-time re-planning based on the sensed environment. Unlike robots that are continuously in motion (such as AUVs), for the problem under consideration the base of the mobile AIRs is stationary during the task execution. This provides the industrial robot, affixed with a high-powered end-effector tool, the stability required to robustly perform a precise, complete coverage task (e.g. in grit-blasting and spray painting). Hence, in this work, the considered uncertainties are related to the base placements of the mobile AIRs.

Authors are from the Centre for Autonomous Systems (CAS) at the University of Technology Sydney, 15 Broadway, Ultimo NSW 2007, Australia
Mahdi.Hassan@student.uts.edu.au

In brief, the overall approach presented in this paper considers the following: (1) appropriate discretization of the search space, (2) a mathematical model that aims at achieving complete coverage while optimizing the AIR team's objectives under uncertainties in base placements, and (3) solving the mathematical model using a multi-objective stochastic optimization algorithm. The main contribution of the paper is the proposed mathematical model. The approach is tested by conducting both simulated and real-world grit-blasting experiments using multiple AIRs and complex objects.

The rest of the paper is organized as follows. Section II defines the problem being addressed. Section III provides the methodology to the proposed approach. The details of the experiments and results are presented in Section IV. Section V provides discussions on results and limitations, and concluding remarks are stated in Section VI.

II. PROBLEM DEFINITION

An example of a large and complex object is shown in Fig. 1. Assume that the complete coverage task of spray painting needs to be carried out on the object using the two AIRs shown. That is, the two AIRs, which are equipped with spray painting nozzles, are required to collectively operate on all accessible surfaces of the objects by following the paths created on the surfaces. In order to achieve complete coverage, it is clear that each AIR is required to operate from multiple base placements. It is assumed that the base of the AIRs is fixed during the spray painting operation so as to achieve stability for precise and uniform coverage.

In order for the AIRs to operate autonomously, they need the capacity to self-monitor, and be aware of the surrounding environments including other AIRs. Each AIR starts by performing exploration and mapping of its local environment [9]. Then, the AIRs will communicate and share the information they have collected so as to create a map of the environment. The AIRs are also expected to have shared information on their capacity, initial position, etc. In this work, it is assumed that the map is given and all AIRs have access to the shared relevant information. This information is then used by each AIR to find the base placements and the visiting sequence such that the AIR team collectively achieves complete coverage of the object. The final stage of AIRs' operation is to execute the task (e.g. spray painting in Fig. 1) by considering a collision-free motion planning [10].

The problem of finding multiple base placements for each AIR to perform the task of complete coverage involves

Fig. 1: Two mobile AIRs spray painting a large object.

the AIRs collaboratively deciding on the following: (1) the base placement of each AIR at each time step of the task execution, and (2) the surface areas of the target objects that will be covered by each AIR. When AIRs are deciding upon these aspects, the uncertainties associated with the AIRs' base placement, and the following team objectives needs to be considered:

- Coverage of the objects must be complete or acceptable (above a threshold).
- Makespan must be minimal.
- Collisions must be avoided by maintaining safe distances from the environment and also between the AIRs.

The uncertainties in base placements of the AIRs can be due to various errors. Examples of the sources of uncertainties are: (1) errors in the collected sensor data (e.g. due to measurement inaccuracies and imprecision), and (2) localization error (e.g. due to the limitations of the algorithm used with respect to the available features from the environment).

The problem can thus be concisely stated as follows: given probability distributions that represent the uncertainties in the AIRs' base placements, how would the AIRs select an optimal number and sequence of base placements such that complete coverage is achieved and the team's objectives are optimized.

III. METHODOLOGY

The proposed mathematical model is presented in the context of a methodology for solving multi-AIR complete coverage. An overview is first presented, followed by a detailed explanation of the mathematical model, and finally an approach to solving the model.

A. Overview of the Approach

An approach to Stochastic Optimization of Multiple Base Placements (Stochastic-OMBP) for each AIR, given the map of the environment, is proposed in this paper (shown in Algorithm 1), with a particular focus on the mathematical modeling. The Stochastic-OMBP first considers discretizing the search space (line 1) to reduce the computation time and complexity, which is acceptable in many industrial applications if a near-optimal solution is achieved. An example of the discretization is shown in Fig. 2. It can be seen that the density of the discrete base placements for each AIR is different, meaning that the AIRs' capacity (e.g. workspace size) is different, as the density is made to correspond to the capacity of the AIR.

The next step of the Stochastic-OMBP is to select a subset of discrete base placements for each AIR (line 2), called the Favored Base Placements (FBPs). FBPs are predicted to be superior in terms of both coverage and collision avoidance. A strategy for determining the FBPs is explained in [3].

