AN IMITATED HUMAN MOTION OPTIMAL B-SPLINE PATH OF A BIPED ROBOT USING GENETIC ALGORITHM.

Rashmi Sahay*, Abhay Kumar, Arun D Udai*** and Dr. A N Sinha*****

***Deptt. of Computer Science **University Polytechnic ***Deptt. Of Mech. Engg

Birla Institute of Technology, Mesra, Ranchi -835215

ABSTRACT

In this work , an attempt is done to imitate the human walking for Biped Robot path trajectory using CCD camera interfacing with Visual Basic software to generate a number of B-spline curve to obtain natural walking path trajectories and then by optimizing it using genetic algorithm to get the best suited path .

Introduction

Robotics is a dynamic interdisciplinary field of study. The study of robotics concerns itself with the desire to synthesize some aspects of human function by use of mechanisms, sensors, actuators etc. Presently different aspects of robotics research are carried out by experts in various fields. At a relatively high level of extraction, robotics can be split into four major areas: Path Trajectory, Kinematics, Computer Vision and Artificial Intelligence.

As mechanical manipulators industrial robots also called robotic manipulator or robotic arms gained much popularity. In recent times biped robotic has become an interesting topic for many researcher. A biped robot is composed of two identical legs and a trunk. Each leg is composed of two links articulated with a knee. Biped robot is mare adaptable than the mobile robot in a variable environment and can have more diverse possibilities in planning the motion. It is more human friendly. However, biped is more prone to fall down, if Its proper design considering zero moment & other factors are not taken into consideration**.** It is difficult to control a biped robot for stable walking. So the generation of optimal walking trajectory is an important problem for biped robot to keep walking stably.

Path planning is the process of calculating a path for a mobile robot to follow so that it can move from one pose (position and orientation) to another [1]. When path of manipulator end is decided in terms of position and orientation of the end effectors in Cartesian space schemes, it can be converted using inverse kinematics to joint space. A method has been proposed based on cubic spline trajectory planning of an industrial manipulator [3]. A constrained time efficient and smooth cubic spline trajectory generation method for industrial robots manipulators have been proposed by B. Cao [2]. An efficient optimal trajectory planner for multiple mobile robots designed by Jason [1] and planning walking patterns for a biped robot by Qiang [4] uses cubic spline. These methods have an inherent disadvantage that the trajectory path cannot be locally modified, which is very much required when the path is to be altered because of any obstruction found. This also prevents from choosing the order of the curve if required. All are designed based on the fact that real time calculation is complex. Moreover, processing of complex path is slow due to heavy computations and numerical methods involved. However, with the advent of fast microcontrollers and processors, the problem may be overcome. Further, the Cartesian path points may be traced back to joint space, i.e. angles of the various joints, using inverse kinematics. The method considers various constraints of biped balancing and locomotion.

In this paper a method is proposed to imitate the human motion by using CCD path trajectory to get the natural path trajectory of stable slow walking of a man. The visual natural path so obtained is to be first represented by a number of B-spline curve and further optimized by Genetic algorithm to fit exactly the natural path imitated by human motion.

Walking Of Biped Robot

There are three major types of walking as discussed below. The movement of foot of each leg while walking can be represented using a curve. The suitable and natural free form curve may be a B-Spline curve that is proposed here to be implemented.

Static Walking

As long as the normal projection of the centre of mass (NPCM) is inside the supported area, the robot is always stable. This means that it could rest in this position at its actual joint angles, without toppling down. It is significant for this type of walking that the gaits look “mechanical”, because the body has to be shifted from the left to right over relatively wide ranges to move the centre of mass above the supported area of the foot that is attached to the ground. To make sure, that the NPCM is always within the mentioned area, the dynamic influences must be reduced. This slows down the forward velocity of the walker. Static walking is characterized by different phases. One half-cycle is sequenced in the “double support phase” when both feet are resting on the ground surface, the “swing phase” when one foot is lifted and swung to the front. This phase is followed by the next double support phase.

Dynamic Walking

The elementary value to control dynamic walking is the Zero Moment Point. The NPCM is allowed to be outside the supported area as long as the statement above is fulfilled. Freezing a robot during this walk would bring the walker to an unstable condition. During dynamic walking, the ZMP must be inside the boundaries of the supported area. Otherwise, it is dynamically unstable. The walking phases are the same as in a static walk.

Running

Running is a kind of dynamic walking at a high velocity. The underlying mathematical requirements are the same as those for dynamic walking. Yet, the phases are different: One cycle comprises a single support phase during which the opposite foot swings to the front. During the following phase both feet are lifted and the whole walker is in a ballistic movement which is absorbed by the next single support phase.

