Coverage Path Re-Planning for Processing Faults

Mykhaylo Nykolaychuk and Frank Ortmeier

Otto-von-Guericke University, Chair of Software Engineering,
Universitätsplatz 2, 39106 Magdeburg, Germany
{mykhaylo.nykolaychuk,frank.ortmeier}@ovgu.de
https://cse.cs.ovgu.de

Abstract. Currently, an automated surface treatment or finishing (e.g., abrasive blasting, cleaning or painting) is performed in two consecutive steps: first processing by tool, second quality evaluation by sensor. Often, a finished surface has defects like areas not properly processed. This is caused by path inaccuracy or errors in tool deployment. The defected areas can be detected only during a subsequent quality evaluation. As a result, a complete surface reprocessing is required which is costly and time-consuming. We propose a new approach that is a combination of surface treatment and quality evaluation processes in a single deployment. In our approach, an initial coverage path for surface treatment by a tool is given or calculated using a state-of-the-art coverage path planning algorithm. In fact, we extend an off-line generated initial coverage path with an ability to react to sensor-based defect recognitions during surface processing by a tool.

Keywords: on-line coverage path planning, path re-planning

1 Introduction

A common task in surface treatment applications is to generate a coverage path for processing a given surface by a tool satisfying specific constraints, e.g., overlapping of adjacent tool footprints in painting. One example is painting a surface with an industrial robot. Computing the optimal path for the robot to paint the surface is the well-known Coverage Path Planning (CPP) problem whose goal it is to find a path that passes a tool through all points of a given surface while avoiding obstacles [9]. In most surface processing applications, a robot follows a path which could be generated either off-line or on-line. Then, the processed surface is inspected for detecting areas not properly processed. Therefore, existing CPP approaches are developed either for an automated surface processing by a tool (e.g., painting gun) [3, 2] or for an automated surface inspection by a sensor (e.g., range sonar, camera) [6, 14]. To the best of our knowledge, the coverage path re-planning for processing faults which requires a combination of two simultaneous actions, namely surface treatment by tool and on-site defect recognition by sensor, is research topic still not addressed. We propose to re-plan an unprocessed part of a given or precomputed initial path whenever a defect is reported in order to reprocess a defected area within current mission. Note that our focus is on a minimal modification of an initial path. We assume that the initial path is feasible and only has to be adapted due to defect reprocessing. In contrast, most off-line and on-line CPP approaches would recompute a new path globally. Using the existing off-line CPP approaches for coverage path re-planning is not feasible since they are time-consuming [6]. Although on-line CPP approaches are fast, they are rather used for global path planning and would re-compute a full remaining part of the initial path. Furthermore, the off-line CPP approaches assume that *a priori* knowledge of working space is enough to generate feasible coverage paths. Hence, possible uncertainties, such as recognized defects, are not considered during the path following.

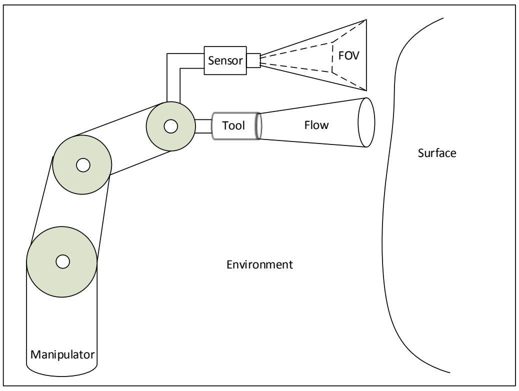


Fig. 1. One possible system setup for surface treatment by a tool and on-site quality evaluation by a sensor.

In this paper, we address the problem of event-based coverage path re-planning according to recognized defects. We classify defects regarding their geometry. Furthermore, we give a definition of the event-based coverage path re-planning problem. Finally, we propose a solution of the defined problem for a random defect on a planar surface. Technically, we need an automated surface processing robot combined with a sensor system for precise quality inspection. One possible system setup is depicted in Fig. 1 where a tool and a sensor are both mounted on the manipulator wrist in some static known relation to each other. Note that the defect recognition problem is out of scope of this work and is assumed to be solved by an existing algorithm, e.g., [13].