Once the FBPs are determined, the proposed mathematical model is solved using a multi-objective stochastic optimization (line 3), which is based on modifications of existing hybrid multi-objective algorithms. The hybrid algorithm uses Genetic Algorithm (GA) and Simulated Annealing (SA) algorithm and will be elaborated upon in Section III-C. Since
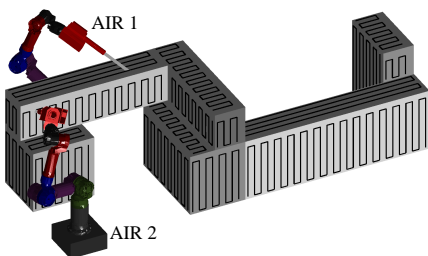
Algorithm 1 Stochastic-OMBP.

1: For each AIR, create η_i discrete base placements around the target objects where i is the AIR index
2: Find favored base placements (FBPs)
3: Perform multi-objective optimization using hybrid GA-SA
4: Select the appropriate/best solution, Z^* from the Pareto front
5: **if** a threshold is not met **then**
6:     Perform further discretization around desirable FBPs
7:     **goto** line 3
8: **end if**
9: **return** solution Z^* for execution

multi-objective optimization is considered, then the output of the optimization is the Pareto front (a set of Pareto-optimal solutions) from which a final solution is selected [3] (line 4). After running the optimization, if a threshold (e.g. minimum coverage) is not reached (line 5), then finer discretization can be iteratively generated (line 6 and 7).

B. Mathematical Modeling

1) Design Variables: Let $B_i = \{b_{i1}, b_{i2}, \ldots, b_{i\eta_i}\}$ be a set of discrete base placements, and $(B_{FBP})_i = \{\beta_{i1}, \beta_{i2}, \ldots, \beta_{i(n_F)_i}\} \subseteq B_i$ be the FBPs for the ith AIR where $i = 1, 2, \ldots, n$. In the case when the rotation of the base about the z-axis effects the coverage performance of the AIR, then each FBP can also be discretized into a number of rotations, and the rotation, θ_z that results in the best coverage can be selected. Thus, each FBP can be defined as the x, y, z position and the θ_z rotation. The design variables are $Z_{ij} \in \{0, 1, \ldots, (n_F)_i\}$, $j = 1, 2, \ldots, (n_b)_i$, $(n_b)_i \leq (n_F)_i$, meaning that the jth base placement of the ith AIR is one of the $(n_F)_i$ FBPs where $(n_b)_i$ is an approximation of the number of FBPs to be visited by the ith AIR. Approximation of $(n_b)_i$ can be made by considering the size of the object and its surface areas. The greater the uncertainty in approximating $(n_b)_i$, the larger its value. $Z_{ij} = 0$ means that the ith AIR will skip to the next, $(j+1)$th base placement decided by Z_{ij+1}.

2) Design Objectives:

a) Design Objective 1 (Maximal Coverage): For each AIR a set of *targets*, O_i are uniformly generated on all surfaces of the objects (e.g. the blue disks on the surfaces of the object shown in Fig. 2), with spacing and radius size that is chosen based on the capacity of the corresponding AIR [3]. At the jth base placement of the ith AIR, a subset of
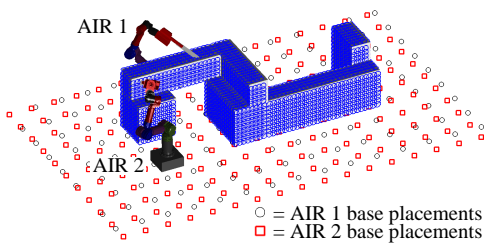

Fig. 2: Discrete base placements of two different AIRs, and target representations of the object.

targets, $O_{ij} = \{o_{ij1}, o_{ij2}, \ldots, o_{ijn_o}\} \subseteq O_i$ are located within the workspace of the AIR. Let $\acute{O}_{ij} \subseteq O_{ij}$ be the *reachable* targets by the AIR at the jth base placement, and let $\tilde{O}_{ij} \subseteq \acute{O}_{ij}$ be the *uncovered* reachable targets (targets not yet covered by any AIR). Therefore, a path for the AIR to cover $\tilde{O}_{ij}$ can be generated [1] by appropriately connecting the center points of the targets. In order for a target, $o_{ijr} \in O_{ij}$ to be counted as reachable by the ith AIR, an AIR pose q_{ijr} needs to be found that reaches the target o_{ijr} with an appropriate end-effector position and orientation.

