Coverage Control of Autonomous Vehicles for Oil Spill Cleaning in Dynamic and Uncertain Environments

Xin Jin† Asok Ray‡

Abstract—In the context of oil spill cleaning by autonomous vehicles in dynamic and uncertain environments, this paper presents a multi-resolution algorithm that seamlessly integrates the concepts of local navigation and global navigation based on the sensory information; the objective here is to enable adaptive decision-making and online replanning of paths. The proposed algorithm provides a complete coverage of the search area for cleanup of the oil spills and does not suffer from the problem of having local minima, which is commonly encountered in potential-field-based methods. The efficacy of the algorithm is tested on a high-fidelity Player/Stage simulator for oil spill cleaning in a harbor, where the underlying oil weathering process is modeled as 2D random-walk particle tracking.

Index Terms—autonomous agents, oil spill, coverage control, uncertain environment, path planning

I. INTRODUCTION

The recent Deepwater Horizon oil spill in the Gulf of Mexico has attracted the attention of world community due to its colossal ecological, economic and social impacts. Over 210 million gallons of crude oil was released and the slicks and sheen of the surface oil directly affected over 180,000 square kilometers of ocean surface [1]. In order to clean this oil spill, over 39,000 personnel, 5,000 vessels, and 110 aircraft were involved, over 700 km of booms were deployed, 275 controlled burns were carried out, approximately 27 million gallons of oily liquid were recovered by skimmers, and more than 1.5 million gallons of chemical dispersant were used in these efforts [2].

In view of the facts that the current oil spill cleaning technology is labor intensive and the toxic chemicals and oil vapors are pernicious to the health of the cleaning crews, there is a pressing need for development and implementation of new technologies for combating oil spills. To mitigate the adverse environmental effects of an oil spill, research efforts focus on development of technologies to remove the oil in situ, minimize operational time, and protect health and safety of the cleaning crew [3]. To this end, several novel methods have been developed to make use of autonomous vehicles for effective oil spill confrontation, such as Seaswarm [4] and Protei [5] that are intended to work as a fleet or “swarm” of vehicles to create an organized system for autonomous ocean-skimming and oil removal. While the current trend emphasizes hardware improvement, advanced navigation algorithms are yet to be developed.

This paper develops a multi-resolution method for autonomously cleaning oil spills in dynamic and uncertain environments, as an augmentation of the authors’ recent work [6] in which an autonomous vehicle explores the unknown and static environment and covers the entire search area. The underlying algorithm [6] relies on the notions of both local navigation and global navigation that depend on the spatio-temporal information needed to make these decisions. However, in general, the environment is dynamic due to the spreading and drift of the spills; therefore, the algorithm [6] is unable to adapt to the weathering process of the oil spill and is thus inadequate for cleaning up the spills.

The current paper overcomes this inadequacy by introducing the capability of dynamic adaptation that, upon detection of the oil spills, would enable the autonomous vehicle to replan its actions online. The proposed algorithm is validated on a Player/Stage platform that is capable of high-fidelity simulation of autonomous vehicles and oil weathering processes for comparison with the benchmark algorithm of back and forth (i.e., zigzag) motion.

II. MODELING OF OIL SPILL PHENOMENA & CLEANING

The a priori information, as needed by autonomous vehicles for oil spill cleaning in dynamic and uncertain environments, is often either incorrect or incomplete. Therefore, time-critical operations of these vehicles require real-time decision-making to facilitate continuous adaptation of the evolving information in situ. The generated information refers to the observed phenomena that relate to dynamic unfolding of the search area (e.g., detection of unknown obstacles and boundaries) and environmental changes (e.g., spreading and drift of oil spills). Although such information can be obtained through remote sensing [7], it may not be always available due to communication constraints and high operational cost. Under these circumstances, the autonomous vehicle is required to scan all points while dynamically discovering new oil spills and avoiding obstacles at unknown locations. This is known as the Complete Coverage Prob-
A variety of algorithms exist in technical literature for coverage control using autonomous vehicles [9]. Although many prototypes of autonomous vehicles have been developed for oil spill cleaning, navigation algorithms for these prototypes are not adequately addressed [4], [5]. For example, several researchers (e.g., [3], [10]) have tested the algorithms for control of autonomous vehicles to perform oil cleaning tasks in dynamic and uncertain environments, where the locations of obstacles and oil spills are a priori unknown. From these perspectives, the operation of the oil spill cleaning process is modeled under the following assumptions.