The remainder of the paper is structured as follows: after an overview of the state-of-the art CPP in Section 2, we give a mathematical problem formulation in Section 3 and discuss the main challenges. We propose a new approach for re-planning an initial coverage path in Section 4 in order to reprocess reported defects during the same mission. We demonstrate our approach on a planar surface and present the results in Section 5. Finally, we conclude in Section 6.

2 Related Work

Many CPP approaches in several robotic applications (e.g., abrasive blasting [13], marine [10], painting robots [2]) were proposed in last decades. For a more detailed review of existing off-line as well as on-line CPP approaches, we refer to the Choset's and Galceran's surveys [4, 9]. Most of them, however, are limited to 2D environments and off-line path planning. In contrast, only few approaches deal with 3D environments and on-line path planning or re-planning, especially in the surface processing applications [6, 10]. The complementary combination of both off-line and on-line approaches recently gained in popularity. This combination is especially beneficial for the tasks where *a priori* knowledge allows off-line path planning, but several uncertainties during path execution require on-line path re-planning [7, 10].

Off-line Coverage Path Planning Computing a trajectory off-line requires exact knowledge about the workspace models including the environment, the surface structure and the robot system behavior. In the off-line CPP, the Morse decomposition [1] is a good choice for complex work spaces which should be decomposed into simple segments. Each defined cell can be covered with simple back-and-forth motions using an adjacency graph. Random sampling-based coverage of 3D structures such as ship hulls is a powerful approach proposed by Englot et al. [6]. The method solved the art gallery problem [15] resulting in a set of views (collected in the graph) that ensure a complete surface coverage. Then, applying the Traveling Salesman Problem (TSP) [12] leads to the minimal-cost walk through the graph. Usually, off-line CPP approaches are time-consuming. They assume working space to be well-known before path planning and expect no faults during execution. Furthermore, in the coverage path re-planning we are primarily interested in a local path adaptation. Using existing off-line approaches for re-planning would lead to global path re-planning. We could, however, use off-line CPP approaches for initial coverage path generation.

On-line Coverage Path Planning or Re-Planning Capturing dynamic properties of the working space by sensor, information extraction and transformation into signals to robot controllers are the major steps of the on-line CPP. Work close to our research was presented by Galceran et al. [10] for inspection of 3D underwater structures. The authors developed a new algorithm for reshaping an initial path on-line for underwater inspection applications according to real-time environment perception by a range sensing sonar. The main objective was to keep an underwater vehicle at an optimal proximity to the sea bed in order to guarantee safe motion and best inspection quality. The approach, based on range measurements, defines sections for re-planning along an initial path and reshapes incrementally perceived sections according to sensor information. By contrast, in the context of coverage path re-planning for recognized defects, an appropriate solution way depends on defect location, defect geometry and other inputs. Each reported defect requires an immediate calculation of an optimal coverage path with minimal deviation from initial one. Hence, straightforward, step-by-step reshaping of an initial path as presented by Galceran et al. [10] is not applicable to our problem.

The approach proposed by Ewert et al. [7] uses an integration of off-line and on-line path planning for real-time trajectory re-planning during assembly. The main planning effort is concentrated in the off-line phase, where for a finite set of process states, a set of feasible movements is precomputed. Then, the proposed algorithm decides on-line about the best preplanned alternative. This approach is not applicable to automated reprocessing of defected areas since an exhaustive prediction of all defect locations, geometries and possible robot configurations is not feasible.

To summarize, existing off-line and on-line approaches are developed in a context very specific and different from ours. Therefore, they are not applicable to the event-based coverage path re-planning problem with the goal to reprocess reported defects during the current mission.

