Path Planning with Smooth Curve Using Electromagnetism-like Mechanism Algorithm

Ming-Wei Hsu[1], Shih-Hua Huang[2], Yih-Guang Leu[3*]

Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan
Email: leuyk@ntnu.edu.tw (*corresponding author)

Abstract— In this paper, we propose a new path planning method by using an electromagnetism-like mechanism algorithm. We use the different encoding method to solve trade-off problem that is encountered in the traditional path planning method. By combining different encoding method with electromagnetism-like mechanism algorithm, path point can be generated without the trade-off problem. In order to connect these points into smooth curve, we compare two path smoothing method, Bezier curve and cubic splines interpolation. Finally, cubic splines interpolation is used because it can generate smoother path.

Keywords— Path Planning, Electromagnetism-like Mechanism Algorithm, Path Smoothing.

I. INTRODUCTION

For all of the mainstream robots which have the movement capability, path planning will be a necessary and worthwhile problem to solve. The traditional path planning method is proposed by computer scientist Dijkstra [1], then A* algorithm [2] solves the problem of direction. The A* algorithm makes it work more efficiently, but these algorithms will encounter the map trade-off problem in the practical application. In a known environment, the map needs to be cut into many grids, and these grids are used to plan the path.

The process of grids is a trade-off problem. If the map is cut into large grids, the relative generated path will be very rough. However, if the map is cut into small girds, it will cost a lot of computing costs. In real case, the obstacles on the map is irregular shape, such that path planning result is not the shortest path. As a result, many algorithms were proposed to compute shorter or more effective path for the trade-off problem.

To solve the problem mentioned above, we need to change the processing method for path planning by using a different map encoding method [3] and connect with the heuristic algorithm to solve the path planning problem. The electromagnetism-like (EM) algorithm [4] is innovative in the heuristic algorithm which is based on the characteristics of electromagnetic attraction and repulsion.

In this paper, instead of applying EM algorithm for optimization, we combine EM algorithm with different encoding method for path planning. It can avoid traditional trade-off problem by spreading points and connecting these points. The algorithm computes the path with path smoothing method. In this paper, cubic spline interpolation is chosen because it can smooth the path without increasing the algorithm's parameters. To demonstrate the proposed method, we simulate the possible map configuration, and use map simulation to verify the feasibility of the algorithm.

II. ELECTROMAGNETISM-LIKE MECHANISM ALGORITHM

Electromagnetism-like Mechanism algorithm [4] is a heuristic algorithm proposed in 2003, and the convergence is presented in [5]. EM algorithm applies the concept of Coulomb's law in the electromagnetic, the better electronic attract bad electrons, and bad electronics will repel other good electronics. The attraction and repulsion forces between electrons accumulate into total force in order to make the path result to better direction. Because EM algorithm has good computing ability, there are many kinds of applications, such as optimization question for communication system [6] [7], optimization the neural networks [8], optimization the learning rule for fuzzy neural networks [9], path-tracking problem [10], optimized tool path planning [11], parameter optimization [12], the routing problem [13], optimization the knapsack problem [14] and other applications [15] [16].

Besides considering the better solution, EM algorithm also consider the worst solution as a contrarian indicator, these multiple reference can let the solution converge faster. On the other hand, the EM algorithm has fewer parameters; it is more intuitive and convenient in parameter adjustment. And it can add more electronics in the map, such that user can adjust parameter based on road condition. It is feasible to choose the EM algorithm for path planning.

The EM algorithm includes four parts [17]: initialize the algorithm (Initiation), search the neighborhood to avoid falling into the optimal solution (Local), calculate the total force (CalcF) for each electronics, and move the electronics along the force direction (Move).

General framework is shown below.

