Assesing demand in stochastic locational planning problems: An Artificial Intelligence approach for emergency service systems.

Yorgos N. Photis and George Grekoussis

University of Thessaly, Department of Planning & Regional Development

2003

ASSESSING DEMAND IN STOCHASTIC LOCATIONAL PLANNING PROBLEMS: AN ARTIFICIAL INTELLIGENCE APPROACH FOR EMERGENCY SERVICE SYSTEMS.

Yorgos GRAIKOUSIS
Phd Candidate
School of Rural and Surveying Engineering,
National Technical University of Athens,
9 Heroon Polytehneiou St. Zografou 15700 Athens
Greece
Tel: +003 210 7722635
Fax: +
E-mail: ggrek@mail.ntua.gr

Yorgos N. Photis
Assistant Professor of Spatial Analysis and Locational Planning.
Department of planning and regional development.
University of Thessaly Volos,
Pedion Areos, 38334
Tel: +003 0421074418
Fax: +003 0421074382
E-mail: yphotis@uth.gr

Abstract:

The efficiency of emergency service systems is measured in terms of their ability to deploy units and personnel in a timely and effective manner upon an event's occurrence. Aiming to exploit stochastic demand, spatial tracing and location analysis of emergency incidents are examined through the utilisation of Artificial Intelligence in two interacting levels.

Firstly, spatio-temporal point pattern of demand is analysed by a new genetic algorithm. The proposed genetic algorithm interrelates sequential events formulating moving events and as a result, every demand point pattern is correlated both to previous and following events. Secondly, the approach provides the ability to predict, by means of neural networks optimised by genetic algorithms, how the pattern of demand will evolve and thus location of supplying centres and/or vehicles can be optimally defined. Neural networks provide the basis for a spatio-temporal clustering of demand, definition of the relevant centres, formulation of possible future states of the system and finally, definition of locational strategies for the improvement of the provided services.

Keywords: Location planning, Spatial analysis, Artificial Intelligence

1 Introduction

Location planning problems can be stated as the optimal location of *p* centres, in a region and the corresponding allocation of demand *d* in these centres. Many system parameters in stochastic locational planning modeling might be taken as uncertain including travel times, construction costs, demand locations, or even time intervals (Owen and Daskin 1998). During these intervals, variance in variables values, defining the location problem might be observed. The study of these changes may lead to their potential prediction (Daskin 1995).

A classical stochastic problem, handling uncertain demand, is the ambulances location in a city (Beckmann 1999). Future phone calls, for emergency location occurrences are not known with certainty. However their prediction should be achieved so that future demand been covered in an optimal way (Daskin1995). Demand is mostly treated as static entry data to a stochastic model. In this paper, demand is treated as spatio-temporal point pattern whose evolution is examined by the utilization of methods and techniques of artificial intelligence.

Artificial intelligence incorporates and simulates basic characteristics of human thought in order to solve practical problems. Neural networks and genetic algorithms are methods and techniques of artificial intelligence that imitate the way that human learns, creates symbols, recognizes and represents elements of reality.

A neural network aims at the recognition of mathematical patterns among data sets (Openshaw 1997). Neural networks are a form of multiprocessor computer system, with highly interconnected processing elements that interact with simple adaptive scalar messages.

As Alp and Drenzer (2003) refer, genetic algorithms (GAs), are heuristic search methods that are designed to mimic the evolution process. GAs are a simulation of natural evolution. As Darwin mentioned in his evolution theory, the fittest living entities survive over weaker. By the same way GAs perform parallel procedures aiming at finding optimal or near optimal solutions, over a set of possible solutions. Although GAs have been applied to hard problems, with very complex objective functions, there are few applications to facility location problems. The spatio-temporal analysis of demand in stochastic locational planning problems is achieved through the proposed genetic algorithm.

The paper is structured as follows. Section 2 analyses the objectives of this paper. In section 3 genetic algorithm's main concepts are presented. Sections

4 and 5 describe the proposed genetic algorithm and the neural network optimized by genetic algorithm respectively. In section 6 the case study and the computational results are presented. Finally section 7 concludes the paper and provides recommendations for further research.

2 Objective

In this paper demand is analysed spatio-temporally and its future evolution is examined. Many models have been developed handling variable demands (Laporate et all 1994, Perl and Ho 1990) such as: price-sensitive demands (Wanger and Falkson 1975), demands weights satisfying a specified distribution (Berman and Parkan 1981), elastic demands with respect to both distance and price (Labbe and Hakimi 1991, Lederer and Thisse 1990).