1) After occurrence of oil spill at a physical location, the spillage stops before initiation of the cleanup task.
2) The location and volume of oil spill are unknown to the autonomous vehicle but the exact location of the vehicle is known through a localization system.
3) The autonomous vehicle uses mechanical cleanup to remove the oil at its current position.

### A. Oil Spill Modeling

Over 50 oil weathering models have been reported in literature. The 2D random-walk particle-tracking model has been adopted in this paper because the model is computationally tractable when simulating a large number of particles online and it predicts the time trajectories of the spill size and the probability distribution of the oil spill.

In the random-walk particle-tracking model, spilled oil consists of a large number of particles, with each particle representing a defined quantity of oil. Effectively, model particles are treated as “mass points”, with their transport determined by tidal currents, wind-driven current, turbulent eddies, gravitational spreading and buoyancy. The 2D update equations [11] for particle positions are given by

$$X^n = X^{n-1} + A(X^{n-1})\Delta t + B(X^{n-1})Z\sqrt{2\Delta t}$$  \hspace{1cm} (1)

where \(\Delta t\) is the time interval, \(X^n\) is the position at time \(n\Delta t\) (i.e., at the step number \(n\)), \(A\) is a forcing vector that models the drift process due to currents and wind, \(B\) is a deterministic scaling matrix, \(Z\) is a vector of two independent random numbers taken from a uniform distribution in the range [-1,1], and \(K\) is a vector of the turbulent coefficients. In this model, the motion of one particle is statistically independent of other particles. As seen in Eq. (1), the displacement of each particle is determined by its previous position, and the effects of drift and spreading. The effects of other weathering processes (e.g., evaporation, natural dispersion and emulsification) are not included in this model.

### B. Multi-Resolution Grid Formulation of the Search Area

The environment to be explored is considered to be a planar area populated with a finite but unknown number of obstacles. The obstacles may have arbitrary shapes and sizes and their exact locations are a priori unknown. The terrain limits are defined either by a hard boundary (e.g., a wall) or by a soft boundary (e.g., subarea of a larger field).

The search area is uniformly partitioned into cells to form a grid map, and a generalized Ising model [12] is constructed over the grid map, which involves a time-varying potential function term to control the movement of the autonomous vehicle in the search area. Let \(\Sigma \triangleq \{T, E, U, O\}\) be a finite set of symbols, which represents all possible states for each grid cell: i) explored and target present, ii) explored and target not present, iii) unexplored, and iv) explored and obstacle detected, respectively. The term target refers to oil spill in this paper.

The concept of multi-resolution navigation is introduced by the authors in [6] to partition the search area at various levels of resolutions and use the corresponding grid map for navigation according to the available spatio-temporal information. At each level, the search area is uniquely and exhaustively partitioned such that the information can be consistently stored by the autonomous vehicles.

Figure 1 shows the switch between the grid maps with different levels of resolution. In Fig. 1, the search area is partitioned at three levels of resolution. The grid map corresponding to level 0 has the finest resolution and the grid cells have the same size or slightly smaller than the autonomous vehicle. The grid cells continue to merge to form level 1 and level 2. The autonomous vehicle first operates in level 0 until no unexplored grid cell remains in its local neighborhood. Then global navigation with level 1 is implemented to find unexplored cells, and the autonomous vehicle moves toward the centroid of the cell that has the most unexplored fine grid cells. If no unexplored grid cells are found, then the autonomous vehicle continues to switch to the coarser level until unexplored cells are found. If no unexplored cells are found at the coarsest level, then the complete coverage task has been accomplished. This formulation avoids unnecessary global calculations and reduces the computational complexity in real-time implementation. The details of local navigation and global navigation are described in Section III.
III. Algorithms of Multi-resolution Navigation

The multi-resolution navigation algorithm is introduced in the authors’ recent work [6] for complete coverage of unknown environments. This section presents a succinct overview of the algorithm and introduces the new capability of dynamic adaptation to facilitate oil spill cleaning.