1: Initiation()

2: ite←1

3: **while** ite *<MAXITE do*

4:    Local(*LITE,α)*

5:    **F ←CalcF()**

6:    Move(**F)**

7:    ite ←ite + 1

8: **end while**

In this paper we use the modified EM algorithm which adds the rearrange function in original EM algorithm. Rearrange function can adjust the position of electrons without altering the path, and it makes the EM algorithm can generate global optimal solution quickly. The flow chart of the EM in this paper is shown in Fig. 1.
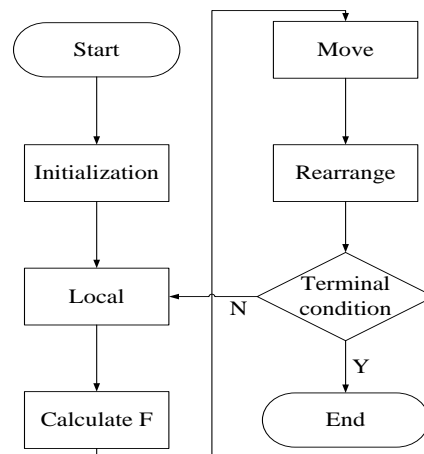

FIG. 1. FLOW CHART IN THIS PAPER

Step1: *Initiation*

Initiation spreads point randomly in the map. Each one of the point is regard as one electrical point, and all of the electronics are connected from start point to the end point, as shown in Fig. 2(a). Then the point is spread, and each electronic evaluates the other electronics' magnetic which has same dimensions, as shown in Fig. 2(b).
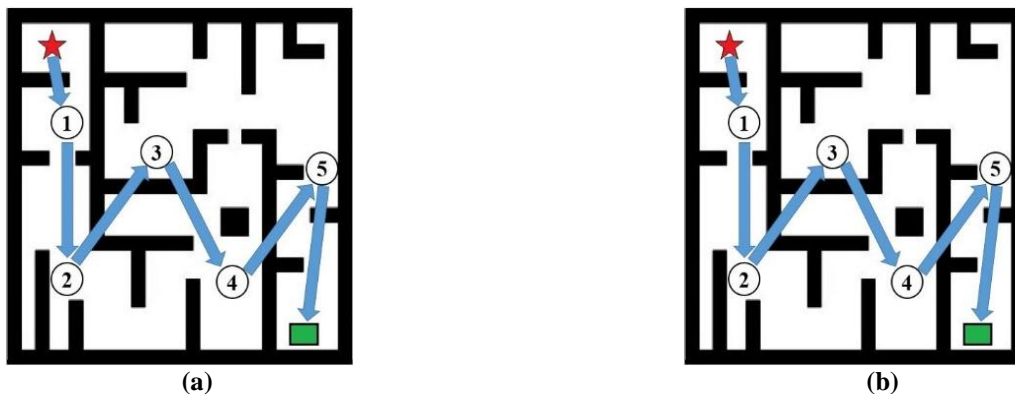

| (a) | (b) |

FIG. 2. (a) LINK THE ELECTRICAL POINT. (b) TWO ELECTRICAL IN THE SAME FIELD

Step2: *Local*

Local makes a provisional electronics for each dimension. The electronic is a replica of the original one, then move randomly. If *Local* finds the better result, it will replace the original one, and finally find out whether there is a better global optimal solution.

Step3: *CalcF*

It is necessary to calculate charge amount for each electrons Q_i before calculating the total force. The attraction and repulsion forces are determined by charge amount. The charge amount of the i th electron is shown in (1).

$$Q^i = \exp(-n \frac{f(e^i) - f(e^{best})}{\sum_{i=1}^{m}(f(e^k) - f(e^{best}))}) \tag{1}$$

where m means the number of particles, e^i means the current electronic, and e^{best} means global optimal solution. The global optimal solution has a greater attraction. Then, calculate the total force F^i for each electron, as shown in (2).

$$F^i = \sum_{j \neq i}^{m} \begin{cases} (e^j - e^i) \dfrac{Q^i Q^j}{\|e^j - e^i\|^2} \\[3mm] (e^i - e^j) \dfrac{Q^i Q^j}{\|e^j - e^i\|^2} \end{cases} \tag{2}$$

Although equation (2) is more accord with physical meaning, it costs more time in algorithm. By [18], we use equation (3) to accelerate the implementation rate of algorithms.

