Title Page

Contents

Authors Profile

***Author**: Hemant Pandey*

***Affiliation**: Presently working as lecturer for IIT JEE training institute TIME.*

Worked for City Montessary School (CMS) –Innovation Wing (IW),Amelox college tutor www.amelox.com

***Job Profile**: Content Developer; Writer*

***Branch**: Main Branch,*

Station Road, Lucknow

***Writing profile**: Currently working on a text book for*

Grade 11-12 of mathematics with Amelox College Tutor; California (USA)

***Qualification**: Masters of Science in Mathematics and currently a Researcher*

***Postal Address**:*

16/732, Indira Nagar,Lucknow.
Uttar Pradesh – 226016(Pin) India
Mobile Number +91-9336291008, +91-9338202560

***Mailing Address**:*

E-mail- ajay_fermat@rediffmail.com
E-mail- ajay_euler@rediffmail.com
E-mail- ajay_gauss@rediffmail.com

<u>Cover Letter</u>

Dear Sir,

My name is hemant pandey and I am a young researcher.

I am proposing an paper on P Vs NP problem for possible publication. The main argument used in the paper is to search an optimal tour for traveling salesman's problem in Euclidean geometry in 2-D, in polynomial time.

This is to certify that the work is original and has not been submitted any where else for publication consideration.

Details are in the pdf attachment.

Sincerely,

Hemant Pandey

2ND JULY 2007

THE MATHEMATICS OF P V NP

THE ABSTRACT

P Vs NP problem is an open problem in the theory of optimization and asks whether two of the important complexity classes, P and NP are same.

The P Vs NP problem directly affects one of the most basic things of our modern day survival, the Internet security. This classic problem in theoretical computer science was formulated by Stephen Cook in 1971.

The RSA ciphering-deciphering technology or public key cryptography has seeds of its success, in assumption of the fact that P is not equal to NP. If we assume truth of this paper's result then newer methods have to be searched for coding public keys, and that is surely an interesting task as if now we have supposed to reach a stagnation point.

The mathematical gain of supposed truth of this result is that it opens a search for solution of the 3000 plus NP complete problems and much more.

The present proof attempts to resolve P=NP by the proposed solution of NP complete Hamiltonians path problem or Euclidean Traveling Salesman Problem, in 2-D, in polynomial time. The proof is using topology, geometry and properties of convex polygons. The proof assumes Euclidean TSP in 2-D case and hence the triangle inequality is to be satisfied.

We have attempted to find an optimal tour for Euclidean travelling salesman problem, by using methods described in the paper in polynomial time of order five i.e. O (5).

Key words: Polynomial time problem, Non-deterministic Polynomial time problem, Hamiltonians path problem, Euclidean Traveling salesman's problem, NP-Complete problem.

MEANING OF SYMBOLS

P - *Polynomial time problem*

NP - *Non-deterministic polynomial time problem*

= - *is equal to*

n - *Pi (180 in trigonometry)*

- *Is not equal to*

n^K - *'n' rose to power 'K'*

n! - *n factorial*

C(n,k) - *Combination of 'n' things taken 'K' at a time.*

I - *Set of Integers.*

HPP - *Hamiltonians path problem*

TSP - *Traveling Salesman Problem*

ETSP - *Euclidean Traveling Salesman Problem*

χ - *This implies that*

∴ - *Therefore*

DEFINITIONS

- *1. P - P means problems whose solution is bounded by a polynomial i.e. whose solution requires size of inputs expressible as a polynomial of the form* Cn^k *,where n are number of inputs, k is an integer and C is an arbitrary constant. Such problems are said to be of order 'n' .Symbol P stands for 'Polynomial'.*

- *2. NP- NP means type of problems which are solvable in polynomial time by a non- deterministic Turing machine only. Symbol NP stands for 'Non-deterministic Polynomial'.*
- *3. NP-Hard - A problem is said to be NP-Hard if an algorithm for solving it could be transformed to solving any other NP problem.*
- *4. NP- Complete- A problem which is both NP and NP-Hard is called NP complete problem.*
- *5. Triangle Inequality- According to the triangle inequality sum of two sides of a triangle is greater than the third side. In almost all cases of Euclidean TSP the triangle is satisfied.*
- 6. *Local optimal tour: A tour may be termed as a local optimal tour if it is the optimal tour w.r.t. to the points existing on the network. This tour may or may not be the optimal tour.*
- *7. Optimal branch: The optimal branch may be defined as the nearest branch chosen according to the lowest sum rule or 'a+b-c rule.'*
- *8.a+b-c rule: Refer page 12*
- *9. Interior local improvements: Local improvements are said to be interior local improvements if we change only the relative positions of points without constructing a virtual segment.*
- *10. Exterior local improvements: If we change position of points by creating a virtual segment we get a external local improvement. Note that once this improvement is introduced we cannot return to our starting point by simply reversing the steps as reversal of a virtual segment is not defined as it is arbitrary.*

1. INTRODUCTION

Computation complexity had its seeds sown way back in 1936 when Turing developed his theoretical computational model. Further developments resulted in 1960's by Hartmanis and Stearns when they coined the idea to measure time and space as a function of the length of the input.