Demand in location problems is defined as a set of distinguishable locations and is depicted by a set of points with known coordinates. The analysis of demand in location planning problems can be performed by spatial point pattern analysis. A point pattern consists of a set of events at a set of locations, where each event represents a single instance of the phenomenon of interest (Sullivan, Unwin 2003). Especially, demand formulates spatio-temporal point patterns, when changes occur diachronically.

Such kind of demand point patterns arise in cases as: spatio-temporal analysis of mobile telephony calls, spatio-temporal analysis of health cases such as virus spread, spatio-temporal analysis of accidents in road network, spatio-temporal analysis of calls accepted by national services or demand analysis in a stochastic locational planning problem. In all the above cases demand is not associated with defined objects that evolve over time, but with events that appear in multiple sets without any linkage information among them.

In spatio-temporal point patterns, multiple sets of events occur. By extending spatial statistical analysis of single point sets to that of multiple point sets, by multivariate point pattern analysis, broader research questions can be addressed about interaction and interrelation. Although multivariate point pattern analysis is used to describe patterns of several distinguishable point sets, it is more or less an extension of traditional point pattern analysis techniques (Lu and Thill 2003). These techniques, estimate spatial association between two events sets by calculating a mixture degree among events. Multivariate techniques do not embed possible pair joining in spatial analysis, failing to address spatial correspondence. The correspondence of events from distinct sets is a key issue in multiple point sets pattern analysis, since these events include some linkage among places based on physical connections, functional interactions or processes relating one place to another, that have to be revealed (Lu and Thill 2003).

However, this correspondence is not always defined. In this case, events from different patterns have to be matched. In this paper, a new genetic algorithm is proposed identifying correspondence among events, by examining the evolution process of spatio-temporal point patterns diachronically. A succession of events is established through time in a way that the diachronic course of the phenomenon in the study region is defined. The genetic algorithm creates spatial time-series of events, corresponding events that do not belong in same time intervals, but in successive ones based on the criterion of spatiotemporal nearest neighbour. As a result each point pattern is correlated to both the previous and following data sets and the trajectory of each event arises. Neural networks are then implemented, in order to predict future states of demand patterns. In these states, location of centres may be achieved, so that minimization of certain objectives is fulfilled.

3 Genetic Algorithms

GAs were proposed by Holland in 1975 as a mathematical transfer of evolution theory, introduced by Darwin, in the positive sciences. In nature, species evolve a complicate mechanism of resistance and behaviour in order to increase their probabilities to survive. As a result, the most powerful species survive while those that are weak either disappear or mutate creating new species. New generations inherit the attributes of their ancestors and as life proceeds leading to improved organisms, attributes are filtered to keep the best. Attributes are registered in the genetic code, the chromosome. Each chromosome is consisted of a set of genes. As species mate and reproduce giving birth to offspring, genes are mixed creating new chromosomes.

By a similar way GAs simulate reality by the following way. An initial solution of problem is coded into a series of genes. Genes receive certain values that are called alleles. Usually the allele of a gene is 0 or 1. Each series of gene is called chromosome and represents a feasible solution. A certain number of chromosomes consists population. The initial population is created randomly. In evolution theory organisms mate and reproduce, in order to survive and improve, while in Gas theory solutions of a problem mate and reproduce in order to provide better solutions.

Chromosomes-solutions from current population are selected and used to form a new population. The selection is based in a factor called fitness. After a cycle of continuous reproductions and generations, the algorithm ends at chromosomes with high fitness. The chromosomes of last generation usually provide the optimal solution.

There are three components that must be defined for a GA to run: define a representation (chromosome), define the fitness function and define the genetic operators (selection, mutation, cross-over)

a) Chromosome representation: Each chromosome represents a solution to the problem and is composed of a string of genes.

b) Initial population: A set of possible solutions that can be created randomly or by using specialized, problem-specific information. It is the starting-point for the genetic algorithm.

