Motional Planning of a Snake-Like Robot Using an Optimized Harmonic Potential Field

Samer Charifa¹, Marwan Bikdash²

Abstract
Snake-like robots have gained popularity in the last three decades for their ability to utilize several gaits in order to navigate through different terrains. They are analogous in morphology to snakes, tentacles, and elephant trunks. We propose a novel method of navigating a snake-like robot based on the Harmonic Field with Optimized Boundary Conditions (HFOBC) and a boundary following algorithm. We apply the HFOBC navigation function using a number of fictitious charges equally spaced on each link. These charges actively follow the potential field towards the target. Furthermore, a generalized mathematical model for an n-link snake-like robot based on Lagrange formulation has also been proposed in this paper.

Keywords
Robotic Motion Planning · Snake-Like Robot · Harmonic Potential Field

Nomenclature

| Symbol | Definition |
|--------|------------|
| $\mathbf{p}_i$ | A vector from the robot to the obstacle $O_i$ |
| $\mathbf{p}_\beta$ | A vector from the robot to the target $\beta$ |
| $U_i(\mathbf{p}_i)$ | A repelling potential field due to the obstacle $i$ |
| $U_\beta(\mathbf{p}_\beta)$ | Attractive potential field due to target $\beta$ |
| $\phi(\mathbf{p})$ | Harmonic potential field value for vector $\mathbf{p}$ |
| $L$ | Minimum distance from the robot to the closest obstacle |
| $R$ | Safety distance |
| $\mathbf{n}$ | A unit vector perpendicular to the boundary in the direction of the robot |
| $\mathbf{t}$ | A unit vector tangential to the boundary in the direction of the negative gradient of the potential field |
| $N_i$ | Number of charges on link $i$ in the snake robot |
| $(x_i, y_i)$ | Position of the center of gravity of the link $i$ |
| $(x_0, y_0)$ | Position of the first charge of the first link |
| $m_i$ | Mass of link $i$ |
| $l_i$ | Length of link $i$ |
| $\theta_i$ | Angle of link $i$ with the global x-axis |
| $\psi_i$ | Angle between link $i$ and link $i-1$ |
| $T$ | Kinetic energy |
| $V$ | Potential energy |
| $k$ | Torsional spring constant |
| $M$ | Inertia matrix |
| $C$ | Coriolis terms vector |
| $G$ | Potential energy vector |
| $q_i$ | Generalized coordinate |
| $Q^a$ | Generalized forces vector due to actuators |
| $Q^g$ | Generalized forces vector due to charges’ forces |
| $Z^i$ | Position of the beginning of link $i$ |
| $\mathbf{P}_j^i$ | Position of charge $j$ on link $i$ |
| $\delta \mathbf{r}_j^i$ | Virtual displacement of the charge $j$ on the link $i$ |
| $\delta w$ | Virtual work |
| $\mathbf{F}_j^i$ | The potential field force acting on charge $j$ on the link $i$ |

1. Introduction

Since the first snake-like robot designed by Hirose in 1972 at the Tokyo Institute of Technology [1], research in snake robots has continued to be improved in terms of the technical design and mathematical modeling. More recent designs of snake-like robots include the JPL Serpentine Robot [2], the ACM-RS and the Perambulator-II [3]. Research has been aimed at mathematical formulation of snake motion, both kinematically [4] and dynamically [5], using robotic snakes with wheels [3], crawler tracks [6] or only the snake body itself [7]. Many useful applications have been recently proposed for snake-like robots such as search-and-rescue in difficult environments [8], subsea applications, pipe inspection, service robots, and even fire-fighting [9].
In the following, we summarize some of the known literature on motion planning of snake-like robots and follow with a short summary of the advantages of our proposed navigation algorithm.

Takanashi et al. [10], [11] proposed a motion planning algorithm compromised of two local sensor based techniques for snake robots. These are Full Shape Modification (FSM) and Partial Shape Modification (PSM). The FSM generates the general shape of the snake robot. The general shape is passed to the PSM which will fine-tune the general shape based on the local sensory information. However, this type of motion planners is limited because of its reactive nature and lack of a global planning stage. Henning et al. [12] proposed a Voronoi-diagram-based roadmap to guide the motion of the snake robot. A major drawback with this type of purely graph theory based motion planners is that the quality of the path is highly dependent on the roadmap, which is typically coarse.

Li et al. [13] proposed a snake-robot motion planning technique based on the solution of the Serpenoid-curve-fitting optimization problem using a two-layer Genetic Algorithm (GA), yet this technique has some drawbacks. This approach requires an expensive computation in order to solve the two-layer GA. Moreover, the convergence is not guaranteed in real-time. Conkur [14] and Conkur and Gurbuz [15] proposed a snake-like robot navigation algorithm using a harmonic potential field. They proposed a path following technique based on “following the path” which means “keeping the ends of the links on the path.” Unfortunately, this is not always a safe assumption particularly when navigating in a cluttered environment. Cheng et al. [16] proposed the so-called Accompanied Snake (A-Snake). The A-Snake is a navigation technique that is applied to mobile robots in general, as well as snake robots. This method applies a rolling window optimization in order to generate elastic paths for robot navigation. It mainly depends on a large-scale camera network distributed on top of the environment. Although this approach can produce fairly good results, it is rather impractical since it requires several cameras mounted on the top of the environment to monitor the gait of the robot throughout its motion. Yagnik et al. [17] proposed using an artificial potential field method with Simulated Annealing (SA) for snake robot navigation in a tail-follow-head fashion. In their algorithm, they used a Gaussian-like potential field function which suffers from the well-known local minima problem. Although they used SA to escape local minima, we argue that this type of heuristic algorithms is not guaranteed to work for complex environment and more links (more than three).