The work of Cook and Karp in early 70's gave birth to the most important and fundamental concept of computational complexity, NP-Completeness and its most fundamental question, whether P= NP.

1.1. THE COMPLEXITY CLASS OF P AND NP

The relationship between the complexity class P and NP is an unsolved question in theoretical computer science.

The relationship between the complexity classes P and NP is studied in computational complexity theory which deals with the resources required to solve a given problem. The resources may be the steps required to solve a problem and space needed for formulation of a solution.

The computational machine in the context is assumed to be deterministic, i.e. it always performs sequential operations, one after another.

Theoretically P class consists of problems that can be solved on a deterministic computational machine in amount of time which assumes polynomial equations in the size of inputs. Mathematically this is measured as order of a problem. For P class this is represented as O (K), where K is a positive integer .We are attempting a solution of order five i.e. O (5).

On the other hand NP class means problems whose solution can only be verified on a deterministic computational machine and can be found only by a Non-deterministic computational machine in polynomial time.

1.2. THE CONCEPT OF NP COMPLETENESS

NP complete problems are those problems which are the 'tough most' and 'hardest' problems in NP.NP complete problems are those NP- hard problems which are in NP.

Precisely a NP-hard problem is one into which any NP problem can be transformed in polynomial time.

The beginning of NP- complete problems attributes to the Boolean satisfiability problem, which was proved to be NP complete by Stephen Cook in early 70's.This is now also known as Cook's theorem. The common NP complete problems are subset sum problem, minesweeper, Traveling salesman's problem and Hamiltonian's path problem.

2. THE PROBLEM

Statement:

The P Vs NP problem has a classic one line statement whether P=NP?

Mathematically P Vs NP states

P = NP or P # NP i.e. whether or not P is equal to NP.

2.1 MEANING AND DEFINITION OF P & NP: -

P states for polynomial time problems, problems that can be effectively solved in polynomial time by using a deterministic computer. Polynomial time means reasonable time in common terms and in technical terms it means that it is expressible in the form of a polynomial equation.

χ *P problems are characterized by a polynomial equation.*

i.e. $P = Cn^K$ *where n is the size of inputs or data and K is a positive integer. We call that these are of order K, i.e., O (K).*

Precisely

P = Polynomial time i.e. time required to solve a P type problem.

C = Arbitrary constant.

n =Size of input or data.

K =Order of P type problem.

Hence P represents a class of polynomial in which total numbers of outcomes are proportional to an integral power of inputs.

NP problems are those in which time required to get a solution is unreasonably large, though the cases are too much, to calculate each case itself may need trivial arithmetic only.

Only problem is number of cases, which are too large for a normal computer to handle fully in polynomial time.

NP literally means non- deterministic polynomial time problem i.e. the problem which can be solved in polynomial time only by a non deterministic computational machine only.

A computer in polynomial or reasonable time cannot handle NP problem.

More often than not there are NP problems that may take centuries for a full solution by brute-force method i.e. by method of checking all options.

There are about 3000 plus NP complete problems.

A NP complete problem is one that is father of all NP problems. It means that if one NP complete problem is solvable in polynomial time so can be any other problem.

Mathematically NP completeness is the generalization of NP problems. In order to prove or disprove P = NP, we have to prove or disprove it for one of those 3000 NP complete general, problems.

2.2 RESULT:

We propose a new result P =NP; We will establish this result for NP complete Hamiltonian's path problem, or Euclidean Traveling salesman's problem. We will find an optimal tour for ETSP with the help of geometrical and topological properties of polygons.

Our proof aims to solve Hamiltonian's path problem or Euclidean Traveling salesman's problem in polynomial time of fifth degree at most.

i.e. for HPP or TSP

We propose $P = Cn^5$ *at most, i.e. NP complete ETSP can be effectively solved in polynomial time of order 5.*

3. THE PROOF

Hamiltonian's path problem HPP or Traveling salesman problem TSP is a well known NP complete problem. We would try to establish that it is solvable in polynomial time of fifth degree at most. Before that we must state TSP or HPP.

