Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Complete coverage path planning for an Arnold system based mobile robot to perform specific types of missions*

Cai-hong LI[†‡1], Chun FANG[1], Feng-ying WANG[1], Bin XIA[1], Yong SONG[2]

[1]*College of Computer Science and Technology, Shandong University of Technology, Zibo 255000, China*

[2]*School of Mechanical, Electrical & Information Engineering, Shandong University, Weihai 264209, China*

[†]E-mail: lich@sdut.edu.cn

Received Oct. 14, 2018; Revision accepted Jan. 17, 2019; Crosschecked Sept. 4, 2019

Abstract: We propose a contraction transformation algorithm to plan a complete coverage trajectory for a mobile robot to accomplish specific types of missions based on the Arnold dynamical system. First, we construct a chaotic mobile robot by combining the variable z of the Arnold equation and the kinematic equation of the robot. Second, we construct the candidate sets including the initial points with a relatively high coverage rate of the constructed mobile robot. Then the trajectory is contracted to the current position of the robot based on the designed contraction transformation strategy, to form a continuous complete coverage trajectory to execute the specific types of missions. Compared with the traditional method, the designed algorithm requires no obstacle avoidance to the boundary of the given workplace, possesses a high coverage rate, and keeps the chaotic characteristics of the produced coverage trajectory relatively unchanged, which enables the robot to accomplish special missions with features of completeness, randomness, or unpredictability.

Key words: Chaotic mobile robot; Arnold dynamical system; Contraction transformation; Complete coverage path planning; Candidate set

https://doi.org/10.1631/FITEE.1800616 **CLC number:** TP242.6

1 Introduction

With the development of artificial intelligence and robotics, the research field in autonomous mobile robots has become a topic of great interest because of the robots' ever-increasing applications across a wide range of activities. This research is needed for a variety of applications such as vacuum cleaning robots (Park et al., 2012), lawn mowers (Ousingsawat and Earl, 2007), harvesting (Oksanen and Visala, 2009), and exploring underwater sources (Galceran and Carreras, 2013; Torres et al., 2016; Zheng et al., 2016). The above applications all involve a key issue, i.e., complete coverage path planning, which addresses the problem of requiring a traverse for every feasible area in a given workplace while avoiding obstacles. However, there exist some other specific types of missions, such as patrolling (Martins-Filho and Macau, 2007a; Hwang et al., 2011; Banerjee et al., 2015; Curiac and Volosencu, 2015a), surveillance (Martins-Filho and Macau, 2007b; Curiac and Volosencu, 2015b), and demining (Prado and Marques, 2014), which require not only complete coverage of the entire terrain, but also finding explosives and intruders that pursue an unpredictable path. These tasks are achieved by designing a chaotic path planner or controller.

Sekiguchi and Nakamura (1999) first constructed a chaotic mobile robot based on the Arnold

[‡] Corresponding author

[*] Project supported by the National Natural Science Foundation of China (Nos. 61473179, 61602280, and 61573213), the Natural Science Foundation of Shandong Province, China (Nos. ZR2017MF047, ZR2015CM016, and ZR2014FM007), and the Shandong University of Technology & Zibo City Integration Development Project, China (No. 2018ZBXC295)

ORCID: Cai-hong LI, http://orcid.org/0000-0003-0255-9249

dynamical system, in which the chaotic behavior of the Arnold dynamical system is imparted to the mobile robot's motion control. Since then, the interaction between mobile robotics and chaos theory has been studied. The main features of chaotic systems are their topological transitivity and their sensitive dependence on the initial conditions (Lorenz, 1997). It is the topological transitivity that guarantees a chaotic mobile robot to patrol the whole surveillance region completely without repetitions. The sensitive dependence on the initial conditions is a desirable characteristic for patrol robots, since the trajectory produced by the chaotic mobile robot is unpredictable or random (Volos et al., 2012a). A chaotic system is very different from a random signal. Chaotic motion is based on determinism, which in the case of mobile robots is an advantage. This happens because the behavior of a robot can be predicted in advance by the system designer.

There are two strategies to construct a chaotic mobile robot. One of them is to use a chaotic dynamical system to obtain the coordinates of the mobile robot directly; this includes two-dimensional (2D) systems, namely a standard map (Martins-Filho and Macau, 2007a; Sooraska and Klomkarn, 2010) and a Henon map (Curiac and Volosencu, 2014). The other method is to use one of the variables of the dynamical chaotic system combined with the kinematic equation of the mobile robot to construct a chaotic path planner or controller for the robot, for example, the one-dimensional (1D) system of a logistic map (Volos et al., 2012b), and the three-dimensional (3D) systems, namely Arnold (Nakamura and Sekiguchi, 2001; Volos et al., 2013), Lorenz (Li et al., 2016), and Chua (Volos et al., 2013). These two methods can both produce complete and unpredictable trajectories to satisfy the requirements of the specific types of missions previously mentioned, but they also have disadvantages. The first method of construction is simple. The trajectory produced is distributed relatively uniformly (Li et al., 2018), or can be improved easily, such as in the Chebychev system, which can be used as two 1D systems to form a 2D system (Li et al., 2017). Nevertheless, the distances between the adjacent sub-goals produced by the constructed path planner may be very large, because they are produced by the chaotic system directly and cannot be controlled by the designer. This phenomenon can cause some problems, such as in the case of obstacle avoidance between two planned sub-goals (Li et al., 2015), path tracking problems, etc. The distance step of the planned path generated by the second approach can be controlled by the designer and executed directly by the robot. One of its main drawbacks is an obstacle avoidance problem along the boundaries of the workplace and the obstacles. Researchers have always used a mirror mapping strategy (Nakamura and Sekiguchi, 2001; Fahmy, 2012; Volos et al., 2013; Liu et al., 2017) to solve this problem although it can cause unevenness and repeatability of the coverage trajectory. While most of the mentioned researchers' work has addressed chaotic surveillance of an area, Curiac and Volosencu (2009, 2012) studied the surveillance of a number of specified locations, and provided random-like sequences of points for fictitious obstacles (Curiac and Volosencu, 2015b; Curiac et al., 2018) based on the chaotic dynamics.

Compared with other well-known low-dimensional continuous chaotic systems, the 3D Arnold equation has some advantages in that the structures of the Arnold equation and the mobile robot equation are similar. It is easy to deal with its variables. Also, it has a higher coverage rate (Nakamura and Sekiguchi, 2001). In this study, we use the Arnold dynamical system integrated with the kinematic equation of the mobile robot to construct a chaotic path planner to achieve complete coverage path planning for specific types of missions. A contracted transformation strategy is designed to form the continuous trajectory and solve the problems that exist in the common mirror mapping method.

2 Arnold dynamical system

The Arnold dynamical system describes a steady solution to the 3D Euler equation which expresses the behaviors of noncompressive perfect fluids on a 3D torus space, which is written as follows:

$$\begin{cases}\dot{x} = A\sin z + C\cos y,\\ \dot{y} = B\sin x + A\cos z,\\ \dot{z} = C\sin y + B\cos x,\end{cases} \tag{1}$$

where A, B, and C are constants, and (x, y, z) denotes the position of a particle. It is known that the Arnold equation shows periodic motion when one of the constants, for example C, is 0 or small and shows chaotic motion when C is large (Nakamura and Sekiguchi, 2001). There exist some combinations that can put the system into a chaotic state. Here we select a commonly used group, namely, [A, B, C]=[1, 0.5, 0.5], to construct the chaotic mobile robot. The following is the model used:

$$\begin{cases} \dot{x} = \sin z + 0.5\cos y, \\ \dot{y} = 0.5\sin x + \cos z, \\ \dot{z} = 0.5\sin y + 0.5\cos x. \end{cases} \tag{2}$$

Fig. 1 shows the 3D phase space and 2D phase space of Eq. (2), where the equation should be discretized first in a iterative form, the initial state is (x_0, y_0, z_0)=(4, 3.5, 0), the iterative step in its discretization form is h=0.1, and the number of iterations is n=2000.