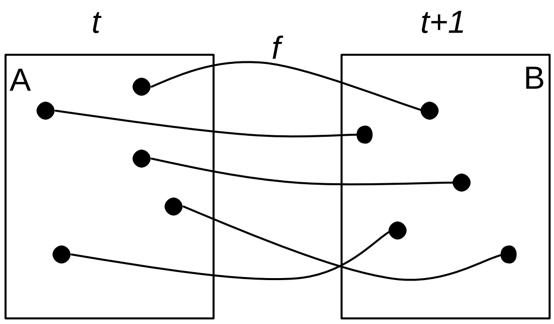
c) Fitness evaluation: Fitness evaluation involves defining a fitness function against which each chromosome is tested. As the algorithm proceeds we would expect the individual fitness of the "best" chromosome to increase as well as the total fitness of the population as a whole.

d) Selection: In order to select chromosomes based on their fitness value from the current population for reproduction, many mechanisms have been proposed such as rank selector, roulette wheel selector, tournament selector etc.

e) Crossover and Mutation: Crossover and mutation are probabilistically applied to create a new population of individuals. Once a pair of chromosomes has been selected, crossover can take place to produce offspring. The parent chromosomes exchange some of their genes creating new chromosomes. A crossover probability of 1.0 indicates that all the selected chromosomes are used in reproduction. Better results are achieved by a crossover probability of between 0.65 and 0.85. Mutation involves flipping a 0 bit to 1 or vice versa. The number of bits to mutate and the specific bits to mutate is chosen in a random manner.

4 Definition of the proposed genetic algorithm

Suppose we have two point patterns in the same region but in two sequential time intervals (graph 1). The proposed genetic algorithm matches these two point patterns by mapping through a bijective function $f: A \to B$, events $S: \{s_1 \ldots s_n\}$ from set A in set B so that for each $s \in: A$ there is one and only one event $f(s) \in A$.



Graph 1. Point matching

4.1 Type of GA

The type of the proposed genetic algorithm is steady-state. This algorithm uses overlapping populations with a user-specifiable amount of overlap. The algorithm creates a population of individuals by cloning the chromosome or population (Wall 1996). For each generation, the algorithm creates a temporary population of individuals, adds these to the previous population, and then removes the worst individuals in order to return the population to its original size. Newly generated offspring are added to the population, and then the worst individuals are destroyed (so the new offspring may or may not make it into the population, depending on whether they are better than the worst in the population).

4.2 Chromosome representation

Chromosome representation is essential to genetic algorithms in order to solve an optimization problem. The selected representation should be able to represent any solution to the problem in a single data structure. Infeasible solutions should be into consideration to the setting of the objective function. The genetic algorithm will create a population of solutions based on a sample data structure.

Each chromosome R_j consists of k genes g_k :

$$R_j = \{ g_1 \ldots g_j \}$$

where:

k is the chromosome length

$j=\{1 \ldots m\}$

m the number of chromosomes in a population.

With the restrictions:

1) $k = n$

where :

n is the number of events *s* in each set *A*,*B*.

2) There is no gene allele in a chromosome with the same value with another one.

Each gene g_i gets as allele the *id* code of points *s*, $g_i=\{1,\dots k\}$.

4.3 The fitness function

The pairs are selected with criterion of minimizing the fitness function *F* that is defined below (1).

$$F = \min \sum_{i,j=1}^{n} (\vec{s}_i^{\,t} - \vec{s}_j^{\,t+1}) \tag{1}$$

Where $\vec{s}_i^{\,t} = \begin{bmatrix} x_i \\ y_i \end{bmatrix}$ is the vector of event *i* in space *A* in time *t* , and $\vec{s}_j^{\,t+1} = \begin{bmatrix} x_j \\ y_j \end{bmatrix}$ the vector of event *j* in space *B* in time *t+1*.

Restrictions:

One to one correspondence:

i = 1,…a,…n

j= 1,…b,…n

If a random pair is (P_a, P_b) then

$\forall (P_i, P_j)$ is

$(P_i, P_j) \neq (P_a, P_j)$

and

$(P_i, P_j) \neq (P_i, P_b)$

The output of the algorithm is the set S_E that includes point matching over time.

4.4 General settings

The population size is set to 150 and the number of generations to 1000.The selection of genes is based in the Tournament selector method that uses roulette wheel method to select two individuals and then picks the one with the higher score.

Each chromosome has two primary operators: crossover, and mutation. The proposed crossover method is ‘Sexual crossover’ that takes four parameters: two parents and two children. The crossover function returns the number of crossovers that occurred as well. The probability for crossover is set to 0.85. The mutation probability is set to 0,1.The mutation operator was programmed so that nodes don’t have the same id after mutation.