In this paper, we propose an optimized harmonic potential field which is proven to be local-minima free. While the current literature indicates that it is better to consider the whole body of the snake robot for motion planning, we have added well-distributed points/colors throughout the body of the robot to account for the motion of each part. Clearly, this addition will result in more accurate and smoother motion planning than tail-follow-head motion planning. Furthermore, the boundary following algorithm will also improve the quality of the motion planning particularly when navigating through narrow corridors, as explained later. Moreover, we present a general n-link navigation framework and show through simulation that our algorithm works well for 3-link and 10-link mobile robots navigating in a cluttered environment.

This paper is organized as follows: We start by describing our proposed mechanical design of the snake robot. The dynamics of the snake robot are described in Section 3. Since the dynamics of a high DOF snake robot are fairly complicated, we first describe the dynamics of a three-link mobile robot and then we proceed to the dynamics of n-link robot. In Section 4, we solve the inverse dynamics problem in the presence of the harmonic field with an optimized boundary conditions. A performance evaluation is described in Section 5. Finally, in Sections 6 and 7, we provide discussion and concluding remarks regarding this work.
3. Dynamics of the Snake-Like Robot

Without loss of generality, we start by illustrating the case of a 3-link mobile robot, then we generalize it to an $n$-link snake-like robot.

3.1. Dynamics of the 3-Link Mobile Robot

Figure 3 shows the top view of a 3-link mobile robot having a number of fictitious charges on it. Since this robotic system has 5 DOF, we have chosen the following five independent variables as generalized coordinates: $x_i$, $y_i$, $\theta_i$, $\theta_{i+1}$, and $\theta_{i+2}$, where $x_i$, $y_i$ are the coordinates of the center of gravity of the link $i$, $\theta_i$ is the angle between the link $i$ and the global $x$-axis, and $Z_i$ is the position of the beginning of link $i$ as shown in Figure 3. From Figure 3, we can write:

\[
\begin{align*}
    x_1 &= x_0 + 0.5l_1 \cos \theta_1, \\
    y_1 &= y_0 + 0.5l_1 \sin \theta_1, \\
    x_2 &= x_1 + l_2 \cos \theta_2 + 0.5l_2 \cos \theta_3, \\
    y_2 &= y_1 + l_2 \sin \theta_2 + 0.5l_2 \sin \theta_3, \\
    x_3 &= x_2 + l_3 \cos \theta_3, \\
    y_3 &= y_2 + l_3 \sin \theta_3.
\end{align*}
\] (1)

For simplicity, we denote $x_0$ and $y_0$ as $x$ and $y$. Also, we assume that all links are geometrically identical and therefore, we write $l_{1,2,3} = l$, and $m_{1,2,3} = m$, and the moment of inertia of each link rotating around its center of gravity is $I_{1,2,3} = \frac{1}{12}l^2m$. Substituting Equation (1) into (3), we obtain the following expression in terms of the generalized coordinates:

\[
T = \frac{1}{2} m (\dot{x}_1^2 + \dot{y}_1^2) + \frac{1}{2} m_l (\dot{x}_2^2 + \dot{y}_2^2) + \frac{1}{2} m_2 (\dot{x}_3^2 + \dot{y}_3^2) + \frac{1}{2} J_1 \dot{\theta}_1^2 + \frac{1}{2} J_2 \dot{\theta}_2^2 + \frac{1}{2} J_3 \dot{\theta}_3^2
\] (3)

In order to prevent the links from folding onto each other excessively, a fictitious torsional spring is added between each two consequent links. The equation describing the potential energy of these springs is:

\[
V = \frac{1}{2} k \left[ (\theta_2 - \theta_1)^2 + (\theta_3 - \theta_2)^2 \right],
\] (5)

where $k$ is the spring constant.