Parametric B – Spline Curve

The B-spline curve is an important 3D curve that can be used for the imitation of natural normal walking of human being after suitably optimized it by any adequate method.

The parametric equation [5] for B – Spline curve is given by

$$\mathbf{p}(u) = \sum_{k=0}^{M} \mathbf{p}_k B_{k,n}(u) \qquad u_{min} \le u \le u_{max} \text{ and } 2 \le n \le M + 1 \qquad (1)$$

The above equation may be summarized as below

- a knot vector **U** = { u_0, u_1, u_2, ……..}.
- n is the order the B – Spline function (degree of curve is n – 1).
- $\mathbf{p}_k$ is the set of M + 1 control points
- $B_{k,n}(u)$ is the B – Spline blending function

The fundamental formula for the B – Spline blending function $B_{k,n}(u)$ is given by

$$B_{k,n}(u) = \frac{u - u_k}{u_{k+n-1} - u_k} B_{k,n-1}(u) + \frac{u_{k+n} - u}{u_{k+n} - u_{k+1}} B_{k+1,n-1}(u) \qquad k = 0, 1, 2 \ldots.M.$$

Any, indeterminate forms like integer division by 0 or 0/0 is assumed as 0.

$$B_{k,1}(u) = \begin{cases} 1, & \text{if } u_k \le u \le u_{k+1} \\ 0, & \text{otherwise} \end{cases}$$

The u_k are again the knot values, which relates the parametric variable u to the $\mathbf{p}_k$ control points and may be defined in ascending numerical order. All the knot values together form the knot vector and integer knot are commonly used for convenience, each blending polynomial of order n is non zero over n intervals of knot vector and thus the total number of values in the vector set of point is M + n + 1.

Trajectory Planning and Balancing

Once the path for ankle [as shown in figure (3)] is decided based on starting point, maximum lift and end point of the step, the intermediate points are calculated based on fitted B - Spline curve parameters i.e. the control points. The intermediate path can be modified if any obstruction is encountered, by modifying these control points or simply by fitting another curve over the shifted intermediate points.

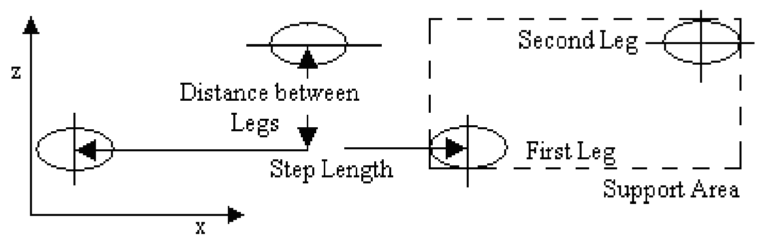

Figure (1): Plan view of path

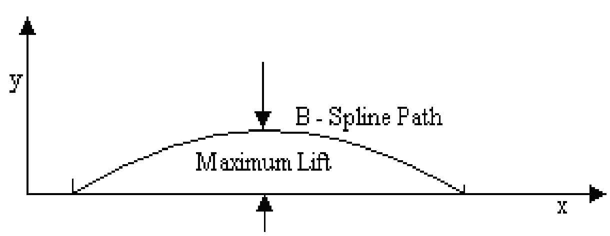

Figure (2): Side view and path of single leg

The hip path is assumed to be in a straight line and the legs are assumed to be hanging with this base line in motion. Support Area or Support polygon is the area surrounded by the corners of the feet [6]. This area is elementary for stability considerations. A statically balanced robot will have Normal Projection of Centre of Mass (NPCM) within the support area. This is also true for slow walking, where body of the robot is required to swing left – right so as to maintain the above condition. Zero Moment Point (ZMP) is the point on the ground surface about which the sum of all the moments of active forces is equal to zero [7]. For dynamic balancing the NPCM can lie outside the support area, in such case ZMP must be inside the support area. All these act as constraints to the body orientation angles, as well as the trajectory path. The foot is to be kept parallel to the ground during complete motion of the ankle joint along the trajectory.

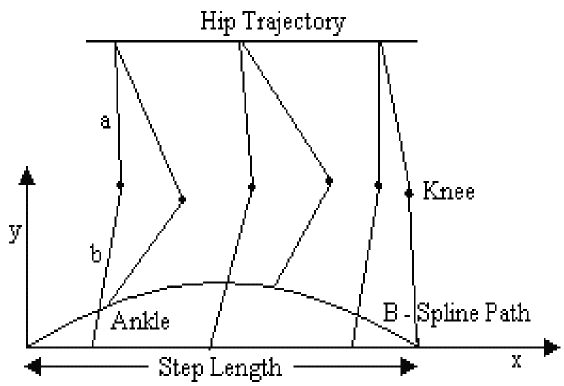

Figure (3): Forward Motion with leg movements for one step.