5 Neural network

The objective of the proposed neural network is the pattern recognition among geographical time series S_E generated by the genetic algorithm and the forecast of their evolution in future. The selection of the appropriate network is a complicate task. A plethora of different combinations, concerning hidden layers, learning rate, learning rule, momentum, step size, epochs should be performed in order for a neural network to be trained successfully. The number of possible combinations is endless and only the multiple trials and tests will lead to a trained network. It is obvious that this process is time-consuming and to a large extent depends on the experience of the researcher. For this reason the optimisation of the neural network is achieved by the utilization of a genetic algorithm.

The genetic algorithm firstly creates randomly a population of entities. These entities are distinguishable neural networks of different architecture. Each neural network constitutes a chromosome and the parameters of its architecture are depicted in the genes. During the initial population, each neural network is trained and tested in a specified number of epochs resulting the mean square error between input and output vector.

Selection operator selects those neural networks that present small mean square error. The parameters of these neural networks are combined by the crossover operator, giving birth to the next generation of neural networks. Moreover, mutation operator is applied changing genes alleles of certain chromosomes, that is to say certain parameters of architecture. The neural networks of this generation are trained and tested. The process continues, selecting the better neural networks and the better parameters, giving new generations of neural networks.

The process terminates when the number of maximum generations or the time duration of algorithm exceeds a certain value. The neural network that presents the minimum mean square error is selected as the one that will produce final prediction.

6 Application

The application concerns in the demand analysis of 520 urgent incidents that fire brigade served during five months (time interval) of the year 2003 in the metropolitan region of Athens. When dealing with public sector institutions, this reflects the significance for state or local officials to determine the optimal locations for emergency stations and vehicles. Fire brigade should serve in the optimum way this demand by locating vehicles inside the city. This location postulates the analysis of the demand in question. The proposed methodology detects relationships among events in different time intervals.

6.1 Event matching (Genetic algorithm)

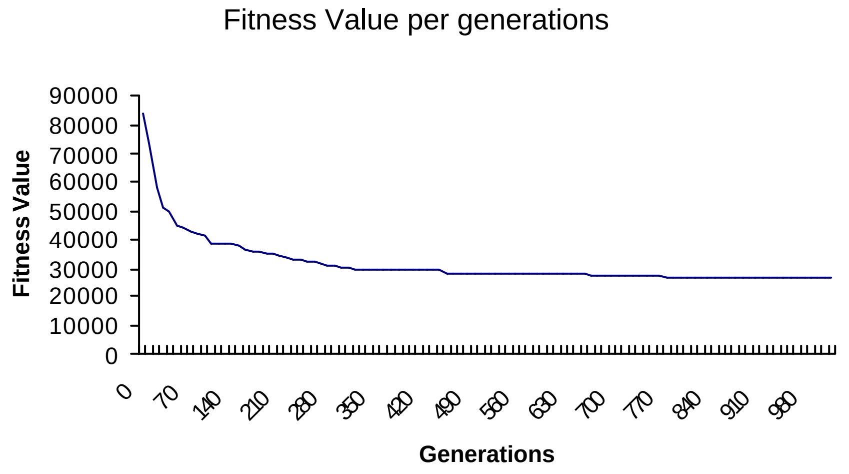
The algorithm runs four times, one per time interval. The results of the algorithm for time interval t_1-t_2 are in table 1.

Table 1. Total statistics after 1000 generations

Total generations	1000
Number of selections since initialization	100000
Number of crossovers since initialization	800000
Number of mutations since initialization	8900
Number of replacements since initialization	100000
Maximum score since initialization	156878
Minimum score since initialization	23561,9
Average of all scores	42291

Average of maximum scores	31902,4
Average of minimum scores	33134,6
Standard deviation of final population	0.0089

Fitness value per generation is depicted in graph 2. The fitness function is minimized in value 23561,9 in the final generation.



Graph 2. Fitness Value per generations

The percentage of the events of time t_1 that have been corresponded with the first nearest neighbor is 57,6% (K-NN=1), while 80% have been corresponded with the first or second nearest neighbor. Furthermore 100% of the events achieve a correspondence distance smaller than average distance. The overall results, for all time intervals are in table 2.