3 Problem Definition

Mathematical formulation Given surface S that is $2D$ manifold in $\mathbb{R}^3$, a tool expressed by its circular footprint area T with radius $r \in \mathbb{R}$, a generated path $P = (p_1, p_2, \dots, p_n)$ where $p_i \in \mathbb{R}^6, 1 \leq i \leq n, n \in \mathbb{N}$ for complete processing of S by a tool T and some defected area D that is $2D$ manifold in $\mathbb{R}^3$ whereby $D \subset S$. At each time step t during path execution, the surface could be divided into already finished part S_f and an unprocessed part S_u. Hence, surface and path could be expressed at any time as $S = S_f \cup S_u$ and $P = P_f \cup P_u$. Every incoming defect information changes the finished and unprocessed surface parts to $S'_f = (S_f \setminus D)$ and $S'_u = S_u \cup D$ respectively. Furthermore, a path part P_u becomes infeasible for processing of S'_u.

Usually, several task-specific constraints and requirements must be satisfied in order to obtain a feasible coverage path. For instance, uniform coverage of a surface in a painting scenario requires a predefined overlapping rate of two successive path strokes. Overlapping constraint of adjacent circular tool footprints which lies on parallel path strokes, defined as $O : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$, is desired at some predefined rate q with allowed deviation ϵ and can be expressed for any footprint pair (T_1, T_2) as $O(T_1, T_2) = q \pm \epsilon$. In some application (e.g., cleaning), overlapping of adjacent tool footprints is not critical from a process quality point of view and can, therefore, be neglected.

The goal is to find a minimal cost adapted path $P' = (p'_1, p'_2, \dots, p'_n)$ to complete processing of unprocessed surface part S'_u considering task-specific constraints C which could be formulated as follows:

$$F(S'_u, P, C) = P'. \tag{1}$$

During path re-planning the goal is to minimize the amount of path turns and/or the total length of the path P' which can be used as a cost function.

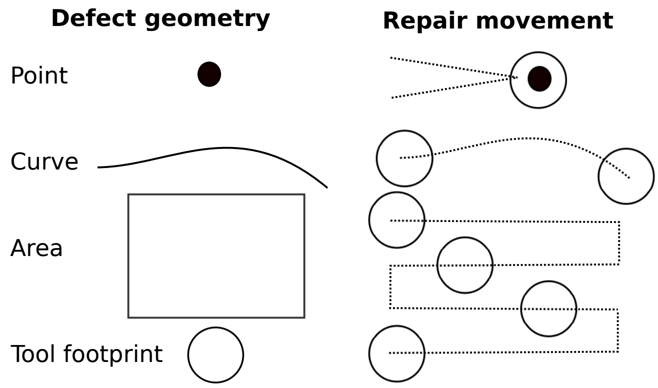


Fig. 2. Three types of defects (*point, curve and area*) and corresponding tool paths (dashed lines) for defect repair.

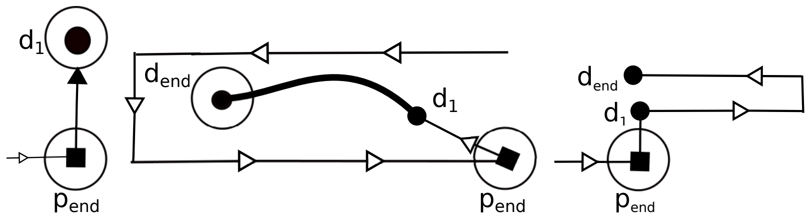
Problem Discussion In order to develop feasible solutions for event-based CPP, it is important to classify defects based on their geometry. Therefore, we propose to distinguish three types of defects, namely point, curve or area as shown in Figure 2. In particular, area is a general defect type including two special cases, point an curve. But a significant difference between general and special cases is that for a special defect type (i.e., point and curve) the coverage path or samples are given and for a general defect type a local coverage path has to be calculated. In order to obtain a feasible (i.e., cost-optimal) path deformation we define the following common requirements which must be satisfied:

- if possible, only local path re-planning for defect repair,
- minimal number of additional path turns (),
- satisfaction of task-specific constraints (e.g., overlapping).

