Path Planning of a Snake-like Robot Based on Serpenoid Curve and Genetic Algorithms

Jinguo Liu, Yuechao Wang and Bin li
Robotics Laboratory of Chinese Academy of Sciences,
Shenyang Institute of Automation (SIA). China
liujinguo@ms.sia.ac.cn

Shugen Ma
Department of Systems Engineering,
Ibaraki University, JAPAN
shugen@dse.ibaraki.ac.jp

Shanghai Jiaotong University also join this research during recent years.

In this paper Serpenoid curve and the SIA snake-like robot were introduced firstly. During the following study we propose a novel control technique for the SIA snake-like robot based on Serpenoid curve. Using real time dual GAs this control technique not only can decide the shortest path and the minimum curvature deviation, but also can limit the motion error’s influence. At last Simulation results show that the technique is effective for the path planning of the SIA snake-like robot.

II. SIA SNAKE-LIKE ROBOT PLATFORM

A. Serpentine Motion

Having these characteristics of catholicity, ground adaptability and high efficiency, serpentine motion (as in Fig. 1) is the basic gait of snake [1,3,4]. Based on experiment and observation over the nature snake, Professor Hirose proposed Serpenoid curve (as in Fig. 2) which changes like a sine wave along the central axis to describe this motion [1]. In this description, the muscle processes the constrigency and extension regularly like sine wave. It matches snake’s usual motion gait.

\[ \rho(s_p) = \frac{2K_p}{L} \alpha_s \sin \left( \frac{2K_p \pi s_p}{L} \right) + k \]  

Fig. 1 Serpentine motion Fig. 2 Serpenoid curve

Hirose has developed various kinds of snake-like robot under the control of Serpenoid curve with satisfaction. So this curve gets approbation and references among the followers [5–8]. The Serpenoid curve is given by the curvature function:

Relative rotation angle of the joints which indicates the locomotion range can be got by integral calculus of the curvature:

\[ \int \rho(s) ds = \frac{2K_p}{L} \alpha_s \sin \left( \frac{2K_p \pi s}{L} \right) + k \]
\[
\theta_i(s) = \int_{\nu_i/2}^{\nu_i/2} \rho(u) \, du = -2\alpha_0 \sin \left(\frac{K_s \pi}{n} \right) \sin \left(\frac{2K_s \pi}{n} s + \frac{2K_s \pi}{n} i \right) + kl
\]

where \(K_s\) gives number of the S-shape, \(\alpha_0\) is the initial winding angle, \(L\) is the whole length of the robot body, \(s_p\) is the body length along the body curve, \(s\) is the ideal displacement along the tail, \(l\) is the single unit length, \(i\) is the unit number and \(k\) is the curvature deviation respectively. While \(k\) changes, the moving direction varies correspondingly. And locomotion velocity can be altered by \(s\)'s change during a fixed period.

B. Snake-like Robot in SIA

With 16 units in total, 8 degrees of freedoms (DOF) in both horizontal and vertical direction, with the length of 1.44m and with the weight of 3.0 Kg, the SIA snake-like robot (as in Fig. 3) has the ability of being accommodating to the environment. Besides implementing serpentine motion, concertina motion, side-winding motion and rectilinear motion, it can climb slope of 20° and get over obstacle about 50 mm.

![Fig. 3 Snake-like robot in serpentine motion](image)

Using CAN bus the snake-like robot's control system is shown in Fig 4. The upper control system is a wireless monitor system which not only can send the commands for motions such as serpentine motion, side-winding motion, rolling and moving forward or backward, but also can change related parameters.

![Fig. 4 The control system](image)

Now the GPS system can distinguish snake-like robot's motion direction, but the accuracy (~10 m) still can't meet the actual need. Therefore both the GPS system and distance sensors are taken into consideration in next snake-like robot. The path planning technique proposed in this paper will provide theory basis for the future snake-like robot.

III. PATH PLANNING TECHNIQUE

A. Nonlinear Path Planning Technique

When the snake-like robot's track is an ideal Serpoid curve, the relative bend angle of each joint can be calculated according to the Serpoid curve function. To see the relationship more clearly, we convert this function into 2D coordinate as follow:

\[
x(s,k) = \int s \cos(\alpha_0 \cos(\frac{2K_s \pi}{L}) + ku) \, du
\]

\[
y(s,k) = \int s \sin(\alpha_0 \cos(\frac{2K_s \pi}{L}) + ku) \, du
\]

where \(\alpha_0\) and \(K_s\) can be decided by the ground information and monitor information. To certain ground they are constants from experiment. \(L\), the length of the body, is also a constant. So to certain object, there are only two unknown parameters \(s\) and \(k\), in (3) and (4). This equation group, (3) and (4), is different from common in that it is nonlinear and cannot be solved directly. For a certain object there are many paths leading to it. Just as in Fig. 5, \(\alpha_0 = \pi / 3\) and \(K_s / L = 1 / (1m)\).