The aim of this objective is to minimize missed-coverage (areas not covered by any AIR). Due to the stochastic nature of the problem, the objective is formulated to minimize the expected value, i.e. :

$$\min_Z f_1(Z) = E[f_c(Z, \Xi_z)] \quad (1)$$

where Z is a set containing all the design variables, and each element in the set Ξ_z, i.e. $(\Xi_z)_{ij} \in \Xi_z$, is dependent on a design variable Z_{ij} and represents the uncertainties in the FBP, $\beta_{i(Z_{ij})}$. Hence, $(\Xi_z)_{ij} \in \Xi_z$ is designed as a random vector with a multivariate normal distribution,

$$(\Xi_z)_{ij} \sim \mathcal{N}(\beta_{i(Z_{ij})}, \Sigma_i), \quad (2)$$

describing the x, y, z position errors and the θ_z orientation error associated with $\beta_{i(Z_{ij})}$. The orientation error can be discarded for an AIR that has a full $360°$ revolute joint at its base. Similarly, the position error along the z-axis can be discarded for an AIR that can't move its base vertically. The function to calculate missed-coverage is expressed as:

$$f_c(Z, \Xi_z) = 1 - \sum_{i=1}^{n} \frac{\sum_{j=1}^{(n_b)_i} S^f(Z_{ij}, \xi_{ij})}{S_i^T} \quad (3)$$

where S_i^T is the number of targets in O_i, and $S^f(Z_{ij}, \xi_{ij})$ calculates the number of uncovered reachable targets $\tilde{O}_{ij}$ based on the design variable Z_{ij} and an independent observation ξ_{ij} of $(\Xi_z)_{ij} \in \Xi_z$.

Since the problem is stochastic, then an estimation of $f_1(Z)$ (in Eq. 1) can be found by using a number of independent simulations (i.e. Monte Carlo Simulations). For each $(\Xi_z)_{ij} \in \Xi_z$, K observations $(\xi_{ij}^1, \xi_{ij}^2, \ldots, \xi_{ij}^K)$ are generated. Let $(\Xi_z)^k$ be a set containing the kth observation from each $(\Xi_z)_{ij} \in \Xi_z$. An estimate of $f_1(Z)$ is obtained using the sample average:

$$\bar{f}_1(Z) = \frac{1}{K} \sum_{k=1}^{K} f_c\left(Z, (\Xi_z)^k\right) \quad (4)$$

where the observations to be taken from Ξ_z of Eq. 3 are the same observations as in $(\Xi_z)^k$.

Note that the targets located in the overlapped areas, i.e. areas that more than one AIR can reach, are only counted once. The overlapped areas can be allocated to an AIR based on a "first come, first served" strategy or based on the area partitioning and allocation method presented in [11].

b) Design Objective 2 (Minimal Makespan): Another important objective is to minimize the makespan (i.e. the overall completion time of the task). The objective is formulated to minimize the expected value, i.e. :

$$\min_Z f_2(Z) = E[f_t(Z, \Xi_z)] \qquad (5)$$

where

$$f_t(Z, \Xi_z) = \max\{t_1(Z, \Xi_z), t_2(Z, \Xi_z), \ldots, t_n(Z, \Xi_z)\}, \qquad (6)$$

and the finish time of the *i*th AIR,

$$t_i(Z, \Xi_z) = \left(\sum_{j=1}^{(n_b)_i} S^f(Z_{ij}, \xi_{ij}) \frac{d_i}{v_i}\right) + (n_b)_i (t_{setup})_i \qquad (7)$$

where d_i is the distance between the center points of two adjacent targets that are on a path of the *i*th AIR, v_i is the speed that the end-effector of the *i*th AIR moves relative to a path, and $(t_{setup})_i$ is the set-up time associated with switching off and on equipment/tools as the *i*th AIR navigates from one base placement to the next. Similar to $f_1(Z)$, an estimation of $f_2(Z)$ can be made using the sample average:

$$\bar{f}_2(Z) = \frac{1}{K} \sum_{k=1}^{K} f_t\left(Z, (\Xi_z)^k\right). \qquad (8)$$

3) Design Constraint (Distance Between Any Two AIRs): At any time *t* of the task execution, the distance between any two AIRs must be greater than a threshold, δ so as to avoid collisions between AIRs and to allow each AIR's manipulator to maneuver more freely within its workspace. The below constraint needs to be met:

$$\|\beta_{AIR_i}(t) - \beta_{AIR_j}(t)\| - (e_i + e_j) > \delta \qquad (9)$$

$\forall i, \acute{j} : i = 1, 2, \ldots, n, \acute{j} = 1, 2, \ldots, n, i \neq \acute{j}$ where $\beta_{AIR_i}(t)$ and $\beta_{AIR_j}(t)$ are the base placements from which the *i*th and the $\acute{j}$th AIRs are operating at time *t*, respectively. e_i and $e_{\acute{j}}$ are the maximum anticipated errors in base placements of the *i*th and $\acute{j}$th AIRs, respectively, which can be, for example, based on the limits of the 99.7% confidence interval (3σ) of the normal distributions representing the uncertainties.