Substituting Equations (4) and (5) into Equation (2) and solving, we obtain the following equations of motion [18]:

\[
M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = Q,
\] (6)

where $M$ is the inertia matrix:

\[
M(q) = m \begin{bmatrix}
3 & 0 & -\frac{1}{2} l \sin \theta_1 & -\frac{1}{2} l \sin \theta_2 & -\frac{1}{2} l \sin \theta_3 \\
0 & 3 & \frac{1}{2} l \cos \theta_1 & \frac{1}{2} l \cos \theta_2 & \frac{1}{2} l \cos \theta_3 \\
-\frac{1}{2} l \sin \theta_1 & \frac{1}{2} l \cos \theta_1 & \frac{1}{2} l^2 \cos (\theta_1 - \theta_2) & \frac{1}{2} l^2 \cos (\theta_2 - \theta_3) \\
-\frac{1}{2} l \sin \theta_2 & \frac{1}{2} l \cos \theta_2 & \frac{1}{2} l^2 \cos (\theta_2 - \theta_3) & \frac{1}{2} l^2 \cos (\theta_3 - \theta_1) \\
-\frac{1}{2} l \sin \theta_3 & \frac{1}{2} l \cos \theta_3 & \frac{1}{2} l^2 \cos (\theta_3 - \theta_1) & \frac{1}{2} l^2 \cos (\theta_1 - \theta_2)
\end{bmatrix},
\] (7)
The matrix \( C(q, \dot{q}) \) represents the Coriolis and centrifugal terms:

\[
C(q, \dot{q}) = \begin{bmatrix}
    -\frac{1}{2} \dot{\theta}_1 \cos \theta_1 - \frac{1}{2} \dot{\theta}_2 \cos \theta_2 - \frac{1}{2} \dot{\theta}_3 \cos \theta_3 \\
    -\frac{1}{2} \dot{\theta}_1 \sin \theta_1 - \frac{1}{2} \dot{\theta}_2 \sin \theta_2 - \frac{1}{2} \dot{\theta}_3 \sin \theta_3 \\
    -\frac{1}{2} \dot{\theta}_1 \sin(\theta_1 - \theta_2) - \frac{1}{2} \dot{\theta}_2 \sin(\theta_2 - \theta_3) - \frac{1}{2} \dot{\theta}_3 \sin(\theta_3 - \theta_1)
\end{bmatrix},
\]

\( G(q) \) represents the forces derived from the potential energy:

\[
G(q) = k \begin{bmatrix}
    0 \\
    \theta_1 - \theta_2 \\
    -\theta_1 + \theta_2 - \theta_3
\end{bmatrix},
\]  

(9)

and \( Q \in \mathbb{R}^3 \) is a vector of generalized forces to be computed later.

### 3.2. Dynamics of the n-Link Snake-Like Robot

In the following, we propose a general formula that can be applied to any \( n \)-link snake-like robot to compute the components of the equations of motion in Equation (6). We apply the same methodology adopted in the previous section, to obtain the components of the classical equation of motion.

#### 3.2.1. Inertia Matrix

The generalized inertia matrix \( M \in \mathbb{R}^{(2n+3) \times (2n+3)} \) and can be divided into three different matrices \( M^{11}, M^{12}, M^{21} \) and \( M^{22} \):

\[
M = \begin{bmatrix}
    M^{11} & M^{12} \\
    M^{21} & M^{22}
\end{bmatrix}.
\]  

(10)

The matrix \( M^{11} \in \mathbb{R}^{2n \times 2n} \) is a diagonal matrix where

\[
M^{11}_{ii} = nm, \ \forall i = j.
\]

The matrix \( M^{12} = M^{11T} = \mathbb{R}^{2n \times n} \) can be written as:

\[
M^{12}_{ij} = \begin{cases}
    \frac{1}{2} + n - j \ell m \sin \theta_j, & \text{if } i = j \\
    \frac{1}{2} + n - j \ell m \cos \theta_j, & \text{if } i > j
\end{cases}
\]

The matrix \( M^{22} \in \mathbb{R}^{n \times n} \) can be written as:

\[
M^{22}_{ij} = \begin{cases}
    \frac{1}{2} + n - j \ell m \cos(\theta_i - \theta_j), & \text{if } i = j \\
    \frac{1}{2} + n - j \ell m \sin(\theta_i - \theta_j), & \text{if } i > j
\end{cases}
\]

3.2.2. Coriolis and Centrifugal Terms Vector

The generalized \( C(q, \dot{q}) \in \mathbb{R}^n \) matrix can be written as:

\[
C = \begin{bmatrix}
    0 & C^1 \\
    0 & C^2
\end{bmatrix},
\]

where \( C^1 \) is a \( 2 \times n \) matrix defined as:

\[
C^1_{ij} = \begin{cases}
    \frac{1}{2} - n + j \ell \cos \theta_i, & \text{if } i = 1, \\
    \frac{1}{2} - n + j \ell \sin \theta_i, & \text{if } i = 2
\end{cases}
\]

and \( C^2 \) is an \( n \times n \) matrix defined as:

\[
C^2_{ij} = \begin{cases}
    n - i + \frac{1}{2} \ell \sin(\theta_i - \theta_j), & \text{if } i \leq j, \\
    n - j + \frac{1}{2} \ell \sin(\theta_j - \theta_i), & \text{if } j > i
\end{cases}
\]

3.2.3. Forces due to the Potential Energy \( G(q) \)

The generalized potential force vector is denoted \( G(q) \in \mathbb{R}^n \). Since we assume that the motion occurs in a 2D plane parallel to the ground, the only potential energy forces are the damping torsional spring forces between each link. Thus, \( G(q) \) and can be written as:

\[
G(q) = [0, 0, G^3]^T,
\]  

(12)

where

\[
G^3 = \begin{cases}
    k(\theta_i - \theta_j), & \text{if } j = 1, \\
    k(\theta_i - \theta_{i-1}), & \text{if } j = n, \\
    k(\theta_{n-1} + \theta_n - \theta_{n+1}), & \text{otherwise}.
\end{cases}
\]

Here \( j \) is the index of each link.