Table 2. K-Nearest neighbors per time interval

K-NN	1	2	3	4	4 >
Time Interval					
t_1-t_2	57.6%	24.4%	7.1%	7.1%	6.6%
t_2-t_3	51.8%	13.9%	13.9%	2.3%	17.9%
t_3-t_4	53.1%	13.6%	25.0%	4.5%	3.6%
t_4-t_5	57.6%	14.2%	18.5%	9.5%	0.0%

6.2 Demand prediction (Neural network)

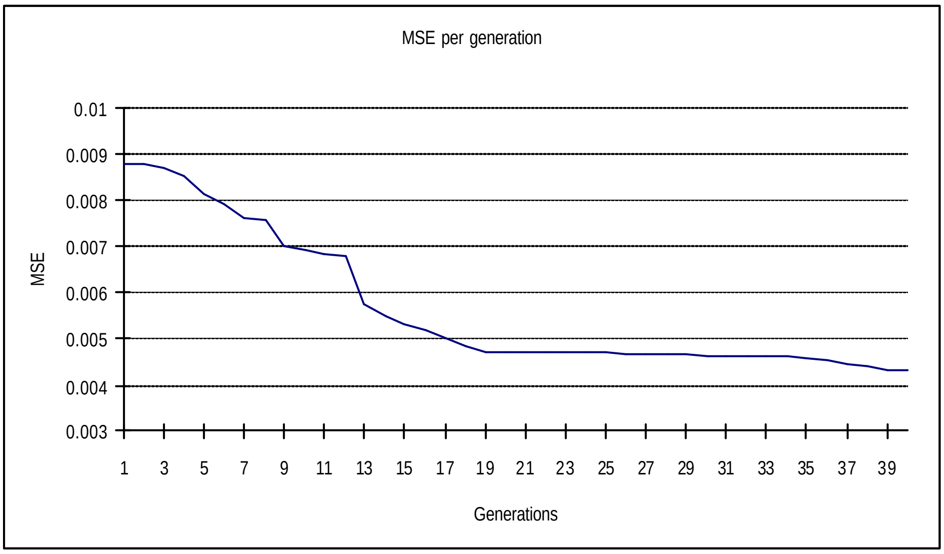
The general type of the neural network is multilayer perceptron. The genetic algorithm defines the number of hidden layers and neurons, the learning rule, the learning rate, the momentum, the transfer functions, and the number of epochs. The genetic algorithm is steady state. Population is defined in 100 and generations in 40. The training epochs in each network are 3000.

Practically, this means that in each generation *100* multilayer perceptron neural networks are created. Those who attribute mean square error under a certain threshold are selected for crossover and mutation. The best neural network is selected as the one that will predict demand pattern. The parameters of the final neural network are presented in the table 3.

Table 3. Neural network and genetic algorithm parameters.

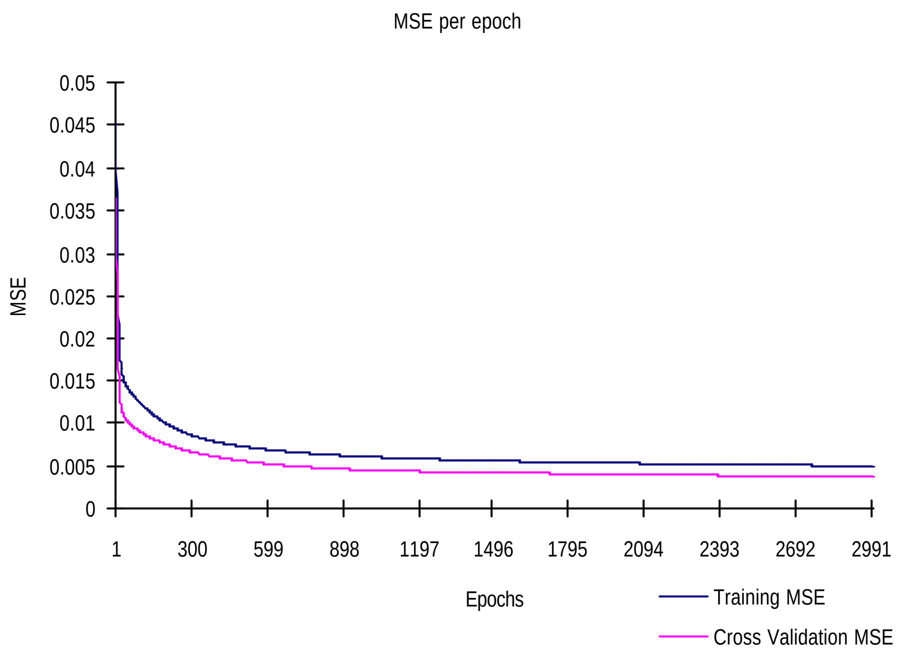
Genetic Algorithm		Neural Network : Multilayer Perceptron					
Type	Steady-State	Hidden layers	Layers	Learning rate	Momentum	Transfer function	Mean square error
Generations	40	2					
Population	100		1st	0.912	0.01	Linear sigmoid	
Selection	Roulette wheel		2nd	0.678	0.012	Linear	
Crossover	0,9		Output	0.781	0.034	Linear	0,004317
Mutation	0,1						