Recall that FBPs are the base placements that are a safe distance away from objects, hence a second constraint is not needed to keep a safe distance between any AIR and the objects in the environment.

C. Multi-objective Stochastic Optimization

In order to solve and test the proposed mathematical model, a suitable multi-objective optimization is required. The multi-objective stochastic optimization used is based on hybrid GA-SA approach. It is inspired by the works in [12], [13] and [14], however certain modifications are made so as to appropriately address the problem under consideration.

The reason for the use of the hybrid GA-SA is to benefit from the advantages of both the GA and the SA, e.g. to benefit from combining the following properties: (1) SA is "one of the fastest and universal probabilistic local procedures" [12]; (2) for complex models, SA has been shown

[14] to be ideally suited for stochastic optimization, and in fact even for non-stochastic models, artificially adding noise improves the performance of SA; (3) GA can create a large number of diverse solutions when creating a population at each generation; and (4) selecting a neighbor solution in SA doesn't guarantee a better solution, however each generation in GA has a higher likelihood of producing better results than its preceding generation.

Algorithm 2 outlines the procedure of the implemented hybrid multi-objective GA-SA for stochastic optimization. Note that the parameters are defined at the top of the algorithm, some comments are added within the algorithm, and explanations of the key lines in the algorithm are provided below. The algorithm takes advantage of utilizing the multinomial probability mass function [13] where an objective is randomly selected to become active in each iteration (i.e. by randomly selecting the value of $\acute{i}$ in line 12), and the varying sample size [14] where the number of observations, *K* gradually increases proportional to the GA iteration number, *l* (i.e. line 6).

To better suit the target problem, Algorithm 2 is designed with certain differences to the algorithms used in the aforementioned literature. Instead of a single objective

---

Algorithm 2 Hybrid multi-objective GA-SA

Input: Objective 1 & 2 initial temperatures: T_1, T_2, Cooling ratios: α_1, α_2, Initial population: P_0, Fitness values for P_0: $F(P_0)$, Population Size: N_{Pop}, No. of generations: N_{Gen}, Max. no. of observations: K_{max}, No. of constant temperature loops: N_{Const}

Output: Pareto-Optimal Solutions (POS)

1: **for** $l = 1$ to N_{Gen} **do**
2:   **for** $m = 1$ to N_{Pop} **do**
3:     **for** $n_c = 1$ to N_{Const} **do**
4:       $\acute{m}_{\acute{i}} \leftarrow \text{random}\{1, 2, \ldots, N_{Pop}\}, \text{ for } \acute{i} = 1, 2, \; \acute{m}_1 \neq \acute{m}_2$
5:       $\rho_{l,m} \leftarrow \text{GenerateChild}(\rho_{l-1,\acute{m}_1} \in P_{l-1}, \; \rho_{l-1,\acute{m}_2} \in P_{l-1})$
6:       $K \leftarrow \text{ceil}(l.K_{max}/N_{Gen})$
7:       $\xi_{ij}^1, \xi_{ij}^2, \ldots, \xi_{ij}^K \leftarrow \text{GenerateObservations}((\Xi_z)_{ij}), \; \forall i, j$
8:       $F(\rho_{l,m}) \leftarrow [\bar{f}_1(\rho_{l,m}), \bar{f}_2(\rho_{l,m})]$ %based on Eqs. 4 & 8
        where $Z = \rho_{l,m}$
9:       **if** $F(\rho_{l,m}) \succ F(\rho_{l-1,\acute{m}_1})$ **or** $F(\rho_{l,m}) \succ F(\rho_{l-1,\acute{m}_2})$ **then**
10:         $P_l \leftarrow P_l \cup \rho_{l,m}$ %add to P_l if $\rho_{l,m}$ dominates a parent
11:         **goto** line 18
12:       **else if** $\text{random}(0, 1) < \exp\left(-(\bar{f}_{\acute{i}}(\rho_{l,m}) - \bar{f}_{\acute{i}}(\rho_{l-1,\acute{m}_{\acute{i}}}))/T_{\acute{i}}\right)$
        where $\acute{i} = \text{random}\{1, 2\}$ **then**
13:         $P_l \leftarrow P_l \cup \rho_{l,m}$
14:         **goto** line 18
15:       **else if** $n_c = N_{Const}$ **then** %if no solution was added to P_l
16:         $P_l \leftarrow P_l \cup \rho_{l-1,\acute{m}_{\acute{i}}}$ %accept a parent as a solution
17:       **end if**
18:     **end for**
19:   **end for**
20:   $[P_l, \text{POS}] \leftarrow \text{Update-NSGA-II}(P_l, P_{l-1}, \text{POS})$
21:   $T_1 \leftarrow \alpha_1 T_1, \quad T_2 \leftarrow \alpha_2 T_2$
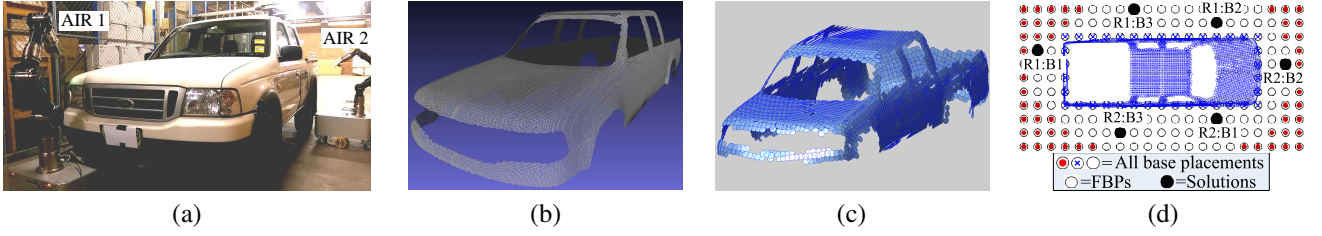22: **end for**
23: **return** POS