![Figure 4](image-url)  

**Figure 4.** The actuator forces and the virtual displacement \( \delta \sigma \) of the three-link robot (please refer to Nomenclature).
3.3. Generalized Forces due to Actuator Forces

After finding $M$, $C$ and $G$ from Equation 6, we proceed to find the generalized forces $Q^a$, due to the actuators. The $j^{th}$ joint is actuated using a bending torque $\overrightarrow{T}_j$ applied between each two links. The traction comes from two omni-directional wheels equipped with traction motors at the two ends of the first link. The two traction forces are denoted $\overrightarrow{p}_1$ and $\overrightarrow{p}_2$, respectively. For an $n$-link snake robot there are $n + 2$ generalized forces associated with $n + 2$ generalized coordinates and $n + 3$ actuator forces. Since the number of actuator forces is more than the number of generalized forces, this robot is an overactuated system.

Let us consider the case of the 3-link robot again. The generalized forces can be found using the virtual work principle [19]. The $k^{th}$ generalized force due to the actuators can be found by computing the work $\delta W_k$ produced by actuator forces when a small virtual displacement $\delta q_k$ is enforced on the system while all other $q_i$ are held constant,

$$\delta q_i = 0 \text{ for } j \neq k.$$  

First, we consider the virtual displacement $\delta \overrightarrow{x} = \delta x \overrightarrow{i}$. Clearly, the only actuator forces which will contribute to the corresponding virtual work $\delta W_k$ will be $\overrightarrow{p}_1$ and $\overrightarrow{p}_2$, see Figure 4.

Thus, we can write:

$$\delta W_k = \overrightarrow{p}_1 \cdot \left( \delta x \overrightarrow{i} \right) + \overrightarrow{p}_2 \cdot \left( \delta x \overrightarrow{i} \right) \implies Q^a_k \delta x = \left( \overrightarrow{p}_1 \right)_x + \left( \overrightarrow{p}_2 \right)_x,$$

and hence,

$$Q^a_k = \left( \overrightarrow{p}_1 \right)_x + \left( \overrightarrow{p}_2 \right)_x.$$  

Similarly,

$$Q^a_k = \left( \overrightarrow{p}_1 \right)_y + \left( \overrightarrow{p}_2 \right)_y.$$  

is the generalized force corresponding to $\delta \overrightarrow{T} = \delta q_j \overrightarrow{i}$. When adding a virtual displacement $\delta \theta_i$, only $\overrightarrow{p}_2$ and $\overrightarrow{p}_1$ will contribute to the virtual work $\delta W_i$, (see Figure 5):

$$\delta W_i = \overrightarrow{p}_2 \cdot \delta \overrightarrow{T}_1 + T_1 \delta \psi_i,$$

where, from Figure 5:

$$\delta \overrightarrow{T}_1 = l \left( - \sin \theta_1 \overrightarrow{i} + \cos \theta_1 \overrightarrow{j} \right) \delta \theta_1,$$

and

$$\psi_i = \theta_2 - \theta_1.$$  

Since $\delta \theta_2 = 0$, $\delta \psi_i$ becomes

$$\delta \psi_i = - \delta \theta_1.$$  

Substituting Equations (16) and (18) into Equation (15),

$$Q^a_{\theta_i} = \left\{ \begin{array}{ll} l \left( - \sin \theta_1 \left( \overrightarrow{p}_2 \right)_x + \cos \theta_1 \left( \overrightarrow{p}_2 \right)_y \right) - T_1 & \text{if } j = 1, \\ T_{j-1} - T_j & \text{if } 1 < j < n, \\ T_{j-1} & \text{if } j = n. \end{array} \right.$$
4. Control Forces Design

In the following, we present our navigational algorithm for the snake robot using an optimized harmonic potential field.

4.1. Harmonic Potential Field with Optimized Boundary Conditions

The harmonic potential field [20] is given by the solution $\psi(r) : \mathbb{R}^2 \to \mathbb{R}$ of Laplace’s Equation: $\nabla^2 \psi = 0$, with the boundary conditions:

$$\psi(r) = c_1 \text{ if } r \in \partial \beta, \quad \text{and} \quad \psi(r) = c_2 < c_1 \text{ if } r \in \partial T,$$

where $\partial \beta$ is the union of the boundaries of all obstacles, $\partial T$ is the target boundary, and $c_1$ and $c_2$ are constants.

The force applied to the robot is then designed to be the negated gradient of the potential field, i.e. $-\vec{F} = -\frac{\partial \psi}{\partial \vec{r}}$.

