Subplanner Algorithm to Escape from Local Minima for Artificial Potential Function Based Robotic Path Planning

Han Ul Yoon¹ and Dong-Wook Lee¹,²
¹Robotics R&D Group, Korea Institute of Industrial Technology, Ansan, Korea
²Department of Robotics and Virtual Engineering, Korea University of Science and Technology, Daejon, Korea

Abstract
This paper proposes a subplanner algorithm, which will be referred to as attractive force rotation and restoration, to compensate the local minima problem under the artificial potential based planner. The key role of the proposed subplanner is to build a navigating path for escaping from the local minima by rotating and restoring the attractive force. A mobile manipulator was adopted as our robotic application to substantiate the capability of the proposed subplanner under various simulation scenarios in which the mobile manipulator finds a path to a target object without collision against stationary/moving obstacles. The simulation results substantiated that the compensated artificial potential based planner successfully found a path to guide the mobile manipulator to the target object in the presence of the local minima.

Keywords: Path planning, Mobile manipulator, Artificial potential based planner, Attractive force rotation and restoration algorithm

1. Introduction
A robotic path planning problem is to find a path that is a continuous mapping from an initial configuration to a goal configuration with a given time interval [1]. Accordingly, solving a path planning problem means that we want to find a continuous curve which lies in free configuration space and connects the initial configuration and the goal configuration [2, 3]. This problem is often solved by a combination of computational geometry, heuristic search algorithms, and game-theoretic decision-making [1, 4, 5]. The applications of the robotic path planning have been explored in a broad range of field such as mobile robots [6-8], aerial robots [9, 10], fork-type lifters [11, 12], humanoid robot [13, 14], etc.

In path planning, a planner normally refers to a method or an algorithm connecting one configuration to the other configuration. The most challenging aspect of a planner design is that a configuration space usually has a high dimensionality. The high dimensional complexity in the configuration space does not allow us to find a solution by using insights from the explicit geometric representation of the free configuration space [2, 15]. Due to the curse of dimensionality, path planning in the high dimensional configuration space usually yields problematic issues, e.g., narrow passage problem [16]. Canny and Reif proved that the problem of designing a path planning algorithm in high dimensional configuration space becomes NP-hard [17].
Artificial potential based planner (APBP) is a compact but strong way to design a planner. Generally, the APBP consists of attractive potential and repulsive potential functions; both potential functions are differentiable real-valued function. These two potential functions generate a force field in which a current configuration is forced to be attracted by a goal configuration and repulsed by non-free configurations \[2\-3\]. One advantage of the APBP is that it eventually guides an initial configuration to a goal configuration. The other advantage is that the two potential functions as well as the corresponding force field can be defined on a workspace as a function of the control points of a robotic system, which in turn, give us geometrical insights about the movement of the control points. However, the critical drawback of the APBP is that it fails to find a path if a configuration falls into local minima (at where the force field is vanished) \[18\].

To mitigate the weak point of the APBP, we propose a sub-planner algorithm to escape from the local minima while a path is being planned for a robotic system under the APBP. The proposed subplanner algorithm partitions the configuration space of the robotic system based on the kinematic features and detects the local minima in the partitioned configuration spaces. Then, it plans a navigating path by manipulating the direction of the attractive force to escape from the local minima. Compared to existing works presented in \[19\-21\] as current state-of-the-art, our approach initiates the use of kinematic feature of the robotic system and deals with the higher dimensional configuration space.

To explain our approach, throughout this paper, we consider the path planning problem of a specific robotic application in which a mobile manipulator robot is instructed to reach at and touch a target object by its end effector while navigating through an area where multiple moving and stationary obstacles are deployed. The organization of this paper is as follows: in Section 2, the APBP to solve a robotic path planning problem is introduced. In Section 3, we discuss the proposed subplanner algorithm to escape from local minima. Section 4 presents simulation results to substantiate the capability of the proposed subplanner. Section 5 will be the conclusion of this paper.

2. Artificial Potential Based Path Planning for a Mobile Manipulator

2.1 A Mobile Manipulator and its Configuration Space

Figure 1 shows a mobile manipulator model that will be considered as our application throughout this paper. The mobile manipulator consists of two main parts: a mobile base part and four-links (three joints) elbow manipulator part. The base frame of the manipulator is attached on the top-center of the mobile base; hence, the configuration of the mobile manipulator, denoted by \(q\), is

\[
q = [x, y, \theta_1, \theta_2, \theta_3, \theta_4]^T,
\]

where \([x, y, \theta_1]^T\) corresponds to the position and orientation of the mobile base, and \([\theta_2, \theta_3, \theta_4]^T\) represents the rotations of manipulator joint. Accordingly, the configuration space of the mobile manipulator is

\[
Q \subseteq \mathbb{R} \times \mathbb{R} \times S \times S \times S \times S = \mathbb{R}^2 \times T^4.
\]