Note that for simplification reasons, we consider recognized defects as fully unprocessed regions without granularity of the processing already done. In order to manage different processing levels of defect areas, additional sensor information about cleanliness or roughness is to be incorporated into the calculation of an adapted path.

4 Coverage Path Re-Planning

The goal of the coverage path re-planning is to find a path adapted for reprocessing of a reported defect. An initial state of the problem is determined by a current tool position on the initial path $p_d \in P$, a defect location d, the remaining part of the initial path P_u and the surface part S'_u to be processed. In general, depending on input we decide whether to perform local path adaptation or to undertake global path re-planning. Essentially, the current tool position



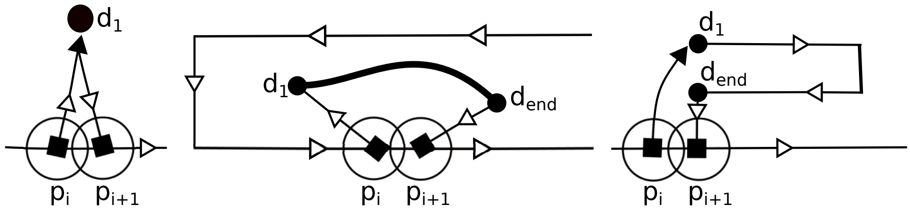
(a) point (b) curve defect (c) area defect

Fig. 3. Reprocessing of different defect types by extension of initial path P_u at the point $p_{end} \in P_u$.

$p_1 \in P_u$ implies four possible re-planning alternatives: 1) immediately stop to follow the initial path and start the local defect repair movement, if p_1 is in close proximity to d_1; 2) keep following the initial path because there is a better location on the path $p_i \in P_u$ for local defect repair; 3) finish surface processing by following the path P_u and then starting from $p_{end} \in P_u$, computing a path for defect reprocessing (see Fig. 3) or 4) re-plan the path P_u globally, if costs of local refinement become infeasible due to task-specific constraints, e.g., no path overlapping. Note that we primarily search for defect repair by local modification of P_u. As a cost function, we use the length of the path, i.e., the sum of the euclidean distances between adjacent path points $\Sigma_{i=1}^{n} dist(p_i, p_{i+1})$. A naive solution to a local path adaptation problem could be:

1. finding way point p_i closest to defect entry point d_1
2. moving from p_i to defect entry point d_1
3. performing defect repair traveling from d_1 to d_{end}
4. leaving d_{end} and returning to the way point p_{i+1}
5. keep following initial path P_u

A solution for the local adaptation of the initial path P_u with respect to different defect types is shown in Fig. 4. Local path adaptation for reprocessing defected areas requires the computation of two suitable points on the initial path $p_i, p_{i+1} \in P_u$ which serve as the starting and the finishing points of the repair movement. For each defect type, a point $p_i \in P_u$ constitutes a location on the initial path with optimal cost (i.e., euclidean distance in combination with full surface coverage requirement and path overlapping constraint) relative to the defect entry point d_1. In order to guarantee full coverage of a given surface and not to introduce new uncovered areas, a tool has to return to the point $p_{i+1} \in P_u$ after finishing defect reprocessing. Fig. 4(a) demonstrates reprocessing a point defect d_1 (filled circle) by moving from point p_1 on initial path (thin



(a) point defect (b) curve defect (c) area defect

Fig. 4. Local path adaptation for repair of different defect types.

line) to d_1, performing defect repair and returning to p_{i+1}. In addition to a local path adaptation for reprocessing curve and area defects requires feasible start and end points of the repair movement, suitable entering d_1 and leaving points d_{end} of the defected region. For a curve defect, we first determine an entry point d_1 based on the current tool position. In Fig. 4(b) a curve defect is reprocessed starting from a path point $p_i \in P_u$, entering the defect area at point $d_1 \in D$, leaving it at point $d_{end} \in D$ in order to finally, finish the defect repair at end point $p_{i+1} \in P'_u$. Repairing an area defect would, first, require generating a coverage path for the defected area and then merging that newly generated path with the existing path P_u as shown in Fig 4(c).