It has been reported in [21] and [22] that the harmonic field with uniform boundary conditions, Eq. (22), does not have a smooth variation outside the target region. In [23], we proposed the Harmonic Field with Optimized Boundary Conditions (HFOBC) combined with Boundary Following Algorithm (BFA) [24]. The HFOBC abandons the uniformity of boundary conditions used in the regular harmonic field, Eq. (22), and instead applies $\psi(r) = \psi(r)$ whenever $r \in \partial \beta$, where $\psi(r)$ is the length of the shortest path from $r$ to the target boundary. By definition $\psi(r) = 0$ on the target boundary. This results in a harmonic field with optimized boundary conditions, which can be written as:

$$\nabla^2 \phi(r) = 0,$$

$$\phi(r) = \psi(r) \text{ for } r \in \partial \beta \cup \partial T.$$

The boundary conditions are optimized in the sense that $\psi(r)$ represents the minimum distance to the target. These distances can be approximated using Dijkstra’s algorithm on a finite-element mesh or other spatial discretization of the configuration space.

Solving the problem in Eq. (23) results in the harmonic potential field whose contours are shown in Fig. 6. Clearly, the resulting field is free of local minima.

4.2. Motion Planning

To apply the HFOBC, the robot boundary is assumed to hold a set of fixed discrete charges interacting with the already developed electrostatic potential field. The physical analogy is not pursued accurately, since the moving robot charges will create a magnetic field that will complicate the physics. These magnetic effects are assumed to be negligible in the adopted analogy. This modification can be made by assuming that a number of fictitious charges are fixed on the robot, as can be seen in Figure 7.

The motion planning algorithm for a snake-like robot through a cluttered environment can be summarized as follows:

1. Apply adaptive allocation of charges.
2. Compute all potential field forces which take the robot toward the target using the HFOBC and the BFA.
3. Compute the actuator’s forces in terms of the potential field forces.
4. Scale the actuators’ forces to accommodate the maximum allowable forces and/or torques that can be applied on the actuators.

4.3. Adaptive allocation of charges

The distribution of the charges on the snake-like robot is important for proper navigation of the robot. If the number of charges on the segment is insufficient, collision may result between a sharp obstacle corner and a long uncharged piece of the robot. This can be avoided by choosing $d_i$, the distance between two consecutive charges, such that:

$$d_i \leq 2d_s,$$

where $d_s$ is the safety distance, as can be seen in Fig. 8. Note that since the safety distance is adaptive, the charge separation can be adaptive as well.
4.4. Generalized Forces for n-Link Snake Robot due to Potential Field Forces

In the following, we find the generalized forces due to the forces acting on the charges shown in Figure 9. We start by the simple example of the 3-link mobile robot. We denote the generalized forces as:

\[ Q' = \begin{bmatrix} Q'_x & Q'_y & Q'_{\theta_1} & Q'_{\theta_2} & Q'_{\theta_3} \end{bmatrix}^T. \]  

(24)

Let us assume that there are 3 charges whose forces are denoted \( F^1_1, F^1_2, \) and \( F^1_3 \) on the first link and other 3 charges whose forces are denoted \( F^2_1, F^2_2, \) and \( F^2_3 \) on the second link and 4 charges whose forces are denoted \( F^3_1, F^3_2, F^3_3, \) and \( F^3_4 \) on the third link. As a result, the total number of forces acting on the 3-link robot is 10, as shown in Figure 9.

In general, the virtual work done by each force \( F^i_k \) when undertaking a virtual displacement \( \delta \mathbf{r}^i_k \) is:

\[ \delta W = \sum_{i=1}^{3} \sum_{k=1}^{3} F^i_k \cdot \delta \mathbf{r}^i_k + \sum_{k=1}^{4} F^3_k \cdot \delta \mathbf{r}^3_k, \]  

(25)

where \( \mathbf{r}^i_k \) represents the position of the charge \( k \) on the link \( i \), and \( \delta \mathbf{r}^i_k \) represents the virtual displacement of charge \( \mathbf{r}^i_k \).

To find the generalized force \( Q'_y \), we apply a virtual increment \( \delta \mathbf{r} = \delta x \mathbf{e}_x = \delta \mathbf{r}^i_1 \) to the first charge \((x_0, y_0)\), as illustrated in Figure 10. Then, every charge’s force contributes a virtual work proportional to its x-component:

\[ Q'_y \delta x = \sum_{i=1}^{3} \sum_{k=1}^{3} \left( F^i_k \cdot \delta \mathbf{r}^i_k \right) + \sum_{k=1}^{4} \left( F^3_k \cdot \delta \mathbf{r}^3_k \right), \]  

\[ = \sum_{i=1}^{3} \sum_{k=1}^{3} \left( F^i_k \right) \delta x + \sum_{k=1}^{4} \left( F^3_k \right) \delta x \]  

and in short:

\[ Q'_y = \sum_{i=1}^{3} \sum_{k=1}^{3} \left( F^i_k \right). \]  

(26)

Similarly, we find the generalized force \( Q'_x \) as:

\[ Q'_x = \sum_{i=1}^{3} \sum_{k=1}^{3} \left( F^i_k \right) y. \]  

(27)