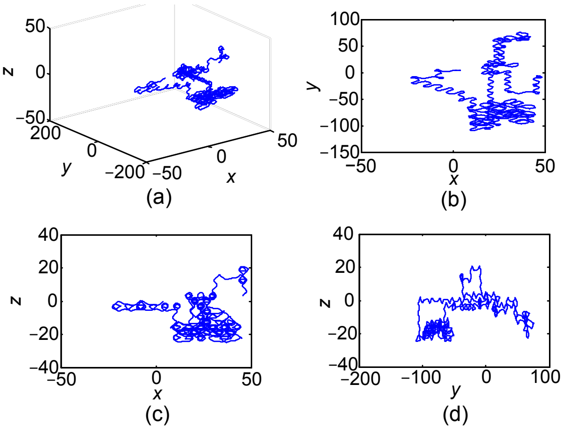

Fig. 1 Phase space: (a) *x*-*y*-*z*; (b) *x*-*y*; (c) *x*-*z*; (d) *y*-*z*

The Lyapunov exponent is used as a measure of sensitive dependence on the initial conditions, which is one of the two characteristics of chaotic behavior (Peitgen et al., 2004). There are n Lyapunov exponents in an n-dimensional state space, and the system is concluded to have sensitive dependence on the initial conditions when the maximum Lyapunov exponent $\lambda_{\max}$ is positive, and the system is concluded to be in a chaotic state. The Lyapunov exponents calculated by different methods are different. Sampling 10 000 time series of variable z, we calculate the Lyapunov exponents of the Arnold equation by its discretization form and definition. The parameters and initial states are as follows:

Coefficients: A=1, B=0.5, C=0.5;

Initial states: x_1=4, x_2=3.5, x_3=0;

Iterative step in the discretization form of the equation: h=0.001;

Number of iterations: t=10 000;

Lyapunov exponents obtained: λ_1=−2.0229×10^{-6}, λ_2=3.8362×10^{-6}, λ_3=−1.8132×10^{-6}.

Since the maximum exponent λ_2 is positive, the Arnold equation has sensitive dependence on the initial conditions. Then Eq. (2) expresses a chaotic dynamical system.

In this study, we assume the chaotic Arnold system (2) to construct the chaotic mobile robot based on the above settings for the initial conditions and the iterative step h.

3 Construction and analysis of the chaotic mobile robot

The Arnold equation is a continuous formula and has three variables. First, the construction method of the chaotic mobile robot is described in detail. Second, the discretization procedure to solve the differential equations of the constructed chaotic mobile robot is introduced. Finally, the coverage efficiency of the chaotic mobile robot constructed by each variable is computed and discussed.

3.1 Construction of the chaotic mobile robot

To generate the chaotic mobile robot, one of the variables of the Arnold equation is employed for integration with the kinematic model of the mobile robot. Here, variable z is used. The mobile robot considered in this work is a typical two-wheeled mobile robot (Sekiguchi and Nakamura, 1999), and the kinematic model is as follows:

$$\begin{cases} \dot{x} = v(t)\cos\theta(t), \\ \dot{y} = v(t)\sin\theta(t), \\ \dot{z} = \omega(t), \end{cases} \tag{3}$$

where $v(t)$ is the linear velocity which can be supposed as a constant value v, $\theta(t)$ is orientation, and $\omega(t)$ is the angular velocity. Choosing one variable from Eq. (2) of the Arnold equation, the chaotic mobile robot can be constructed by replacing $\theta(t)$ with it. There are two strategies to design it according to whether the linear velocity $v(t)$ is a constant or variable.

If $v(t)$ is a variable, the designed chaotic mobile robot is as follows:

$$\begin{cases} \dot{x} = \sin z + 0.5\cos y, \\ \dot{y} = 0.5\sin x + \cos z, \\ \dot{z} = 0.5\sin y + 0.5\cos x, \\ \dot{x}_{\mathrm{r}} = v(t)\cdot\cos z, \\ \dot{y}_{\mathrm{r}} = v(t)\cdot\sin z, \\ \dot{\theta} = \omega(t), \end{cases} \tag{4}$$

$$\begin{cases} v(t) = \dfrac{1}{2}\big(v_{\mathrm{r}}(t) + v_{\mathrm{l}}(t)\big), \\ \omega(t) = \dfrac{1}{L}\big(v_{\mathrm{r}}(t) - v_{\mathrm{l}}(t)\big), \end{cases} \tag{5}$$

where $(x_{\mathrm{r}}, y_{\mathrm{r}})$ is the coordinate position of the robot, $v_{\mathrm{r}}(t)$ is the linear velocity of the right wheel of the mobile robot, $v_{\mathrm{l}}(t)$ is the linear velocity of the left wheel, and L is the distance between the left and right wheels. The chaotic mobile robot constructed based on Eq. (4) includes six first-order differential equations. It produces the coverage trajectory by controlling the values of $v_{\mathrm{r}}(t)$, $v_{\mathrm{l}}(t)$, and z. If the linear velocity $v(t)$ is supposed as a constant value v, the designed chaotic mobile robot can be simplified as follows:

$$\begin{cases} \dot{x} = \sin z + 0.5\cos y, \\ \dot{y} = 0.5\sin x + \cos z, \\ \dot{z} = 0.5\sin y + 0.5\cos x, \\ \dot{x}_{\mathrm{r}} = v\cdot\cos z, \\ \dot{y}_{\mathrm{r}} = v\cdot\sin z. \end{cases} \tag{6}$$

Five first-order differential equations are included and the coverage trajectory is produced by controlling the angular velocity of the robot, namely variable z. All of the states evolve in a five-dimensional (5D) space according to Eq. (6), or in a six-dimensional (6D) space according to Eq. (4), where both include the 3D subspace of the Arnold system. Thus, the models constructed possess chaotic characteristics. Since we only want to verify the efficiency of the path planning strategy, the simpler model constructed based on Eq. (6) is used.

In Eq. (6) the same procedure is used to design the chaotic mobile robot with x and y of the Arnold equation as it does with z, and z is replaced in the fourth and fifth equations of Eq. (6) with x or y.

3.2 Solution of the chaotic mobile robot

Eq. (6) includes five first-order differential equations. The five variables, besides the robot covering trajectory, can be computed by discretizing the differential equations. For the Arnold equation, the second-order Runge Kutta method is enough to obtain accurate results of the variables. Its common model is as follows:

$$\begin{cases} Y_{n+1} = Y_n + 0.5k_1 + 0.5k_2, \\ k_1 = hf(X_n, Y_n), \\ k_2 = hf(X_n + h, Y_n + k_1). \end{cases} \tag{7}$$

Then the discretization parameters of the differential equations in Eq. (6) are

$$\begin{cases} K_{1,n}^{x} = h(A\sin z_n + C\cos y_n), \\ K_{1,n}^{y} = h(B\sin z_n + A\cos y_n), \\ K_{1,n}^{z} = h(C\sin z_n + B\cos y_n), \\ K_{1,n}^{x_{\mathrm{r}}} = hv\cos z_n, \\ K_{1,n}^{y_{\mathrm{r}}} = hv\sin z_n, \\ K_{2,n}^{x} = h\big(A\sin(z_n + K_{1,n}^{z}) + C\cos(y_n + K_{1,n}^{z})\big), \\ K_{2,n}^{y} = h\big(B\sin(x_n + K_{1,n}^{z}) + A\cos(z_n + K_{1,n}^{z})\big), \\ K_{2,n}^{z} = h\big(C\sin(y_n + K_{1,n}^{z}) + B\cos(x_n + K_{1,n}^{z})\big), \\ K_{2,n}^{x_{\mathrm{r}}} = hv\cos(z_n + K_{1,n}^{z}), \\ K_{2,n}^{y_{\mathrm{r}}} = hv\sin(z_n + K_{1,n}^{z}). \end{cases} \tag{8}$$