Once the trajectory is generated with the above constraints, it is converted to joint space parameters i.e. the individual angles of various joints using inverse kinematics.

Optimal Shape modification of B-spline curves:

There are two ways to alter the shape of the B-spline curve

- By repositioning of control points
- By Modification of the knot vector.

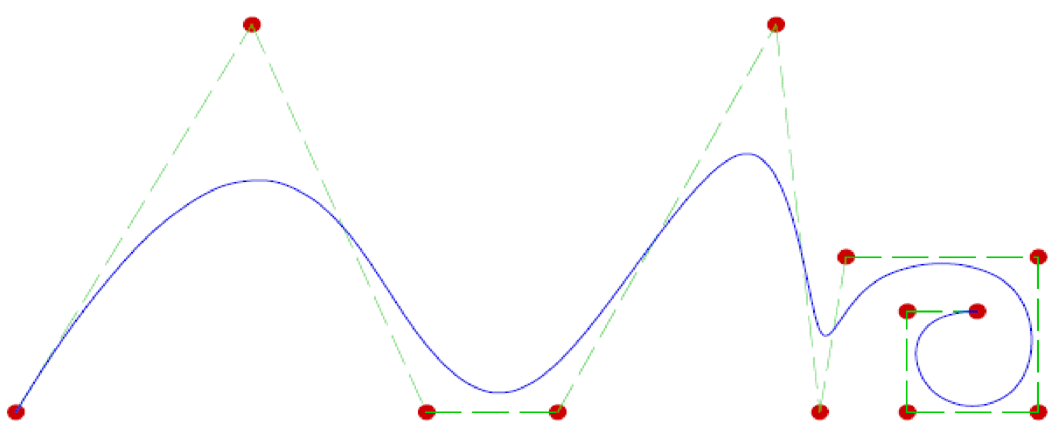

Figure 4 A B-spline curve and its corresponding control polygon.

B-spline curve is briefly defined by the user by means of so called control points. These points can be regarded as vertices of a polygon, the control polygon. The curve does not necessarily interpolate the control point (which is the case for an ordinary spline curve). Normally the B-spline curve is given such parameters that actually make it interpolate the first and last control points. The intermediate points operate as magnets on the curve, see figure 4. The control points will from here be designated P_i, the number of control points will be designated $n+1$. Other parameters that control the shape of the curve is: the degree of the curve, and the so called knot vector. The higher the degree the smother curve as shown in figure 5. The lowest degree, 1, gives straight lines between the control points. Degree 2 gives an approximating curve that will touch and share tangent with the control polygon between the control points a higher degree gives more smooth curves. The order is always the degree+1. The degree of the curve is denoted by p (power).

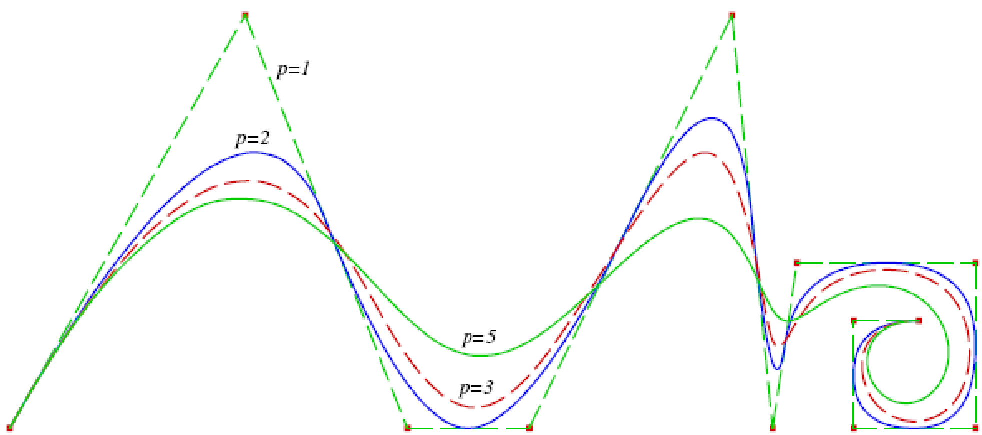

Figure 5: B-spline curves with same control polygon, but different degree. From the polygon itself; degree=1, to the smoothest curve, degree=5.