Next, we proceed by computing the generalized forces corresponding to virtual angular displacement, \( Q'_{\theta_1}, Q'_{\theta_2}, \) and \( Q'_{\theta_3} \). Note the two examples of adding the virtual displacement \( \delta \theta \) in Figure 11 and the virtual displacement \( \delta \theta \) in Figure 12:

First, a general expression for the vector \( \mathbf{r}^i_k \) is derived:

\[ \mathbf{r}^i_k = \frac{k}{N} l \left( \cos \theta_i \mathbf{e}_i + \sin \theta_i \mathbf{e}_j \right) + \mathbf{z}_j, \]  

(28)

where \( \mathbf{z}_j \) is the end-point vector (shown in Figure 9), which can be written as:

\[ \mathbf{z}_j = \begin{cases} (x_0 \mathbf{e}_i + y_0 \mathbf{e}_j) + \sum_{k=1}^{i-1} l \left( \cos \theta_k \mathbf{e}_i + \sin \theta_k \mathbf{e}_j \right) & \text{if } j > 1, \\ (x_0 \mathbf{e}_i + y_0 \mathbf{e}_j) & \text{if } j = 1. \end{cases} \]  

(29)
Substituting Equation (30) into Equation (25), we find the generalized bra, we find a general expression for

From Equation (28) and (29), and after applying straightforward algebra, we apply the same approach of finding the generalized forces of the 3-link robot to the n-link mobile robot where each link \( j \) has \( N_j \) charges acting on it. We let the generalized forces vector be denoted as

Next, we proceed by computing \( \delta \mathbf{T}^i_k \) in terms of \( \delta \theta_j \). Obviously,

From Equation (28) and (29), and after applying straightforward algebra, we find a general expression for \( \frac{\partial \mathbf{T}^i_k}{\partial \theta_j} \) as follows:

Substituting Equation (30) into Equation (25), we find the generalized force \( Q_{\theta_j} \) as:

Similarly, we find expressions for \( Q_{\theta_j} \) and \( Q_{\theta_l} \) as follows:

Now, we can find the generalized forces for the \( n \)-link mobile robot, and we obtain the following:

We apply the same approach of finding the generalized forces of the 3-link robot to the \( n \)-link robot, and we obtain the following:

and for \( j = 1, \ldots, n - 1, \)

The last link is a special case,

Note that the number of charges on the last link cannot be equal to 1.

Figure 11. A diagram of the three-link robot shows the virtual displacement \( \delta \Theta_1 \).

Figure 12. A diagram of the three-link robot shows the virtual displacement \( \delta \Theta_2 \).
In this section, we compute the forces of the actuators in terms of the forces of the charges. To find actuators’ forces, we equate the generalized forces corresponding to the actuator forces and the generalized forces corresponding to the charges’ forces. For the 3-link robot, we obtain the following set of five equations:

\[
\begin{align*}
(\bar{p}_1)_x + (\bar{p}_2)_x &= Q_1', \\
(\bar{p}_1)_y + (\bar{p}_2)_y &= Q_2', \\
-l \sin \theta_1 (\bar{p}_2)_x + l \cos \theta_1 (\bar{p}_2)_y - T_1 &= Q_1'', \\
T_1 - T_2 &= Q_2'', \\
T_2 &= Q_3'',
\end{align*}
\] (39)

where the six unknowns \(p_{1x}, p_{1y}, p_{2x}, p_{2y}, T_1, T_2\) This is an over-actuated system, and there are infinitely many solutions.

Rearranging Equation (39), we find expressions for the torques \(T_1\) and \(T_2\) as follows:

\[
\begin{align*}
T_1 &= Q_2' + Q_1', \\
T_2 &= Q_3'.
\end{align*}
\] (40)

Therefore, we write Equation (39) as follows:

\[
\begin{align*}
(\bar{p}_1)_x + (\bar{p}_2)_x &= Q_1', \\
(\bar{p}_1)_y + (\bar{p}_2)_y &= Q_2', \\
-l \sin \theta_1 (\bar{p}_2)_x + l \cos \theta_1 (\bar{p}_2)_y &= Q_1'' + Q_2' + Q_3'.
\end{align*}
\] (42)

We will seek a least square solution, which minimizes the amount of traction as measured by \(\|\bar{p}_1\| + \|\bar{p}_2\|\). In short,

\[
\begin{align*}
\min_{p_1, p_2} J &= p_1^T p_1 + p_2^T p_2, \quad \text{s. t.} \\
p_1 + p_2 &= F, \\
R^T p_2 &= T_q,
\end{align*}
\] (43)

where,

\[
\begin{align*}
F &= \begin{bmatrix} Q_1' & Q_2' \end{bmatrix}^T, \\
T_q &= Q_1'' + Q_2' + Q_3', \\
R &= \begin{bmatrix} -l \sin \theta_1 & l \cos \theta_1 \end{bmatrix}^T,
\end{align*}
\]

are all assumed to be known at a given configuration.