$$F^i = \sum_{j \neq i}^{m} \begin{cases} (e^j - e^i)Q^i Q^j & if \ f(e^j) < f(e^i) \\ (e^i - e^j)Q^i Q^j & if \ f(e^j) \geq f(e^i) \end{cases} \tag{3}$$

Step4: *Move*

After calculating the total force F^i, electrons are designed to move a random distance based on the direction of the total force, and the movement method is shown on equation (4).

$$e_k^i = e_k^i + \gamma \frac{F^i}{\|F^i\|}(RNG) \tag{4}$$

where γ means a uniform distribution between 0 and 1. It also can use other types of distributions in calculating the step length. RNG means a random number for step length. It can determine a nonzero probability to move to unvisited search space.

Step5: *Rearrange*

If the electronics are overly concentrated, the EM algorithm will occur sharp point phenomenon when it applied to path planning, as shown in Fig.3. Therefore, we need the *Rearrange* function to redistribute the electrons. As a result, the global optimal solution is generated more quickly.
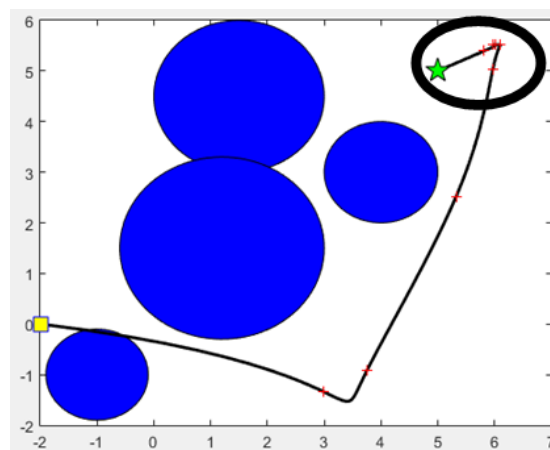

FIG. 3. SHARP POINT PHENOMENON

The general framework of the *Rearrange* function is described as follows.

```
 1:  γ ← RNG(0,1)
 2:  if γ < T * det  then
 3:      j ← 1
 4:      for i = 1 to h do
 5:          if length(i) = L*j / n
 6:              e_k^i = position(i)
 7:          if i = n+1
 8:              break
 9:          end if
10:          j ← j+1
11:      end if
12: end for
```

γ means a uniform distribution between 0 and 1. T means the threshold which makes the path rearrange. *det* is a parameter decrease when iteration number increases. We use *det* to control the probability which let algorithm execute the *Rearrange* function. L is the length of the path, n is the number of sections.

III.   PATH SMOOTHING

The result of the path planning often has a non-smooth turning characteristic. It is caused by the rough segmentation of the map. This makes the path planning result possess sharp angle turning in the actual application. Therefore, the vehicle must need a lot of deceleration, or even stop the vehicle to make a turn.

After the path smoothing process, it makes the smooth path planning result. Therefore, the vehicle can make a turn without a lot of deceleration in actual application.

However, when we use high-order polynomial to approximate the path planning result, Runge's phenomenon will occurs. Runge's phenomenon occurs when using polynomial interpolation with polynomials of high degree over a set of equispaced interpolation points. It is a problem of oscillation at the edges of an interval. To avoid Runge's phenomenon, multi-sections are used rather than high-order interpolation method for path smoothing. This paper will introduce two methods of smoothing path in follow.

3.1   Bezier Curve [19]

Given point D_0 and D_1, the Bezier Curve L_0 is a straight line between those two points. The curve is given by equation (5):

$$L_0 = (1-t)D_0 + tD_1 \tag{5}$$

If Bezier Curve starts from D_0 to D_2 and has a relay point D_1, the second order Bezier Curve $C_2(t)$ is shown as equation (6):

$$C_2(t) = (1-t)^2 D_0 + 2t(1-t)D_1 + t^2 D_2 \tag{6}$$

The general form of a n degree Bezier curve defined by the control points D_i (where $i = 0, \cdots, n$) is

$$C(t) = \sum_{i=0}^{n} B_{i,n}(t)D_i \tag{7}$$

where $B_{i,n}(t)$ is called Bernstein polynomials that are defined in (8).