Fig. 3: (a) Set-up of the experiment, (b) Point cloud of the vehicle, (c) Target representation, (d) A solution selected from the Pareto front.

GA, a controlled elitist NSGA-II [15], which is a multi-objective non-dominated sorting genetic algorithm, is used. An advantage of NSGA-II is that unlike the diversification strategy used in [12], the diversity in Algorithm 2 is achieved through NSGA-II itself [15] (i.e. when updating the current population in line 20). Additionally, since NSGA-II is a multi-objective optimization algorithm, then Pareto-Optimal Solutions (POS) can be obtained (line 23) and a solution can then be conveniently selected for the target application. However, POS need to be updated (line 20) after each generation. Termination criteria is also decided by NSGA-II rather than SA, e.g. if the maximum number of generations, N_{Gen} is reached, or if a threshold on the average change in POS over certain number of generations is met. As a result, since SA doesn't terminate the algorithm, then there is the convenience of each objective having its own annealing scheme (line 21), and hence unlike in [12], there is no need for normalizing the objectives in order to have a single temperature applied to the equation used in line 12.

Note that in each generation, a population is created by iteratively generating children (line 5) through crossover and mutation operators, but only accepting those that dominate their parents (lines 9 to 11) or satisfy the acceptance criteria used by SA (lines 12 to 14). If the mutation operator is used to generate a child, $\rho_{l,m}$ then only one of the input parents ($\rho_{l-1,\acute{m}_1}$ or $\rho_{l-1,\acute{m}_2}$) will be used, and consequently only the used parent will be checked for dominance (i.e. in line 9).

IV. EXPERIMENTS

Six case studies are conducted using a real-world object (Fig. 3(a)) and a complex simulated object (Fig. 4(a)), and the purpose is to test and compare the Stochastic-OMBP for different scenarios and conditions. The AIRs used in the case studies comprise of a RGBD camera affixed to the end-effector of a 6DOF Schunk industrial robot mounted on a mobile Neobotix MP700 base. Computation is carried out on a PC with 2.8GHz Intel Xeon E5-2680, however since the code is single-thread then only one core of the CPU is used. The *gamultiobj* function from the Matlab R2013a optimization toolbox, which is based on NSGA-II, was used. Parameters are set as default, however modifications were made to accommodate for the hybrid GA-SA approach.

The AIRs are tasked with the grit-blasting operation [3] and they mimic the cleaning of the external metal surfaces of a vehicle in a real-world environment (Fig. 3) and a complex

simulated object (Fig. 1). As was mentioned in Section III-A, exploration and mapping [9] result in a point cloud (Figs. 3(b)) and a target representation (Figs. 3(c)). Further details are presented in [9].