In order to obtain an automated coverage path re-planning approach, we adapt the approach for obstacle avoidance based on attractive and repulsive potentials [11]. A local or global adaptation of the initially planned path is based on the potential function $U : \mathbb{R}^n \rightarrow \mathbb{R}$ which is defined as the sum of attractive and repulsive potentials [5]. We define a defected area as a goal which attracts all points on the initial path P_u. Note that each point on the initial path $p_i \in P_u$ is associated with attractive and repulsive potentials. This artificial force is defined as a quadratic function of distance (Eq. 2) between p_i and defect location d_1 in case of point defect or, in case of curve or area defect a principal point d_p. By contrast to the original approach, our attractive potentials decrease with distance from defect point d_1. This different behavior of attracting potentials allows to preserve an initial position of those points far away from defect d_1. Possible obstacles or predefined restrictions (e.g., no path overlapping) repel potentials. Note that we do not consider obstacle avoidance in this paper.

Let $dist(d_1, p_i)$ be the euclidean distance between defect entry point d_1 and some path point $p_i \in P_u$. Then, an attractive potential at some point $p_i \in P_u$ is computed as follows:

$$U_{att}(p_i) = \begin{cases} \frac{\frac{1}{2}\zeta}{dist^2(d_1,p_i)}, if\ dist(d_1, p_i) > dist^* \\ \frac{dist^*\zeta}{dist^2(d_1,p_i)}, if\ dist(d_1, p_i) \leq dist^*, \end{cases} \quad (2)$$

where $\zeta \in [0; 1]$ is a scaling factor and $dist^* \in \mathbb{R}$ is a threshold distance for stronger attractive potentials near the defect location and weaker ones that are further away. We compute threshold distance $dist^*$ dynamically, depending on the distance from the path point $p_i \in P_u$ closest to defect point d_1. If $dist(d_1, p_i)$ is smaller than the distance between two adjacent path strokes, then our threshold $dist^*$ is within close proximity to the defect. Otherwise, a defect and a nearest point on the initial path are separated by at least one or more path strokes. This leads to $dist^*$ being grater and therefore, influencing more points on the initial path. As a result, the artificial potentials will attract path points stronger to the defect. Note that with increasing distance $dist(d_1, p_i)$, where $p_i \in P_u$ is the nearest path point to the defect location, applying attracting potentials leads to a higher deviation of the re-planned path from initial path. In such cases the re-planned path could be refined by smoothing and/or adding additional path points.

To summarize, our proposed approach based on artificial potentials looks as follows:

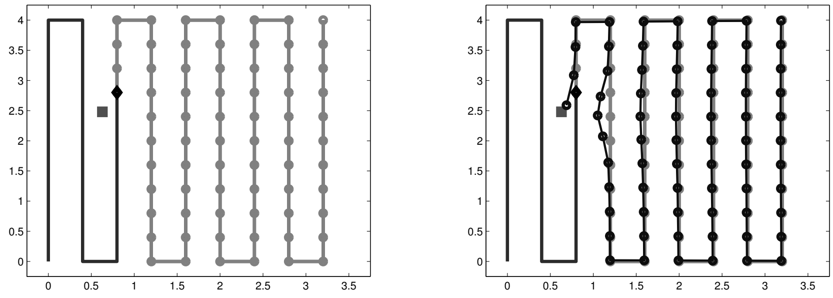
1. Input:
   initial path P_u
   surface S
   defect D
   current tool position $p \in P_u$
2. obtain artificial potentials for all $p_i \in P_u$ based on Eq. 2
3. apply calculated potentials to initial points $p_i \in P_u$
4. calculate costs for re-planned path
5. if costs not acceptable, use new scaling factor ζ for artificial potentials

In the Section 5 we test our approach on a specific application scenario.

