A complex network approach to cloud computing

Gonzalo Travieso, Carlos Antônio Ruggiero, Odemir Martinez Bruno and Luciano da Fontoura Costa

Instituto de Física de São Carlos, Universidade de São Paulo, 13560-970 São Carlos SP, Brazil
E-mail: gonzalo@ifsc.usp.br

Received 21 September 2015, revised 15 December 2015
Accepted for publication 6 January 2016
Published 24 February 2016

Online at stacks.iop.org/JSTAT/2016/023402
doi:10.1088/1742-5468/2016/02/023402

Abstract. Cloud computing has become an important means to speed up computing. One problem influencing heavily the performance of such systems is the choice of nodes as servers responsible for executing the clients’ tasks. In this article we report how complex networks can be used to model such a problem. More specifically, we investigate the performance of the processing respectively to cloud systems underlaid by Erdős–Rényi (ER) and Barabási-Albert (BA) topology containing two servers. Cloud networks involving two communities not necessarily of the same size are also considered in our analysis. The performance of each configuration is quantified in terms of the cost of communication between the client and the nearest server, and the balance of the distribution of tasks between the two servers. Regarding the latter, the ER topology provides better performance than the BA for smaller average degrees and opposite behaviour for larger average degrees. With respect to cost, smaller values are found in the BA topology irrespective of the average degree. In addition, we also verified that it is easier to find good servers in ER than in BA networks. Surprisingly, balance and cost are not too much affected by the presence of communities. However, for a well-defined community network, we found that it is important to assign each server to a different community so as to achieve better performance.

Keywords: network dynamics, random graphs, networks
1. Introduction

With the Internet boom, an impressive mass of computing resources, encompassing both machine and data, have become widely available. At the same time, the number of users has grown greatly, implying a growing demand for Internet collaborative access on a number of machines and platforms. Cloud computing has emerged as a natural integration of these two trends. The basic idea in this paradigm is to define integrated, distributed servers capable of supplying services to users through the Internet. In addition, since the data in the cloud system have to be widely accessible in many places and for many users, multiple servers are required. As a consequence of its reliance on the Internet, cloud systems tend to have complex topologies, which compounds the choice of where in the network the servers should be placed. In particular, the distribution of the servers should lead to small communication times between users and servers, without overloading any of the servers.

Complex networks have become an important subject in science and technology because of their ability to represent and model a large number of complex systems such as society, protein interaction and transportation, among many others [1–3]. In computer science, complex networks have been used, for instance, in the study of the topology of the Internet [4], the Web [5], email communications [6], the complexity of software systems [7] and modelling grid computing [8–12]. In the latter field, complex networks were used to represent task execution in grid computing environments, with the tasks being supplied by a master, on demand from worker processors, which were distributed along the network topology. On the contrary, in cloud computing several users concur for access to a small number of servers.
In the present work we extend the use of complex networks to modelling and evaluating the performance of multiple-server cloud computing environments. More specifically, we quantify the effects of different topologies, namely Erdős–Rényi [13], Barabási–Albert [14] and a modular model, with respect to the positioning of servers in the network topology. For the sake of simplicity we consider only pairs of servers in cloud environments. In our model, cloud servers are considered as a single node in the network. In fact, the servers are networks of computers interconnected to efficiently reply to requests. This simplification is justified by the much lower communication latencies in the internal network of the servers, as compared to the connections from clients to servers. Other works emphasise the internal network [15, 16] or use cloud computing to study computer networks [17].

This article starts by presenting the basic concepts and methods adopted, and follows by presenting how cloud environments can be represented in complex networks, and investigating the performance of such configurations for different placements of servers in the network topology. We found that the distribution of servers in cloud computing environments is determinant for their performance, quantified in terms of communication costs and balance. In addition, the best configurations depend strongly on the network topology.

2. Methodology

We consider here a network that provides a communication infrastructure for agents placed on its nodes. Some agents (called ‘servers’) will be chosen to provide services for the remaining (the ‘clients’). A request for a service is forwarded by a client to the closest server following the shortest path, and the response from the server follows the same path in the reverse order. Once the servers are placed in the network, each client is assigned to the closest server. Thus, for good efficiency on the delivery and execution of the services, the servers must be placed in the network such that they are relatively close to their clients, and each server is responsible for answering requests from about the same number of clients. Figure 1 shows two contrasting situations regarding the placement of two servers in the same network. In the left-hand part of the figure, a good balance is achieved because each server is associated with a similar number of clients. On the contrary, on the right, one of the servers resulted with only six clients, while a much larger number of clients is associated with the other server.

To quantitatively evaluate the above aspects, given a choice of servers we compute two measurements: the average cost and the balance, defined as follows.