Finally, the discretization formula of model (7) to the chaotic mobile robot is

$$\begin{cases} x_n = x_{n-1} + 0.5K_{1,n-1}^{x} + 0.5K_{2,n-1}^{x}, \\ y_n = y_{n-1} + 0.5K_{1,n-1}^{y} + 0.5K_{2,n-1}^{y}, \\ z_n = z_{n-1} + 0.5K_{1,n-1}^{z} + 0.5K_{2,n-1}^{z}, \\ x_{\mathrm{r},n} = X_{n-1} + 0.5K_{1,n-1}^{X} + 0.5K_{2,n-1}^{X}, \\ y_{\mathrm{r},n} = Y_{n-1} + 0.5K_{1,n-1}^{Y} + 0.5K_{2,n-1}^{Y}. \end{cases} \tag{9}$$

Given an initial state $(x_0, y_0, z_0, x_{\mathrm{r}.0}, y_{\mathrm{r}.0})$, the five variables $(x_n, y_n, z_n, x_{\mathrm{r}.n}, y_{\mathrm{r}.n})$, including the covering trajectory $(x_{\mathrm{r}.n}, y_{\mathrm{r}.n})$ of the mobile robot, can be computed step by step. Here, (x_0, y_0, z_0) is the initial state of the Arnold equation, and (x_0, y_0, z_0)=(4, 3.5, 0). It remains unchanged in this study. $(x_{\mathrm{r}.0}, y_{\mathrm{r}.0})$ is the start point of the robot in a given workplace, which can be given arbitrarily in the running domain.

Fig. 2 shows the covering trajectories produced by the chaotic mobile robot constructed based on Eqs. (8) and (9) at different numbers of iterations, where $(x_{r.0}, y_{r.0})$=(1, 1). In the figures, the blue lines show the covering trajectories, the red "O" denotes the start point, and the black "◇" expresses the end point. From Fig. 2, we can see that the chaotic mobile robot can produce a continuous covering trajectory based on a given initial state.

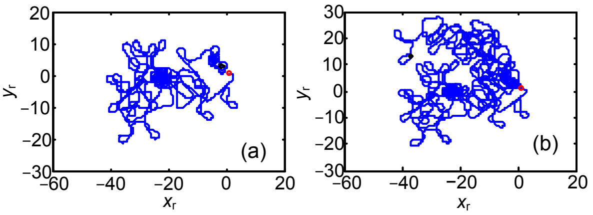

Fig. 2 Coverage trajectories produced by the proposed chaotic mobile robot: (a) *n*=800; (b) *n*=1500
References to color refer to the online version of this figure

In the following verification, only the initial state $(x_{r.0}, y_{r.0})$ of the robot and the iteration number n are changed according to the experimental requirements.

3.3 Coverage efficiency discussion of each variable

All the variables x, y, and z in the chaotic Arnold equation can be used to construct the chaotic mobile robot. Which one has the best coverage efficiency demonstrated by the constructed chaotic mobile robot? The coverage rate is commonly used to evaluate the coverage efficiency. It is a ratio of the covered terrain Ω_C to the total workplace Ω_{mn}, defined as

$$\text{Coverage\_rate} = \Omega_C / \Omega_{mn} \times 100\%. \tag{10}$$

The workplace is commonly discretized into a grid map to simplify the computation of the coverage rate. In this circumstance, the coverage rate is the total number of grids covered:

$$\text{Coverage\_rate} = \frac{1}{M}\sum_{i=1}^{M} I(i), \tag{11}$$

$$I(i) = \begin{cases} 1, & \text{when the grid is covered,} \\ 0, & \text{when the grid is not covered,} \end{cases} \tag{12}$$

where $I(i)$ (i=1, 2, ..., M) is the coverage situation for each grid in the divided workplace, and M is the total number of divided grids in the workplace.

Even though the parameters of the Arnold system and the iterative step are all the same, the coverage rate can be very different at different start points. We analyze the coverage rate of the three variables based on 10 identical start points, at n=800. The results are shown in Table 1. The mean and standard deviation of the coverage rates in Table 1 are calculated and illustrated in Table 2.

Table 1 Coverage rates of the three variables

Start point	Coverage_rate (%)		
$(x_{r.0}, y_{r.0})$	x	y	z
(0, 0)	56.76	44.25	41.25
(1, 1)	66.25	35.75	58.50
(2, 2)	27.75	38.75	40.75
(3, 3)	38.25	42.25	35.25
(4, 4)	34.75	53.25	56.00
(5, 5)	39.25	46.00	47.50
(6, 6)	33.25	40.00	40.50
(7, 7)	43.25	46.50	57.00
(8, 8)	54.75	22.00	49.00
(9, 9)	60.00	28.75	50.25

Table 2 Statistics for the coverage rates of the three variables

Statistics variable	x	y	z
Mean (%)	45.426	39.75	47.60
Standard deviation	13.0405	9.1074	8.0062

4 Characteristics of the coverage trajectory

The coverage trajectory produced by the chaotic mobile robot constructed also has the two main chaotic features, namely sensitivity to the initial conditions and topological transitivity. These features can guarantee generation of the unpredictable and complete coverage trajectory for the mobile robot under the specific types of missions.

4.1 Sensitivity to the initial conditions

Sensitivity to the initial conditions means that two arbitrarily close start points can produce very different iterative trajectories after several iterative steps. Fig. 3 exemplifies the feature based on two produced trajectories which have basically the same start points. The start point of the blue line is $(x_{r.0}, y_{r.0})$ =(1, 1), and the purple dotted line is $(x_{r.0}, y_{r.0})$=(1.01, 1.01). The two red circles "O" denote the start points at almost the same position. However, after several iterations, the two trajectories end at very different positions, expressed by the black "◇."

From this, the following conclusions can be obtained:

1. The trajectory produced by the chaotic mobile robot has sensitivity to the initial conditions. Thus, it meets the needs of the special missions for the mobile robot due to its randomness or unpredictability.

2. The constructed system can produce rich iterative trajectories. Just tiny differences in the initial conditions of the robot are able to generate very different trajectories. This can satisfy different coverage requirements.

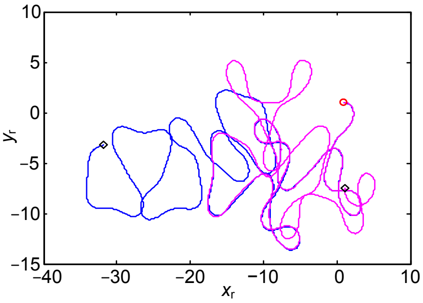

Fig. 3 Sensitivity to the initial conditions of the produced trajectories

References to color refer to the online version of this figure

4.2 Coverage characteristics

The system constructed has topological transitivity, so it can produce the coverage trajectory to survey the whole workplace. Yet, the trajectory is not limited to a constant area, which is shown in Fig. 2. With an increase in the number of iterations n, the covered area increases. Fig. 2b has a larger iteration number, and also possesses a larger covered area, but the boundary of the workplace changes.