The augmented cost is (in matrix notation):

\[L = p_1^T p_1 + p_2^T p_2 + \lambda^T (p_1 + p_2 - F) + \mu (R^T p_2 - T_q),\] (45)

where \(\lambda = \begin{bmatrix} \lambda_1 & \lambda_2 \end{bmatrix}^T \in \mathbb{R}^2\) and \(\mu\) are the Lagrange multipliers. The solution of this minimization problem must satisfy:

\[
\begin{bmatrix}
2I & 0 & I \\
0 & 2I & I \\
I & 0 & I
\end{bmatrix}
\begin{bmatrix}
p_1 \\
p_2 \\
\lambda
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
F
\end{bmatrix} + \begin{bmatrix}
0 \\
0 \\
T_q
\end{bmatrix}. \tag{46}
\]

This solution is suboptimal in the sense that it minimizes the efforts \(\|\bar{p}_1\|\) and \(\|\bar{p}_2\|\) but may lead to forces exceeding \(F_{\text{max}}\), the maximum allowable force on each actuator. Next, we scale down all forces until all forces are deliverable.

The full solution is given by Equations (40)-(41) and by the solution of Equation (46):

\[
\begin{align*}
p_1 + p_2 &= F, \\
R^T p_2 &= T_q, \\
2p_1 + \lambda &= 0, \\
2p_2 + \lambda + \mu R &= 0.
\end{align*}
\] (47)

4.6. Scaling of Forces

Due to the high repulsive nature of the boundary following algorithm, we scale down all forces/torques with the same scaling factor, so that all forces/torques can be delivered. Assume that the maximum deliverable torque is \(T_{\text{max}}\) and the maximum deliverable driving force/traction force for the first and the second link is \(p_{\text{max}}\). Hence,

\[
\begin{align*}
\bar{p}_1 &= \frac{p_1}{y}, \\
\bar{p}_2 &= \frac{p_2}{y}, \\
\bar{T}_1 &= \frac{T_1}{y}, \\
\bar{T}_2 &= \frac{T_2}{y},
\end{align*} \tag{48}
\]

where \(y = \max\left(\frac{\|\bar{p}_1\|}{p_{\text{max}}}, \frac{\|\bar{p}_2\|}{p_{\text{max}}}, \frac{\max\|\bar{T}_i\|}{T_{\text{max}}}, i = 1...n\right)\).

5. Performance Evaluation

We have simulated our algorithm on two types of robots. A three-link mobile robot and a ten-link snake-like robot.

5.1. Path Planning of 3-Link Mobile Robot

Here, we show the motion planning results of a three-link mobile robot. Each link is 5 m in length. Clearly, the robot is able to navigate through all the obstacles until it reaches the target. The robot maintains a safe distance to all boundaries, as can be seen from Figure 13.

As an example, we consider the speed profile for the first node in the robot. Figure 14 shows a practical speed profile. Note that the desired speed for the robot navigating without having any interaction with other boundaries is 5 m/sec. This value drops when the robot is interacting with a wall or a corner or with multi-wall-corner boundaries. This can be seen in Figure 14.
5.2. Path Planning for 10-Link Snake-Like Mobile Robot

Here, we show the simulation results for a 10-link snake-like robot. Figure 15 shows excellent navigation of the snake robot in the obstructed environment. This means excellent obstacle avoidance, no stagnation in local minima, no vibration or oscillation in the presence of obstacles, and practical speed profile. The safety distances to all boundaries are exactly as expected in the boundary following algorithm. Each link has three charges on it. Figure 16 shows the speed profile for the first charge. Clearly, the robot slows down when turning a corner or passing between a corner and a wall.

6. Discussion and Future Work

In this section, we discuss the main advantages of the proposed work and present several future plans for further development. In this research, the gait of the robot is computed using an optimized harmonic artificial potential field. This potential field is free of local minima, which makes it appealing for robust obstacle avoidance. Furthermore, the introduction of the well-distributed charges makes the planned motion very smooth. This approach introduces improvement to the tail-follow-head approach adopted in the literature [17], where the potential field is computed only for the head. Moreover, we present a novel presentation of the inverse dynamics of the snake-like robot derived from the potential field forces.

As future work, we plan to extend this work to 3D environments. Also, we plan to implement this work on a wirelessly controlled snake-robot equipped with laser and vision sensors.

7. Conclusions

We have developed a novel method for the navigation of a snake robot. The navigation method is based on the optimized harmonic field and boundary following algorithm. Fictitious charges well-distributed on each link of the snake robot have been used to navigate the robot us-
ing the HFOBC. This produces smooth motion and excellent obstacle avoidance. A generalized mathematical model for an n-link snake-like robot based on Lagrange formulation has been developed. The inverse dynamics problem for the proposed mechanism of a snake-like robot has been resolved. The simulation results show excellent navigation in the presence of obstacles and the speed profiles seem practical and correspond with the desired speed limit around the obstacles.

References

[1] S. Hirose, Biologically Inspired Robots: Snake-Like Locomotors and Manipulators, Oxford University Press, Oxford, 1993.

[2] E. Paljug, T. Ohm, and S. Hayati, “The JPL serpentine robot: a 12 dof system for inspection,” IEEE Int. Conf. on Robotics and Automation, Nagoya, Aichi, Japan, vol. 3, May 1995, pp. 3143-3148.