2.2 A Mobile Manipulator Path Planning: Problem Definition

Let \(q, \ Q, \ Q_{\text{free}}, \) and \(Q_{\text{obs}}\) be a configuration, a configuration space, a free configuration space, and an obstacle-occupied configuration space, respectively. For our mobile manipulator, given an initial time \(t_0\) and a final time \(t_f\), we want to find a path \(\gamma\) that is a continuous mapping

\[
\gamma : [t_0, t_f] \rightarrow Q,
\]

which satisfies

\[
\gamma(s) \in Q_{\text{free}}, \forall s \in [t_0, t_f] \quad \text{and} \quad \gamma(t_0) = q_{\text{init}}, \quad \gamma(t_f) = q_{\text{goal}},
\]

where \(Q_{\text{free}} = Q - Q_{\text{obs}}\). Note that the role of the APBP is to assign a free configuration \(q\) to \(\gamma(s)\); thus

\[
\gamma(s) \leftarrow q \quad \text{and} \quad q \in Q_{\text{free}}.
\]
2.3 Designing an Artificial Potential Based Planner

2.3.1 Setting the control points of a mobile manipulator in a workspace

A term “control points” refers to the points of our mobile manipulator on where the artificial potentials have an effect. To set the control points, we start with defining $p_i(q)$ that maps a given configuration $q$ in a configuration space $Q$ onto a three dimensional position vector $p_i$ in a workspace $W$:

$$p_i(q) : \mathbb{R}^2 \times T^1 \rightarrow \mathbb{R}^3. \quad (5)$$

Now, we want to set five control points on the mobile manipulator as follows: i) the origin of the mobile base frame, ii) the origin of the joint#1, iii) the origin of the joint#2, iv) the origin of the joint#3, and v) the origin of the end effector. Given $q$, we can specify $p_1(q)$ through $p_5(q)$ with the Denavit-Hartenberg (DH) parameters of the mobile manipulator (see Figure 1).

$$p_1(q) = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}, \quad p_2(q) = \begin{bmatrix} x \\ y \\ d_1 \end{bmatrix},$$

$$p_3(q) = \begin{bmatrix} x + a_2 c_1 c_2 \\ y + a_2 s_1 c_2 \\ a_2 s_2 + d_1 \end{bmatrix},$$

$$p_4(q) = \begin{bmatrix} x + a_3 c_1 c_2 c_3 - a_3 c_1 s_2 s_3 + a_2 c_1 c_2 \\ y + a_3 s_1 c_2 c_3 - a_3 s_1 s_2 s_3 + a_2 s_1 c_2 \\ a_3 s_2 c_3 + a_3 c_2 s_3 + a_2 s_2 + d_1 \end{bmatrix},$$

$$p_5(q) = \begin{bmatrix} \left( x + a_4 c_1 c_2 c_3 c_4 - a_4 c_1 c_2 s_3 s_4 + a_3 c_1 c_2 c_3 \\ - a_4 c_1 s_2 s_3 c_4 - a_4 c_1 s_2 s_3 s_4 - a_3 c_1 s_2 s_3 \\ + a_2 c_1 c_2 \right) \\ \left( y + a_4 s_1 c_2 c_3 c_4 - a_4 s_1 c_2 s_3 s_4 + a_3 s_1 c_2 c_3 \\ - a_4 s_1 s_2 s_3 c_4 - a_4 s_1 s_2 s_3 s_4 - a_3 s_1 s_2 s_3 \\ + a_2 s_1 c_2 \right) \\ \left( a_4 s_2 c_3 c_4 - a_4 s_2 s_3 s_4 + a_3 s_2 c_3 + a_4 c_2 s_3 c_4 \\ + a_4 c_2 s_3 s_4 + a_3 c_2 s_3 + a_2 s_2 + d_1 \right) \end{bmatrix},$$

where $c_i$ and $s_i$ are shorthand notations for $\cos \theta_i$ and $\sin \theta_i$, respectively.

2.3.2 Attractive potential in workspace

The role of attractive potential is to attract a current control point $p_i(q)$ to a goal control point $p_i(q_{goal})$ by yielding a vector field in a workspace $W$. The most common choice of attractive potential function is the combination of conic potential and quadratic potential for two conditions stating whether $p_i(q)$ is close to $p_i(q_{goal})$ or not $[1, 2]$. Let $\parallel \cdot \parallel$ denote $L_2$-norm. We first define the two conditions as

$$\parallel p_i(q) - p_i(q_{goal}) \parallel > \epsilon, \quad (c1)$$

$$\parallel p_i(q) - p_i(q_{goal}) \parallel \leq \epsilon, \quad (c2)$$

where $\epsilon$ is a threshold distance from $p_i(q_{goal})$ where a planner switches between conic and quadratic potentials. Now, the attractive potential function $U_{att,i}$ can be defined by

$$U_{att,i}(p_i(q)) = \begin{cases} \epsilon \parallel p_i(q) - p_i(q_{goal}) \parallel - \frac{1}{2} \epsilon \epsilon^2, & \text{if } (c1), \\ \frac{1}{2} \epsilon \parallel p_i(q) - p_i(q_{goal}) \parallel^2, & \text{if } (c2) \end{cases} \quad (7)$$

where a scaling factor $\epsilon$ is a small positive constant.