ETSP: Suppose there is a salesperson that has to visit several cities in order to sell business. He has the specified map of all the cities that come in his way. Obviously his problem is to find shortest possible route or the optimal tour that covers all the cities. We assume Euclidean TSP onwards so triangle inequality is satisfied and all the maps are drawn on a 2-D plane.

Obviously we can name all the routes and get the answer instantaneously. But the bone in the dish is not summing the distances from city to city. It is the number of such cases.

For 'n' cities total cases turn out to be n! , which is a whopping number even for values of 'n' as small as 100.

Therefore even for modest 100-city tour there are 100! cases.

These cases are too large for a deterministic computer to handle. It may take decades for a fastest computer on earth to find optimal tour or shortest possible route for say 1000 cities only.

Actually computers can handle polynomial time processes i.e. where $P = Cn^k$.

These Polynomials doesn't grow that fast if 'n' is the variable or size of data.

Here 'n' = Number of cities or size of data or input.

$P = Cn^{10}$ *(say)*

Doesn't grows as fast as say $P = C.3^n$

Here latter are called exponential time processes. After them comes NP processes.

Now we will prove that HPP or ETSP is solvable in polynomial time using geometrical & topological properties of polygons applied on topologically equivalent maps.

Mathematically we will show that total cases for ETSP are reducible to Cn^5 *from n!, which means that the solution becomes polynomial.*

Our solution is geometrical in nature and assumes ETSP on topologically equivalent maps.

For a start we assume that maps available are topologically correct i.e. in which relative distances matter and no scaling is required. The emphasis is on the property exhibited by each point and its relative position.

For e.g. in Fig 1 below

$d(A_1A_2) < d(A_1A_3) < d(A_1A_4)$ *etc.*

Here $d(A_i A_j)$ *is usual distance function measuring distance between any arbitrary points* A_i *and* A_j *relative to distance between other arbitrary points* A_m *and* A_n *(say).*

Space for Fig.1

These maps are topological maps only. We again state that the distances are relative only and emphasis is on the property exhibited by each point not on their actual position.

3.1 THE ISSUE OF SHORTEST ROUTE-SPECIAL CASE

POINTS ON THE PERIPHERY OF CONVEX POLYGON

We will state and prove a general theorem about shortest route through the periphery of a standard convex polygon.. We start with few definitions.

Standard convex Polygon: A standard convex polygon or SCP for short is one in which all the internal angles are between $90^0$and 180^0. A peculiar property of SCP is that all diagonals are greater than the two sides forming it, or adjacent sides to it. It is easy to establish since in a right triangle hypotenuse is diagonal or greatest side and as the opposite angle grows the diagonal side dilates. So if one angle is larger than 90^0 then one side i.e. side opposite to the before said angle is the largest side.

Now we are in a position to state our former result.

3.2 THEOREM

For all points lying on the periphery of a SCP, the shortest route between them is through the peripheral path.

This can be established without any trouble. Any other route other than peripheral route will include one or more diagonals. As stated before in SCP the diagonals are larger than the forming sides. Hence if three diagonals replace three sides they would increase the net distance.

We can prove it rigorously too as follows: -

Space for Fig. 2

Let the original route value along periphery be 'N'

Case 1: When a diagonal is joined between two consecutive points

Let A14 is joined to A12, so the point A13 is now out of network. [Refer Fig.2].

Now since we have to cover each point of the network, A13 has to be joined to some other point. Let A13 is joined to A3 and A4.These points are arbitrary .The important point is not the point but the property exhibited by each point. If A13 is joined to any other point the property exhibited by the point would be the same as with this point. Note we are talking of topological properties where only the relative position matters.

∴ ***Now new network distance is***

N-A14A13-A13A12+A14A12+A3A13A4A13-A3A4

Now A14A12 >A14A13 (A14A12 is the adjacent diagonal of SCP and by the definition of SCP it is greater than the side forming it)

Further A3A13 >A13A12 (Since by the definition of SCP the shortest distance from a point on the periphery is next point to it on either side, all other branches from emerging from it are the diagonals)

Finally A4A13>A3A4 (A4A13 is the adjacent diagonal of SCP and by the definition of SCP it is greater than the side forming it)

x ***The Net network distance increases as sum of the adding distances is greater than the subtracting distances.***

Hence for the points laying on a standard polygon the shortest route or the optimal tour is along the periphery.

Case 2: When a diagonal is joined between any two non consecutive points

We now consider the case when a diagonal is joined between non consecutive points. The proof is similar. Let us take any arbitrary point .Let a diagonal be joined between A5A10.So points from A6 to A9 are abundant. Let these points be joined to segment A1A15.