By repositioning of the control points

The simplest technique to modify the shape of B-spline curve is by repositioning the one or more control points. If a control point is moved, it will not affect the whole curve, but a limited region around that control point as shown in the figure 4.3. How much of the curve that will bee affected depends on the degree. A lower degree gives more local control (only part near the point is moved). This property of B-spline curves is called the “local control property”. Maximal allowed degree for a B-spline curve is $n =$ number of control points +1. Such a curve is equivalent to a Bezier curve with the same degree and same control points. Moving a control point in a Bezier curve will affect the whole curve, i.e. the local control property is lost when the B-spline curve degree is increased to its maximal value.

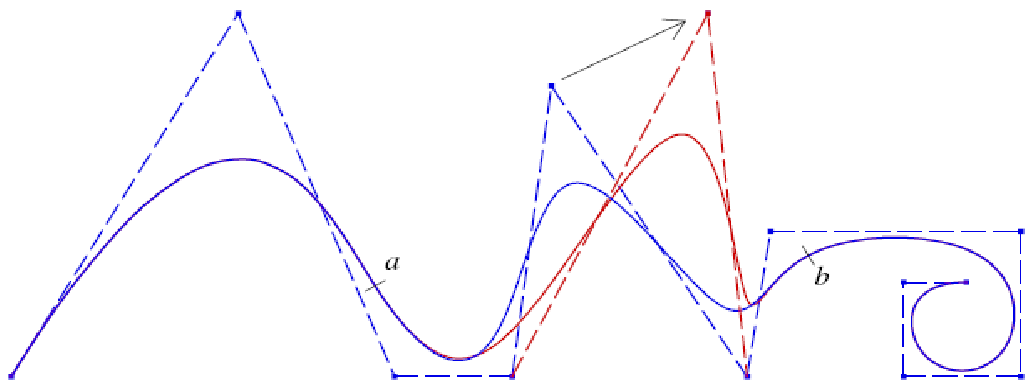

Figure 6: The local control property of B-spline curves. The figure shows an order 4 B-spline curve with 12 control points. Moving one control point only affects the curve between a and b.

Genetic algorithms:

Genetic algorithms are search and optimization algorithms based on principles of natural genetics and natural selection. The basic elements of natural genetics: reproduction, crossover, mutation are used in genetic search procedure. Although genetic algorithms were first presented systematically by Holland, the basic ideas of analysis and design based on concepts biological evolution can be found in the work of Rechenberg, and more recently reviewed and enhanced by Goldberg, Davis and many others. Philosophically, GAs is based on Darwin's theory of survival of fittest.

An overview of natural selection

In nature, the organisms that are best suited to competition for scanty resources (e.g., food, space) survive and mate. They generate offspring, allowing the transmission of their heredity by means of genes contained in their chromosomes. Adoption to a changing environment is essential for the perenity of individuals of each species. Therefore, natural selection leads to the survival of the fittest genes. The reproduction process allows diversification of the gene pool of a species. Evolution is initiated when chromosomes from two parents recombine during reproduction. New combination of genes is generated from previous ones and therefore a new gene poll is created. Segments of two parent's chromosomes are exchanged during crossover, creating the possibility of the 'right' combination of genes for better individuals. Mutation introduces sporadic and random changes in the chromosomes. Repeated selection, crossover and mutation cause the continuous evolution of the gene pool of species and the generation of individuals that survive better in a competitive environment.

Working procedure of genetic algorithm

A genetic algorithm operates through a simple cycle of following steps:

- Encoding
- Creation of initial population

- Evaluation and fitness assignment
- Reproduction
- Crossover
- Mutation

Encoding

The application of a genetic algorithm to a problem starts with the encoding. The encoding is a mapping that transforms a possible solution to the problem into a structure containing a collection of decision variables those are relevant to the problem at hand. A particular solution to the problem can then be represented by a specific assignment of values to the decision variables. The set of all possible solutions is called the search space, and a particular solution represents a point in that search space. In practice, these structures can be represented in various forms, including among others, strings, trees, and graphs. There are also a variety of possible values that can be assigned to the decision variables, including binary, *k*-ary, real, and permutation values. Basically it depends on the nature of the problem variables.

Creation of initial population

Genetic algorithm operates on a population of solutions. The initial populations of solutions are usually created randomly. When designing a genetic algorithm, it is necessary to decide what the population size must be. General wisdom dictates that increasing size increases its diversity and reduces the probability of a premature convergence to a local optimum. However, this strategy also increases the time required for the population to converge to the optimal regions in the search space.