5 Results

We implemented our approach in Matlab. We consider an application (e.g., cleaning) which is not sensitive to overlapping or to repeated visiting of surface parts already processed. A 2D rectangular surface is planar and without obstacles. We assume the tool orientation to be normal to the surface and to be represented by its circular footprint (with radius r) projected on the surface. An initial coverage path is a simple back and forth pattern.

We consider a case study regarding the point defect with different positions of the tool relative to defect location. The path re-planning is performed for two different tool positions on the initial path: 1) near the defect and 2) far from defect. Fig. 5(a) demonstrates for case one an initialization of the coverage path re-planning problem where tool position $p_i \in P_u$ (black filled diamond) is near the defect location d_1 (gray square). Gray circles on the initial path represent those way points $p_i \in P_u$ that have to be visited in order to cover the remaining surface. The result of applying our approach based on the artificial potentials is presented in Fig. 5(b). In this case a scaling factor $\zeta = 0.11$ was obtained in the second iteration of our approach. As a result, the black circles connected with the black line show the coverage path after re-planning. The biggest deviation from the initial points is located in the close proximity to the defect, i.e., on the

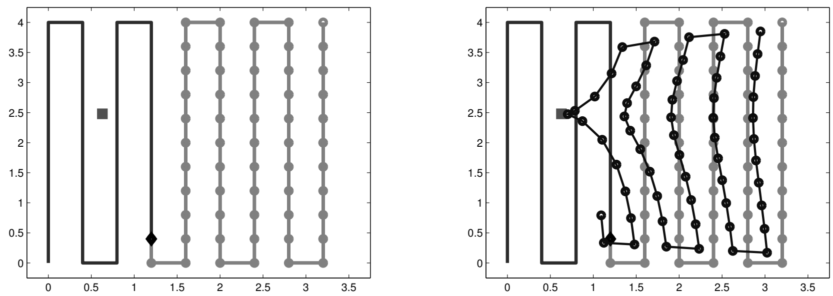


(a) Problem initialization (b) Solution of Tool near defect

Fig. 5. Coverage path re-planning for "tool near point defect" based on artificial potentials.

two closest path strokes. This corresponds very well with the predefined distance threshold $dist^*$. Following this adapted path allows to cover defect as well as the remaining surface during the current mission.

In the second case, the current location of the tool is far from the defect position, i.e., the defect was recognized significantly later than in the first case. Fig 6(a) shows the initialization of the "tool far from point defect", where the



(a) Problem initialization (b) Solution of Tool far from defect

Fig. 6. Coverage path re-planning for "tool far from point defect" based on artificial potentials.

grey square represents the same defect location d_1 and the black diamond on the initial path indicates a different current tool position $p_i \in P_u$. In this case, straightforward usage of artificial potentials leads to a strong path deformation (see Fig. 6(b)). This is caused by the big scaling factor $\zeta = 0.4$ and the distance threshold which particularly implies strong attractive potentials to all points on the initial path. this means that the resulting path is not feasible, since an expected full coverage of the remaining surface is not guaranteed. This result shows that an additional adaptation of potentials is required in order to obtain a re-planned path with less deviation from the initial path P_u.

Summarizing, a direct use of attractive potentials for coverage path re-planning can be a good choice for case "tool near defect location", but could fail in the case "tool far from defect". Hence, an additional adaptation of the method based on attractive potentials is required.

6 Discussion

The combination of surface treatment by a tool and quality evaluation by a sensor in a single deployment have not been addressed yet. In this paper we defined an event-based coverage path re-planning problem. Its main difference to the classic CPP problem is that the surface can be extended by a defect area during the surface treatment process. As a consequence, the coverage path (initially given or calculated) becomes infeasible and has to be adapted according to on-site information. A path re-planning strategy depends on the defect structure, a current tool position as well as on task-specific constraints and can be either local adaptation of initial path or global path re-planning. The solution to event-based coverage path re-planning is an optimal coverage path adapted on-line according to defect information. We proposed local and global solution for coverage path re-planning based on straightforward local adaption and artificial potentials respectively. However, last approach needs further investigation in order to perform feasible path re-planning for case "tool far from defect".