A. Generalized Ising Model for Local Navigation

A four-state generalized Ising model is constructed by extending the earlier work of Gupta et al. [12]. A local energy term \( E_\xi \) at a grid cell \( \xi \) is defined as:

\[
E_\xi(t) = \sum_{(\xi,\nu)_{\kappa_1}} J_{\xi\nu} \Psi(\gamma_\xi(t), \gamma_\nu(t)) + \Phi(B_\xi(t), \gamma_\xi(t))
\]  

(2)

where \( \langle \xi, \nu \rangle_{\kappa_1} \) implies summation over a \( \kappa_1 \)-neighborhood of \( \xi \), for some \( \kappa_1 \in \mathbb{N} \). The \( \kappa \)-neighborhood of a grid cell \( \xi \) is defined as

\[
\mathcal{N}_\kappa(\xi) = \{ \nu : max(|\xi_x - \nu_x|, |\xi_y - \nu_y|) \leq \kappa \},
\]  

(3)

where \( \xi_x, \xi_y \in \mathbb{N} \) and \( \nu_x, \nu_y \in \mathbb{N} \) denote the \( x \) and \( y \) coordinates of grid cells \( \xi \) and \( \nu \), respectively. The computation of Eq. (2) is carried out in \( \mathcal{N}_{\kappa_0}(\mu) \), where \( \mu \) is the grid cell occupied by the autonomous vehicle, and \( \kappa_0 \) is the distance between \( \mu \) to the boundary of local navigation area.

The first term in the right hand side of Eq. (2) defines the total interaction potential due to the sum of the effects of neighbors on the state at a grid cell \( \xi \). This term is called adaptation term because the effects of the observed states in the neighborhood cause changes in the resultant energy potential at a grid cell \( \xi \), which enables real-time adaptation in the navigation path trajectory. The coefficient \( J_{\xi\nu} \) denotes the interaction strength between two distinct grid cells \( \xi \) and \( \nu \). For \( \eta = \max(|\xi_x - \nu_x|, |\xi_y - \nu_y|) \), i.e., the distance between two neighborhood grid cells, \( J_{\xi\nu} \) is given as

\[
J_{\xi\nu} = \begin{cases} 
\eta^{-\alpha}, & \forall \xi \neq \nu \text{ and } \eta \in \{1, \ldots, \kappa_1\} \\
0, & \text{otherwise}
\end{cases}
\]  

(4)

where \( \alpha \in (0, \infty) \) is a control parameter. The (implicitly time-dependent) interaction function \( \Psi \) is defined as

\[
\Psi(\gamma_\xi(t), \gamma_\nu(t)) = \begin{cases} 
\psi_T, & \text{for } \gamma_\xi(t) = U, \gamma_\nu(t) = T \\
\psi_E, & \text{for } \gamma_\xi(t) = U, \gamma_\nu(t) = E \\
0, & \text{otherwise}
\end{cases}
\]  

(5)

where \( \psi \) defines the influence of a grid cell \( \nu \) on an unexplored grid cell \( \xi \).

The second term in the right hand side of Eq. (2) defines the navigation control function \( \Phi \) that depends on an exogenous time-varying potential field \( B_\xi(t) \) and the state \( \gamma_\xi(t) \) at a grid cell \( \xi \). The function \( \Phi \) is defined as:

\[
\Phi(B_\xi(t), \gamma_\xi(t)) = \begin{cases} 
\phi_T, & \text{for } \gamma_\xi(t) = T \\
\phi_E, & \text{for } \gamma_\xi(t) = E \\
\phi_O, & \text{for } \gamma_\xi(t) = O \\
B_\xi(t), & \text{for } \gamma_\xi(t) = U
\end{cases}
\]  

(6)

where the constants \( \phi_T \leq 0, \phi_E \leq 0 \) and \( \phi_O < 0 \) correspond to low-energy states of the explored grid cells in the presence (i.e., \( \gamma_\xi(t) = T \)) and absence (i.e., \( \gamma_\xi(t) = E \)) of a target and the presence of obstacle (i.e., \( \gamma_\xi(t) = O \)), respectively. The potential field \( B_\xi(t) \) defines the time-varying potential at unexplored grid cells (i.e., \( \gamma_\xi(t) = U \)) and is given as

\[
B_\xi(t) = B^*_\xi - C_{\xi,\mu}(t)
\]  

(7)

where \( B^*_\xi \) represents the constant potential field constructed to navigate the autonomous vehicle with no \textit{in situ} adaptation. The relative cost potential function \( C_{\xi,\mu}(t) \) defines the total decrease in potential at a grid cell \( \xi \) due to travel and turn costs that are incurred to reach the grid cell \( \xi \) from a current position with the grid cell coordinate \( \mu \) at time \( t \).