Evaluation and fitness assignment

Once an initial population of solutions is created, the individuals of the population are exposed to an evolution function that plays the role of environmental pressure in the Darwinian evolution. Evaluation means calculating the objective function value and constraint violations. Thereafter, a metric must be defined by using the objective function value and constraint violation values to assign a relative merit to the solution called the fitness. In the obscene of the constraints, the fitness of a solution is a function of the solutions objective function value.

Reproduction or selection

The primary objective of the reproduction is to make duplicates of good solutions and eliminate bad solutions in a population, while keeping population size constant. Based on each solution's fitness, a mechanism of selection determines the mates for the genetic manipulation process. In this process individual with higher fitness values have grater chance of being selected for mating and subsequent genetic action. Consequently, highly fit individuals live and reproduce, and less fit individuals die (survival of fittest). The selection policy is ultimately responsible for ensuring survival of the best fitted individuals. This is achieved by performing the following tasks:

- Identify good (usually above average) solutions in the population.
- Make multiple copies of good solutions.
- Eliminate bad solutions from the population so that multiple copies of good solutions can be placed in the population.

There exists a number of ways to achieve the above tasks. Some common methods are:

- Tournament selection,
- Proportionate selection, and
- Ranking selection.

In this present work tournament selection scheme is used. It is an ordinal selection scheme. Tournament selection makes n tournaments to choose n individuals for s-wise tournament selection, tournaments are played between s individuals and the better solution (individual) is choose and place in the mating pool. S other solutions are again picked and another slot in the mating pool is filled with better solution. If carried out systematically, each solution can be made to participate in exactly s tournaments. The best solution in the population will win s times, there by making s copies of it in new population. Using a similar argument, the worst solution will lose in s tournaments and will be eliminated from the population. In this way, any solution in the population will have zero, one, two, ------, or s copies in the new population. Literature survey shows that tournament selection has better or equivalent convergence and less computational complexity properties when compared to any other reproduction operator that exists in the literature.

Crossover

Crossover operator is applied next to the selection. A little thought will indicates that the reproduction operator cannot create any new solutions in the population. It only makes multiple copies of good solutions at the expense of not so good solutions. The creation of new solutions is performed by crossover and mutation operators. Like the reproduction operator, there exist a number of crossover operators in the GA literature, but in almost all crossover operators, two solutions are picked from the mating pool at random and some portions of the strings are exchanged between the solutions to create two new solutions. For example in a single point crossover operator, this is performed by randomly choose a crossover site along the string and exchange all bits on the right side of the crossing site as shown in figure 7.

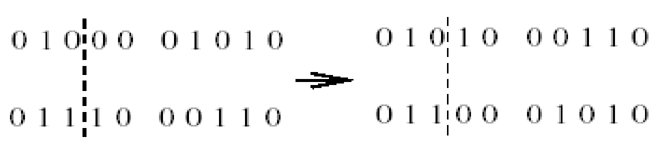

Figure 7: An illustration of the single-point crossover operator**.**

Mutation

Mutation induces sporadic and random alterations to strings. The need for mutation is to keep diversity by reintroducing divergence into a converging population so that the algorithm proceeds to global optimum instead of may be converging upon a local optimum. The biological inspiration behind this operator is the way in which a chance mutation in natural chromosome can lead to the development of desirable traits which give the individual displaying these characteristics an advantage over its competitors. The procedure of bit wise mutation is shown in figure.8.

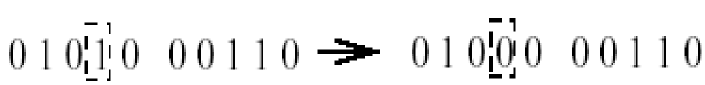

Figure 8: An illustration of the mutation operation.

The solutions created by the genetic manipulation process constitute the next generation to be evaluated. The genetic algorithm cycle is repeated until a satisfactory solution to the problem is found. The flow chart describes the working principle of genetic algorithm:

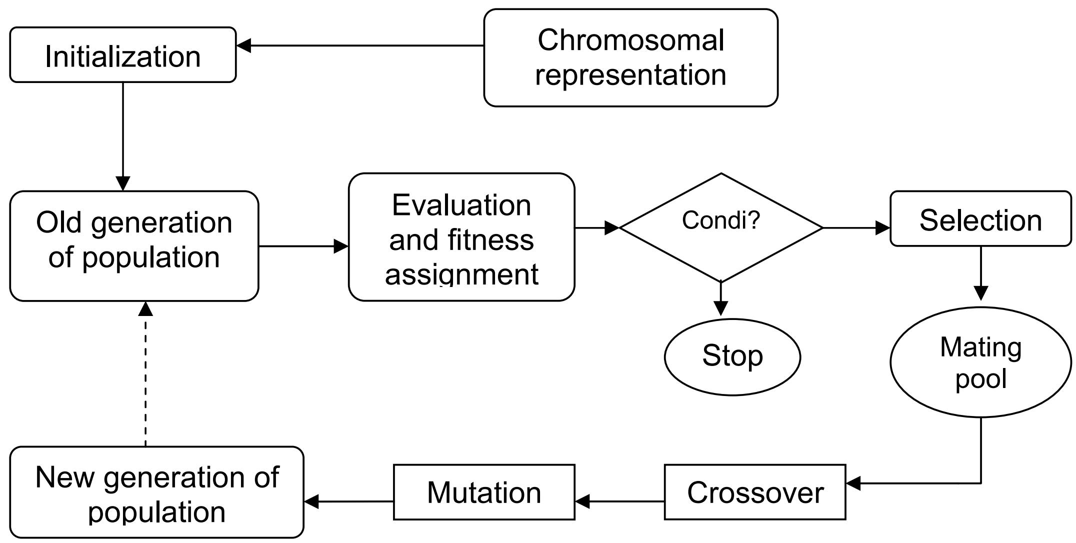

Figure 9: A flow chart of working principle of a GA.

Imitated Human Motion Path Capturing:

The human motion is captured by CCD video camera which is interfaced Visual Basic (VB) software to get the Cartesian coordinates of the path as shown below. The interfacing program is written in Visual basic.

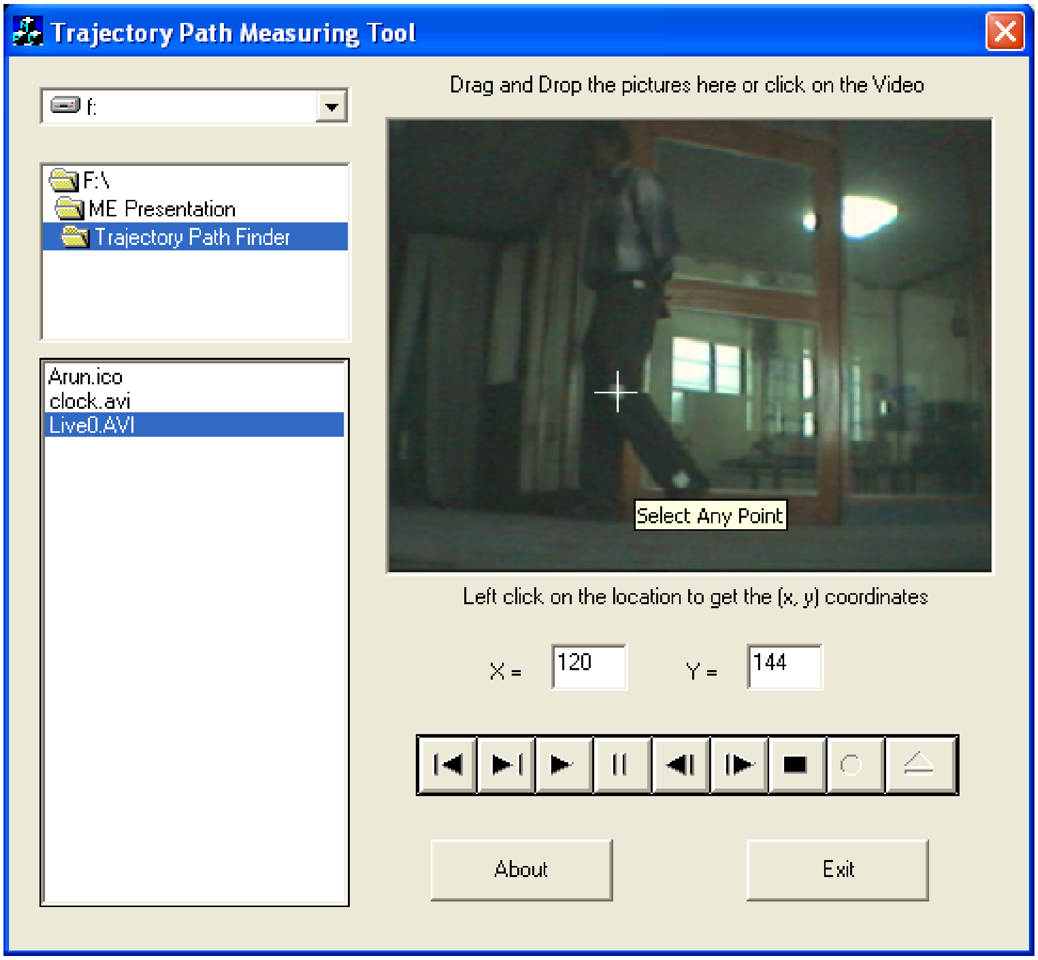

Figure: Capturing Natural Trajectory.