The corresponding attractive force field $F_{att,i}$ can be yielded by taking a partial derivative with respect to $p_i(q)$ and negation

$$F_{att,i}(p_i(q)) = -\nabla_{p_i(q)} U_{att,i}(p_i(q))$$

$$= \begin{cases} -\epsilon \parallel p_i(q) - p_i(q_{goal}) \parallel, & \text{if } (c1), \\ -\epsilon \parallel p_i(q) - p_i(q_{goal}) \parallel, & \text{if } (c2) \end{cases} \quad (8)$$

which converges to a zero vector as $p_i(q)$ approaches to $p_i(q_{goal})$ by switching $U_{att,i}$ in Eq. (7) from the conic potential (c1) to the quadratic potential (c2). Figure 2 presents the example of the attractive potential and the corresponding attractive force field in $\mathbb{R}^2$.

2.3.3 Repulsive potential in workspace

The role of repulsive potential is to repulse a current control point $p_i(q)$ when it approaches toward any $j^{th}$ obstacle in a workspace $W$. Let $o_j$ denote a point on the $j^{th}$ obstacle that
where a scaling factor yields the shortest distance from \( p_i(q) \). We first define two conditions stating whether \( p_i(q) \) is within \( \delta \) boundary from the shortest point of the \( j \)th obstacle

\[
\| p_i(q) - o_j \| < \delta, \quad \text{(c3)} \\
\| p_i(q) - o_j \| \geq \delta. \quad \text{(c4)}
\]

Now, the repulsive potential function \( U_{\text{rep},i,j} \) can be defined as

\[
U_{\text{rep},i,j}(p_i(q)) = \begin{cases} 
\eta \left( \frac{1}{\| p_i(q) - o_j \|} - \frac{1}{\delta} \right)^2, & \text{if (c3)}, \\
0, & \text{if (c4)},
\end{cases}
\]

where a scaling factor \( \eta \) is a positive constant. The corresponding repulsive force \( F_{\text{rep},i,j} \) is equal to

\[
F_{\text{rep},i,j}(p_i(q)) = -\nabla p_i(q) U_{\text{rep},i,j}(p_i(q)) = \begin{cases} 
\eta \left( \frac{1}{\| p_i(q) - o_j \|} - \frac{1}{\delta} \right) \frac{1}{\| p_i(q) - o_j \|^2} \frac{(p_i(q) - o_j)}{\| p_i(q) - o_j \|}, & \text{if (c3)}, \\
0, & \text{if (c4)},
\end{cases}
\]

Figure 3 shows the example of the repulsive potential and the corresponding repulsive force field in \( \mathbb{R}^2 \).

2.3.4 Building a path in configuration space by using APBP

To map two artificial forces, \( F_{\text{att},i} \) and \( F_{\text{rep},i,j} \), in a workspace \( \mathcal{W} \) onto a force \( F_\mathcal{Q} \) in a configuration space \( \mathcal{Q} \), we define Jacobian \( J_i(q) \in \mathbb{R}^{3 \times 6} \) for \( i \)th control points \( p_i(q) \)

\[
J_i(q) = \frac{\partial p_i(q)}{\partial q}, \quad (11)
\]

for example

\[
J_3(q) = \frac{\partial p_3(q)}{\partial q} = \begin{bmatrix} 1 & 0 & -a_2 s_1 c_2 & -a_2 c_1 s_2 & 0 & 0 \\
0 & 1 & a_2 c_1 c_2 & -a_2 s_1 s_2 & 0 & 0 \\
0 & 0 & 0 & a_2 c_2 & 0 & 0 \end{bmatrix}.
\]

Now, \( F_{\text{att},i} \) and \( F_{\text{rep},i,j} \) can be mapped onto \( F_\mathcal{Q} \) via \( J_{i,t}^T(q) \)

\[
F_\mathcal{Q}(q) = \sum_i J_{i,t}^T(q) F_{\text{att},i}(p_i(q)) + \sum_j \sum_i J_{i,t}^T(q) F_{\text{rep},i,j}(p_i(q)), \quad (12)
\]

which plans a next configuration \( q' \) at a current configuration \( q \) by

\[
q' \leftarrow q + F_\mathcal{Q}(q). \quad (13)
\]

To build a path \( \gamma \), the next configuration is assigned to the path for a time stamp \( s + \Delta s \)

\[
s' \leftarrow s + \Delta s, \quad \gamma(s') \leftarrow q', \quad (14)
\]

then we update the time stamp, the current configuration, and the control points by

\[
s \leftarrow s', \quad q \leftarrow q', \quad p_i(q) \leftarrow p_i(q'). \quad (15)
\]

By repeating a procedure Eq. (12) through Eq. (15) until \( q \) is equal to \( q_{\text{goal}} \), we can build a path \( \gamma(s) \) for \( s \in [t_0, t_f] \) under the APBP (of course, Eq. (12) requires to calculate Eq. (9) through Eq. (11)). The mobile manipulator path planning algorithm by using the APBP is summarized in Table I.