Fitness function per generation, which is the minimum mean square error of each neural network per population, is depicted in graph 3. Fitness function is minimized in generation 40 (0.004317)



Graph 3 Minimum Mean Square Error per generation.

The training mean square error per epoch, as well as the cross validation mean square error per epoch for the best neural network are depicted in graph 4



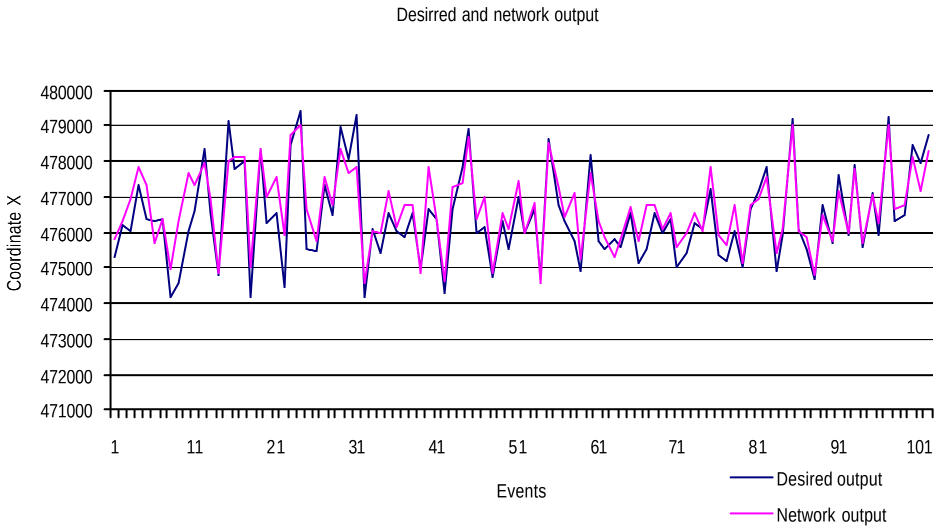
Graph 4 Mean square error

Neural network's statistical results are presented in table 4. The normalized mean square error expresses the mean square error divided by the variance of the desired output. Values close or smaller to 0.2 are regarded satisfactory.