Let \( s(i) \) be the server associated with client \( i \) (i.e. the server that is closest to \( i \)). The average cost is defined as

\[
c = \frac{2}{n - n_s} \sum_i d(i, s(i)),
\]

where \( d(i, j) \) is the shortest path length from node \( i \) to node \( j \) in the network, \( n \) is the number of nodes in the network and \( n_s \) is the number of servers. Factor 2 is included to account for the request and response communication costs. The sum runs over all clients \( i \).
The balance should quantify whether all the servers receive work from approximately the same number of clients. Let \( A_j \) be the set of clients associated with server \( j \), and \( |A_j| \) its cardinality. We define the balance as the ratio from the smaller to the larger of these sets:

\[
b = \frac{\min_j |A_j|}{\max_j |A_j|},
\]

where the \( \min \) and \( \max \) run over all servers \( j \).

We want to evaluate the effect of network topology on this dynamical process. For simplicity and computational efficiency, we consider the case of only two servers. Given a pair of servers, we choose with which server the clients are associated, using the distance matrix of the network and choosing the nearest server for each client. Afterwards, the values of average cost and balance are computed for this pair of servers using the above expressions. The process is repeated for all possible pairs of servers in the network. A good pair should have simultaneously a large value of balance and a small value of average cost. We define the \textit{elite} of server pairs as the intersection of the pairs within the 20\% with the best (smallest) values of average cost and the 20\% with the best (largest) value of balance. For each evaluated network we compute: the smallest values of average cost and the largest value of balance for all the pairs, the threshold values of average cost and the balance needed to include a pair in the elite, the average values of the average cost and the balance for all the pairs, the average values of the average cost and balance for the pairs in the elite, and the number of pairs in the elite.

We now consider the effect of community structure of the network on the balance and average cost. We want to quantify the effects of difference in sizes and separation of communities. The network model used consists of \( N \) nodes, each associated with one of two communities, \( C_1 \) and \( C_2 \), with \( \lfloor \alpha N \rfloor \) nodes in \( C_1 \) and \( N - \lfloor \alpha N \rfloor \) nodes in \( C_2 \), where \( 0 < \alpha < 1 \) and \( \lfloor x \rfloor \) means rounding \( x \) to the closest integer. Without losing generality, in the following we choose \( C_1 \) to be the smallest, and therefore \( 0 < \alpha \leq \frac{1}{2} \). Each pair of nodes is connected according to the following:

**Inside** \( C_1 \) If both nodes are from \( C_1 \), they are connected with probability

\[
p_1 = \frac{1 - \delta \langle k \rangle}{\alpha N},
\]

where \( \alpha \) is the community fraction.
where $\langle k \rangle$ is the desired average degree and the parameter $\delta$ controls the community structure as will be discussed below.

**Inside $C_2$** When both nodes are from $C_2$, the connection probability is

$$p_2 = \frac{1 - \alpha - \alpha \delta \langle k \rangle}{(1 - \alpha)^2} \frac{1}{N}.$$  \hfill (4)

**Between communities** The probability of inter-community connection is given by

$$p_i = \frac{\delta \langle k \rangle}{1 - \alpha} \frac{1}{N}.$$  \hfill (5)

Different values are chosen for the probability in the two communities to achieve the same average degrees for all nodes. If the same value of probability is used for a small and a large community, each node in the smaller one will have fewer connections as compared to the nodes in the large one. For the values in equations (3)–(5), the average degree of nodes in $C_1$ is (for large values of $N$):

$$p_1 \alpha N + p_1 (1 - \alpha) N = (1 - \delta) \langle k \rangle + \delta \langle k \rangle = \langle k \rangle.$$  

For $C_2$, the average degree is:

$$p_2 (1 - \alpha) N + p_2 \alpha N = \frac{1 - \alpha - \alpha \delta}{1 - \alpha} \langle k \rangle + \frac{\alpha \delta}{1 - \alpha} \langle k \rangle = \langle k \rangle.$$  

The value $\delta$ is a community strength parameter and quantifies how much of the existing connectivity in the $C_1$ is used for inter-community connections. Note that, if $\delta = 0$, then $p_i = 0$ and there are no connections between communities. Therefore, values of $\delta$ near zero result in a pronounced community structure. On the other hand, if $\delta = 1$, we have $p_1 = 0$, and all links from $C_1$ are to $C_2$. In this last case, if the two communities are of the same size ($\alpha = 1/2$), all links from the nodes in $C_2$ go to the nodes in $C_1$, and the network is bipartite. In the general case, links still exist between the nodes in the largest community. A value of $\delta = 1/2$ corresponds to the case where half of the links in $C_1$ go to the same community, and half to the other and is the largest value of interest to us here.