Now adding distance =A5A10 +A1A6+A9A15

And subtracting distance=A5A6+A9A10+A1A15

Now A5A10 > A5A6 (A5A10 is the adjacent diagonal to A5A6 and by definition of SCP former is greater than the latter)

A1A6 > A1A15 (Same reason as above)

& A9A15 > A9A10 (Same reason as above)

As stated before this proof is general since the relative position of points and property exhibited by the point matters.

IMPORTANT: Although this theorem is a new result but the proof works very well without the assumptions of the proof. This proof may save few steps but it does in no way affect the truth of the result (given in next section) or the order of given problem. This is provided only as a guideline for the shortest route if the points lie on the periphery of a SCP.

6. THE ISSUE OF SHORTEST ROUTE OR THE OPTIMAL TOUR-GENERAL

4. THE ISSUE OF SHORTEST ROUTE OR THE OPTIMAL TOUR-GENERAL CASE

4.1 THE GENERAL DOMAIN

How can we use the before proved theorem or otherwise, to get the shortest route or the optimal tour between the points?

Here is a possible answer.

Consider the general domain of points shown above. The orientation of the points is arbitrary. The important point is not the points or their placement but the property exhibited by each point and its relative position. Our basic approach for the shortest route is that we start from the shortest and keep it shortest all the while. With the help of this approach we will get a shorter tour which is at least locally optimal, i.e. optimal w.r.t. to the starting points. After that we apply corrections or arrays of corrections to get the optimal (Universal) tour. Even if the previous result is not used in general we start from any route and with the process of constantly improving our route and discarding longer routes in the process we reach at the shortest route. The method used is basically the method of elimination of longer routes and careful selection of shorter routes.

We start with the ouster most mesh of one map and join them so the maximum numbers of destinations lie on a standard convex polygon. From theorem the shortest route lies on the periphery for these cities. Even if it does not hold good then also we join them to all the exterior points and proceed.

Our next object is to join to these branches the points which are nearest to them than any other two points, branch or segment. For this we calculate ' a +b – c' for all 'n' cities for all the branches of Outer mesh if ' a +b – c' is minimum for any of the branches we join it to the branch. This may be termed as nearest or cheapest insertion to the outer convex shell.

We would like to define ' a + b – c' rule. In the Fig.[3] if point O is added to the network to the segment A1A2 then

a = Adding distance on the segment of the network due to new point O and point A1 of line segment A1A2.

b = Adding distance on the segment of the network due to new point O and point A2 of line segment A1A2.

c = Subtracting distance on the network due to the segment A1A2.

Space for Fig.3

∴ *a +b – c*

= Net addition to the existing network due to new point 'O'.

As seen above for the section A1O, A2O is the adding distances & A1A2 is the subtracting distance from the network, see [Fig.3]. We find this value for all segments

We repeat the process for new joined branches till we reach a network that looks like [Fig.4]

Space for Fig. 4

The above network has following properties.

This is the shortest route or the optimal tour (local optimal tour) between the points on the network joined so far. No confusion about the term local optimal tour should stem out. This is the optimal tour for the points joined so far w.r.t. themselves but this is a local optimal tour w.r.t. the points all the points as better combination may exist between these and other points in the optimal tour. We would take this case under the heading virtual segments or hypothetical diagonals. The virtual segment case puts each point under testimony, and each point is considered vulnerable to a change in position, after application of point to segment (Section 6.1) and segment to segment rule (Section 6.2).

All the points that are left are either nearer to themselves or to branches other than on the network. These may be called hypothetical diagonals or virtual segments. The name pops up as they are hypothetical diagonals or virtual segments which can still be joined between the points on the already existing network of [Fig.4]

4.2 THE NEXT NETWORK CASE

After we have the original network intact we start with other independent points, independent in the sense they are nearer to themselves than to any of the points on the existing network. We repeat the same process of the general domain till all the points gets exhausted [refer to Fig. 5].

Space for Fig. 5

So our net shortest route may now look like fig. 5. We have taken four networks for simplicity.

The four networks are respectively the shortest route between the particles of the corresponding networks .We now use segment rule to join these networks.

It is that the networks are joined via the closest segment.

The segment length is calculated as follows. (For details refer section 6.4)

‘a + b –c –d ’; Here a ,b are adding distance & c , d are subtracting distances.

Suppose we have to join A1A2 to B2B3 [Refer Fig. 6].

The net adding distance is

a =A1B2

b =A2B3 &

Net subtracting distance is A1A2 & B2B3. Similarly we check for other segment B3B4 (say).

For whichever two segments the ‘a +b –c –d’ is minimum we join them.

Next case is the case of hypothetical diagonals. Now our shortest route may look like