Shortest Paths between two Points in Space for Self Managed Digital Systems

Manoj Kumar
Assistant Professor
Govt. College of Teacher Education Dharamshala
Kangra, Himachal Pradesh, India

ABSTRACT

In this paper, It has been proved that there can be more than one shortest path between two different points in space for self managed digital systems with finite memory space. Also, a formula has been given, that can be used to find total number of different paths between given two points.

General Terms:

Self Managed Digital Systems, Shortest Paths, Robotics

Keywords:

Shortest Paths, Robotics, Self Managed Digital Systems, Space Movement of Digital Systems, Robot Motion, Minimal Movement Problem

1. INTRODUCTION

There are various scenarios where we need an intelligent digital system to perform some task, they can vary from observation of environment on other planets to maintaining our houses. In such cases, we require a digital system that can move in space efficiently to perform the given task. Such type of digital systems can encounter some problems in performing the given task. They must be able to handle such problems on their own. They must have alternative ways to complete the given task with maximum efficiency. While moving from one point to another point in space. It is always required that system should follow a path with minimum distance in the given homogeneous conditions. To improve their reliability, we must find some alternative paths having the same minimum distance[4].

2. DIGITAL SYSTEMS

Digital Systems are the machines which take digital signals as inputs, process them according to a set of instructions, and produce output in the form of digital signals. These digital signals are encoded with two distinct symbols i.e. 0 (zero) and 1 (one) also called bits. The set of instructions, required to process inputs, are also encoded with bits. Digital systems have a storage unit that can store any information which can be encoded in the form of bits. Storage unit can store input values, processing instructions and output produced. Storage unit is also called memory unit. It is a physical component which consists of digital circuits[3].

3. SELF MANAGED DIGITAL SYSTEMS

Self managed digital systems are the systems, which can manage themselves according to any change in their environment. They can observe environment, and change their state accordingly, on their own. In the process of creating self managed digital systems which can move in space, we need to develop more efficient algorithms with more reliability. To provide such reliability we must find all possible solutions in given set of problems. Finding a shortest path from one point to another point in space is one such problem. To provide more reliability, we must find other alternative possible solutions with same efficiency.

4. MOVEMENT OF SELF MANAGED DIGITAL SYSTEM IN SPACE

We consider a digital system with a finite memory. That is made to move in space on its own. It can observe its surrounding environment, process it, to take decisions, to make required movements. In order to manage its movements it is required to represent space coordinates in its memory. Suppose we can use a maximum of n bits to store the position of its adjacent point(s) that it can move onto, from its current position i.e. it can store a maximum of 2^n distinct points it can move onto[3].

As n is a finite number, 2^n will also be a finite number. So there will be some finite positions in space that it can move onto and there always will be some positions in space that it cannot move onto. Hence, *A digital system that uses a finite memory to represent space coordinates, cannot move continuously in space*.

However to move from one point to another it has to travel the intermediate distance physically. When it lies between two adjacent points, it will be at a position, which is not stored in its memory, and hence it can't be managed. So, it is called in an unstable state. And when machine is at a point that is stored in its memory, it is considered in a *stable state*.

In order to move in space, it has to move from its current position to next adjacent position by travelling a finite minimum distance, say d, in a given direction, say $\hat{v}$. In other words, in a given direction $\hat{v}$, it cannot be at a stable state by travelling a distance less than d. Any such movement can be represented as

$$\vec{m} = d.\hat{v}$$

Now, consider it has to move from point P_1 to another point P_2 by travelling a minimum distance between these two points. As it can move through given finite points only. The path travelled by it can be represented as

$$\vec{P_1P_2} = a_1d_1\hat{v}_1 + a_2d_2\hat{v}_2 ... + a_kd_k\hat{v}_k \tag{1}$$

where
a_i is the total number of steps taken in the direction $\hat{v}_i$ from P_1 to P_2 for $1 \leq i \leq k$.
d_i is minimum possible finite distance that can be travelled in the direction $\hat{v}_i$ for $1 \leq i \leq k$.
As we know that vector addition is associative, so we can write this equation in other forms, by rearranging the individual terms, without changing its physical significance [2]. These equations represent other shortest paths between P_1 and P_2 . Hence,
There exist more than one shortest path between two different points in space for self managed digital systems, with finite memory space.
We can write equation 1 in number of different ways given by

$$w = \frac{N!}{a_1!a_2!a_3!...a_k!} \tag{2}$$

where
a_i is the total number of steps taken in the direction $\hat{v}_i$ from P_1 to P_2 for $1 \leq i \leq k$.
and
$N = a_1 + a_2 + a_3 + ... + a_k$ i.e. total number of steps taken in all directions. It gives us the total number of different paths between given two points.

5. ILLUSTRATION

Let us consider a system, represented by a point on a computer screen, such that, it can move from one pixel to next adjacent pixels only. Consider, currently its postion is $P_1(x_1, y_1)$. And it can move onto its adjacent pixels, as shown in Figure 1.

