ISSN: 2321-7782 (Online)
Volume 2, Issue 4, April 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Paper / Case Study
Available online at: www.ijarcsms.com

An Improved system for efficient clustering using ant colony algorithm

Minal Chaudhari[1]
Department of Information Technology
Maharashtra Institute of Technology
Pune – 411038
India

Vidya Deshpande[2]
Department of Information Technology
Maharashtra Institute of Technology
Pune – 411038
India

Abstract: ***A simulated network models a typical distribution of calls between nodes; those nodes which carry extra traffic can become congested, and that will result in loosing calls. In addition to calls, the network also supports a population of simple peer node with behaviours modelled on the trail laying abilities of Dynamic ants. The Dynamic ants can move in the network randomly between chosen pairs of nodes; as they move they deposit simulated pheromones as a function of their distance from their source node, and the congestion encountered on their journey. They select their path on every node depending on the distribution of simulated pheromones at every node. Calls between nodes are routed as a function of the pheromone distributions at each intermediate node. The performance of the network is measured by the average no of hops taken to complete the calls. In this paper, the results of using the Dynamic ant- based control (D (DABC)) are compared with those achieved by using k-means algorithm used in network management. The (DABC) system is shown to result in fewer call failures than the other methods, while exhibiting many attractive features of distributed control.***

Keywords: Ant colony clustering, dynamic ant based control system.

I. INTRODUCTION

Clustering in a network means to spread work between two or more computers, network links, CPUs, hard drives, or other resources, so that we should get optimal resource utilization, maximize throughput, and minimize response time.

It can be useful when dealing with redound Dynamic ant communications links. For example, a company may have multiple Internet connections ensuring network access even if one of the connections should fail. A failover management would mean that one link is designated for normal use, and second link is used only if the first one fails. With load balancing for FTP Application, so that both the links should get utilized fully for long time.

A device or program decides which of the available links to send packets along, it sends the packets to the link which is not failed previously & because of using multiple links it simultaneously increases the available bandwidth in the network. Major telecommunications companies have multiple routes through their networks or to external networks. They use more sophisticated load balancing for FTP Application to shift traffic from one path to another to avoid network congestion and eventually it minimizes the cost of transit across external networks to improve network reliability.

The notion of complex collective behaviour emerging from the behaviour of many relatively simple units, and the interactions between them, is fundamental to the field of artificial life. The growing understanding of such systems offers the prospect of creating artificial systems which are controlled by such emergent collective behaviour; in particular, we believe that the exploitation of this concept might lead to completely new approaches for the management of distributed systems, such as load balancing for FTP Application in telecommunications networks. In such networks, Calls between two points are typically routed through a number of intermediate switching stations, or nodes; in a large network, there are many possible routes for

each such call. It is thus possible to relieve actual or potential local congestion by routing calls via parts of the network which have spare capacity.

Load balancing for FTP Application is essentially the construction of call -routing schemes which successfully distribute the changing load over the system and minimise lost calls. Of course it is possible to determine the shortest routes from every node to every other node of the network. In this way the average utilisation of nodes will be minimised, but this is not necessarily the ideal way to avoid node congestion, as this has to do with how the traffic on the network is distributed. Controlling distributed systems like these by means of a single central controller has several disadvantages. The controller usually needs current knowledge about the entire system, necessitating communication links from every part of the system to the controller.

These central control mechanisms scale badly, due to the rapid increase of processing and communication overheads with system size. Failure of the controller will often lead to failure of the complete system. There is the additional practical commercial requirement that centrally controlled systems may need to be owned by one single authority. Further, the nature of distributed systems like these is highly dynamic, complex and stochastic, and their behaviour can neither be predicted nor explained by reducing it to a single central controllable factor.[1]

II. Related Work

The Partitional Clustering method seeks to partition a collection of documents into a set of non-overlapping groups, so as to maximize the evaluation value of clustering. The general criterion for a good partition is that objects of different cluster are “far apart” and objects within the same cluster are “close” to each other.

One of the issues with Partition algorithms is their high complexity, as some of them exhaustively enumerate all possible groupings and try to find the global optimum. Even for a small number of objects, the number of partitions is huge. That’s why, common solutions start with an initial, usually random, partition and proceed with its refinement. A better practice would be to run the partition algorithm for different sets of initial _ points (considered as representatives) and investigate whether all solutions lead to the same final partition. Partitional Clustering algorithms try to locally improve a certain criterion. First, they compute the values of the similarity or distance, they order the results, and pick the one that optimizes the criterion. Hence, the majority of them could be considered as greedy-like algorithms.[2]

The best-known partitioning clustering algorithm is the K-means algorithm and its variants. This algorithm is simple, straightforward and is based on the firm foundation of analysis of variances. The K-means algorithm clusters a group of data vectors into a predefined number of clusters. It starts with a random initial cluster centre and keeps reassigning the data objects in the dataset to the cluster centres based on the similarity between the data object and the cluster centre. The reassignment procedure will not stop until a convergence criterion is met (e.g., the fixed iteration number or the cluster result does not change after a certain number of iterations).[3]

III. Problem Definition

In recent years high bandwidth broadband connections have become available to everyone, the demands for online things are constantly increasing. Data-on-demand systems like BitTorrent and YouTube are highly successful, where the user has the possibility to In contrast to data on demand; live video streams are only available at one particular time. The most important task here is to create a continuous data stream. The momentary used systems are mainly constructed as a server to client solution, where one single source distributes the content to its receivers.