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i} \tag{8}$$

the cubic Bezier curve $C_3(t)$ is given in (9).

$$C_3(t) = (1-t)^3 D_0 + 3t(1-t)^2 D_1 + 3t^2(1-t)D_2 + t^3 D_3 \tag{9}$$

3.2   Cubic Splines Interpolation [20]

Spline interpolation uses segmented polynomial to interpolation, and it can achieve the smaller interpolation error by using low-order polynomial, and avoid Runge's phenomenon.

First, define $k_i = C''(t_i)$ for $i = 1, 2, ..., n-1$, $C_i(t_i) = w_i$, $C_i(t_{i+1}) = w_{i+1}$ and $k_0 = 0, k_n = 0$. And it Lagrange form for $C''(t_i)$ is

$$C_i''(x) = \frac{k_{i+1}(x - t_i)}{(t_{i+1} - t_i)} + \frac{k_i(t_{i+1} - x)}{(t_{i+1} - t_i)} \tag{10}$$

Then the $C_i'(x)$ and $C_i(x)$ are given as

$$C_i'(x) = \frac{k_{i+1}(x - t_i)^2}{2(t_{i+1} - t_i)} + \frac{k_i(t_{i+1} - x)^2}{2(t_{i+1} - t_i)} + A_i + B_i \tag{11}$$

$$C_i(x) = \frac{k_{i+1}(x - t_i)^3}{6(t_{i+1} - t_i)} + \frac{k_i(t_{i+1} - x)^3}{6(t_{i+1} - t_i)} + A_i(x - t_i) + B_i(t_{i+1} - x) \tag{12}$$

where A_i and B_i are shown in equation (13) and (14).

$$B_i = \frac{w_i}{(t_{i+1} - t_i)} - \frac{(t_{i+1} - t_i)k_i}{6}, \tag{13}$$

$$A_i = \frac{w_{i+1}}{(t_{i+1} - t_i)} - \frac{(t_{i+1} - t_i)k_{i+1}}{6} \tag{14}$$

Therefore, C_i and C_i' are generated as following equation (15) and (16).

$$C_i(x) = \frac{k_{i+1}(x - t_i)^3}{6(t_{i+1} - t_i)} - \frac{k_i(x - t_{i+1})^3}{6(t_{i+1} - t_i)} + \left[\frac{w_{i+1}}{t_{i+1} - t_i} - \frac{(t_{i+1} - t_i)k_{i+1}}{6} \right](x - t_i) - \left[\frac{w_i}{(t_{i+1} - t_i)} - \frac{(t_{i+1} - t_i)k_i}{6} \right](x - t_{i+1}) \tag{15}$$

$$C_i'(x) = \frac{k_{i+1}(x - t_i)^2}{2(t_{i+1} - t_i)} - \frac{k_i(x - t_{i+1})^2}{2(t_{i+1} - t_i)} + \frac{w_{i+1} - w_i}{t_{i+1} - t_i} - \frac{k_{i+1} - k_i}{6}(t_{i+1} - t_i) \tag{16}$$

Also, $C_i'(t_i)$ requires the following condition.

$$C_{i-1}'(t_i) = C_i'(t_i) \quad i = 1, 2, \cdots, n-1$$

Then we can get equation (17).

$$C_i'(t_i) = -\frac{1}{2} k_i(t_{i+1} - t_i) + \frac{w_{i+1} - w_i}{t_{i+1} - t_i} - \frac{k_{i+1} - k_i}{6}(t_{i+1} - t_i) \tag{17}$$