The table given below shows the Cartesian coordinates of the imitated human slow walk.

X (× 10 cm)	**Y** (× 10 cm)
0	0
1.5	2.5
3.0	1.5
4.5	1.0
6.0	0

Figure 10 shows the original recorded movement of the human leg at the actual movement

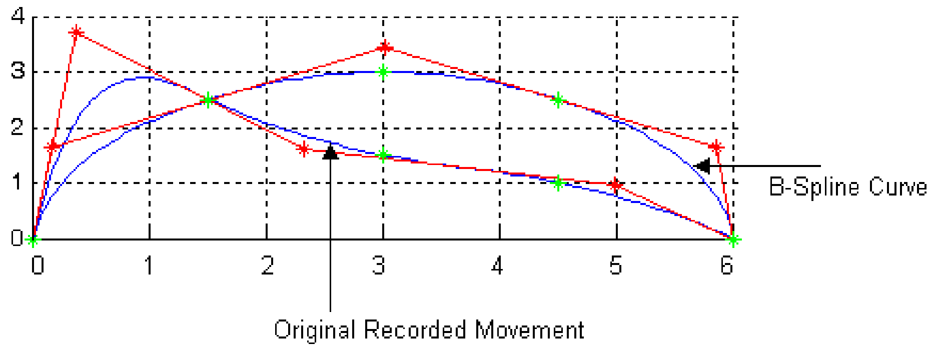

Figure10: Leg Movement (Scale: 1:10cm)

and a B – spline curve drawn with the help of control point having the same span between 0 & 6. However, it can be seen that path traced by B- spline curve is not suitably match the recorded path. This can be done by optimizing the path by genetic algorithm method.

Path trajectory profile optimization by genetic algorithm:

In this section, the principle of the path trajectory profile optimization process based on genetic algorithm is described step by step. As stated earlier, genetic algorithm is a computer-based problem solving method, which uses a computational model of the evolution in nature. The principles and concepts of genetic algorithm are derived from the theory of Darwinian evolution.

The core of our system is a mathematical model of path trajectory mechanism and a simulator program based on this model. Genetic algorithm is used to produce automatically a set of alternative path trajectory profiles for this program, run the simulator and finally evaluate the path trajectory profiles on the basis of simulator out put. The objective is to find an optimum path trajectory profile with respect to imitated human model of the path trajectory mechanism.

An overview

The following step by step description outlines, how this kind of evolutionary path trajectory profile optimization by genetic algorithm works.

Path trajectory profile representation (encoding)

To optimize the movements of the path trajectory follower, the profile of the path trajectory must be optimized to produce optimal path trajectory characteristics. In this case it is most practical to optimize the path in term of its displacement. In this present method B- spline curve is used to represent the imitated human curve. The B-spline curve and its derivatives up to $(k\text{-}2)^{th}$ derivative are continuous. In order to satisfy the continuity requirements at least a 3^{th} degree curve is required for representation of path. Also in accordance with problem specific knowledge the path trajectory should start and end with at least c^2 continuity. Meeting these continuity requirements and other boundary conditions leads to the use of 4-8 control points and their values are fixed through out the simulation runs. .

1. Initial population:

When path trajectory profile optimization process started, we create a population of path profiles. The population is initialized by generating the profiles for initial population randomly. The size of the population is N. Those path trajectory profiles are individuals of the first generation. Figure 11 shows the path profiles generated randomly.

2. Evaluation

The next step is to evaluate each individual path profile in the initial population with the original recorded path The idea is to test the generated path with recorded path. Based on the evaluation results the difference on path violation values are calculated for each path profile.

4. Selection

The next step is to select the good path trajectory profiles of initial population and then place them in the mating pool. The selection is a random process. In this process two path trajectory profiles are taken randomly from the initial population, compare the two path trajectory profiles. If the two are different then select the one with less deviation of the path. choose one with large crowding distance value and place it in the mating pool. This procedure repeat until the mating pool is filled by the path trajectory profiles or N times..