If the produced trajectory is bound to a given workplace, what will happen? Suppose the given workplace is Ω_{mn}, and the size is $m\times n$=20×20. The start point of the robot is $(x_{r.0}, y_{r.0})$=(10, 10). Fig. 4 illustrates the coverage trajectories at different numbers of iterations. It is shown that in the beginning the covered area increases with the increase in the number of iterations n. When n>4000, the covered area in Ω_{mn} remains unchanged due to the fact that most of the coverage trajectory runs out of the given workplace and never comes back. Fig. 4d has the same covered area as Fig. 4c, though its number of iterations displays real differences. This phenomenon can reduce the work efficiency of the mobile robot, and it is difficult to obtain a 100% coverage rate of a given workplace. Other start points in the workplace have a similar problem, or the situation is worse wherein most of the trajectories run out of the boundary and never come back.

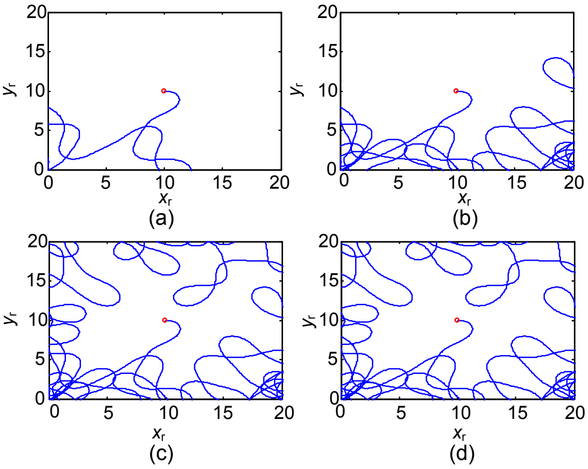

Fig. 4 The coverage trajectories in a given workplace: (a) n=100; (b) n=3000; (c) n=4000; (d) n=10 000

Is it possible to obtain a high coverage rate while compressing the whole coverage trajectory into a given workplace? The answer is no. Fig. 5 exhibits the compressed results for all of the coverage trajectories produced in Fig. 4. It shows that the coverage rate is growing step by step with an increase of n in the beginning. When n is larger than a certain value, the coverage rate is decreasing. Twenty data samples of the coverage rate at different n's are listed in Fig. 6 and Table 3. Fig. 6 shows the change of the coverage rate very intuitively. High values are obtained when n is not very large, while a very small value is obtained when n is larger than a certain value. Table 3 shows that the coverage rate reaches the maximum value 66% at n=9300, while it reaches the sub-maximum value 62.5% at n=1200. The coverage rates of the coverage trajectories produced by other start points demonstrate a similar feature. Fig. 7 shows an example for $(x_{r.0}, y_{r.0})$=(3, 3). It also obtains a higher coverage rate at a smaller number of iterations.

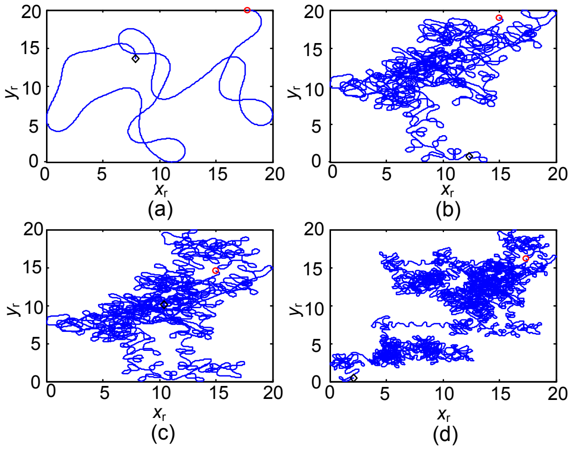

Fig. 5 Compressed results of the coverage trajectories: (a) n=100; (b) n=3000; (c) n=4000; (d) n=10 000

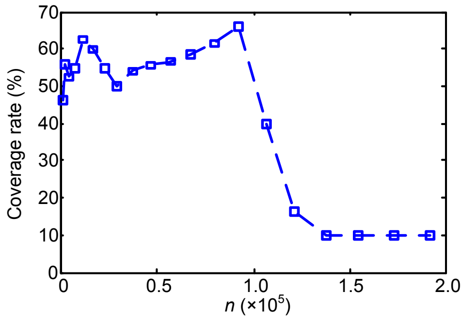

Fig. 6 The coverage rate vs. the iteration number n at $(x_{r.0}, y_{r.0})$=(10, 10)

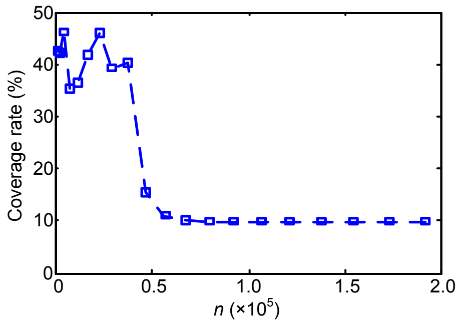

Fig. 7 The coverage rate vs. the iteration number n at $(x_{r.0}, y_{r.0})$=(3, 3)

Given the above discussions, the problems existing in coverage path planning for the constructed chaotic mobile robot are as follows:

1. The robot may run out of the workplace and never come back while using the coverage trajectory produced by the constructed chaotic mobile robot directly.

Table 3 Coverage rate at different iteration number n at $(x_{r.0}, y_{r.0})$=(10, 10)

Iteration number n	Coverage_rate (%)	Iteration number n	Coverage_rate (%)
200	46.25	5700	56.50
300	55.75	6800	58.50
500	52.50	8000	61.50
800	54.50	9300	66.00
1200	62.50	10 700	39.75
1700	59.75	12 200	16.25
2300	54.75	13 800	10.00
3000	49.75	15 500	10.00
3800	54.00	17 300	9.75
4700	55.50	19 200	9.75

2. The coverage rate can reach a high value at a certain number of iterations while using the compressed coverage trajectory, but not 100%. This may not satisfy the complete coverage need of the special tasks.

3. Not only does the number of iterations influence the coverage rate, but the start point of the chaotic mobile robot does. However, a real robot can start from any initial point in a given workplace. A random start point cannot guarantee having a higher coverage rate.

A strategy of complete coverage path planning should be considered to enable the constructed mobile robot to accomplish the specific types of missions, with a relatively small iteration number and a higher coverage rate in a given workplace.

5 Contraction transformation strategy

To overcome the above problems, a contraction transformation strategy is proposed to accomplish complete coverage path planning for the special mission. It includes two parts:

1. Construction of the candidate sets of the start points with a higher coverage rate

There are rich trajectories with a higher coverage rate that can be produced by the start points in a given workplace due to the characteristics of sensitivity to the initial conditions. Theoretically, it is possible to find the start points that are distributed in all parts of the running workplace to satisfy the need. The selected start points $(x_{r.0}, y_{r.0})$ are recorded in the candidate set Set$(x_{r.0}, y_{r.0})$. The coverage trajectory produced by the start points in Set($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) is mapped to the given workplace. Then start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) after mapping are recorded in candidate set Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$).

2. Contraction transformation

For a given target point in the running workplace, the coverage trajectory produced by one start point ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) which is nearer the target point is contracted to it. This can link two coverage trajectories to form a continuous trajectory for the coverage task.

Given the above two-part strategy, a continuous trajectory with a higher coverage rate can be formed to achieve complete coverage path planning.

5.1 Construction of the candidate sets

Suppose there is a given running workplace, Ω_{mn}, and the size is $m\times n$=20×20. The coverage trajectories produced by the chaotic mobile robot constructed are mapped to Ω_{mn}, and their start points are denoted as ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$), where $x_{\mathrm{r}.0}\in[0, 10]$, $y_{\mathrm{r}.0}\in[0, 10]$. After mapping, the corresponding start point ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) changes to a certain point in Ω_{mn}, expressed as ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$), where ($x'_{\mathrm{r}.0}$)∈[0, 20], ($y'_{\mathrm{r}.0}$)∈[0, 20]. The candidate set Set($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) includes the start points ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$), which have higher coverage rates (here suppose Coverage_ rate≥40%). After mapping, the corresponding set is changed to Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$), Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$)={($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$), n, Coverage_ rate$_n$}. The final goal of Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) is to contain enough start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) to cover each area of Ω_{mn}.