Certain main parameters are the same for the AIRs: (1) targets are $0.04m$ in radius and each has an approx. 30% overlap with a neighboring target, (2) discrete base placements are generated with $0.35m$ spacing, (3) the set-up time $(t_{setup})_i$ in Eq. 7 is chosen to be a conservative value of 10mins, (4) δ in Eq. 9 is set to $1m$, and (5) in Eq. 2, the covariance $\Sigma_i = \Sigma$ for all i where Σ is a 2×2 identity matrix (I_2) multiplied by $\sigma^2 = 0.01m^2$. Note that the first joint of the AIRs which rotates about the z-axis is a full $360°$ revolute joint, and the base of the AIRs can't move vertically.

Note that at each base placement, the aim is to find the reachable targets rather than planning a trajectory for the AIR which requires separate planner during the task execution (i.e. after running the Stochastic-OMBP). An independent lookup table approach [3] was used to generate AIR poses for each target representing the object.

A. Case Study 1: Two AIRs Grit-blasting Vehicle's Surfaces

The Stochastic-OMBP is carried out for the real-world scenario shown in Fig. 3(a). 3270 targets represent the surfaces of the vehicle as shown in Fig. 3(c). The end-effector speed, v_i in Eq. 7 is set to $0.056m/s$ for both AIRs. Fig. 3(d) shows a solution chosen from the Pareto front. The selection strategy for choosing a solution from the Pareto front is similar to that explained in [3]. In the figure, 179 discrete base placements are shown from which 87 are the FBPs. The red filled circles (base placements) are poor performing in terms of surface coverage and the blue crossed circles have high likelihood of collision with the object. The filled black circles are based on the chosen solution where the notation Ri:Bj represents the jth base placement of the ith AIR.

Based on the chosen solution, on average, 94.8% of the accessible targets can be covered with a makespan of 3183 seconds. The difference in completion time of the two AIRs is less than 3%, meaning that a near-optimal makespan has been achieved since the AIRs will be expected to complete the task at approximately the same time. It was found that a solution with coverage greater than 99% can be achieved using the model, however this results in a significantly larger makespan. In the case when a slightly larger threshold for coverage is needed (e.g. 97%), then at least one AIR will

TABLE I: Multi-objective hybrid GA-SA vs. NSGA-II.

	Coverage	SD	Makespan	SD
Hybrid GA-SA	92.82 %	0.0057	3167 sec	30.79
NSGA-II	89.81 %	0.0189	3176 sec	101.01

need to visit an additional base placement, which results in the makespan being significantly increased since the value of $(t_{setup})_i$ is chosen to be conservative (10mins).

B. Case Study 2: Checking Solution Convergence

To ensure that the hybrid stochastic optimization algorithm used is robust and suitable for solving the proposed model, Case Study 1 was repeated 10 times and for each run the best solution from the Pareto front was selected. For each of the 10 solutions, the sample average for each of the two design objective was calculated (i.e. based on Eqs. 4 and 8) by considering 100 observations. Then, the average of the 10 solutions was calculated. The results are shown in the middle row of Table I. Each of the 10 solutions is a near-optimal solution and the low standard deviation (SD) in Table I shows that the Pareto front converges upon optimal solutions.

C. Case Study 3: Comparing Hybrid GA-SA with NSGA-II

To further demonstrate that the hybrid GA-SA algorithm is suitable, the proposed mathematical model was solved (for Case Study 1) using NSGA-II alone (i.e. without incorporating simulated annealing). The same default settings were used for the *gamultiobj* function of the Matlab optimization toolbox. Once again, similar to the previous case study where the hybrid GA-SA was used, the optimization was repeated 10 times and the average of the 10 solutions was calculated. 100 observations were considered in each solution. The results are shown in the Table I. It can be seen that the solutions obtained using the hybrid GA-SA algorithm are better than using NSGA-II alone. For both algorithms, on average, the optimization terminates at 105 GA iterations. Note that analyzing sensitivity, convergence rate and tuning optimization parameters is outside the scope of this paper.

D. Case Study 4: Testing for Various Levels of Uncertainties

To examine the extent to which uncertainties in base placements effects the results, a solution, Z^* is tested for different uncertainties, i.e. for different values of Σ ($\Sigma = I_2\sigma^2$) in Eq. 2, as shown in Table II. The solution, Z^* is obtained while not accounting for uncertainties in the proposed mathematical model. Z^* has an exceptional performance for the scenario discussed in the first case study (94%
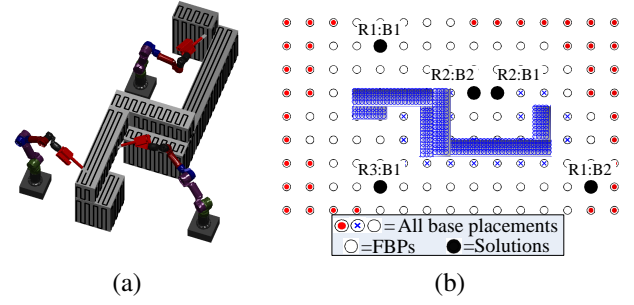