3. Results and discussion

3.1. ER and BA networks

Figure 2 shows the result of this evaluation for the Erdős–Rényi (ER) and Barabási–Albert (BA) network models with varying values of average degree. This choice was made to evaluate the effect of degree heterogeneity. Each network has 200 nodes and we generate 100 networks for each model/parameter combination to compute the average and standard deviation of each measurement.
Figure 2. Balance, cost and number of pairs in the elite for BA and ER models. The points are the averages of 100 networks, each with 200 nodes. The error bars show one standard deviation. All the pairs of each network are evaluated.
First we note that the theoretical maximum value of balance is achieved for some
node pairs in most networks. Also, with the exception of small values of average degree,
the balance achieved by the pairs in the elite is close to the maximum in both models.
It also can be seen that the values of balance (network and elite averages, as well as
threshold) for the ER networks are slightly better than for the BA networks. This is
possibly due to the excessive influence of the hubs in the BA topology, making it more
sensitive to the choice of pairs.

The situation is different with regard to communication costs, where BA networks
are better (with the exception of networks with a high average degree). It also is inter-
esting to note that the difference in costs for the best and average pairs is much larger
in the BA networks. This is due to the fact that in these networks, the hubs are central
(in the closeness centrality sense) and therefore have small average distances to the
other network nodes. If two hubs are chosen in a pair, the communication costs for the
pair will be small. But pairs with two hubs are a small minority of all the possible pairs,
and therefore do not significantly affect the averages.

The networks have $N = 200$ nodes, and therefore there are about $N^2/2 = 20000$
distinct pairs. For the elite, we choose the pairs that are in the 20% better in cost and
in balance. If the two criteria were unrelated, the expected number of pairs in the elite
would be $0.04 N^2/2 = 800$. As can be seen in figure 2, the number of pairs in the elite
of the ER networks is close to this expected value, with significant differences only for
small average degrees. On the other hand, in the BA networks the number of pairs in
the elite is much lower, about half of the number in the ER networks. This suggests
that in topologies with a strong degree of heterogeneity the efficiency is much more
sensitive to the choice of the pair of servers.

3.2. Communities

Figure 3 shows the impact of changes in community sizes ($\alpha$) on the balance and cost
for $\delta = 0.1$ (strong community separation). We see that cost is statistically independent
of $\alpha$, while there is a small effect for balance: greater differences in community sizes
(smaller alpha) reduce the average balance, but has a lesser effect on the performance
of the best pairs. As expected, for $\delta = 1/2$ there is no influence of the division of nodes
in communities, as they are not well separated. For intermediate values of $\delta$ some
influence of $\alpha$ can be seen, but it disappears at about $\delta = 1/4$ (we do not show this
results in the figure). Values for balance and cost for larger $\delta$ are close to the ones for
$\alpha = 0.5$ in the figure.

Figure 4 shows the effect of varying $\delta$, fixing $\alpha = 0.25$. As the community separation
gets stronger (smaller values of $\delta$), the balance and cost get worse, but only slightly.
For communities of similar sizes ($\alpha$ closer to 1/2) even this small effect disappears and
there is almost no influence of $\delta$ (results not shown in the figure). This means that a
network with communities of different sizes and strong community separation is not
well suited for this kind of dynamics, but also that this is only important in extreme
cases (both $\alpha$ and $\delta$ small).

The previous results show that there is only a small influence of communities in the
performance, but they assume the best pairs of servers are selected. We complement
the results with figure 5, where we fix $\alpha = 1/2$ and change $\delta$ (left), or fix $\delta = 0.1$ and
change $\alpha$ (right), and evaluate the number of pairs in the elite (top) and the fraction of these elite pairs where each element is in a different community (bottom). On the top left we see that, for communities of the same size, the number of pairs in the elite is not

Figure 3. Balance and cost in a community model as a function of size differences. The network has two communities, with nodes distributed between them according to parameter $\alpha$ (values of $\alpha$ close to 1/2 imply communities of similar size, see the text). The connectivity between the nodes in the two networks is controlled by parameter $\delta$ (larger values of $\delta$ imply more connections between communities, see the text). The results are averages of 100 networks, each with 200 nodes; the error bars show one standard deviation.