Figs. 6 and 7 show that the chaotic mobile robot can obtain a higher coverage rate with 1000 iterations. Thus, we sample the start points around this number of iterations. To improve the sampling efficiency, 12 data samples of the start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) are collected. When the coverage rate for the first data sample is greater than 40%, these group start points are collected in Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$). The construction procedures for Set($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) and Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) are shown in Algorithm 1.

Function(Path_controller, ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$)) is the chaotic mobile robot constructed. It produces the coverage trajectory ($x_{\mathrm{r}.n}$, $y_{\mathrm{r}.n}$) based on the given start point ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$). The coverage trajectory ($x_{\mathrm{r}.n}$, $y_{\mathrm{r}.n}$) produced above is mapped to the running workplace Ω_{mn} by function Map(($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$), Ω_{mn}), and the trajectory ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) is obtained. Ω_{mn} is discretized into a grid map, and the number of grids is $m\times n$. Then the coverage rate for the start point of the compressed coverage trajectory ($x'_{\mathrm{r}.n}$, $y'_{\mathrm{r}.n}$) is calculated by Function([$x'_{\mathrm{r}.n}$, $y'_{\mathrm{r}.n}$], $m\times n$) based on Eqs. (9) and (10).

Algorithm 1 Construction of the candidate sets

```
Set($x_{r.0}$, $y_{r.0}$)=[]; Set($x'_{r.0}$, $y'_{r.0}$)=[];
Value$_{min}$=0; Value$_{max}$=10; Value$_{step}$=0.1;
$n_{min}$=800; $n_{max}$=1900; $n_{step}$=100;
for $x_{r.0}$=Value$_{min}$:Value$_{step}$:Value$_{max}$;
  for $y_{r.0}$=Value$_{min}$:Value$_{step}$:Value$_{max}$;
    [$x_{r.n}$, $y_{r.n}$]=Function(Path_controller, ($x_{r.0}$, $y_{r.0}$));
    [$x'_{r.n}$, $y'_{r.n}$]=Map(($x_{r.0}$, $y_{r.0}$), $\Omega_{mn}$);
    Coverage_rate$_n$=Function([$x'_{r.n}$, $y'_{r.n}$], $m\times n$);
    if Coverage_rate$_n$≥40%
      for $n$=$n_{min}$:$n_{step}$: $n_{max}$
        Set($x_{r.0}$, $y_{r.0}$)←{($x_{r.0}$, $y_{r.0}$), $n$};
        Set($x'_{r.0}$, $y'_{r.0}$)←{($x'_{r.0}$, $y'_{r.0}$), $n$, Coverage_rate$_n$};
      end for
    end if
  end for
end for
return {Set($x_{r.0}$, $y_{r.0}$), Set($x'_{r.0}$, $y'_{r.0}$)};
```

When ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$)=(0, 0), the produced coverage trajectory is shown in Fig. 8a, at n=800, and Fig. 8b is the mapped figure in Ω_{mn}. After mapping, the position of the start point changes, denoted as ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$). The computed coverage rate in Fig. 8b is 41.25%. Since it is greater than 40%, the chaotic mobile robot produces 12 coverage trajectories in turn based on the start point (0, 0), from n=800. Each time n increases by 100. Then the 12 coverage trajectories produced are mapped to Ω_{mn} and 12 different start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) emerge. They are collected into Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$).

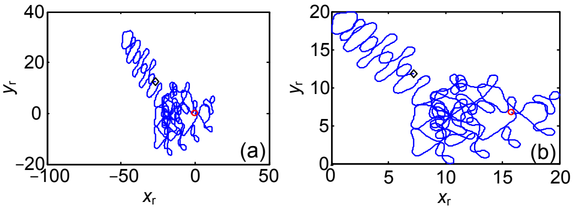

Fig. 8 The coverage trajectories at ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$)=(0, 0), n=800: (a) coverage trajectory produced by the chaotic mobile robot; (b) coverage trajectory mapped in Ω_{mn}

Fig. 9 shows the coverage rates of the 12 mapped coverage trajectories, n=800–1900. Fig. 10 illustrates the 12 start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) in Ω_{mn}. The black “O” denotes the first start point, at n=800. The number of iterations of the points in other positions can be deduced based on the first start point, and the difference is 100 in turn.

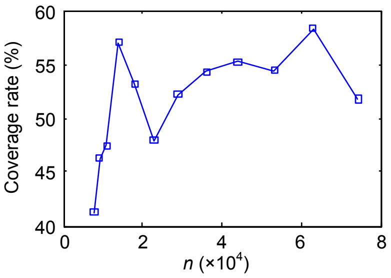

Fig. 9 The coverage rates at ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$)=(0, 0), n=800–1900

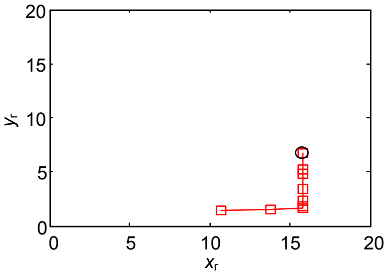

Fig. 10 The 12 start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) in Ω_{mn}

The parts for the constructed candidate sets Set($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) and Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) are shown in Fig. 11. Fig. 11a illustrates the start points ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) which can produce the coverage trajectories with higher coverage rates. Based on the selected start point, 12 coverage trajectories with different numbers of iterations are produced and mapped to Ω_{mn}. Then the 12 start points ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) of the mapped coverage trajectories are depicted in Fig. 11b. The marks labeled with the same color in Fig. 11 indicate the corresponding sampled groups.

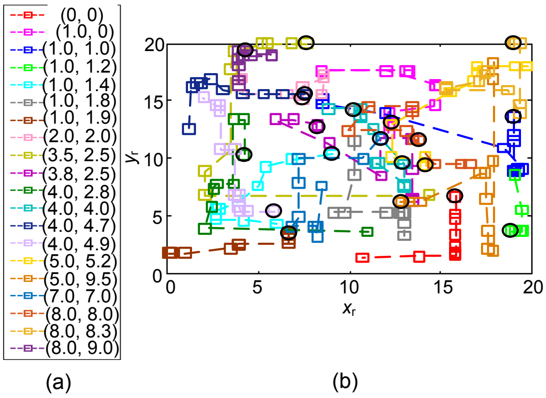

Fig. 11 The selected start points in Set($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$) (a) and the corresponding start point group in Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) (b)

5.2 Contraction transformation algorithm

The robot can start from any position in a given environment Ω_{mn}. Suppose the position is Rob$_{\_\mathrm{start}}$ ($x_{\mathrm{r}.0}$, $y_{\mathrm{r}.0}$). A start point ($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$) in Set($x'_{\mathrm{r}.0}$, $y'_{\mathrm{r}.0}$), which is closer to Rob$_{\_\mathrm{start}}$, is selected to produce the coverage trajectory and contracted to the position of Rob$_{\_\mathrm{start}}$. The contracted trajectory is used as part of the real trajectory of the robot. It has an end point that is regarded as the target contracted point of the next time. The above procedure is repeated until the whole coverage task finishes.

The goal of contraction transformation is to contract the start point Trej$_{2\_\mathrm{start}}$($x_{\mathrm{r}2\_\mathrm{start}}$, $y_{\mathrm{r}2\_\mathrm{start}}$) of the second trajectory Trej$_2$($x_{\mathrm{r}2}$, $y_{\mathrm{r}2}$) to the end point Trej$_{1\_\mathrm{end}}$($x_{\mathrm{r}1\_\mathrm{end}}$, $y_{\mathrm{r}1\_\mathrm{end}}$) of the first trajectory. After contraction, the second trajectory Trej$_2$($x_{\mathrm{r}2}$, $y_{\mathrm{r}2}$) changes to the third trajectory Trej$_3$($x_{\mathrm{r}3}$, $y_{\mathrm{r}3}$).

