Volume 2, Issue 8, August 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Survey On Nearest Node Selection Using Differential Evolution Algorithm

Ravi Kumar Patel
CSE,CSVTU University
India

Mr. Rohit Raja
CSE,CSVTU University
India

Abstract— **The performance, Search efficiency and functionality, and scalability of the application is highly effected by the peer to peer topology. This paper is consisted of proposing a Differential Evolution (DE) approach that can be applied to Neighbor Selection (NS) in P2P Networks. Differential Evolution (DE) algorithm is a new heuristic approach mainly having three advantages; Finding the true global minimum regardless of the initial parameter values ,fast convergence ,and using few control parameters .DE algorithm is a population based . Upper half of the peer-connection matrix is encoded by each candidate. It is done through the undirected graph to reduce the search space dimension. The result outcome is that for large scale problems, DE usually required shorter time to obtain better results than the Genetic Algorithm (GA).**

Keywords— **Optimization, Genetic Algorithm, Undirected graph, Artificial Intelligence, Differential Evolution, Test Functions, Neighbor Selection**

I. INTRODUCTION

There are many number of P2P systems which are used in corporate networks for public prosperity. P2P is composed of peers and the connection between those peers. Those connection may have different weights and maybe directed or undirected. Many properties of peer-to-peer topology based architecture are affected by the connection of these nodes. This forces to change the performance, functionality, search efficiency and scalability of systems a major problem called by the dynamic participation at peer host lies in the current peer-to-peer systems. Dynamic membership of peer host causes difficulty in the P2P systems these days. This results in a constant reorganization of the topology [15].

Kurmanowytsch et al. propose a new peer-to-peer system for reducing the information and content through the observation of peer-to-peer topology and its applications these systems provide us a very good service such as direct communication between neighbour nodes and distributed search . These systems are good for certain task like alter the files but it was predefined topology. Koulouries et al. after some time introduced a new topology for maintaining the peer-to-peer networks This topology provides for self organization and good control over dynamic computing, which is being necessary for services like routing, access, and topology forming and resource management.

In a P2P system, all participating peers form a P2P network on top of an underlying physical network. A P2P network is an abstract, logical network called an overlay network. Based on existing research [11] we formulate the neighbor selection problem for P2P overlay networks in this Section. As given by Liu,et al. a P2P network can be mo deled based on the following assumptions:

- •An overlay connection between a pair of peering nodes consists of a number of physical links which form a shortest path between the pair of end no des in the physical topology, and Internet paths are relatively stable.
- •The same size packets traversing the same physical link in a short period of time will have similar delay, as assumed by many other measurement applications

DE optimizes a problem by maintaining a population of individuals solutions called candidates and moving these candidates around in the search-space according to simple formulae. The movements of the candidates are guided by the best found positions in the search-space, which are continually updated as better positions are found by the candidates. DE optimizes a problem by maintaining a population of candidate solutions called candidates and moving these candidates around in the search-space according to simple formulae. The movements of the candidates are guided by the best found positions in the search-space, which are continually updated as better positions are found by the candidates.

II. PROBLEM FORMULATION

Kooa et al. presents a new topology for selecting a node and trying to find the connection between those nodes through used of undirected graph [6]. Given a fixed number of *N* peers, we use a graph $G = (V;E)$ to denote an overlay network, where the set of vertices $V = \{v_{1,\ldots\ldots\ldots\ldots,}v_N\}$ represents the *N* peers and the set of edges $E = \{e_{ij} \epsilon \{0,1\}, i\ ,\ j, = 1, \ldots\ldots, N\}$ represents their connectivity : $e_{ij} = 1$ if peers *i* and *j* are connected, and $e_{ij} = 0$ otherwise. For an undirected graph, it is required that $e_{ij} = e_{ij}$ for all $I \neq j$, and $eij = 0$ when $i = j$. Let *C* be the entire collection of content pieces, and we denote *{ci*