5. Crossover

The next step is crossover, in this process two individual path trajectory profiles in the mating pool are selected at random and then exchange the portions of two path trajectory profiles. This will produce two new path trajectory profiles called offspring's. In other words the crossover of better path trajectory profiles will produce best path trajectory profiles.

6.Mutation

After crossover, the randomly chosen path trajectory profiles will become subjects of mutation. This means that a random selection of randomly chosen path trajectory profile will be modified a little. If we use the crossover operator only, we are only able to recombine existing path trajectory sections. The purpose of mutation operator is to bring new information and new properties in to the population.

7. Evaluation.

The next step is to evaluate each individual path trajectory profile of the mate population with the imitated recorded path. Based on the evaluation results the best population of the path is chosen with the help of the least distance evaluation out of the many path population as shown below.

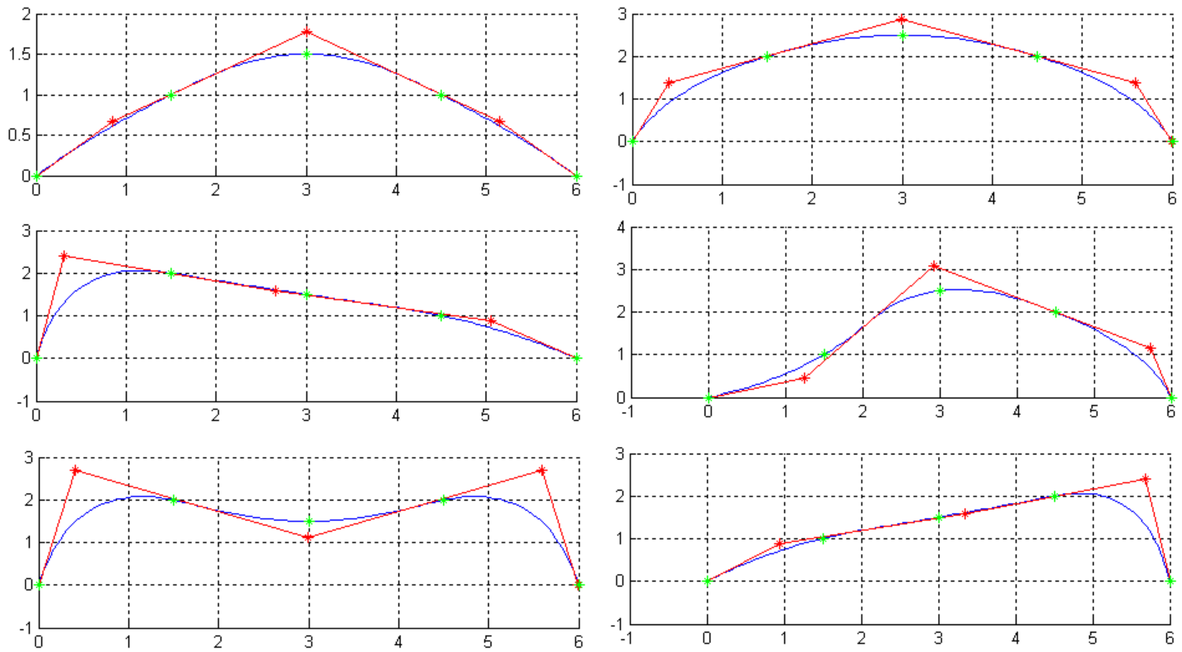

Figure 11. Random Population of B- spline paths

Conclusion: The method proposed in this work will enable the robot walking on the natural gait manner as an human being to grater extent suitable to the configuration of the robot size and wait. B- spline curve is powerful tool to represent any path after it is optimized by genetic algorithm for the different combinations of control points and knot vectors .

REFERENCES

[1] Shafaat Ahmed Bazaz, Bertand Tondu, "3 – Cubic Spline for On-line Cartesian space trajectory planning of an industrial manipulator", IEEE, 1998, AMC '98 – COIMBRA, pp: 493 – 498.

[2] B. Cao, G. I. Dodds, G. W. Irwin, "Constrained time-efficient and smooth cubic spline trajectory generation for industrial robots", IEE Proc. – Control Theory Appl., Vol. 144, No. 5, September 1997, pp: 467 – 475.

[3] Jason Thomas, Alan Blair, Nick Barnes, "Towards an efficient optimal trajectory planner for multiple mobile robots", Accessed through Google search 2005.

[4] "Planning Walking Patterns for a Biped robot", Qiang Huang, Kazuhito Yokoi, Shuuji Kajita, Hirojiko Arai, Noriho Koyachi, Kazuo Tanie, IEEE Transaction on Robotics and Automation, Vol. 17, No. 3, June 2001.