3. A Method to Escape from Local Minima

3.1 The Existence of Local Minima under the APBP

From Eq. (12), we have seen that \( F_{\text{att},i} \) and \( F_{\text{rep},i,j} \) can be mapped onto \( F_\mathcal{Q} \) via \( J_{i,t}^T(q) \). By rearranging the right side of Eq. (12), we can obtain

\[
F_\mathcal{Q}(q) = J_{i,t}^T(q) \left( \sum_i F_{\text{att},i}(p_i(q)) + \sum_j \sum_i F_{\text{rep},i,j}(p_i(q)) \right), \quad (16)
\]
Accordingly, the all control points of the mobile manipulator implies that if there exists a configuration $q$ such that

$$\sum_i F_{\text{att},i}(p_i(q)) + \sum_j \sum_i F_{\text{rep},i,j}(p_i(q)) = [0, 0, 0, 0]^T,$$  \hspace{1cm} (17)

then

$$F_Q(q) = [0, 0, 0, 0, 0]^T.$$ \hspace{1cm} (18)

By recalling Eq. (13), we can know that a next configuration $q'$ will be the same as $q$. In other words, a path $\gamma$ is stopped at the configuration $q$ which means that the APBP fails to build a path. Accordingly, the all control points of the mobile manipulator $p_i(q)$ are trapped in one pose. We refer to these problematic configurations as local minima.

### 3.2 Detecting Local Minima in Partitioned Configuration Spaces

Suppose that we give a task for a mobile manipulator to grasp an object on the table by its end effector (gripper). If the mobile base part cannot approach to the target object or the elbow manipulator part get stuck at some point, the mobile manipulator certainly fails to complete the given task. In this sense, if positional variations in the mobile base part or/and the elbow manipulator part related to $F_Q(q)$ in Eq. (16) are vanished as illustrated in Figure 4, then our mobile manipulator will never reach to a goal configuration. Therefore, we focus on detecting local minima caused by the mobile base part and the elbow manipulator part to investigate whether they are trapped for long iterations, respectively.

To detect those local minima, we first partitions a configuration space $Q \subseteq \mathbb{R}^2 \times \mathbb{T}^4$ into two parts: $Q_m$ includes the configuration of the mobile base $q_m = [x, y]^T \in \mathbb{R}^2$ and $Q_e$ consists of the configuration of the elbow manipulator $q_e = [\theta_1, \theta_2, \theta_3, \theta_4]^T \in \mathbb{T}^4$. Then, the two types of local minima caused by the mobile base part and/or the elbow manipulator part can be detected easily by investigating the following two conditions. For the local minima caused by the mobile base part, we first project a current configuration $q$ and a goal configuration $q_{\text{goal}}$ in $Q$ onto the partitioned configuration space $Q_m$; thus,

$$q = [x, y, \theta_1, \theta_2, \theta_3, \theta_4]^T \rightarrow \bar{q} = [x, y, 0, 0, 0, 0]^T,$$

$$q_{\text{goal}} = [x^*, y^*, \theta^*_1, \theta^*_2, \theta^*_3, \theta^*_4]^T \rightarrow \bar{q}_{\text{goal}} = [x^*, y^*, 0, 0, 0, 0]^T.$$

Then, we calculate the summation of the attractive force and the repulsive forces and investigate whether

$$F_{\text{att},1}(p_1(\bar{q})) + \sum_j F_{\text{rep},1,j}(p_j(\bar{q})) \rightarrow [0, 0, 0]^T.$$ \hspace{1cm} (19)

If Eq. (19) converges to $[0, 0, 0]^T$, then it implies that the mobile base part falls into local minima even in the absence of the elbow manipulator part. Next, $q$ and $q_{\text{goal}}$ in $Q$ is projected onto $Q_e$ to detect the local minima caused by the elbow manipulator part.
We call this subplanner algorithm as an attractive force rotation (AFRO) which means that the elbow manipulator part itself eventually traps in the local minima even if the position of the mobile base part is fixed.

3.3 Subplanner Algorithm to Escape from Local Minima: Attractive Force Rotation and Restoration

The existence of local minima is critical issue when we employ the APBP to solve a path planning problem. In general, it is even difficult to visualize or infer the location of local minima in high dimensional configuration space. Fortunately, we could characterize the two types of local minima, which belong to a mobile base and an end effector, by utilizing insights from the kinematic feature of our mobile manipulator.

When a path is planned under the APBP, a current configuration is attracted to a goal configuration by an attractive force \( F_{\text{att}} \). Therefore, if we can design a subplanner algorithm which makes a mobile manipulator to escape from local minima by temporarily changing and restoring the direction of \( F_{\text{att}} \) until the positions of the mobile base and the end effector reaches at the point being apart from any obstacles with a safety margin. We call this subplanner algorithm as an attractive force rotation and restoration (AFRORE) algorithm. The algorithmic foundation of the AFRORE algorithm is introduced below by two processes: attractive force rotation (AFRO) and attractive force restoration (AFRE), respectively.

3.3.1 Attractive force rotation

To explain the AFRO, we exemplify a case of which a mobile base part is trapped by a local minimum to show the AFRO illustratively as in Figure 5. For clarity, let us start with defining shorthand notations

\[
F_{\text{att}}, i (p_i(q)) := F_{\text{att}}, \\
F_{\text{rep}}, i, j (p_i(q)) := F_{\text{rep}}, \\
\text{and } F_{\text{sum}} = F_{\text{att}} + F_{\text{rep}}.
\]

At a local minimum, \( F_{\text{att}} \) is equal to \( F_{\text{rep}} \); hence \( F_{\text{sum}} \) becomes \([0, 0, 0]^T\).