This approach scales up to certain point; until the server’s full capacity is reached. A Unicast system, where each client has a direct connection to the server, has the advantage that it is easy to implement, and live and on-demand streaming is not a problem. The disadvantage the server has to sustain the entire data distribution work load. This means the server needs a high bandwidth

connection and the operator has to carry all the costs of maintaining the functionality of the system. Peer-to-Peer networks now offer the possibility to distribute data content to an unlimited number of users and reducing the bandwidth bottleneck on the source.[4]

IV. Proposed Architecture and Working

Compared to the traditional algorithms, the swarm algorithms usually are flexible, robust, decentralized, and self-organized. These characters make the swarm algorithms potentially suitable for solving dynamic problems, such as document collection clustering. In the following session, we will introduce three nature inspired approaches for document clustering analysis. These clustering algorithms use stochastic and heuristic principles discovered from observing bird flocks, fish schools and ant food forage. Unlike other partition clustering algorithm such as K-means, these nature inspired algorithms either generate more accurate clustering results or does not require initial Partitional seeds. [5]

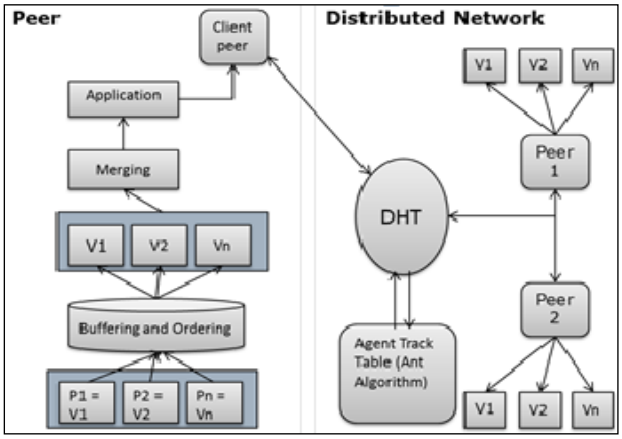

Fig.1 Proposed Architecture

Architecture and working of proposed system is as follows:

- Proposed system basically focused on implementing a real time application for downloading data, The purpose of this proposed work is to lay down the system architecture and detailed design for the work entitled "File sharing Using Ant clustering Algorithm". It contains the overall description of the work.
- The peers wishing to request a video, forms a network by getting registered with Distributed Hash Table (DHT). Information like IP address, port number, name of the file and size of file content is provided to DHT which stores that information in database. DHT gets the bandwidth of peer with the help of Network Hash Table. When a new user/peer requests a file, it contacts the DHT.
- The DHT in turn searches the database for requesting video. If requested video is available in single peer, then DHT sends that peer information to the requesting peer. If multiple peers have that same requested video, then the DHT sorts its peer's information based on bandwidth.
- Then, the highest bandwidth peer information is transferred to the requester. Requesters establish a connection to provider and get video through streaming. In some cases, some of the requested video parts are available in one provider peer eg: v1, v2, v3 and remaining parts eg: v4, v5 available in some other provider peer.
- Then the requester needs to establish a thread to both peer and obtain the video segments in parallel. Video provider information is updated in trace table of peers in the path through which the video is streamed.
- When a new peer request for video that is already requested, the peer uses the updated information in trace table and gets the video. The provider splits the video into several parts and assigns a sequence number to each part. The requester while receiving the video chunks; checks sequence number, merge video parts and play it in the media player. If a

particular video part is missed, requester waits for it, even if other parts are received. After got the video parts, it merges at the receiver.

The proposed system uses ant clustering algorithm in track table which is given in following table1. The ant clustering algorithm is one type of bottom-up approach for solving the data clustering and classification problems. Ant Colony Optimization is not just a data clustering application of the Ant Colony Optimization Algorithm, it is another bio inspired algorithm derived from several ant species' natural behaviours. This algorithm and ant colony optimization algorithm are commonly be considered belong to the same ant swarm algorithm category.