The contraction procedure includes two parts: contraction of the horizontal coordinate and contraction of the longitudinal coordinate for the produced coverage trajectory. Suppose the parameter of contraction of the horizontal coordinate is k_x, and the parameter of contraction of the longitudinal coordinate is k_y. The detailed procedure is as follows.

5.2.1 Contraction of the horizontal coordinate

If Trej$_{2\_\mathrm{start}}$ is on the right side of Trej$_{1\_\mathrm{end}}$, the second trajectory Trej$_2$ should be contracted to the left side, while keeping its minimum value of the horizontal coordinate unchanged until the values of the horizontal coordinate of the two points are equal.

$$k_x = x_{\mathrm{r}2\_\mathrm{start}} \big/ x_{\mathrm{r}1\_\mathrm{end}}, \tag{13}$$

$$x_{\mathrm{r}3} = x_{\mathrm{r}2} \big/ k_x. \tag{14}$$

Otherwise, the second trajectory Trej$_2$ should be contracted to the right side, while keeping its maximum value of the horizontal coordinate unchanged until the values of the horizontal coordinate of the two points are equal.

$$k_x = (m - x_{\mathrm{r}1\_\mathrm{end}}) \big/ (m - x_{\mathrm{r}2\_\mathrm{start}}), \tag{15}$$

$$x_{\mathrm{r}3} = m(1 - k_x) + k_x \cdot x_{\mathrm{r}2}. \tag{16}$$

5.2.2 Contraction of the longitudinal coordinate

If Trej$_{2\_\mathrm{start}}$ is below Trej$_{1\_\mathrm{end}}$, the second trajectory Trej$_2$ should be contracted to the downside, while keeping its minimum value of the longitudinal coordinate unchanged until the values of the horizontal coordinate of the two points are equal.

$$k_y = y_{\mathrm{r2\_start}} / y_{\mathrm{r1\_end}}, \tag{17}$$

$$y_{\mathrm{r3}} = y_{\mathrm{r2}} / k_y. \tag{18}$$

Otherwise, the second trajectory Trej$_2$ should be contracted to the upside, while keeping its maximum value of the longitudinal coordinate unchanged until the values of the horizontal coordinate of the two points are equal.

$$k_y = (m - y_{\mathrm{r1\_end}}) / (m - y_{\mathrm{r2\_start}}), \tag{19}$$

$$y_{\mathrm{r3}} = m(1-k_y) + k_y \cdot y_{\mathrm{r2}}. \tag{20}$$

Fig. 12 shows the flowchart of the algorithm.

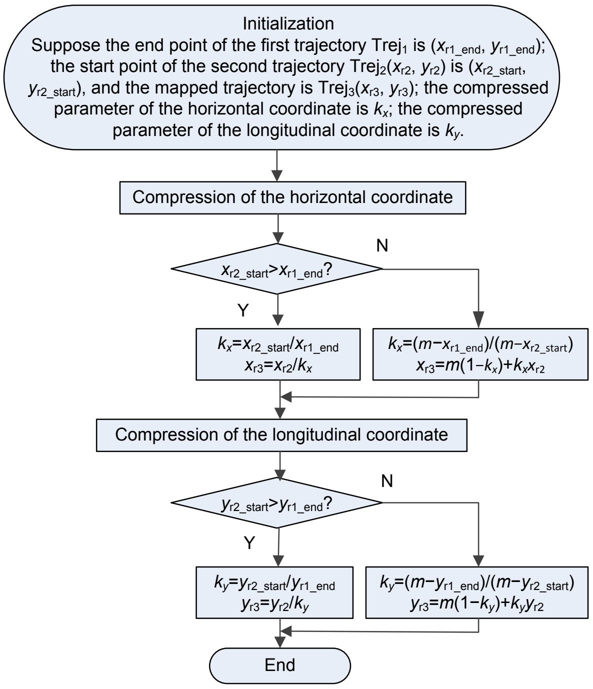
Initialization
Suppose the end point of the first trajectory Trej$_1$ is ($x_{\mathrm{r1\_end}}$, $y_{\mathrm{r1\_end}}$); the start point of the second trajectory Trej$_2$(x_{r2}, y_{r2}) is ($x_{\mathrm{r2\_start}}$, $y_{\mathrm{r2\_start}}$), and the mapped trajectory is Trej$_3$(x_{r3}, y_{r3}); the compressed parameter of the horizontal coordinate is k_x; the compressed parameter of the longitudinal coordinate is k_y.
Compression of the horizontal coordinate
$x_{\mathrm{r2\_start}}>x_{\mathrm{r1\_end}}$?
N
Y
$k_x=x_{\mathrm{r2\_start}}/x_{\mathrm{r1\_end}}$
$x_{\mathrm{r3}}=x_{\mathrm{r2}}/k_x$
$k_x=(m-x_{\mathrm{r1\_end}})/(m-x_{\mathrm{r2\_start}})$
$x_{\mathrm{r3}}=m(1-k_x)+k_x x_{\mathrm{r2}}$
Compression of the longitudinal coordinate
$y_{\mathrm{r2\_start}}>y_{\mathrm{r1\_end}}$?
N
Y
$k_y=y_{\mathrm{r2\_start}}/y_{\mathrm{r1\_end}}$
$y_{\mathrm{r3}}=y_{\mathrm{r2}}/k_y$
$k_y=(m-y_{\mathrm{r1\_end}})/(m-y_{\mathrm{r2\_start}})$
$y_{\mathrm{r3}}=m(1-k_y)+k_y y_{\mathrm{r2}}$
End

Fig. 12 Flowchart of the contraction transformation algorithm

Suppose a start point Rob$_{\mathrm{start}}$($x_{\mathrm{r.0}}$, $y_{\mathrm{r.0}}$) of the robot is (8, 8). The coverage trajectory produced for the chaotic mobile robot is shown in Fig. 13a for n=800. Fig. 13b shows the mapped trajectory whose coverage rate is 49%. The trajectory is denoted as the second trail Trej$_2$, and its start point Trej$_{2\_\mathrm{start}}$($x_{\mathrm{r2\_start}}$, $y_{\mathrm{r2\_start}}$) changes to (13.8, 11.6). Assume that the end point Trej$_{1\_\mathrm{end}}$($x_{\mathrm{r1\_end}}$, $y_{\mathrm{r1\_end}}$) of the first trajectory Trej$_1$ is (11, 11). To link the two trajectories Trej$_1$ and Trej$_2$ to form a continuous path for the robot, the contraction transformation task is to contract Trej$_{2\_\mathrm{start}}$ to the position of Trej$_{1\_\mathrm{end}}$. The contracted trajectory Trej$_3$(x_{r3}, y_{r3}) is shown in Fig. 13c for the designed contraction transformation algorithm, and its coverage rate changes to 41%. The formed trajectory Trej$_3$(x_{r3}, y_{r3}) is shown in Fig. 13d when Trej$_{2\_\mathrm{start}}$ is contracted to another trajectory, Trej$_{1\_\mathrm{end}}$($x_{\mathrm{r1\_end}}$, $y_{\mathrm{r1\_end}}$)=(15, 16), and its coverage rate changes to 25%. The above discussions show the following:

1. The designed contraction transformation algorithm is feasible. In a given workplace, the start point of a trajectory can be contracted to any end point of another trajectory, but the coverage rate of the contracted trajectory decreases.

2. The largest the distance between two points, the smaller the coverage rate of the contracted trajectory. Therefore, we had better choose the nearest start point and end point to carry out the contraction transformation.