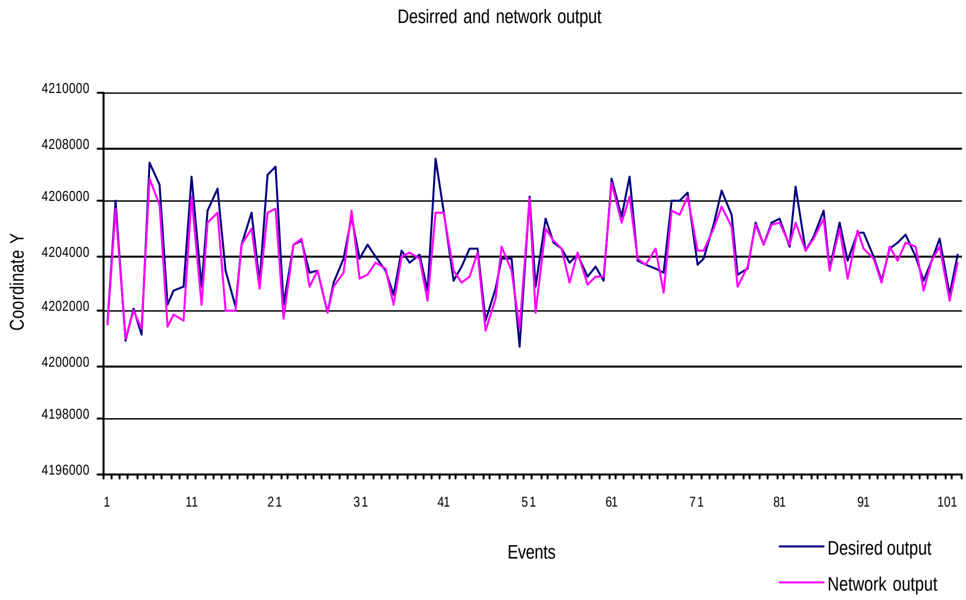
Table 4 Best neural network performance

Best Neural Network	Output 1	Output 2
	X	Y
NMSE	0.10128429	0.098288033
MSE	353.5318543	301.5770232
r	0.962526944	0.970335912

Mean square error expresses the mean difference among real distances between points and predicted. The value of 353 meter per X axe and 301 per Y axe are satisfactory for the specific case study. Finally correlation coefficient, as an indicator for a trained network is close to 1. The final results per variable X and Y are depicted in graphs 5,6. Real and predicted events are depicted in map 1.

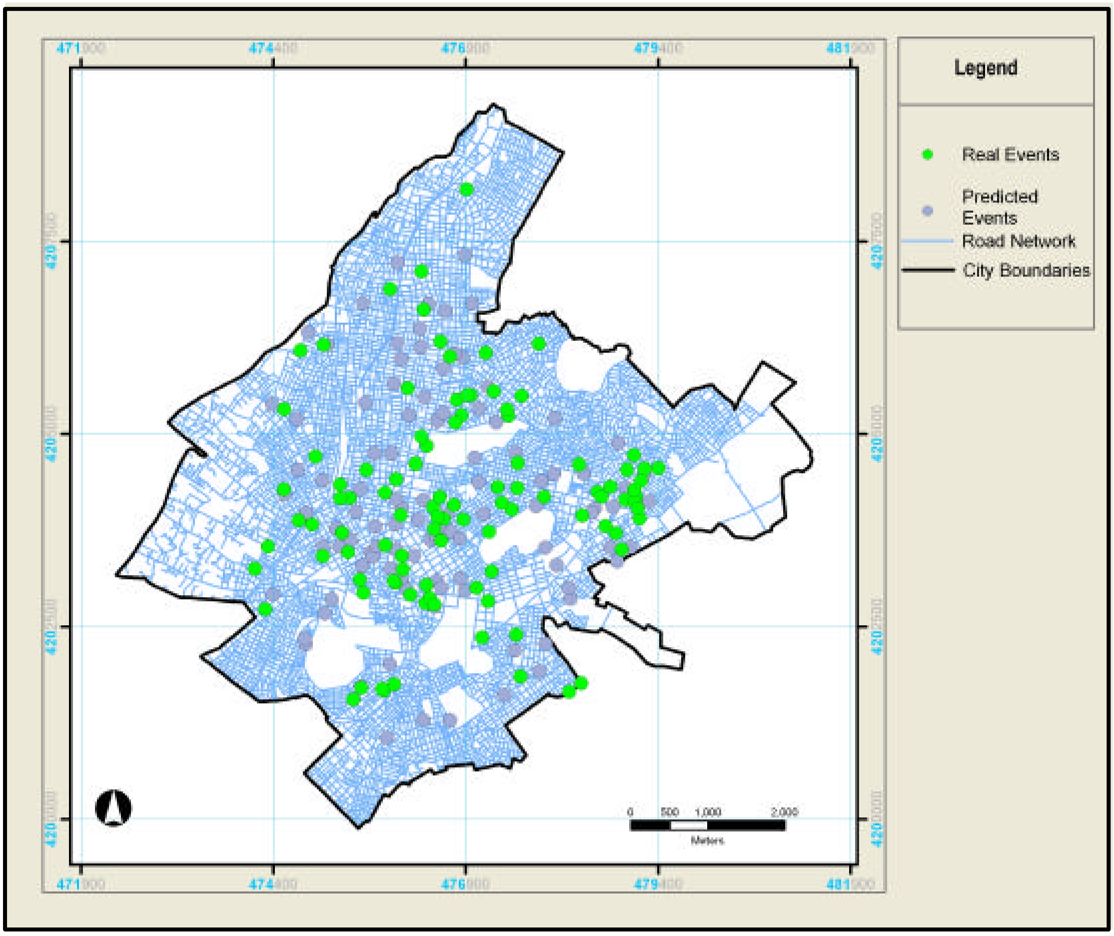


Graph 5 Desired and network output for variable X



Graph 6. Desired and network output for variable y

Map 1. Real and predicted events.