Table 1

Randomly deploy every data object i on an un-occupied cell of the grid

Randomly place every artificial ant k on an un-occupied cell of the grid

LOOP

FOR every ant k

Move to a randomly selected neighbour un-occupied cell

If object i in the ant's neighbourhood cell

If ant k doesn't carry any data object

Compute f(oi) and Pp(i)

Pickup item i with probability Pp(i)

If ant k carrying data object j AND there are at least one un-occupied cell in the neighbourhood

Compute f(oj) and Pd(j)

Drop object j at the nearest un0occupied cell with probability Pd(j)

END

The ant clustering algorithm inspired from entomologists' observation that some species of ants have the capability to sort large corpses into clusters.

Deneubourg and his colleagues (Deneubourg et al., 1991) proposed a "Basic Model" to explain the ants' behavior of piling corpses and eggs. In their study, a population of ant-like agents randomly moved in a 2 dimensional grid. Each agent only follows one simple rule: moving randomly in the grid and establishing a probability of picking up the data object it meets if it is free of load, or establishing a probability of dropping down the data object if it is loading the data object. After several iterations, a clustering result emerges from the collective activities of these agents. Lumer, Faieta and other researchers (Lumer & Faieta, 1994) extended this "Basic Model" and introduced the standard Ant Clustering Algorithm for multi-dimensional data clustering analysis. [6]

In this standard ant clustering algorithm, the input data set composed of N d-dimensional vector objects are spread randomly over a bi-dimensional grid of size m*m where m depends on number of items and each grid cell can only be occupied by a single

data object at any time. Each cell in the grid contents at most one item. A colony of artificial ants is allowed to randomly move on the grid for picking up, carrying, and dropping off data objects based on some probabilistic rules. The ants can perceive the neighbourhood in the environment. At this initial stage, each ant does not "carry" any item. [7]

V. Conclusion and Future Scope

Intelligence analysts are currently overwhelmed with the tremendous amount of information streams generated every day. There is a lack of comprehensive tools that can real-time analyse the information streams. Clustering analysis plays an important role in improving the accuracy of information. However, most clustering technologies can only be applied for analysing the static document collection because they normally require a large amount of computation resource and a long time to get accurate results. It is very difficult to cluster dynamic changed information streams on an individual computer.

The ant clustering algorithms report can continually refine the clustering result and quickly react to the change of individual data. The ant clustering algorithms are suitable for clustering dynamic changed information. As it is dynamic we can download the data more efficiently and ultimately increases speed of downloading.

In dynamic ant colony clustering algorithm ant can carry only one item at a time. In many cases multiple objects need to be moved to purify ant's local neighbourhood, but the ant can only move one item at a time and has to repeat search steps. So if one ant can carry multiple objects at a time we can increase the clustering speed.

References

1. B. Barán, M Almirón, E. Chaparro. 'Dynamic ant Distributed System for Solving the Traveling Salesman Problem'. pp. 779-789. Vol. 215 th Informatic Latinoamerican Conference-CLEI,. Paraguay (2010).
2. S. Bak, J. Cobb, E. Leiss. 'Load Balancing for FTP Application Routing via Randomization'. pp. 999-1010. Vol. 215 th Informatic Latino american Conference-CLEI. Paraguay (1999).
3. M. Dorigo, V. Maniezzo, A. Colorni. 'The Dynamic ant System: Optimization by a colony of cooperating agents'. pp. 1-13. Vol. 26-Part B. IEEE Transactions on Systems, Man, and Cybernetics,. Number 1 (1996).
4. Dorigo, G. Di Caro. 'Dynamic antNet. Distributed Stigmergetic Control for Communications Networks'. pp. 317-365 Journal of Artificial Intelligence Research. Number 9 (1998).
5. Schoonderwoerd et al 97, R. Schoonderwoerd, O. Holland, J. Bruten. 'Dynamic ant-like agents for load balancing for FTP Application in telecommunications networks', Hewlett-Packard Laboratories, Bristol-England (1997).
6. Beckers, R., Holland, O.E., & Deneubourg, J.L. (1994). From local actions to global tasks: Stigmergy and Collective Robotics. In R.A. Brooks, & P. Maes (Eds.) Artificial Life IV, Cambridge, MIT Press. p. 181-189.
7. Goss. S., Beckers, R., Deneubourg, J.L., Aron, S., & Pasteels, J.M. (1990). How trail-laying and trail following can solve foraging problems for Dynamic ant References colonies. In R.N. Hughes (Ed.) NATO AS1 Series.

Author(s) Profile

Minal Chaudhari is currently pursuing her Master of Engineering from Maharashtra Institute of Technology and she has completed her Bachelor of Engineering from Pune University.

Vidya Deshpande received her M.S. degree in System and Information from BIT's Pilan. Her teaching area includes Operating System, System Software, Design and Analysis of Algorithms, Compiler Construction, Computer Graphics, Data Structures and Files, Discrete Structures, Software Testing and Quality Assurance. Her research area includes Algorithms and System Software.