Therefore, Eq. (2) describes the total energy potential at a grid cell \( \xi \), which is the sum of: i) neighborhood interaction potential due to nearby target locations, and ii) a time-varying field that depends on an externally applied potential and the traveling and turning costs. The autonomous vehicle computes the values of the energy potentials \( E_\xi(t) \) for all \( \xi \in \mathcal{N}_{\kappa_0}(\mu) \), and sets the centroid of the grid cell \( \xi^*(t) \) that has the highest potential as the goal for local navigation:

\[
\xi^*(t) = \arg\max_{\xi \in \mathcal{N}_{\kappa_0}(\mu)} E_\xi(t)
\]  

(8)

B. Probability Vectors for Global Navigation

Global navigation is usually operated over large search area that involves a large number of the finest-level grid cells. Calculation of the energy terms for all the grid cells requires high computation power, especially when the search area is large. This may undermine the real-time implementation capability of the algorithm and its further application in the multi-agent cooperation. To resolve this problem, a light weight probability vector is used to store the environment information of each cell at the coarse levels.

Each coarse cell partition is assigned a probability vector that records the states of the fine cells located in it

\[
p_i = [\gamma_\xi(t) = T, |\gamma_\xi(t) = E|, |\gamma_\xi(t) = U|, |\gamma_\xi(t) = O|]^{T}
\]  

(9)

where \( |\gamma_\xi(t) = \sigma_j| \) with \( j = 1, 2, 3, 4 \) signifies the number of fine cells in that region with state \( \sigma_j \in \Sigma \), and \( p_i \) refers to the \( i \)th cell of the \( \ell \)th level. Evidently, the sum of the four elements of the vector is the number of cells in the region. Then, the probability of finding a cell with given state \( w \) in region \( i \) in level \( l \) is given by:

\[
p_i^l(\gamma_\xi(t) = w) = \frac{|\gamma_\xi(t) = w|}{\sum_{j=1}^{4} |\gamma_\xi(t) = \sigma_j|}
\]  

(10)

This probability vector is very lightweight and extremely easy to store. In global navigation, the probability vectors of all the grid cells at the current coarse level are calculated, and the one with highest probability of unexplored cells is set as the goal.
IV. VALIDATION ON A SIMULATION TEST-BED
This section presents validation of the multi-resolution algorithm for oil spill cleaning on a simulation test-bed. The test-bed is built upon the Player/Stage platform that is a high-fidelity open source robotic simulator [13].

A. Player/Stage Simulator
The oil spill is modeled as a 2D random walk process. The weathering processing of the oil spill is modeled by using a large number of oil particles in the simulation. The navigation algorithm and the oil spill simulation are implemented in separate subroutines such that the autonomous vehicle has no access to the distribution of the oil spill until it enters the grid cell that is occupied by oil particles.

The oil weathering process is simulated by following the model in Eq. (1), where the simulated scenario is an oil spill incident in a harbor. The oil spillage is assumed to stop before the cleanup process is initiated; however, the spill starts to spread and drift due to tidal current and wind, and would eventually hit the port if preventive actions are not taken. In the oil spill simulation, \( N = 5,000 \) oil particles, forcing vector \( \mathbf{A} = [2 \times 10^{-5}, 5 \times 10^{-5}]^T \), scaling matrix \( \mathbf{B} = [2 \times 10^{-2}, 0, 0, 5 \times 10^{-2}]^T \), and vector of turbulent coefficients \( \mathbf{K} = [0.5, 0.5]^T \) are used. The vector \( \mathbf{Z} \) of two independent random numbers is obtained from a uniform distribution with range \([-1, 1]\). Each time step approximately corresponds to \( \Delta t = 0.3 \) sec in real time.