When the mobile manipulator is trapped into the local minimum, we can identify that this local minimum is due to a mobile base part by investigating Eq. (19). A key idea of the AFRO is to keep rotating \( F_{\text{att}} \) to one direction, say clockwise by \( \theta_R \) (note: \( \theta_R \) is negative value) along the \( z \)-axis of a current local frame of the mobile manipulator for an iteration \( s + \Delta s \leftarrow s \)

\[
F_{\text{att}} |_{s + \Delta s} \leftarrow \begin{bmatrix} c_{\theta_R} & -s_{\theta_R} & 0 \\ s_{\theta_R} & c_{\theta_R} & 0 \\ 0 & 0 & 1 \end{bmatrix} F_{\text{att}} |_s,
\]

until \( F_{\text{sum}} \) guides the control point \( p_1(q) \) of the mobile manipulator to the point lying on a \( \delta \)-boundary from an obstacle at where \( F_{\text{rep}} \) vanishes. Figure 5 illustrate the AFRO process for the local minimum caused by the mobile base part.

Similarly, when the elbow manipulator part falls into a local minimum, which can be identified by investigating Eq. (20), the AFRO keeps rotating \( F_{\text{att}} \) by \( \theta_L \) along the \( y \)-axis of a current local frame of the mobile manipulator

\[
F_{\text{att}} |_{s + \Delta s} \leftarrow \begin{bmatrix} c_{\theta_L} & -s_{\theta_L} & 0 \\ s_{\theta_L} & c_{\theta_L} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{\theta_L} & 0 & s_{\theta_L} \\ 0 & 1 & 0 \\ -s_{\theta_L} & 0 & c_{\theta_L} \end{bmatrix} F_{\text{att}} |_s,
\]

until \( F_{\text{sum}} \) leads the control point \( p_2(q) \) through \( p_3(q) \) of the mobile manipulator to the point lying on the \( \delta \)-boundary from the obstacle.

Under the AFRO process, the generated \( F_{\text{sum}} \) allows both the mobile base part and the elbow manipulator part to contribute to escape local minima. For instance, when Eq. (22) is processed to escape from local minima related to off-ground obstacles and the elbow manipulator part, the generated \( F_{\text{sum}} \) acts on all five control points; hence, the mobile base part is also pushed away from the off-ground obstacles. This advantageous characteristic of the AFRO helps and ensures the mobile manipulator to escape from the local minima caused by the mobile base part or/and the elbow manipulator part. The AFRO is followed by
Figure 5. Attractive force rotation (AFRO) process: (a) the mobile base part of a mobile manipulator falls in a local minimum, $F_{\text{sum}} = 0$, (b) the subplanner keeps rotating $F_{\text{att}}$ clockwise by $\theta_R$, (c) $F_{\text{att}}$ is resurrected, and the mobile manipulator moves along with it, and (d) the mobile manipulator has escaped from a local minimum.

an AFRE process introduced below.

### 3.3.2 Attractive force restoration

Suppose that $p_1(q)$ of the mobile base reaches at a point lying on $\delta$-boundary from the closest obstacle by the AFRO as shown in Figure 5. At this point, if the mobile base traces the circumference of a circle $R$ of which radius is larger than $\delta$ toward counter-clockwise direction, then the mobile base never goes into $\delta$-boundary. Therefore, a key idea of AFRE process is that we restore $F_{\text{att}}$ by keep rotating it by $\theta_L (= -\theta_R)$, along the $z$-axis of a current local frame of the mobile manipulator

$$F_{\text{att}}|_{s+\Delta s} \leftarrow \begin{bmatrix} c_{\theta_L} & -s_{\theta_L} & 0 \\ s_{\theta_L} & c_{\theta_L} & 0 \\ 0 & 0 & 1 \end{bmatrix} F_{\text{att}}|_s. \quad (23)$$

Figure 6 illustrates the AFRE process in which the mobile base follows a circular trajectory lying on the circumference of a circle with a radius $R$.

The radius $R$ determines the curvature of the circular trajectory and can be defined as a function of $\theta_L$. From Eq. (8) and Eq. (10), we know that $F_{\text{sum}}$ becomes a vector of which size is $\|\epsilon\zeta\|$. AS shown in Figure 6(a), if Eq. (23) is executed $2\pi/\theta_L$ times while the mobile base is being moved by $\|\epsilon\zeta\|$, then the circumference is completely traced. Therefore, we can calculate $R$ by

$$\|\epsilon\zeta\| \cdot \frac{2\pi}{\theta_L} = 2\pi R \quad \Rightarrow \quad R = \frac{\|\epsilon\zeta\|}{\theta_L}. \quad (24)$$

During the AFRE process, if a distance between $p_1(q)$ and the closest obstacle becomes greater than $\alpha\delta$, where $\alpha$ is a positive constant to set up a safety margin, then the AFRE is terminated; finally, a path planner is switched to the APBP as the mobile manipulator has successfully escaped from the local minima.

Similarly, when the local minima is caused by the elbow manipulator part, the AFRE is performed along the $y$-axis of a current local frame of the mobile manipulator

$$F_{\text{att}}|_{s+\Delta s} \leftarrow \begin{bmatrix} c_{\theta_L} & s_{\theta_L} & 0 \\ -s_{\theta_L} & c_{\theta_L} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{\theta_R} & 0 & s_{\theta_R} \\ 0 & 1 & 0 \\ -s_{\theta_R} & 0 & c_{\theta_R} \end{bmatrix} F_{\text{att}}|_s. \quad (25)$$

If the shortest distance from any control points of the elbow manipulator part, $p_2(q)$ through $p_5(q)$, to the closest obstacle is greater than $\alpha\delta$, then a path planner is switched to the APBP.