By using $s_i = t_{i+1} - t_i$, and $d_i = \frac{w_{i+1} - w_i}{t_{i+1} - t_i}$, equation (17) has

$$C_i'(t_i) = -\frac{1}{6} s_i k_{i+1} - \frac{1}{3} s_i k_i + d_i \tag{18}$$

$$C_{i-1}'(t_i) = \frac{1}{6} k_{i-1} s_{i-1} + \frac{1}{3} k_i s_{i-1} + d_{i-1} \tag{19}$$

By $C_{i-1}'(t_i) = C_i'(t_i)$, then equation (20) is given.

$$\begin{cases} s_{i-1} k_{i-1} + 2(s_{i-1} + s_i)k_i + s_i k_{i+1} = 6(d_i - d_{i-1}) \\ k_0 = k_n = 0 \end{cases} \tag{20}$$

Bezier curve is widely used in computer graphics and other path smoothing method. It is a simple and practical way. However, distinguishing between control point and path point in path planning algorithm is more complicated.

In the EM algorithms, cubic splines interpolation is used in doing path smoothing algorithm so that the number of parameters does not increase. Regarding each point as one path point, the algorithm enhances the relevance and thus enhances the computing performance.

IV. SIMULATION RESULT

The simulation parameter is shown on Table.1, and the simulation result is shown in Fig. 4. As shown in Fig. 4, the square is the start point, the star is the end point, and the circle is the obstacle. After the path planning and smoothing, the curve is the algorithm result. We calculate the cost function by using the Pythagorean Theorem. When the path contacts the obstacle, its cost function will be multiplied 100 as a penalty. Thus we can always find the best cost without touching the obstacle.

TABLE 1
SIMULATION PARAMETERS

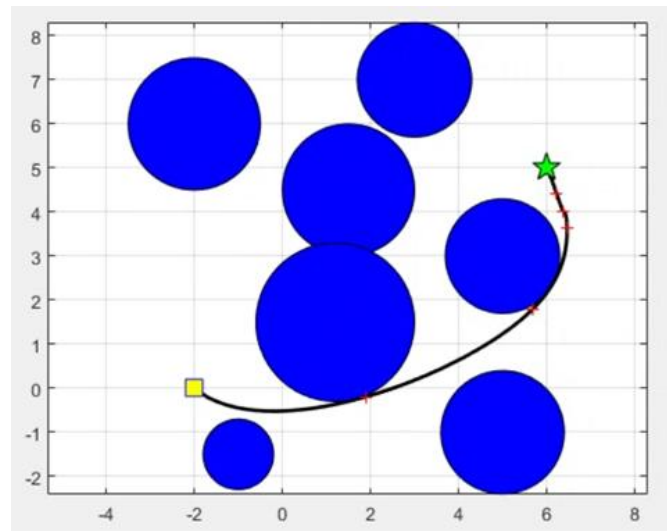| Parameters	Max Iteration(*MAXITE*)	Local Iteration(*LITE*)	Number of population	Number of Control Point(m)	Delta(*det*)	Rearrange threshold(T)	Penalty
Values	50	100	10	5	0.4	0.1	100

FIG.4. SIMULATIONAL RESULT

V. CONCLUSIONS

This paper has presented a path planning algorithm to solve the trade-off problem that is encountered in the traditional path planning method. Comparing two path smoothing method, Bezier curve and cubic splines interpolation, cubic splines interpolation is selected to smooth the path without increasing the number of parameters. By using the path smoothing method, the vehicle could go through the angle smoothly like reality.

ACKNOWLEDGEMENTS

REFERENCES

[1] Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". *Numerische Mathematik* I: 269-271

[2] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. on Systems Science and Cybernetics, vol. 4, no. 2*, pp. 100-107, 1968.

[3] Yanrong Hu and Simon X. Yang, "A Knowledge Based Genetic Algorithm for Path Planning of a Mobile Robot," *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, New Orleans, USA, Pages 4350 – 4355, 2004.

[4]　S.I. Birbil and S.-C. Fang, "An electromagnetism-like mechanism for global optimization," *J. Glob. Opt.*, vol. 25, no. 3, 263-282, 2003.