$\subseteq$ *C, I=1,……, N} to* be the collection of the content pieces each peer *i* has. And further assume that each peer i will be connected to a maximum of d_i neighbors, where $d_i< N$. The disjointness of contents from peer *i* to peer *j* is denoted by $c_i \backslash c_j$,which can be calculated as:

$$c_i \backslash c_j = ci - (ci \cap c_j) \qquad (1)$$

Where\ denotes the intersection operation on sets. This disjointness can be interpreted as the collection of content pieces that peer *i* has but peer *j* does not. In other words, it denotes the pieces that peer *i* can upload to peer *j*. Moreover, the disjointness operation is not commutative, i.e., $c_i \backslash c_j \neq c_j \backslash c_i$, we also denote mod $c_i \backslash c_j$ to be the cardinality of $c_i \backslash c_j$, which is the number of content pieces peer *i* can contribute to peer *j*. In order to maximize the disjointness of content, we want to maximize the number of content pieces each peer can contribute to its neighbors by determining the connections e_{ij}'s. Define e_{ij}*'s* to be sets such that e_{ij} = *C* if e_{ij} = 1, and e_{ij} =Ø (null set) otherwise. Therefore we have the following optimization problem:

$$\max_E \sum_{i=1}^{N} \left| \bigcup_{n=1}^{m} (ci \cap\ cj) \cap eij \right| \qquad (2)$$

Subject to

$$\sum_{j=1}^{N} e_{ij} \le d_i \ for\ all\ i$$

III. Differential Evolution Implementation

For applying Differential Evolution algorithm one of the key issue is mapping of the problem solution into the candidate space , for doing these its need to state a problem S=(N,C,M,f) where the N is used for the total number of peers, and C shows the entire collection of contain piece, M is represented the maximum number of peers which each peer can contain steadily the session, *f* is to goal the number of swap pieces, i.e. to maximize Equation (2), which directly affects its feasibility and performance. Usually, the candidate's position is encoded to map each dimension to one directed connection between peers, i.e. the dimension is *N* * *N*. But the neighbor topology in P2P networks is an undirected graph, i.e. e_{ij} = e_{ij} for all $i \neq j$. We set up a search space of *D* dimension as $N *(N -1) =2$. Accordingly, each candidate's position is represented as a binary bit string of length *D*. Each dimension of the candidate's position maps one undirected connection. The domain for each dimension is limited to 0 or 1.

The differential Evolution method is perhaps the fastest evolutionary computational procedure yielding most accurate solutions to continuous global optimization problems. It consists of three basic steps[17]:

- Generation of (large enough) population with individuals in the m-dimensional space, randomly distributed over the entire domain of the function in question and evaluation of the individuals of the so generated population by finding f(x), where x is the decision variable.
- Replacement of this current population by a better fit new population.
- Repetition of this replacement until satisfactory results are obtained or the given criteria of termination are met.

An optimization task consisting of D parameters can be represented by a D-dimensional vector. In Differential Evolution, a population of NP solution vectors is randomly created at the start. This population is successfully improved by applying mutation, crossover and selection operators.

The differential evolution consists of a population of candidate, which are initialized with a population of random candidate solutions. They move iteratively through the *D*-dimension problem space to search the new solutions, where the fitness *f* can be measured by calculating the number of condition attributes in the potential reduction solution. Each candidate has a position represented by a position vector $\sim p_i$ (*i* is the index of the candidate), and a velocity represented by a velocity-vector $\sim v_i$. Each candidate remembers its own best position so far in a vector *~p# i* , and its *j*-th dimensional value is *p# ij* . The best position- vector among the swarm so far is then stored in a vector *~p**, and its *j*-th dimensional value is *p*j* . When the candidate moves in a state space restricted to zero and one on each dimension, the change of probability with time steps is defined as follows:

$$P(p_{ij}(t) = 1) = f(p_{ij}(t\text{-}1);v_{ij}(t\text{-}1);p^{\#}_{ij}(t\text{-}1);\ p^{*}_{j}\ (t\text{-}1)) \qquad (3)$$

where the probability function is

$$sig\left(v_{ij}(t)\right) = \frac{1}{1+e^{v_{ij}(t)}} \qquad (4)$$

At each time step, each candidate updates its velocity and moves to a new position according to Eqs.(5) and (6):

$$v_{ij}(t) = wv_{ij}(t-1) + c_1 r_1 \left(p_{ij}^{\#}\ (t-1) - p_{ij}(t-1)\right) + c_2\ r_2\ \left(p_j^{*}(t-1) - p_{ij}(t-1)\right) \qquad (5)$$

$$p_{ij}(t) = \begin{cases} 1, & if\ \rho < sig(v_{ij}(t)) \\ 0, & otherwise \end{cases} \qquad (6)$$