Figure 4. Effect of the community separation on balance and cost, for $\alpha = 0.25$. Separation is controlled by parameter $\delta$. The results are averages of 100 networks, each with 200 nodes; the error bars show one standard deviation.
affected by the strength of the separation of communities. The bottom left-hand plot shows that for small values of $\delta$ almost all the pairs in the elite have nodes in different communities. This means that, under a strong community structure, good efficiency can only be achieved by putting one server in each community. On the top right we see that in the case of a relatively strong community structure ($\delta = 0.1$), the number of pairs in the elite decreases as $\alpha$ is decreased from 0.5 to 0.25, but increases again afterwards. As $\alpha$ decreases, the communities are of different sizes, and it becomes more difficult to find pairs of nodes that at the same time are close to the client nodes and equally divide those clients between themselves. The increase below $\alpha = 0.25$ can be explained by looking at the bottom right-hand plot, where we see that fraction of elite pairs with nodes in different communities sharply decreases as $\alpha$ decreases. This means that, as one of the communities decreases in size, it becomes advantageous to put both server nodes in the largest community, as the increased cost for the smaller one is of little importance.

Figure 5. Number of pairs in the elite and fraction of those pairs that have a node in each community. The results are shown fixing the number of nodes in each community through the parameter $\alpha = 1/2$ and varying the strength of connectivity between the networks by changing $\delta$, or fixing $\delta = 0.1$ and varying $\alpha$. 

doi:10.1088/1742-5468/2016/02/023402
3.3. Dynamical evaluation of cost and balance

In the preceding, we used cost and balance, two topological measurements, to quantify the performance of server pairs. The actual performance will depend on other dynamical processes happening in the network such as packet routing, traffic fluctuations and congestion. A full analysis of these dynamical factors is beyond the scope of this work. Nevertheless we present here some results for a simple simulation model to help assess the impact of cost and balance in the execution of tasks.

The model used in our simulations is as follows: each node is at the same time a router for packet traffic and a client or server for the computation. We assume that a packet sent from node $i$ to node $j$ is routed through the shortest path (if there are multiple such paths one is chosen randomly). The execution of cloud tasks occurs simultaneously with background packet traffic (presumably generated by other network operations). In each node, the background packets are generated as a Poisson process with an average interarrival time of $t_p$ having as a destination a randomly chosen node. Each client generates a request for a task execution as a Poisson process with an average interarrival time of $t_t$; this time is computed after the arrival of the response to the last requested task. All the tasks have the same processing time $L$ (load) in the servers, and the server can only process one task at a time. If a new task arrives when the server is busy, it is stored in a waiting queue from which the server retrieves tasks (in arrival order) when it becomes free. The time taken for a packet to be routed from a node to the next is used as our time unit (all other times are specified in multiples of this unit). A router can only send one packet per time unit. If another packet arrives when the router is busy, it is stored in a waiting queue (of unlimited size), from which packets are retrieved in order of arrival when the router becomes free.

The results are shown in figure 6, and include times for 30 networks in each scenario, with 10 pairs of nodes selected to be servers for each network according to the criterion described.

To evaluate the impact of cost measurement, we use BA networks with 1000 nodes and average degree 2. The use of a small average degree increases the distances and helps us thus evaluate the impact of cost. For each network, we evaluate the 10 best and 10 worst pairs according to the cost measure. Figure 6(a) shows the results for a situation where the background traffic is low (top) and another where it is high (bottom). We see that there is a clear difference in the task completion times distributions, with low-cost pairs having lower times. Increased traffic makes the distinction between the best and worst pairs even clearer. A possible explanation is that with increased traffic, the packets that need to go through many routers have increased the probability of landing in a waiting queue.

To evaluate balance, we now work with networks of average degree 10. This higher average degree (in comparison with the previous simulations) reduces the distances, and makes cost less important, the results being thus determined mostly by balance. We select the best and worst pairs according to balance. In figure 6(b) we see that if there is a light load on the servers (top), there is almost no distinction between good and bad pairs, as expected, because balance is not important if each server could deal with all of the load generated by the clients. On the other hand, if we increase the load
Figure 6. Histograms of task completion times for various simulation scenarios. (a) Simulations on a BA network with $n = 1000$ $\langle k \rangle = 2$, $L = 20$, $t_t = 10^5$. Top: $t_p = 10^5$; bottom: $t_p = 900$. (b) Simulations on a BA network with $n = 1000$ $\langle k \rangle = 10$, $L = 20$, $t_p = 10^5$. Top: $t_t = 10^6$; bottom: $t_t = 3 \cdot 10^4$. (c) Simulations on a BA network with $n = 1000$ $\langle k \rangle = 6$, $L = 20$. Top: $t_p = 10^5$ and $t_t = 10^6$; bottom: $t_p = 900$ and $t_t = 3 \cdot 10^4$. 

doi:10.1088/1742-5468/2016/02/023402
(bottom), the task completion times for the worst pairs increase significantly more than for the best pairs.