Table 2 presents the AFRORE algorithm. The APBP for the mobile manipulator path planning with the subplanner AFRORE for escaping from local minima is summarized in Table 3.
Table 2. Subplanner AFRORE for escaping from local minima

| Function: AFRORE(localmin, type) |
|----------------------------------|
| / * Atractive Force Rotation */ |
| WHILE localmin |
| SWITCH type |
| case $Q_m$ |
| • $F_{at} | s + \Delta \leftarrow Rot_z(\theta_1) \cdot F_{at} | s$ | Eq. (21) |
| • Do Eq. (12), Eq. (13), Eq. (14), and Eq. (15) |
| • checkpoint $\leftarrow p_1(q)$ |
| break |
| case $Q_e$ |
| • $F_{at} | s + \Delta \leftarrow Rot_z(\theta_1) \cdot Rot_z(\theta_R) \cdot F_{at} | s$ | Eq. (22) |
| • Do Eq. (12), Eq. (13), Eq. (14), and Eq. (15) |
| • checkpoint $\leftarrow \{p_2(q), p_3(q), p_4(q), p_5(q)\}$ |
| break |
| END |
| IF $\min ||checkpoint - o_j|| \geq \delta$ |
| localmin $\leftarrow$ false |
| break |
| ELSE |
| repeat WHILE |
| END |

/* Atractive Force Restoration */

WHILE $\min ||checkpoint - o_j|| \leq \alpha \delta$

SWITCH type

case $Q_m$

• $F_{at} | s + \Delta \leftarrow Rot_z(\theta_1) \cdot F_{at} | s$ | Eq. (23) |
• Do Eq. (12), Eq. (13), Eq. (14), and Eq. (15) |
• checkpoint $\leftarrow p_1(q)$ |
break

case $Q_e$

• $F_{at} | s + \Delta \leftarrow Rot_z(\theta_1) \cdot Rot_z(\theta_R) \cdot F_{at} | s$ | Eq. (25) |
• Do Eq. (12), Eq. (13), Eq. (14), and Eq. (15) |
• checkpoint $\leftarrow \{p_2(q), p_3(q), p_4(q), p_5(q)\}$ |
break
END

IF $\min ||checkpoint - o_j|| \geq \alpha \delta$
break
ELSE
repeat WHILE
END

Table 3. APBP for a mobile manipulator path planning with subplanner AFRORE for escaping from local minima

| Function: Atractive Potential Based Planner |
|------------------------------------------|
| Given $q_{init}$, $q_{goal}$, localmin and $t_0$ |
| • Initialize: $q \leftarrow q_{init}$, $\gamma(t_0) \leftarrow q_{init}$, localmin $\leftarrow$ false |

WHILE $q$ is not $q_{goal}$

• Calculate $F_\Omega(q)$ from $F_{at,i}(p_i(q))$ and $F_{rep,i,j}(p_i(q))$ | Eq. (12) |

IF $F_\Omega(q) = [0, 0, 0, 0, 0]^T$

• localmin $\leftarrow$ true
• Detect local minimum type: type $\in\{Q_m, Q_e\}$ | Eq. (19) and Eq. (20) |
• Run AFRORE(localmin, type) (see Table 2)
ELSE

Plan a next configuration: $q' \leftarrow q + F_\Omega(q)$ | Eq. (13) |
Increase a time stamp: $s' \leftarrow s + \Delta s$ | Eq. (14) |
Assign $q'$ to a path: $\gamma(s') \leftarrow q'$ | Eq. (14) |
Update: $s \leftarrow s'$, $q \leftarrow q'$, $p_i(q) \leftarrow p_i(q')$ | Eq. (15) |
END

• Finalize: $t_f \leftarrow s$ and $\gamma(t_f) \leftarrow q_{goal}$
RETURN $\gamma$

4. Simulation Results

The capability of the proposed planner, the APBP with the AFRORE algorithm, was tested under various environmental scenarios. Considering the main role of the mobile manipulator as a co-working robot or a service robot, we created the environmental scenarios in which the mobile manipulator tries to reach a goal configuration to pick up a target object. We set stationary obstacles as well as moving obstacles which can be regarded as moving people, round tables, etc. For all scenarios, the initial and the goal configuration of the mobile manipulator were set to $q_{init} = [5, 50, 0.5\pi, 0.5\pi, 0.5\pi]^T$ and $q_{goal} = [93.597, 49.245, 0.157\pi, 0.304\pi, 0.185\pi, -0.430\pi]^T$, respectively, which correspond to the end effector position $[x, y, z]^e_{init} = [5, 50, 7]^T$ and $[x, y, z]^e_{goal} = [95, 50, 6]^T$. Stationary and moving obstacles are presented as black (black) and green (gray) circles, respectively.