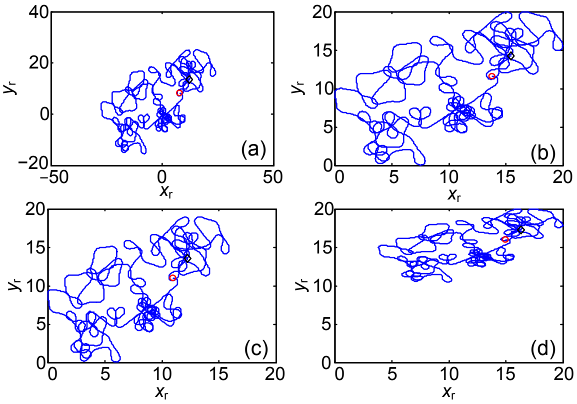

Fig. 13 Contraction transformation algorithm: (a) coverage trajectory produced by the chaotic mobile robot; (b) mapped trajectory in the given workplace; (c) contracted trajectory to the end point (11, 11); (d) contracted trajectory to the end point (15, 16)

6 Implementation of complete coverage path planning

Given a surveillance workplace Ω_{mn} without obstacles, an algorithm is designed to achieve complete path planning with a smaller number of iterations and a higher coverage rate. The procedure is as follows:

1. In Ω_{mn}, select the start point $\text{Rob}_{\text{start}}(x_{\text{r.0}}, y_{\text{r.0}})$ of the mobile robot randomly. The start point is also the end point Trej_{end} of the first trajectory.

2. A start point $\text{Trej}_{\text{start}}$ nearer Trej_{end} is selected from the candidate set $\text{Set}(x'_{\text{r.0}}, y'_{\text{r.0}})$.

3. The produced coverage trajectory based on $\text{Trej}_{\text{start}}$ is contracted to the position of Trej_{end} and forms the new part of the planned coverage trajectory for the mobile robot, and the end point of the trajectory is updated as the new Trej_{end}.

4. The coverage rate of the planned trajectory in Ω_{mn} is computed.

5. If the coverage rate satisfies the need of the requirement, the procedure ceases; otherwise, go to step 2.

Suppose the start point $\text{Rob}_{\text{start}}(x_{\text{r.0}}, y_{\text{r.0}})$ is (10, 10) in Ω_{mn}. The size of Ω_{mn} is 20×20. The task of the robot is to cover the workplace Ω_{mn} quickly with a higher coverage rate, and here it is 90%. Fig. 14 shows the planned coverage trajectory using the designed algorithm, where the purple "O" expresses the start point, the black "◇" denotes the end point, and the blue lines are the planned path. The designed path is continuous and includes six sections of the contracted trajectories. Its coverage rate is 91%, and n=7100. This shows that the algorithm is effective and can satisfy the requirement of the coverage task with a smaller number of iterations and a higher coverage rate. It needs no obstacle avoidance along the boundary of the workplace. For comparison, the mirror mapping approach shown in Fig. 15 is implemented, where m_i is the current position of the robot, m_i' is the next point that is outside the boundary, m_{i+1} is the mirror mapped position for the mobile robot, m_{cross} is the intersection point of the connecting line between the inside and outside robot coordinates with the boundary, and the brown bold line expresses the boundary. In a real space, the mobile robot moves as if it were reflected by the boundary (Fig. 16). In the figure, the start point is the same as that in Fig. 14, denoted as the purple "O." The red "*" on the boundary are the intersection points when the robot collides with the boundary. From the generation process of the trajectory (Fig. 16), the trajectory is formed clockwise along the boundary of the workplace. This circumstance spoils the chaotic characteristics of the chaotic mobile robot and loses the unpredictability of the trajectory. In addition, as can be seen from Figs. 16c and 16d, there are multiple reflections at the boundaries, which may decrease the planning efficiency. In Fig. 16d, the number of iterations is almost the same as that in Fig. 14, while the number of reflections n_{f} is 1500, and the coverage rate is 62.5%, lower than that of the designed algorithm.

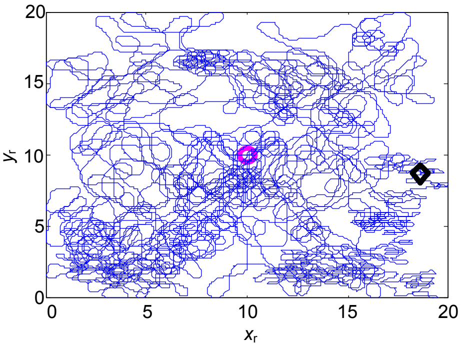

Fig. 14 The planned coverage trajectory produced by the designed algorithm

References to color refer to the online version of this figure

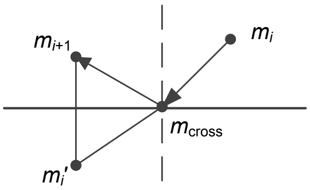

Fig. 15 Mirror mapping

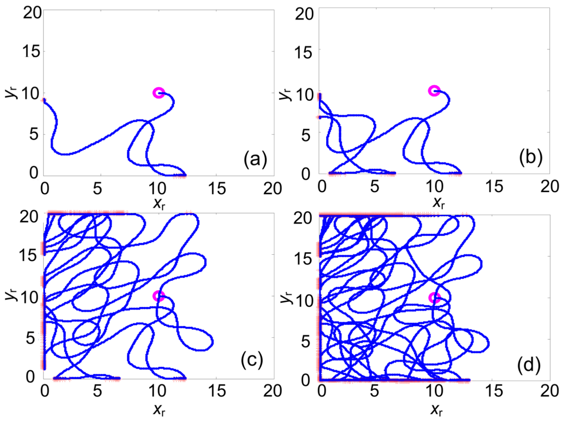

Fig. 16 The planned coverage trajectories produced by the mirror mapping method: (a) n_{f}=20, n=418, Coverage_rate=12%; (b) n_{f}=100, n=779, Coverage_rate=16.75%; (c) n_{f}=500, n=3401, Coverage_rate=55.75%; (d) n_{f}=1500, n=7121, Coverage_rate=62.50%

References to color refer to the online version of this figure

7 Conclusions

A chaotic mobile robot has been constructed using variable z which possesses the highest coverage rate of the three variables based on the Arnold dynamical system. The contraction transformation strategy has been designed to link the contracted coverage trajectories to a continuous path to achieve the coverage task with completeness, randomness, or unpredictability. The algorithm can maintain the chaotic characteristics in a relatively unchanged manner. The coverage rate of the planned path can reach a higher value with a smaller number of iterations. The advantages of the algorithm are:

1. Variable z is used to construct the chaotic mobile robot with the kinematic model of the mobile robot. The designed chaotic system can produce the coverage trajectory with a higher coverage rate.

2. In a given workplace, the trajectory produced can be contracted to any start point in it and form a continuous trajectory with a high coverage rate based on the construction of the candidate set and the design of the contraction transformation strategy.

3. No obstacle avoidance is needed to avoid collisions of the workplace boundaries. The produced trajectory is bounded in the workplace and the coverage rate and efficiency are improved.

4. All the coverage trajectories are produced by the designed chaotic mobile robot and possess the chaotic characteristics, namely completeness, randomness, or unpredictability, to meet the requirements of specific types of tasks.

5. The designed algorithm can be applied to other 3D chaotic systems to construct a chaotic mobile robot that produces a patrol or surveillance continuous coverage path.

In the future we will consider pursuing further work in the following areas:

1. At present, there are relatively few sampled start points in the constructed candidate set. More points should be collected by testing a large number of data samples to cover the whole workplace evenly and densely.

2. The designed algorithm works in a bounded environment without obstacles inside. The influence of obstacles should also be considered.