Fig. 4: (a) 3 AIRs grit-blasting a complex object, (b) A solution selected from the Pareto front.

coverage, and 3126 seconds makespan), however this is only true for the condition where there are no uncertainties in base placements. The solution Z^* was tested for different levels of uncertainties and as shown in Table II, as the uncertainties in base placements of the AIRs increase, the result of Z^* worsens. Hence, by accounting for uncertainties as per the proposed Stochastic-OMBP approach, other solutions can be found that will have better performance, as shown in Table II. Note that as per previous case studies, the solutions are calculated based on 100 observations.

E. Case Study 5: Three AIRs with Different Capabilities Grit-blasting a Complex Object

This case study is constructed to test the Stochastic-OMBP for a complex structure shown in Fig. 4(a), and for an increased number of AIRs (3 AIRs) that have different capabilities in terms of end-effector speed, i.e. $v_1 = 0.056m/s$, $v_2 = 0.05m/s$, and $v_3 = 0.04m/s$. In this case study, the acceptable threshold of coverage is set very high (99%). The object is $3 \times 1 \times 0.8$ meters in size. The workspace size of the AIRs is the same, meaning that the discrete base placements shown in Fig. 4(b) are the same for all AIRs. 66 of the 135 discrete base placements are the FBPs.

The results are shown in Fig. 4(b), which is based on a solution chosen from the Pareto front. AIR 3 is assigned only one base placement so as to minimize the overall completion time of the task, since this AIR is the slowest AIR. On average, the completion time of AIRs 1 to 3 is 2060, 2060, and 1520 seconds, respectively. On average, the AIRs can cover 99.4% of the 2279 targets representing the surfaces.

To check for convergence, the same procedure as in Case Study 2 is performed. The average of the 10 solutions is 99.3% for coverage with a SD of 0.0032%, and 2109 seconds for makespan with a SD of 94 seconds. On average, the optimization terminates at 104 GA iterations.

F. Case Study 6: Testing for Various Levels of Uncertainties

The same procedure as in Case Study 4 is carried out for the scenario discussed in Case Study 5. The results are shown in Table III. Solution Z^* is optimal with 100% coverage and a makespan of 2708 seconds, however this is only true for the condition where there are no uncertainties in base placements. Solution Z^* becomes unacceptable in

TABLE II: Testing for different values of $\Sigma = I_2\sigma^2$ (for the scenario where 2 AIRs are grit-blasting a vehicle).

		$\sigma^2=0.01$	$\sigma^2=0.02$	$\sigma^2=0.03$
For solution Z^*	Coverage (%)	91.25	88.60	86.16
	Makespan(sec)	3131	3116	3108
Incorporating uncertainties	Coverage (%)	92.82	91.75	89.66
	Makespan(sec)	3167	3107	3105

TABLE III: Testing for different values of $\Sigma = I_2\sigma^2$ (for the scenario where 3 AIRs are grit-blasting a complex object).

		$\sigma^2{=}0.01$	$\sigma^2{=}0.02$	$\sigma^2{=}0.03$
For solution Z^*	Coverage (%)	98.90	97.94	97.16
	Makespan(sec)	2150	2163	2179
Incorporating Uncertainties	Coverage (%)	99.26	99.49	99.23
	Makespan(sec)	2109	2161	2325

meeting the 99% coverage threshold for all 3 levels of uncertainties considered. By accounting for uncertainties, as per the proposed model, acceptable solutions are found for all 3 levels of uncertainties.

V. DISCUSSION

The case studies have verified the proposed approach, however there are a number of limitations that require a brief discussion. For the applications under consideration the mobile base of the AIRs must be kept static during the task execution, and when an AIR has completed its task at a base placement, it moves to its next base placement using an existing and simple path planner. However, in certain foreseeable future applications it may be necessary for an AIR's base to move while the AIR is executing the task. This would require additional parameters to be considered during the planning stage. The computational complexity of the algorithms has been found to be near linear as the number of robots and targets increase, although it would be interesting to verify this more thoroughly in the future.

A number of features were used with the aim of improving the computation time, e.g. discretization of the search space and the AIRs' end-effector path, varying sample size in the optimization algorithm, and the use of a lookup table to find AIR poses. However, the main focus of this work has been on testing the mathematical model for achieving optimal base placements rather than improving time efficiency. Potential efficiency improvements include: (1) using surface fitting to estimate the fitness values for each observation, (2) allowing the AIRs to start executing the task with a reasonable solution, and if a better solution is found at a later stage, then the current solution merges smoothly to the improved solution at an appropriate time interval, (3) optimizing the parameters used in the hybrid GA-SA algorithm, and (4) designing a criteria for assessing the effectiveness of the search space discretization method.