Note that all presented simulation results were also uploaded as video files and can be found at [https://www.dropbox.com/s/ea2xqyqqkzvb/MobileManiPlanning.html](https://www.dropbox.com/s/ea2xqyqqkzvb/MobileManiPlanning.html)
4.1 Scenario 1: Stationary On-Ground Obstacles Deployed like Stepping Stones

As illustrated in Figure 7, five stationary obstacles were placed as the stepping stones. In this scenario, we considered an office room in which five round tables are placed. Figure 7(a) shows the mobile manipulator is heading to the goal configuration. Since there existed no local minima, the AFRORE algorithm was not executed. Under APBP, the trace in Figure 7(b)–(d) depicts that the mobile manipulator keeps passing through obstacles without any collision against the obstacles, and (d) the mobile manipulator reached at the goal.

4.2 Scenario 2: Moving-to-Stationary On-Ground Obstacles with a Local Minimum

Figure 8 shows the second scenario. In this scenario, we considered a situation in which five people meet and start chatting at the center of a room. We allowed the obstacles to meet around the center area after the mobile manipulator took off from the initial position. As the obstacle were deployed around the center, they created a local minimum by which the mobile manipulator could be trapped under simple APBP.

In Figure 8(a), the mobile manipulator is moving toward the goal configuration, and the obstacles is gathering around the center. After the mobile manipulator falls into the local minimum, it detects that the local minimum is caused by the mobile base part; therefore, it executed the AFRO of Eq. (21).
4.3 Scenario 3: Off-Ground Stationary Obstacle and Moving-to-Stationary On-Ground Obstacles with Local Minima

In the third scenario illustrated in Figure 9, we set an off-ground stationary obstacle (a monkey bar), five on-ground moving-to-stationary obstacles (chatting people), and two on-ground stationary obstacles (round tables). By this environmental setup, local minima for the elbow manipulator part and the mobile base part were created, respectively. Also, the chatting people would meet at the center of a room; in contrast to the second scenario, the gap between chatting people became narrower in this scenario.

Figure 9(a) shows that the elbow manipulator part is trapped in local minima. To escape from this local minima, the AFRO of Eq. (22) and the AFRE of Eq. (25) are activated in sequel after detecting the local minima caused by the elbow manipulator part, which in turn, serves to pass underneath the monkey bar as illustrated in Figure 9(b)–(c). After passing the monkey bar, the mobile base part falls into local minima created by the chatting people. Indeed, there exist multiple local minima due to the narrowly deployed chatting people. Figure 9(d)–(e) shows that the mobile manipulator is escaping from the local minima by performing the AFRO of Eq. (21) and the AFRE of Eq. (23) multiple times. Finally, the planner is switched to the APBP, and the mobile manipulator position reaches at the goal as presented in Figure 9(f).

4.4 Scenario 4: Room Crowded by Moving On-Ground Obstacles with Possible Local Minima

In the fourth scenario illustrated in Figure 10, the mobile manipulator is supposed to reach the goal configuration after passing a room crowded by randomly moving on-ground obstacles. This scenario corresponds to a situation in which the mobile manipulator is performing a given task while people are moving in workplace. Figure 10(a)–(f) shows that the mobile manipulator successfully reaches to the target object. However, even though
5. Conclusion and Future Works

In this paper, we proposed a subplanner algorithm, which was referred to as the AFRORE algorithm, to escape local minima which interferes the APBP to build a desired path. The proposed subplanner algorithm used insight from the kinematics of the robotic system; specifically, it allowed the APBP to overcome weakness by detecting local minima in partitioned configuration spaces and subplanning a path to escape from the detected local minima. The capability of the proposed subplanner was substantiated by the simulation results in which a mobile manipulator reached a target object successfully under various environmental conditions. A limitation of the proposed approach might be that we should priorly have an insight about local minima related to the kinematics of the employed robotic system, which is rather difficult for the highly complex robotic systems.

As future works, the capability of the proposed subplanner algorithm should be investigated for more complex robotic systems such as dual-arm robot, humanoid, etc., of which both kinematics and configuration space has higher complexity. Also, the dynamic tuning of scaling factors in artificial potential functions can be formulated as an optimization problem to find a path with desired criteria, e.g., a shorter but riskier path or a longer but safer path. Lastly, enhancing the main planner and the proposed subplanner by combining with other types of planners, e.g., geometry-based planner or sampling-based planner, to solve more complex problem would be an interesting topic as well.

Acknowledgement

This work was supported by the Korea Institute of Industrial Technology under “Development of Soft Robotics Technology for Human-Robot Coexistence Care Robots (No. EO180026).”

References

[1] J. C. Latombe, Robot Motion Planning. Boston, MA: Kluwer Academic Publishers, 2003.

[2] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, Principles of Robot
Motion: Theory, Algorithms, and Implementations. Cambridge, MA: The MIT Press, 2005.

[3] M. W. Spong, S. Hutchinson, and M. Vidyasagar, Robot Modeling and Control. Hoboken, NJ: John Wiley and Sons, 2006.

[4] J. Canny, “A Voronoi method for the piano-movers problem,” in Proceedings of IEEE International Conference on Robotics and Automation, St. Louis, MO, 1985, pp. 530-535. [http://doi.org/10.1109/ROBOT.1985.1087297]

[5] D. Halperin, O. Salzmann, and M. Sharir, “Algorithmic motion planning,” in Handbook of Discrete and Computational Geometry. Boca Raton, FL: CRC Press, 2017, pp. 1311-1342.