A Pioneer 2AT robot is modeled in the simulator. Although we use Pioneer robot for validation, the navigation algorithm presented in the paper is meant to be generic and not platform dependent. A 30m \( \times \) 30m harbor map has been used to test the performance of the proposed algorithms. The size of the grid cell at the finest level is \( 1m \times 1m \), which is slightly larger than the size of the Pioneer 2AT robot. The selection of the starting point of the operation depends on the direction of tidal current and wind. In this simulation exercise, the starting point is the bottom left corner because the wind and tidal current make the oil spill drift toward the harbor.

Since the initial location of the oil spill is unknown to the autonomous vehicle and both the size and the location of the oil spill change with time due to spreading and drift, the autonomous vehicle needs to cover the entire search area to assure complete cleanup of the oil particles. The typical back and forth motion is optimal for searching an area in terms of minimum number of turns when no adaptation to target and obstacle avoidance is needed. Therefore, for area coverage planning, the exogenous potential field \( B^x \) in Eq. (7) is designed for back and forth motion such that the potential field has a decreasing magnitude from column to column, starting from a maximum value of magnitude 10,000 at the start point, while having equipotential grid cells on each column. The other parameters in Eq. (2) have been selected to be \( \kappa_0 = 3, \kappa_1 = 2, \phi_T = 0, \phi_V = 0, \phi_O = -40,000, \psi_T = 2,000, \psi_E = 0, \) and \( \alpha = 0.8 \) in the simulation exercises; however, the specific values of these parameters do not have very significant effects on the algorithm performance as long as they are of the same relative order of magnitude.

In order to comparatively evaluate the performance of the proposed multi-resolution navigation algorithm, a benchmark algorithm is also tested in the same scenario. In the benchmark algorithm, the autonomous vehicle implements back and forth motion and avoids obstacles as needed. When no unexplored grid cells are in its local neighborhood, the vehicle moves toward the direction of the most unexplored grid cells. In essence, the benchmark algorithm is a simplified version of the multi-resolution navigation algorithm without the adaptation term as shown in Eq. (2). Without the adaptation term, the benchmark algorithm is not affected by the detection of the oil spill in the current grid cell and thus does not deviate from the back and forth motion to search the neighborhood of current grid cell for oil spills.

B. Performance Metrics
Three different performance metrics are used to compare the effectiveness of the proposed algorithm with that of the benchmark algorithm in oil spill cleaning:

- \( T_{\text{total}} \): total time to cover the entire search area
- \( T_{\text{clean}} \): time to clean up all the oil spills
- \( \text{AUC} \): area under the curve in oil spill cleaning profile

The algorithm that has smaller \( T_{\text{total}} \) and \( T_{\text{clean}} \) is considered to be more effective. In the case that these two metrics contradict with each other, the algorithm with smaller \( T_{\text{clean}} \) is preferred because the major task in this application is to clean up oil spills in the shortest time.

To keep track of the history of oil spill cleaning, a profile is generated to record the remaining oil spill at each time step. The metric Area Under the Curve (AUC) has been used to quantitatively compare different profiles. AUC emphasizes the efficiency of cleaning oil spills. The algorithm with smaller AUC is considered to be more effective.

The performance metrics \( T_{\text{clean}} \) and AUC are related. The cleanup time \( T_{\text{clean}} \) shows how fast the algorithm is able to finish cleaning all oil spills, and the AUC takes the remaining oil spill at each time step into consideration.

C. Simulation Results
This simulation exercise aims to validate the proposed algorithms for a real-life oil spill cleaning scenario, and compare their performance with the benchmark algorithm. A map with the layout of a typical harbor is designed. The harbor consists of several ports and has one entrance that connects to the open sea, as shown in the top row of Fig. 2. The initial oil spill is located at the entrance, probably due to a maritime accident outside the harbor or drifting of the oil spill from the site of an offshore platform accident.\(^1\)