In this work we defined an event-based coverage path re-planning problem that only deals with geometrical path adaptation under consideration of task-specific constraints. Nevertheless, we envision several directions for future work. The algorithm presented in this paper works under the assumption that an initial coverage path is given. A more convenient way would be to develop a new off-line CPP algorithm in order to suit an initial coverage path P for on-line re-planning, e.g., to extract surface features (e.g., curvature) and use this for a task-dependent subdivision of a complex surface.

Furthermore, a freedom of execution can be used as an additional degree of freedom for coverage path planning or re-planning. As a rule, CPP assumes that the paint gun is normal to the surface. In reality, however, it might often have some deviation, which is not harmful for industrial process. From et al. [8] showed that small errors (up to 20 degrees) in the paint gun orientation does not affect the painting quality. This tool orientation freedom forms a spherical cone for each surface point. The path does not need to visit all the points, but can visit only an inner point of that cone. This freedom allows to obtain paths with smaller costs.

References

[1] Ercan U Acar et al. "Morse decompositions for coverage tasks". In: *The International Journal of Robotics Research* 21.4 (2002), pp. 331–344.

[2] Mayur V Andulkar, Shital S Chiddarwar, and Akshay S Marathe. "Novel integrated offline trajectory generation approach for robot assisted spray painting operation". In: *Journal of Manufacturing Systems* (2015).

[3] Prasad N Atkar et al. "Uniform coverage of automotive surface patches". In: *The International Journal of Robotics Research* 24.11 (2005), pp. 883–898.

[4] Howie Choset. "Coverage for robotics - A survey of recent results". In: *Annals of Mathematics and Artificial Intelligence* 31 (2001), pp. 113–126.

[5] Howie Choset et al. *Principles of robot motion: theory, algorithms, and implementation.* MIT press, 2005.

[6] Brendan Englot and Franz S Hover. "Sampling-Based Coverage Path Planning for Inspection of Complex Structures". In: *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS.* 2012.

[7] Daniel Ewert et al. "A Graph Based Hybrid Approach of Offline Pre-planning and Online Re-planning for Efficient Assembly under Realtime Constraints". English. In: *Intelligent Robotics and Applications.* Ed. by Honghai Liu et al. Vol. 6425. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2010, pp. 44–55.

[8] P.J. From and J.T. Gravdahl. "A Real-Time Algorithm for Determining the Optimal Paint Gun Orientation in Spray Paint Applications". In: *IEEE Transactions on Automation Science and Engineering* 7.4 (Oct. 2010), pp. 803–816.

[9] Enric Galceran and Marc Carreras. "A survey on coverage path planning for robotics". In: *Robotics and Autonomous Systems* 61.12 (2013), pp. 1258–1276.

[10] E. Galceran et al. "Coverage path planning with realtime replanning for inspection of 3D underwater structures". In: *IEEE International Conference on Robotics and Automation (ICRA).* 2014, pp. 6586–6591.

[11] Oussama Khatib. "Real-time obstacle avoidance for manipulators and mobile robots". In: *The international journal of robotics research* 5.1 (1986), pp. 90–98.

[12] Gilbert Laporte. "The traveling salesman problem: An overview of exact and approximate algorithms". In: *European Journal of Operational Research* 59.2 (1992), pp. 231–247. ISSN: 0377-2217.

[13] Pedro Javier Navarro et al. "Sensors Systems for the Automation of Operations in the Ship Repair Industry". In: *Sensors* 13.9 (2013), pp. 12345–12374.

[14] P Olivieri et al. "Coverage path planning for eddy current inspection on complex aeronautical parts". In: *Robotics and Computer-Integrated Manufacturing* 30.3 (2014), pp. 305–314.

[15] T.C. Shermer. "Recent results in art galleries (geometry)". In: *Proceedings of the IEEE* 80.9 (Sept. 1992), pp. 1384–1399.