[6] F. Duchon, A. Babinec, M. Kajan, P. Beno, M. Florek, T. Fico, and L. Jurisica, “Path planning with modified a star algorithm for a mobile robot,” Procedia Engineering, vol. 96, pp. 59-69, 2014. [https://doi.org/10.1016/j.proeng.2014.12.098]

[7] R. Zhao, D. H. Lee, and H. K. Lee, “Fuzzy logic based navigation for multiple mobile robots in indoor environments,” International Journal of Fuzzy Logic and Intelligent Systems, vol. 15, no. 4, pp. 305-314, 2015. [http://dx.doi.org/10.5391/IJFIS.2015.15.4.305]

[8] D. H. Kim, “Integrated path planning and collision avoidance for an omni-directional mobile robot,” International Journal of Fuzzy Logic and Intelligent Systems, vol. 10, no. 3, pp. 210-217, 2010. [http://dx.doi.org/10.5391/IJFIS.2010.10.3.210]

[9] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart, “Three-dimensional coverage path planning via viewpoint re-sampling and tour optimization for aerial robots,” Autonomous Robots, vol. 40, no. 6, pp. 1059-1078, 2006. [https://doi.org/10.1007/s10514-015-9517-1]

[10] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes, “Uniform coverage structural inspection path-planning for micro aerial vehicles,” in Proceedings of 2015 IEEE International Symposium on Intelligent Control, Sydney, Australia, 2015, pp. 59-64. [http://dx.doi.org/10.1109/ISIC.2015.7307280]

[11] S. Feyzabadi and S. Carpin, “Risk-aware path planning using hierarchal constrained Markov decision processes,” in Proceedings of 2014 IEEE International Conference on Automation Science and Engineering, Taipei, Taiwan, 2014, pp. 297-303. [http://dx.doi.org/10.1109/CoASE.2014.6899341]

[12] J. H. Cho and Y. T. Kim, “Design of autonomous logistics transportation robot system with fork-type lifter,” International Journal of Fuzzy Logic and Intelligent Systems, vol. 17, no. 3, pp. 177-186, 2017. [http://dx.doi.org/10.5391/IJFIS.2017.17.3.177]

[13] L. Jaillet and J. M. Porta, “Path planning with loop closure constraints using an atlas-based RRT,” in Robotics Research. Cham: Springer, 2016, pp. 345-362. [https://doi.org/10.1007/978-3-319-29363-9_20]

[14] Y. Wei, W. Jiang, A. Rahmani, and Q. Zhan, “Motion planning for a humanoid mobile manipulator system,” 2018. Available [https://arxiv.org/abs/1806.07349]

[15] S. Lavalle, Planning Algorithms. Cambridge, UK: Cambridge University Press, 2006, pp. 154-205.

[16] D. Hsu, L. E. Kavraki, J. C. Latombe, R. Motwani, and S. Sorkin, “On finding narrow passages with probabilistic roadmap planners,” in Robotics: The Algorithmic Perspective: 1998 Workshop on the Algorithmic Foundations of Robotics. Natick, MA: A K Peters, 1998, pp. 141-153.

[17] J. Canny and J. Reif, “Lower bounds for shortest path and related problems,” in Proceedings of the 28th Annual Symposium on Foundations of Computer Science, Los Angeles, CA, 1987, pp. 49-60.

[18] Y. Koren and J. Borenstein, “Potential field methods and their inherent limitations for mobile robot navigation,” in Proceedings of IEEE International Conference on Robotics and Automation, Sacramento, CA, 1991, pp. 1398-1404. [https://doi.org/10.1109/ROBOT.1991.131810]

[19] V. Sezer and M. Gokasan, “A novel obstacle avoidance algorithm: follow the gap method,” Robotics and Autonomous Systems, vol. 60, no. 9, pp. 1123-1134, 2012. [https://doi.org/10.1016/j.robot.2012.05.021]

[20] Y. Ma, G. Zheng, W. Perruquetti, and A. Qiu, “Local path planning for mobile robots based on intermediate objectives,” Robotica, vol. 33, no. 4, pp. 1017-1031, 2015. [http://dx.doi.org/10.1017/S0263574714000186]
[21] Y. Singh, S. Sharma, D. Hatton, and R. Sutton, “Optimal path planning of unmanned surface vehicles,” *Indian Journal of Geo Marine Sciences*, vol. 47, no. 7, pp. 1325-1334, 2018.

**Han Ul Yoon** received his Ph.D. degree from the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign in 2014. Dr. Yoon served as a postdoctoral research associate and an assistance professor of instruction in the Department of Mechanical Engineering at Texas A&M University in 2014-2016 and 2017, respectively. He is currently a senior researcher in Robotics R&D Group at the Korea Institute of Industrial Technology (KITECH). He is mainly interested in assistive human-robot interaction, biomechanical analysis and control, and robotic path planning.

E-mail: huyoon@kitech.re.kr

**Dong-Wook Lee** received his Ph.D. from the Department of Control and Instrumentation Engineering at Chung-Ang University in 2000. Dr. Lee was a research professor at Chung-Ang University from 2002 to 2004. He was a post-doctoral researcher at the University of Tennessee at Knoxville from 2004 to 2005. He is currently a principal researcher in Robotics R&D Group at Korea Institute of Industrial Technology (KITECH). His research interests include android robot, social human-robot interaction, and evolutionary computation.

E-mail: dwlee@kitech.re.kr