Where $c1$ is a positive constant, called as coefficient of the self-recognition component, $c2$ is a positive constant, called as coefficient of the social component. $r1$ and $r2$ are the random numbers in the interval [0,1]. The variable *w* is called as the inertia factor, which value is typically setup to vary linearly from 1 to near 0 during the iterated processing. ½ is random number in the closed interval [0, 1]. From Eq.(5), a candidate decides where to move next, considering its current state, its own experience, which is the memory of its best past position, and the experience of its most successful candidate in the population. The candidate has a priority levels according to the order of peers. The sequence of the peers will be not changed during the iteration. Each candidate's position indicates the potential connection state.

DE [2] is a population-based global optimization algorithm that uses a floating-point (real-coded) representation. The *i*-th individual of the population at generation (time) *G* is a *D*-dimensional vector containing a set of *D* optimization parameters:

$$\vec{Z}_{i,G} = [\,Z_{i,1,G} .\ Z_{i,2,G}, \ldots\ldots Z_{i,D,G}]$$

Now, in each generation, a donor $G\ Y_{i,j}$, is created. The method of creating this donor vector demarcates between the various DE schemes. In one of the earliest variants of DE, now called DE/rand/1 scheme, three other parameter vectors (say the *r1*, *r2*, and *r3*-th vectors such that r1, r2 r3 $\in$[1, NP] and r1$\neq$ r2$\neq$ r3 are chosen at random from the current population. Next the difference of any two of the three vectors is multiplied by a scalar number *F* and the scaled difference is added to the third one, whence we obtain the donor vector $Y_{i,G}$. The process for the *j*-th component of the *i*-th vector may be expressed as:

$$Y_{i,j,G} = Z_{r1,j,G} + F.(\,Z_{r2,j,G} - Z_{3,j,G})$$

Next a crossover operation takes place to increase the potential diversity of the population. We use 'binomial 'crossover in which case the number of parameters inherited from the mutant has a (nearly) binomial distribution. Thus for each target vector $Z_{i,G}$ a trial vector $R_{i,G}$ is created in the following fashion:

$$R_{i,j,G} = Y_{i,j,G}\ ,\text{if(} rand_{i,j}(0,1) \leq Cr \text{ or } j=j_{rand})$$

$$Z_{i,j,G}\ ,\text{otherwise}$$

for j = 1, 2, ….., D and *rand* j (0, 1) ∈ [0,1] is the j-th evaluation of a uniform random number generator j [1,2,...., D] *rand* Î is a randomly chosen index which ensures that $R\ i\ (t)$ gets at least one component from $Z\ i(t)$.

Next step of the algorithm calls for selection in order to determine which one between the target vector and trial vector will survive in the next generation i.e. at time $t = t+1$. If the trial vector yields a better value of the fitness function, it replaces its target vector in the next generation; otherwise the parent is retained in the population:

$$\left.\begin{aligned} \vec{Z}_{i(t+1)} &= \vec{R}_{i(t)} && \text{if } f(\vec{R}_{i(t)}) \leq f(\vec{Z}_{i(t)}) \\ &= \vec{Z}_{i(t)}) && \text{if } f(\vec{R}_{i(t)}) \geq f(\vec{Z}_{i(t)}) \end{aligned}\right\}$$

where f(.) is the function to be minimized.

IV. PARAMETER FOR ALGORITHM

Population Numbers(*NP*)	The size of population decides the total number of the solution vectors in the same iteration
Mutation Factor(*F*)	This factor decides how many perturbation ratios the solution can acquire. If the value is greater, the magnitude of jump will be increase
Crossover Rate(*CR*)	Crossover rate decides the swap probability between trial vector and target vector

V. CONCLUSIONS

In this paper, the problem of neighbor selection in peer-to-peer networks is studied in the presented paper. It made use of the differential evolution approach during solving the problem. In the mentioned approach the upper half of the peer connection is encoded. That is done through undirected graph. It substantially reduces the dimensions of search graph.