VI. CONCLUSION

Multiple mobile and autonomous industrial robots can be deployed to carry out tasks such as grit-blasting and spray painting. In this paper, an approach was presented with the aim of maximizing the coverage of objects' surfaces under uncertainties of the robots' base placements. The principal behind the approach enables the robots to collaborate in a manner that minimizes the effect of uncertainties on the performance of the robots, while optimizing the robot team's objectives. A mathematical model has been developed to deal with the stochastic nature of the problem and to achieve complete coverage. The mathematical model is solved using a stochastic optimization approach that considers hybrid multi-objective GA-SA to optimize coverage and makespan. The presented approach was tested using both simulated and real-world experiments and verified by means of several comparative studies using different scenarios and conditions.

As future work, computation time and complexity of the approach can be studied in detail. Extending the approach to be applied to more complicated scenarios with varying distributions representing uncertainties in robots' base placements (e.g. in underwater environments) can also be investigated.

REFERENCES

[1] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258 – 1276, 2013.

[2] M. Hvilshoj, S. Bogh, O. Skov Nielsen, and O. Madsen, "Autonomous industrial mobile manipulation (AIMM): past, present and future," *Industrial Robot: An International Journal*, vol. 39, no. 2, pp. 120–135, 2012.

[3] M. Hassan, D. Liu, G. Paul, and S. Huang, "An approach to base placement for effective collaboration of multiple autonomous industrial robots," in *IEEE International Conference on Robotics and Automation*, May 2015, pp. 3286–3291.

[4] M. F. Aly, A. T. Abbas, and S. M. Megahed, "Robot workspace estimation and base placement optimisation techniques for the conversion of conventional work cells into autonomous flexible manufacturing systems," *International Journal of Computer Integrated Manufacturing*, vol. 23, no. 12, pp. 1133 – 1148, 2010.

[5] P. Sotiropoulos, N. Aspragathos, and F. Andritsos, "Optimum docking of an unmanned underwater vehicle for high dexterity manipulation," *IAENG International Journal of Computer Science*, vol. 38, no. 1, pp. 48 – 56, 2011.

[6] G.-C. Vosniakos and E. Matsas, "Improving feasibility of robotic milling through robot placement optimisation," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 5, pp. 517–525, 2010.

[7] T. Bretl and S. Hutchinson, "Robust coverage by a mobile robot of a planar workspace," in *IEEE International Conference on Robotics and Automation*, May 2013, pp. 4582–4587.

[8] E. Galceran, R. Campos, N. Palomeras, D. Ribas, M. Carreras, and P. Ridao, "Coverage path planning with real-time replanning and surface reconstruction for inspection of three-dimensional underwater structures using autonomous underwater vehicles," *Journal of Field Robotics*, 2014.

[9] G. Paul, S. Webb, D. Liu, and G. Dissanayake, "Autonomous robot manipulator-based exploration and mapping system for bridge maintenance," *Robotics and Autonomous Systems*, vol. 59, no. 78, pp. 543–554, 2011.

[10] M. Clifton, G. Paul, N. Kwok, D. Liu, and D.-L. Wang, "Evaluating performance of multiple RRTs," in *IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications*, Oct 2008, pp. 564–569.

[11] M. Hassan, D. Liu, S. Huang, and G. Dissanayake, "Task oriented area partitioning and allocation for optimal operation of multiple industrial robots in unstructured environments," in *13th International Conference on Control Automation Robotics Vision*, Dec 2014, pp. 1184–1189.

[12] B. Cakir, F. Altiparmak, and B. Dengiz, "Multi-objective optimization of a stochastic assembly line balancing: A hybrid simulated annealing algorithm," *Computers and Industrial Engineering*, vol. 60, no. 3, pp. 376 – 384, 2011.

[13] R. Banos, J. Ortega, C. Gil, A. Fernandez, and F. de Toro, "A simulated annealing-based parallel multi-objective approach to vehicle routing problems with time windows," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1696 – 1707, 2013.

[14] X. Gu (Gracie), "The behavior of simulated annealing in stochastic optimization," p. 64, 2008, m.S thesis, Iowa State University, Ames, Iowa.

[15] K. Deb and T. Goel, "Controlled elitist non-dominated sorting genetic algorithms for better convergence," in *Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, E. Zitzler, L. Thiele, K. Deb, C. Coello Coello, and D. Corne, Eds. Springer Berlin Heidelberg, 2001, vol. 1993, pp. 67–81.