\(^1\)A video of this simulation exercise is available at the following URL \( \text{http://goo.gl/R0SXw} \) for downloading or viewing online.
The multi-resolution algorithm is used to navigate an autonomous vehicle to clean up the oil spill. Four snapshot are taken and shown in Fig. 2(a)-(d). The top row shows the layout of the harbor, the oil spill, and the trajectory of the autonomous vehicle, while the bottom row shows the environment map in the vehicle’s onboard memory. Figure 2(a) shows the vehicle follows the offline plan and implements back and forth motion to explore the search area. Figure 2(b) shows the vehicle deviates from the offline plan once it detects the oil spill and explores the neighborhood of the grid cells where oil spill is detected. Figure 2(c) shows the moment when the vehicle successfully cleans up all the oil spills in the search area, and Fig. 2(d) shows the exploration of the remaining search area.

In the environment maps, the yellow box around the search area in Fig. 2(a) is the buffer specified to prevent the vehicle from leaving the search area. As the exploration goes on, the vehicle detects the jetty and the port, which are marked...
in dark green in the environment map. Upon detection of the jetty and the port, the buffer is automatically labeled around them to keep a safe distance and avoid collision. The grid cells where oil spill is discovered are labeled in a different color, and the vehicle spends more time to search the neighboring area of the cells. As seen in Fig. 2(d), the complete perimeter of the harbor is mapped in dark green.

The benchmark algorithm is also implemented in the same map to compare with the proposed algorithm, and the results are shown in Fig. 3. The only difference between Fig. 2 and Fig. 3 is that the autonomous vehicle does not adapt to the neighborhood of the grid cells with oil spill detected. Instead, it follows the offline plan and implements the back and forth motion until it covers the entire search area. As a result, the trajectory recorded in Fig. 3 is neater than the one in Fig. 2.

The profiles of oil spill cleaning are shown in Fig. 4. The y-axis in Fig. 4 indicates the normalized number of the remaining oil particles, and the x-axis is the time steps in the simulation. The proposed method takes much less time in cleaning up all the oil spills, although it takes slightly more time to finish scanning the entire search area. This is understandable since in the proposed method the autonomous vehicle deviates from the offline plan and exploits the neighboring areas when it detects oil spills.

The proposed method and the benchmark method are comparatively evaluated using the performance metrics, and the results are shown in Table I. Although it takes 11.1% more time to finish scanning the entire search area, the proposed method is much more efficient in oil spill cleaning and takes 16.6% less time to clean up all the oil spills. As indicated by $T_{\text{clean}}$ and AUC, the proposed method significantly reduces the impact of the oil spill to the environment by dynamically adapting to the oil spill and replanning online.

### Table I

<table>
<thead>
<tr>
<th>Method</th>
<th>$T_{\text{total}}$</th>
<th>$T_{\text{clean}}$</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark</td>
<td>9783</td>
<td>3387</td>
<td>2578</td>
</tr>
<tr>
<td>Proposed</td>
<td>10865</td>
<td>2826</td>
<td>1618</td>
</tr>
<tr>
<td>Improvement</td>
<td>$-11.1%$</td>
<td>$16.6%$</td>
<td>$37.2%$</td>
</tr>
</tbody>
</table>

V. SUMMARY, CONCLUSIONS AND FUTURE WORK

This paper presents a multi-resolution navigation algorithm for oil spill cleaning in dynamic and uncertain environments using autonomous vehicles. The concepts of local and global navigation are integrated for adaptive decision making according to the available spatio-temporal information. The local navigation provides a reduced computational complexity in local decision-making while the global navigation is organized in a hierarchical manner to prevent the robot from being stuck into a local minima. Dynamic adaptation significantly improves the cleaning efficiency and reduces the impact of the oil spill to the environments.

The proposed algorithm has been validated in a harbor example. With the multi-resolution navigation algorithm, the autonomous vehicle manages to explore the complex and unknown environment, and cleans up all the oil spills in a timely manner. Compared to the benchmark algorithm that uses back and forth motion and obstacle avoidance algorithm, the proposed multi-resolution algorithm is more efficient in oil spill cleaning and significantly reduces the impact of the oil spill to the environment.

While there are many research issues that need to be resolved before exploring commercial applications of the proposed algorithms, the following topics are under active research:

- Validation of the algorithm on hardware platform;
- Extension of the algorithm for multi-agent cooperation.

### REFERENCES