3. The chaotic mobile robot is constructed only with variable z. While variable x can produce the coverage trajectories at a high coverage rate based on some start points, the hybrid chaotic mobile robot can be considered to be constructed by variables x and z together to improve the coverage rate and efficiency.

4. Because the designed chaotic mobile robot defines the position goal in each step, a closed-loop controller should be designed to track each sub-goal; otherwise, the coverage rate will be affected and disturbed by various factors.

Compliance with ethics guidelines

Cai-hong LI, Chun FANG, Feng-ying WANG, Bin XIA, and Yong SONG declare that they have no conflict of interest.

References

Banerjee C, Datta D, Agarwal A, 2015. Chaotic patrol robot with frequency constraints. Proc IEEE Int Conf on Research in Computational Intelligence and Communication Networks, p.340-344. https://doi.org/10.1109/ICRCICN.2015.7434261

Curiac DI, Volosencu C, 2009. Developing 2D chaotic trajectories for monitoring an area with two points of interest. Proc 10th WSEAS Int Conf on Automation & Information, p.366-369.

Curiac DI, Volosencu C, 2012. Chaotic trajectory design for monitoring an arbitrary number of specified locations using points of interest. *Math Probl Eng*, 2012:940276. https://doi.org/10.1155/2012/940276

Curiac DI, Volosencu C, 2014. A 2D chaotic path planning for mobile robots accomplishing boundary surveillance missions in adversarial conditions. *Commun Nonl Sci Numer Simul*, 19(10):3617-3627. https://doi.org/10.1016/j.cnsns.2014.03.020

Curiac DI, Volosencu C, 2015a. Imparting protean behavior to mobile robots accomplishing patrolling tasks in the presence of adversaries. *Bioinspir Biomim*, 10(5):056017. https://doi.org/10.1088/1748-3190/10/5/056017

Curiac DI, Volosencu C, 2015b. Path planning algorithm based on Arnold cat map for surveillance UAVs. *Def Sci J*, 65(6):483-488. https://doi.org/10.14429/dsj.65.8483

Curiac DI, Banias O, Volosencu C, et al., 2018. Novel bioinspired approach based on chaotic dynamics for robot patrolling missions with adversaries. *Entropy*, 20(5):378. https://doi.org/10.3390/e20050378

Fahmy AA, 2012. Performance evaluation of chaotic mobile robot controllers. *Int Trans J Eng Manag Appl Sci Technol*, 3(2):145-158.

Galceran E, Carreras M, 2013. Planning coverage paths on bathymetric maps for in-detail inspection of the ocean floor. Proc Int Conf on Robotics and Automation, p.4159-4164. https://doi.org/10.1109/ICRA.2013.6631164

Hwang KS, Lin JL, Huang HL, 2011. Dynamic patrol planning in a cooperative multi-robot system. Proc 14th FIRA RoboWorld Congress on Next Wave in Robotics, p.116-

123. https://doi.org/10.1007/978-3-642-23147-6_14

Li CH, Song Y, Wang FY, et al., 2015. Chaotic path planner of autonomous mobile robots based on the standard map for surveillance missions. *Math Probl Eng*, 2015:263964. https://doi.org/10.1155/2015/263964

Li CH, Song Y, Wang FY, et al., 2016. A bounded strategy of the mobile robot coverage path planning based on Lorenz chaotic system. *Int J Adv Robot Syst*, 13(3):107. https://doi.org/10.5772/64115

Li CH, Song Y, Wang FY, et al., 2017. A chaotic coverage path planner for the mobile robot based on the Chebyshev map for special missions. *Front Inform Technol Electron Eng*, 18(9):1305-1319. https://doi.org/10.1631/FITEE.1601253

Li CH, Wang ZQ, Fang C, et al., 2018. An integrated algorithm of CCPP task for autonomous mobile robot under special missions. *Int J Comput Intell Syst*, 11:1357-1368. https://doi.org/10.2991/ijcis.11.1.100

Liu P, Sun JJ, Qin HZ, et al., 2017. The area-coverage path planning of a novel memristor-based double-stroll chaotic system for autonomous mobile robots. Proc Chinese Automation Congress, p.6982-6987. https://doi.org/10.1109/CAC.2017.8244036

Lorenz EN, 1997. The Essence of Chaos. Liu SD, translator. China Meteorological Press, Beijing, China, p.186-189 (in Chinese).

Martins-Filho LS, Macau EEN, 2007a. Patrol mobile robots and chaotic trajectories. *Math Probl Eng*, 2007:61543. https://doi.org/10.1155/2007/61543

Martins-Filho LS, Macau EEN, 2007b. Trajectory planning for surveillance missions of mobile robots. In: Mukhopadhyay SC, Gupta GS (Eds.), Autonomous Robots and Agents. Springer Berlin Heidelberg, p.109-117. https://doi.org/10.1007/978-3-540-73424-6_13

Nakamura Y, Sekiguchi A, 2001. The chaotic mobile robot. *IEEE Trans Robot Autom*, 17(6):898-904. https://doi.org/10.1109/70.976022

Oksanen T, Visala A, 2009. Coverage path planning algorithms for agricultural field machines. *J Field Robot*, 26(8):651-668. https://doi.org/10.1002/rob.20300

Ousingsawat J, Earl MG, 2007. Modified lawn-mower search pattern for areas comprised of weighted regions. Proc American Control Conf, p.918-923. https://doi.org/10.1109/ACC.2007.4282850

Park E, Kim KJ, del Pobil P, 2012. Energy efficient complete coverage path planning for vacuum cleaning robots. In: Park JJ, Leung VCM, Wang CL, et al. (Eds.), Future Information Technology, Application, and Service. Springer, Dordrecht, p.23-31. https://doi.org/10.1007/978-94-007-4516-2_3

Peitgen HO, Jürgens H, Saupe D, 2004. Chaos and Fractals: New Frontiers of Science. Springer, New York, NY. https://doi.org/10.1007/b97624

Prado J, Marques L, 2014. Energy efficient area coverage for an autonomous demining robot. In: Armada MA, Sanfeliu A, Ferre M (Eds.), ROBOT2013: First Iberian Robotics Conf. Springer, Cham, p.459-471. https://doi.org/10.1007/978-3-319-03653-3_34

Sekiguchi A, Nakamura Y, 1999. The chaotic mobile robot. Proc IEEE/RJS Int Conf on Intelligent Robots and Systems, p.172-178. https://doi.org/10.1109/IROS.1999.813000

Sooraska P, Klomkarn K, 2010. "No-CPU" chaotic robots: from classroom to commerce. *IEEE Circ Syst Mag*, 10(1): 46-53. https://doi.org/10.1109/MCAS.2010.935740

Torres M, Pelta DA, Verdegay JL, et al., 2016. Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. *Expert Syst Appl*, 55:441-451. https://doi.org/10.1016/j.eswa.2016.02.007

Volos CK, Kyprianidis IM, Stouboulos IN, 2012a. A chaotic path planning generator for autonomous mobile robots. *Robot Auton Syst*, 60(4):651-656. https://doi.org/10.1016/j.robot.2012.01.001

Volos CK, Bardis NG, Kyprianidis IM, et al., 2012b. Implementation of mobile robot by using double-scroll chaotic attractors. Proc 11th Int Conf on Applications of Electrical and Computer Engineering, p.119-124.

Volos CK, Kyprianidis IM, Stouboulos IN, 2013. Experimental investigation on coverage performance of a chaotic autonomous mobile robot. *Robot Auton Syst*, 61(12):1314-1322. https://doi.org/10.1016/j.robot.2013.08.004

Zheng Z, Liu Y, Zhang XY, 2016. The more obstacle information sharing, the more effective real-time path planning? *Knowl-Based Syst*, 114:36-46. https://doi.org/10.1016/j.knosys.2016.09.021