REFERENCES

[1]. Ajith Abraham ·Hongbo Liu, Aboul Ella Hassanien "Multi swarms for neighbor selection in peer-to-peer overlay Networks" © Springer Science+Business Media, LLC 2010.

[2]. Swagatam Das, Ajith Abraham, Senior Member, IEEE, and Amit Konar, Member, IEEE "Automatic Clustering Using an Improved Differential Evolution Algorithm" IEEE Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, Vol. 38, No. 1, January 2008

[3]. A Particle Swarm Optimization Algorithm for Neighbor Selection in Peer-to-Peer Networks Shichang Sun, Ajith Abraham, Guiyong Zhan3, Hongbo Liu 6th International Conference on Computer Information Systems and Industrial Management Applications (CISIM'07) 0-7695-2894-5/07 $20.00 © 2007 IEEE

[4]. Peer-to-Peer Neighbor Selection Using Single and Multi-objective Population-Based Meta-heuristics , Hongbo Liu1,2, Ajith Abraham3, and Fatos Xhafa Envi., SCI 146, pp. 323–340, 2008. Springerlink.com c Springer-Verlag Berlin Heidelberg 2008

[5].M A Simple and Global Optimization Algorithm forengineering Problems: Dierential evolutionalgorithm Dervis KARABOGA and Selcuk ¨OKDEM Turk J Elec Engin, VOL.12, NO.1 2004, c TUBITAK

[6]. A Particle Swarm Optimization and differentialevolution Algorithms: Technical Analysis,Applications and Hybridization Perspectives Swagatam Das, Ajith Abraham, and Amit Konar Springer-Verlag Berlin Heidelberg 2008

[7]. Differential Evolution – A Simple and efficientheuristic for Global Optimization over Continuous Spaces .RAINER STORN And KENNE Journal of Global Optimization 11: 341–359, 1997 TH PRICE 1997 Kluwer Academic Publishers. Printed in the Netherlands.

[8]. Employing qos Driven Neighbor Selection for Heterogeneous Peer-to-Peer Streaming , Laizhong Cui, Yong Jiang and Jianping Wu IEEE ICC 2011 proceedings

[9]. Discrete Differential Evolution with Local Search to Solve the Traveling Salesman Problem João , Guilherme Sauer and Leandro dos Santos Coelho April 7, 2008 IEEE

[10]. Lua E K, Crowcroft J, Pias M, Sharma R and Lim S (2005) A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. IEEE Communications Surveys& Tutorials, 7(2):72–93

[11]. Belmonte M V, Conejo R, D´iaz M and P´erez-de-la-Cruz J L (2006) Coalition Formation in P2P File Sharing Systems. Lecture Notes in Artificial Intelligence, CAEPIA'05

[12]. Cho H (2007) An Update Propagation Algorithm for P2P File Sharing over Wireless Mobile Networks, Lecture Notes in Computer Science, ICCS'07, 4490:753–760

[13]. Pianese F, Perino D, Keller J and Biersack E W (2007) PULSE: An Adaptive, Incentive-Based, Unstructured P2P Live Streaming System. IEEE Transactions on Multimedia, 9(8):1645–1660

[14]. Sen S and Wang J (2004) Analyzing Peer-to-Peer Traffic Across Large Networks. IEEE/ACM Transactions on Networking, 12(2):219–232 44

[15]. Androutsellis-theotokis S and Spinellis D (2004) A Survey of Peer-to-Peer Content Distribution Technologies. ACM Computing Surveys, 36(4):335–371

[16]. Meo M and Milan F (2008) qos Content Management for P2P File sharing Applications. Future Generation Computer Systems, 24:213–221

[17].Completing Correlation Matrices of Arbitrary Order by Differential Evolution Method of Global Optimization: A Fortran Program SK Mishra

[18]. Improving the Performance of Differential Evolution Algorithm with, Modified Mutation Factor , Ching-Wei Chien , Zhan-Rong Hsu , Wei-Ping Lee 2009 International Conference on Machine Learning and computingipcsit vol.3 (2011) © (2011) IACSIT Press, Singapore