[5]　S. I. Birbil, S.-C. Fang, and R.-L. Sheu, "On the convergence of a population -based global optimization algorithm," *J. Glob. Opt.*, vol. 30, no. 2, 301-318, 2004.

[6]　Tsai, Chuen-Yih, Ho-Lung Hung, and Shu-Hong Lee. "Electromagnetism-like method based blind multiuser detection for MC-CDMA interference suppression over multipath fading channel." *Computer Communication Control and Automation (3CA), 2010 International Symposium on.* Vol. 2, pp. 470-475, 2010.

[7]　J. C. Chen, "Partial transmit sequences for PAPR reduction of OFDM signals with stochastic optimization techniques," *IEEE Trans. Consumer Electron., vol. 56*, no. 3, pp. 1229-1234, Aug. 2010.

[8]　Q. Wu, et al., "Training neural networks by electromagnetism-like mechanism algorithm for tourism arrivals forecasting," in *Bio-Inspired Computing: Theories and Applications (BIC-TA), 2010 IEEE Fifth International Conference on,* 2010, pp. 679-688.

[9]　P. Wu, K. Yang, and Y. Hung, "The study of electromagnetism-like mechanism based fuzzy neural network for learning fuzzy if-then rules," *Knowledge-Based Intelligent Information and Engineering Systems*, pp. 907–907, 2005.

[10]　Yaonan, Wang, Yang Yimin, Yuan Xiaofang, Yin Feng, and Wei Shuning. "A Model Predictive Control Strategy for Path-Tracking of Autonomous Mobile Robot Using Electromagnetism-Like Mechanism", *2010 International Conference on Electrical and Control Engineering*, 2010

[11]　Kuo, C. L., Chu, C. H., Wu, Q., Li, X., Gao, L. and Li, Y., "Electromagnetism-like Algorithms for Optimized Tool Path Planning in 5-Axis Flank Machining," *Computers and Industrial Engineering*, Vol. 84, pp. 70-78, 2014.

[12]　F.-K. Chang and C.-H. Lee, Design of Fractional PID Control via Hybrid of Electromagnetism-Like and Genetic Algorithms, *in Eighth International Conference on Intelligent Systems Design and Applications*, pp 525–530, 2008.

[13]　P. Wu, K.-J. Yang, and B.-Y. Huang, "A Revised EM-like Mechanism for Solving the Vehicle Routing Problems," *in Proceedings of the Second International Conference on Innovative Computing, Information and Control*, pp. 181-184, 2007.

[14]　C. C. Chang, C. Y. Chen, C. W. Fan, H. C. Chao and Y. H. Chou, "Quantum-Inspired Electromagnetism-Like Mechanism for Solving 0/1 Knapsack Problem." *in IEEE conference on ITCS,* pp.1-6, 2010.

[15]　Yun, Shang. "Research of Big Data Analysis on Rough Set and Electromagnetism-Like Mechanism Algorithm." *in IEEE conference on CIT*, pp. 923-926 , 2014.

[16]　C. Y. Chen, C. H. Chiu, and H. C. Chao, "Classical and Quantum-Inspired Electromagnetism-Like Mechanism and its Applications," *IET Control Theory & Applications,* vol. 6, no. 10, 2012, pp. 1424–33.

[17]　Zhang, C., Li, X., Gao, L., and Wu, Q., "An improved electromagnetism-like mechanism algorithm for constrained optimization," *Expert Systems with Applications*, Vol. 40, No. 14, pp. 5621-5634, 2013.

[18]　Kuo, C. L., Chu, C. H., Wu, Q., Li, X., Gao, L. and Li, Y., "Electromagnetism-like Algorithms for Optimized Tool Path Planning in 5-Axis Flank Machining," *Computers and Industrial Engineering*, Vol. 84, pp. 70-78, 2014.

[19]　A. R. Forrest, "Interactive interpolation and approximation by Bezier polynomials" *Comput. J.*, vol. 15, no. 1, pp. 71–79, 1972.

[20]　McKinley, Sky, and Megan Levine." Cubic spline interpolation." *College of the Redwoods*, 1998.