![Fig. 5 Many kinds of path leading to the object](image)

So there are many kinds of solution in above equation group. This is a typical nonlinear path planning problem which is to find the Serpoid curve having the point close to the object. The optimization function can be described as follows:

\[
\min f_i(s,k) = \sqrt{(x(s,k) - x_o)^2 + (y(s,k) - y_o)^2}
\]

![Fig. 6 (a) Contour plane (b) 3D surface graphics](image)

With the same parameters in Fig. 5, the contour plane and 3D surface graphics of function (5) are shown in Fig 6. It is clear that both local optimization and overall optimization exist. To select a best path, more optimization functions are added as follow:
\begin{align}
(1) \text{min} s; \quad (2) \text{min} |k|.
\end{align}

Where term (1) is the shortest path and term (2) is the minimum curvature deviation. Experiment shows that movement error of the snake-like robot increases as curvature deviation increases. Objection function (5) and objection function (6) form the multi-objective optimization (MO) for snake-like robot’s path planning.

B. Range of k and s

In function (5), both k and s are indeterminate. It is known from their definitions that k, curvature deviation, decides the path’s macroscopic bending and \( \alpha \), the initial winding angle, decides the microcosmic bending. Both affect the length of the actual path.

As in Fig. 5, the beginning point is origin and the beginning center axis is the x-axis. The influence of Serpentine curve wave’s amplitude is ignored. That is, if \( \alpha_0 = 0 \), (3) and (4) can be simplified approximately as follow:

\begin{align}
x_d &= \int \cos(k\ u)du \\
y_d &= \int \sin(k\ u)du
\end{align}

(7) (8)

We can obtain:

\begin{align}
x_o = \sin(k\ s')/k' \\
y_o = (1 - \cos(k\ s'))/k' \\
k' = 2y_o/(x_o^2 + y_o^2) \\
s' = \sin^{-1}(k\ x_o)/k' + 2m\pi (m \in Z)
\end{align}

(9) (10) (11) (12)

Calculations show that to arrive at the object with the minimum curvature the range of k can be settled as:

\begin{align}
\left| y_o/(x_o^2 + y_o^2) \right| \leq |k| \leq \left| y_o/(x_o^2 + y_o^2) \right|
\end{align}

(13)

While \( k_o = 3y_o/(x_o^2 + y_o^2) \).

\begin{align}
\left| k_o \right| \leq |k| \leq 3\left| k_o \right|
\end{align}

(14)

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{fig7.png}
\caption{Begin at a random point}
\end{figure}

When to get the shortest path the range of s can be settled as:

\begin{align}
\sqrt{x_o^2 + y_o^2} \leq s \leq 3\sqrt{x_o^2 + y_o^2}
\end{align}

(15)

While \( s_o = \sqrt{x_o^2 + y_o^2} \).

\begin{align}
s_o \leq s \leq 3s_o
\end{align}

(16)

To any beginning point, as \((x_o, y_o)\) in Fig.7, the novel object \((x'_d, y'_d)\) in the novel coordinate \(x' y'\) is given by:

\begin{align}
x'_d = \sqrt{(x_d - x_o)^2 + (y_d - y_o)^2} \cos \theta \\
y'_d = \sqrt{(x_d - x_o)^2 + (y_d - y_o)^2} \sin \theta \\
\theta = \arctan((y_d - y_o)/(x_d - x_o)) - \alpha
\end{align}

(17) (18) (19)

In the novel coordinate the range of k and s can be obtained when \((x'_d, y'_d)\) substitutes for \((x_d, y_d)\) from (7) to (16). In this section we decide the range of k and s to for: (1). without further research we can determine the direction and length of the path to the object; (2). ambit of k and s is limited in the follow algorithm.

C. Real Time Dual GAs

Usually MO can be defined as working out the max value or min value for multi-objective function under a set of constraints. When conflict exists between these objective functions, there is no optimization solution satisfying all. Pareto solution is useful in solving this problem [9]. That is, to transform multi-objective function into one objective function being universal to all.

This model is a compromise model. Under the same constraints, we can get the compromise solution from the compromise model. In fact the compromise solution is the practical effective solution. It can be obtained from three techniques as follow:

(1). setting compromise model by adding all these weighted objective functions;
(2). taking all these objective functions into consideration at first while only the most important one at last;
(3). using human-machine interaction to select the solution;

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{fig8.png}
\caption{Flow chart of real time dual GAs}
\end{figure}

Because of inevitable error of snake-like robot’s locomotion and local optimization of path planning, real time dual GAs has been proposed as the flow chart shown in Fig.8. In the first layer optimization solution of the single objective
function, (5), sets the bound for the optimization solution of the multi-objective function. In the second layer compromise model can be settled by adding all these weighted objective functions and its optimization solution is the satisfied solution. The algorithm is explained in details as follow:

1. Genetic algorithm is a kind of searching technique based on nature selection and genetics principle for global optimization. In particular it is applicable to complicated nonlinear problem which cannot be solved by traditional technique and it is widely applied in mobile robot's path planning and MO [10–12]. Simple GA includes four parts such as initialization, selection, crossover and mutation. Its basic steps are [9]:

   Step 1 Produce binary codes for the independent variable at random and initialize the population;
   Step 2 Decode binary codes into decimalization codes and calculate their fitness;
   Step 3 Reproduce above binary codes having larger fitness, so the superiority can be inherited and pass down;
   Step 4 Crossover above binary codes randomly at some probability;
   Step 5 Mutate above binary codes;
   Step 6 Evaluate every individual in the novel generation population forming in step 5;
   Step 7 Repeat above steps till the best individual meeting the need or the last generation or the fitness increasing not any more.

2. Real time iterative algorithm is used to adjust the object position without interruption. The snake-like robot’s motion is influenced by many indeterminate factors such as ground condition, mechanism's vibration and motor voltage's variety. In this study we propose the real time control technique for the SIA snake-like robot. That is, when the sensor information, wireless control information and GPS information change, the main control unit adjusts the parameters in (3) and (4).

3. Dual GAs are used to decide the path with the shortest length and the minimum curvature deviation to the object. In the first layer single objective optimization, (\( s_{op1}, k_{op1} \)), can be obtained from function (5). When the MO function is constructed, both cause and effect relationship and parataxis relationship exist between function (5) and function (6). Therefore in the second layer GAs, \( s_{op2} \) and \( k_{op2} \) are considered as the upper bound of \( s \) and \( |k| \) respectively.

   Ambit of \( k \) and \( s \) are modified as:

\[
| k | \leq | k_{op1} |
\]

\[
 s_0 \leq s \leq s_{op1}
\]

According to optimize objective function (5) and (6), the weighted objective function for the second layer GAs is expressed as follow:

\[
\min f_c(s, k) = \lambda_1 \left( f_1(s, k)/\epsilon - 1 \right) + \lambda_2 \left( s / s_{op2} - 1 \right) + \lambda_3 \left( |k| / k_{op2} - 1 \right)
\]

where \( \sum_{i=1}^{3} \lambda_i = 1, \lambda_i \geq 0 \) is the weight coefficient and \( \epsilon \) is the distance error precision decided by actual need. The compromise optimization solutions in the second layer GAs (\( s_{op2}, k_{op2} \)) are the ideal results, the shortest length and the minimum curvature deviation. The limited range of \( k \) and \( s \) also shown in Fig. 9.

IV. SIMULATION

To verify the above algorithms, assume that object is (4, 6) and related parameters are set as:

\( a. K_e = 1.0, \alpha_0 = 1.0, L = 1.44m \)
\( b. K_e = 0.8, \alpha_1 = 0.8 \), and \( L = 1.44m \).

In GA-I, chromosome population is 20, crossover probability is 0.6, mutation probability is 0.08 and max evolution generation is 50. While in GA-II, chromosome population is 20, crossover probability is 0.5, mutation probability is 0.05, max evolution generation is 30, \( \lambda_1 = \lambda_2 = \lambda_3 = 1/3 \), and \( \epsilon = 0.001m \).

Table 1 Comparison of the simulation results

<table>
<thead>
<tr>
<th>( s_{op1} )</th>
<th>( k_{op1} )</th>
<th>( s_{op2} )</th>
<th>( k_{op2} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td>11.0817</td>
<td>0.18331</td>
<td>10.9217</td>
</tr>
<tr>
<td>b.</td>
<td>10.0815</td>
<td>0.20009</td>
<td>9.9215</td>
</tr>
</tbody>
</table>

If the snake-like robot achieves object (4, 6) without error, \((s_{op1}, k_{op1})\) and \((s_{op2}, k_{op2})\) can be calculated as in Table 1. It is seen from the table that \( s_{op1} \geq s_{op2} \) and \( k_{op2} > k_{op1} \). When \( \alpha_0 \) and \( K_e \) change, so do \( k \) and \( s \). To see feasibility of the algorithm more clearly the Serpenoid curves from origin to any object are shown in Fig.10.

Fig. 10 Simulation of the path planning technique

(a) the objects on a curve (b) the objects in a line
Related parameters are set the same as group b. Simulation of the path planning technique shows that it can achieve any kind of object with satisfaction.

V. CONCLUSION

In this paper we propose a novel control technique for the SIA snake-like robot based on Serpemoid curve. Using real time dual GAs this control technique not only can decide the shortest path and the minimum curvature deviation but also can limit the motion error's influence. Simulation results show that the technique is effective for the path planning of the SIA snake-like robot. This work will provide theory foundation for the future snake-like robot research.

ACKNOWLEDGMENT

This research is supported partly by China National 863 Program No.2001AA422360.

REFERENCES