7 CONCLUSIONS

The problem of demand spatio-temporal analysis in locational planning problems is examined. The proposed approach is differentiated from existing ones in that it does not analyze spatial pattern of demand as one entity examining its evolution over time, but specializes in the analysis of events as separate entities. Demand spatial pattern is henceforth checked partially and changes can be examined over time in any part of space. By this way it is likely to localize sub areas in the region of study that present different behavior from the rest. The localization of these regions offers a complete comprehension for the evolution of spatial pattern.

Moreover the analysis is based on combinational correspondence of successive spatial patterns, by the utilization of a new genetic algorithm. The algorithm performs point matching by the correspondence of points among successive time intervals producing geographical time-series. Events of each time interval are corresponded with their spatio-temporal nearest based on the one to one restriction.

The algorithm is applied in case study of Athens. At least 65% of events, in all time intervals, are corresponded with first or second class nearest neighbour and 100% of events achieved distance correspondence better than the average. Neural network, optimized by genetic algorithm predicts future state of demand pattern with high accuracy.

Further research on spatial analysis of demand patterns could be made in several directions. Firstly, the genetic algorithm initialization should be optimized in order to achieve better final results. Furthermore point matching should be extended to cover the matching of different number of events occurrences in sequential time intervals. Finally spatial outliers should be detected in order to achieve better spatial matching.

Acknowledgments

This paper is part of PhD research, which is financially supported by scholarship ‘Herakleitos: Scholars in Basic Research’. The Project is co-funded by the European Social Fund (75%) and National Resources (25%).

The genetic algorithm is programmed in visual C++ 7 using the library “Galib”. This library is developed by Matthew Wall at the Massachusetts Institute of Technology (MIT) and is available at no cost for non-profit purposes.

REFERENCES

Alp O., Erkut E., Drenzer Z.(2003) An efficient genetic algorithm for the p-median problem, **Annals of Operational Research 122, 21-42**, Kluwer Academic Publishers

Berman O.,and Parkan C. (1981) A facility location problem with distance dependent demand **Decision Sciences**, 12: 653-632.

Beckmann M., (1999), **Lectures on location theory**, Springer – Verlag.

Daskin M., (1995), **Network and Discrete Location**, J.Wiley & Suns, N.Y

Fischer M., and Leung Y., (1998) A genetic-algorithms based evolutionary computational neural network for modeling spatial interaction data. **Annals of Regional Science** 32:437–458

Goldberg D.,E. (1989) **Genetic Algorithms in Search, Optimization, and Machine Learning**, Addison-Wesley Pub Co

Labbe M., and Hakimi S.L.,(1990) Market and locational equilibrium for two competitors, **Operational Research** 39: 749-756.

Laporate G., Louneaux F.V.,and Van Hamme L.(1994) Exact solution of a stochastic location by an integer L-shaped algorithm. **Transportation Science**, 28:95-103.

Lederer P.J., and Thisse J.F., (1990) Competitive location on network under delivered pricing, **Operational Research**. Lett., 9:147-153.

Lu Y., Thill J-C. (2003) Assessing the cluster correspondence between paired point locations, Geographical **Analysis, Vol35, No4**. The Ohio State University.

Perl J., and Ho P.K., (1990) Public facilities location under elastic demand, **Transportation Science,24:**117-136.

Openshaw S., and C.Openshaw (1997), **Artificial Intelligence in Geography**. John Wiley & Sons Ltd. England.

Owen S.H, Daskin M.S. (1998), Strategic facility location: A review, **European Journal of Operational Research 111**. pp 423-447.

O'Sullivan, D., and D. Unwin (2003) **Geographic Information Analysis**. Hoboken, NJ: John Wiley & Sons.

Wall M., (1996) GAlib: **A C++ Library of Genetic Algorithm Components version 2.4** Documentation Revision B. Massachusetts Institute of Technology

Wanger J.L, and Falkson L.M., (1975) The optimal nodal location of public facilities with price-sensitive demand, **Geographical Analysis, 7:** 69-83.