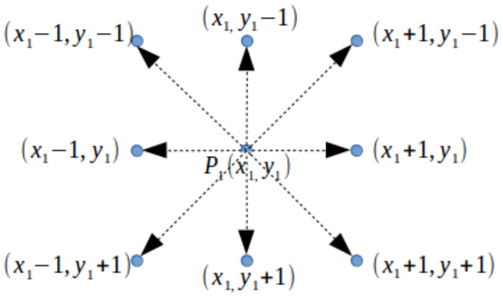

Fig. 1. Adjacent Points to P_1

Suppose it has to move to another pixel P_2 as shown in Figure 2. If we move closest to a straight line from P_1 to P_2 it gives us a shortest path given by:

$$\vec{P_1P_2} = d_1\hat{v}_1 + d_2\hat{v}_2 + d_1\hat{v}_1 \tag{3}$$

where
$d_1 = |\vec{P_1I_1}| = |\vec{I_2P_2}|$,

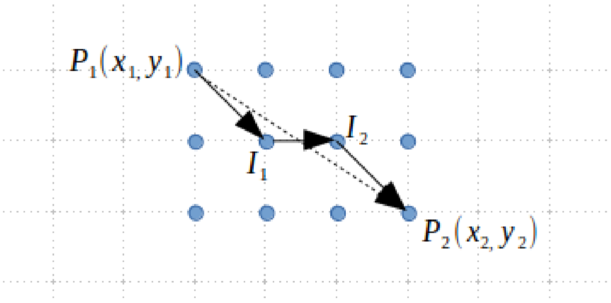

Fig. 2. A Shortest Path (Closest to straight line)

$d_2 = |\vec{I_1I_2}|$
and
$\hat{v}_1 = \frac{\vec{P_1I_1}}{|\vec{P_1I_1}|} = \frac{\vec{I_2P_2}}{|\vec{I_2P_2}|}$,
$\hat{v}_2 = \frac{\vec{I_1I_2}}{|\vec{I_1I_2}|}$
Also, in equation 3 , we have,
$a_1 = 2, a_2 = 1$
and
$N = 2 + 1 = 3$
Now, according to equation 2 , we can write this equation in number of ways given by:
$w = \frac{N!}{a_1!a_2!} = \frac{3!}{2!1!} = 3$
i.e.

$$\vec{P_1P_2} = d_1\hat{v}_1 + d_1\hat{v}_1 + d_2\hat{v}_2 \tag{4}$$

$$\vec{P_1P_2} = d_1\hat{v}_1 + d_2\hat{v}_2 + d_1\hat{v}_1 \tag{5}$$

$$\vec{P_1P_2} = d_2\hat{v}_2 + d_1\hat{v}_1 + d_1\hat{v}_1 \tag{6}$$

The equations 4, 5 and 6 represent all three possible paths, with shortest distance, from P_1 to P_2 for the given system. Path represented by equation 4 is shown in Figure 3. Similarly, other paths can be obtained using equations 5 and 6 . [1].

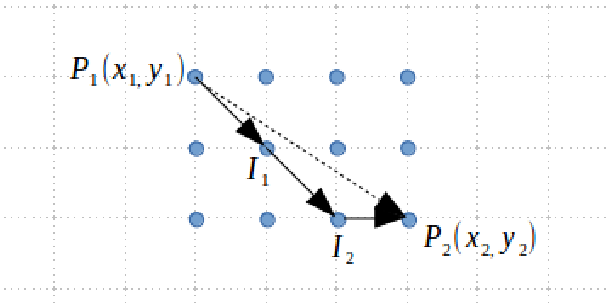

Fig. 3. An Alternative Shortest Path

6. CONCLUSION

It has been proved that, there can be more than one shortest path, in space, for self managed digital systems with finite memory. Also, a formula has been given that can be used to calculate total number of paths possible between given two points. These results can be used in designing systems that are to be made to move in space. It will provide more reliability and efficiency, specially, in dynamic environments. In dynamic environments conditions may change at any time, so, the system will be able to select any other alternative path with minimum possible distance, out of all possible paths available, accordingly.

7. ACKNOWLEDGEMENTS

My thanks to everyone and everything I learned from, and I am, specially, thankful to my parents for their support.

8. REFERENCES

[1] Donald Hearn and M. Pauline Baker. *Computer Graphics C version*. Dorling Kindersley (India) Pvt. Ltd., Licensees of Pearson Education in South Asia, 2008.

[2] Edgar A. Kraut. *Fundamentals of Mathematical Physics*. Dover Publications, 2007.

[3] V. Rajaraman and T. Radhakrishnan. *An Introduction to Digital Computer Design*. Prentice-Hall of India Private Limited New Delhi, fourth edition, June 1998.

[4] James A. Storer and John H. Reif. Shortest paths in the plane with polygon obstacles. *Journal of the Association for Computing Machinery*, 41, No. 5:982–1012, september 1994.