[3] S.H. Yamada, M. Chigisaki, K. Mori, K. Takita, K. Ogami, S. Hirose, “Development of amphibious snake-like robot ACM-RS,” Proc. 36th Int. Symp. Robotics, Tokyo, Japan, Nov. 2005, pp. 163-175.

[4] M. Nilsson, “Serpentine locomotion on surfaces with uniform friction,” Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Sendai, Japan, Sep. 2004, pp. 1751-1755.

[5] P. Liljeback, Ø. Stavdahl, and K.Y. Pettersen, “Modular pneumatic snake robot: 3D modelling, implementation and control,” Proc. 16th IFAC World Congress, Prague, Czech Republic, July 2005, pp. 1-5.

[6] T. Kamegawa, T. Yamazaki, H. Igarashi, and F. Matsumo, “Development of the snake-like rescue robot ‘Kohga’,” Proc. IEEE Int. Conf. Robotics and Automation, New Orleans, LA, April 2004, pp. 5081-5086.

[7] S.A. Fjerdingen, J.R. Mathiassen, H. Schumann-Olsen, and E. Kyrkebe, “Adaptive Snake Robot Locomotion: A Benchmarking Facility for Experiments,” European Robotics Symposium, Czech Republic, vol. 44, March 2008, pp. 13-22.

[8] A. Wolf, H. Choset, H.B. Brown, and R. Cazciola, “Design and Control of a Mobile Hyper-Redundant Urban Search and Rescue Robot,” International Journal of Advanced Robotics, vol 19, no. 8, 2005, pp. 221-248.

[9] P. Liljeback, Ø. Stavdahl, and A. Beitnes, “SnakeFighter: development of a water hydraulic fire-fighting snake robot,” Proc. IEEE Int. Conf. Control, Automation, Robotics, and Vision, Dec. 2005, pp. 1-6.

[10] N. Takanashi, H. Choset, and J. Burdick, “Experimental Results for Sensor Based Planning for Hyper-redundant Manipulators,” IEEE/RSJ International Conference on Intelligent Robots and Systems, Yokohama, Japan, 1993, pp. 636-643.

[11] H. Choset and J. Burdick, “Extensibility in Local Sensor Based Planning For Hyper Redundant Manipulators (Robot Snakes),” Conference on Intelligent Robotics in Field, Factory, Service and Space (CIRFFSS 1994), Houston, Texas, 1994, pp. 820-829.

[12] W. Henning, F. Hickman, and H. Choset, “Motion Planning for Serpentine Robots,” ASCE Space and Robotics, Albuquerque, New Mexico, 1998, pp. 1-7.

[13] Jinguo Liu, Yuechao Wang, Bin li, and Shugen Ma, “Path planning of a snake-like robot based on serpensoid curve and genetic algorithms,” World Congress on Intelligent Control and Automation, vol. 6, June 2004, pp. 4860-4864.

[14] E.S. Conkur, “Path planning using potential fields for highly redundant manipulators,” Robotics and Autonomous Systems, vol. 52, August 2005, pp. 209-226.

[15] E.S. Conkur and R. Gurbuz, “Path Planning Algorithm for Snake-Like Robot,” Information Technology and Control, 2008, vol.37, no.2, pp. 159-162.

[16] Y. Cheng, P. Jiang, and Y.F. Hu, “A-Snake: Integration of path planning with control for mobile robots with dynamic constraints,” The 2nd International Conference on Computer and Automation Engineering, Feb. 2010, pp. 127-134.

[17] D. Yagnik, R. Jing, and R. Liscano, “Motion planning for multi-link robots using Artificial Potential Fields and modified Simulated Annealing,” IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications, July 2010, pp 421-427.

[18] J.E. Slotine and W. Li, “On the Adaptive Control of Robot Manipulators,” The International Journal of Robotic Research, vol. 6, no. 3, Sep. 1987, pp. 49-59.

[19] J.H. Ginsberg, Advanced Dynamics, New York, NY, Cambridge University Press, 1998.

[20] C. Connolly, R. Weiss, and J. Burns, “Path Planning Using Laplace’s Equation,” IEEE Int. Conf. Robotics and Automation, Cincinnati, OH, May 1990, pp. 2102-2106.

[21] D. Alvarez, J.C. Alvarez, and R.C. Gonzalez, “Online Motion Planning using Laplace Potential Fields,” IEEE Int. Conf. Robotics and Automation, Taipei, Taiwan, Sep. 2003, pp. 3347-3352.

[22] S. Charifa and M. Bikdash, “Comparison of geometrical, kinematic, and dynamic performance of several potential field methods,” IEEE Southeast Conference, Atlanta, GA, March 2009, pp. 18-23.

[23] M. Bikdash and S. Charifa, “Rigid Body Navigation Using a Combined Optimized Potential Field and a Boundary Following Algorithm,” Mathematical and Computational Applications, accepted, 2010.

[24] S. Charifa and M. Bikdash “Adaptive boundary-following algorithm guided by artificial potential field for robot navigation,” IEEE Workshop on Robot Intelligence in Informationally Structured Space, Nashville, TN, March 2009, pp. 38-45.