Finally, we compare elite and non-elite pairs (using both cost and balance). We use the BA networks of average degree 6, compute all elite pairs, then randomly select 10 pairs each from the elite and non-elite sets. The results are shown in figure 6(c) for light loads and traffic (top) and heavy loads and traffic (bottom). We see that in both cases the elite pairs tend to outperform the non-elite pairs. The difference is not so pronounced here as in the previous results because we are selecting randomly from the sets, and the elite set is bound to contain some pairs of not so good performance (we used the same criterion to choose the elite as in the previous sections: being in the 20% best in both cost and balance) and the total number of pairs is high.

4. Conclusion

This article has investigated the effect of the distinct distribution of servers in cloud computing environments with respect to three network topologies, namely ER, BA and modular. In order to better discuss and organise the investigation, we classified as elite the pairs of servers with top performance regarding both communication cost and balance.

Several results were obtained. First, we have that ER generally provides a better balance in detriment to communication cost, while BA provides complementary characteristics. In addition, the elite pairs of servers are more populous in the ER than in the BA networks, and the difference between the best and average pairs is larger in the latter. The investigation of the modular networks was performed while varying the number of nodes in each community and the strength of connection between them. Although the balance is affected by their relative size, little effect has been observed regarding communication cost. Also, for communities with a similar size, the strength of the interconnection between them was not found to influence either communication cost or balance. However, if the communities have different sizes, less interconnection between them worsens both the balance and cost. When the separation between the communities is pronounced, most of the pairs with better performance will have each of its servers in different communities. All in all, we have confirmed that the distribution of servers in cloud computing environments can be critical for the performance in terms of communication cost and balance, with the best configurations depending heavily on the network topology.

Future works could address more than a pair of servers and other topologies, and consider the effect of specific network features on the performance and the influence of dynamical effects related to traffic and task arrival times for many scenarios.

Acknowledgment

OMB is grateful to CNPq (307797/2014-7 and 484312/2013-8). LdFC is grateful to FAPESP (2011/50761-2), CAPES, NAP-PRP-USP, and CNPq (307333/2013-2).
References

[1] Barabási A-L 2002 Linked 1st edn (Cambridge, MA: Perseus)
[2] Newman M 2010 Networks: An Introduction (Oxford: Oxford University Press)
[3] da Fontoura Costa L, Rodrigues F A, Travieso G and Villas Boas P R 2007 Characterization of complex networks: a survey of measurements Adv. Phys. 56 167–242
[4] Faloutsos M, Faloutsos P and Faloutsos C 1999 On power-law relationships of the internet topology Comput. Commun. Rev. 29 251–62
[5] Baldi P, Frasconi P and Smyth P 2003 Modeling the Internet and the Web: Probabilistic Methods and Algorithms (New York: Wiley)
[6] Tyler J R, Wilkinson D M and Huberman B A 2005 Email as spectroscopy: automated discovery of community structure within organizations Inform. Soc. 21 143–53
[7] Ma Y, He K and Du D 2005 A qualitative method for measuring the structural complexity of software systems based on complex networks 20th Asia-Pacific Software Engineering Conference (APSEC) pp 257–63
[8] da Fontoura Costa L, Travieso G and Ruggiero C A 2005 Complex grid computing Eur. Phys. J. B 44 119–28
[9] de Angelis A F, Travieso G, Ruggiero C A and da Fontoura L 2008 On the effects of geographical constraints on task execution in complex networks Int. J. Mod. Phys. C 19 847–53
[10] Travieso G and da Fontoura Costa L 2011 Effective networks for real-time distributed processing J. Syst. Sci. Complexity 24 39–50
[11] Travieso G, Ruggiero C A, Bruno O M and da Fontoura Costa L 2013 Predicting efficiency in master–slave grid computing systems J. Complex Netw. 1 63–71
[12] Prieto-Castillo F, Astillero A and Botón-Fernández M 2015 A stochastic process approach to model distributed computing on complex networks J. Grid Comput. 13 215–32
[13] Erdős P and Rényi A 1959 On random graphs Publ. Math. 6 290–7
[14] Barabási A-L and Albert R 1997 Emergence of scaling in random networks Science 286 509–12
[15] Abts D, Marty M R, Wells P M, Klausler P and Liu H 2010 Energy proportional datacenter networks SIGARCH Comput. Archit. News 38 338–47
[16] Saad A and El-Mahdy A 2013 Network topology identification for cloud instances Third Int. Conf. on Cloud and Green Computing (Karlsruhe, Germany, 30 Sept–2 Oct 2013) (Piscataway, NJ: IEEE) pp 92–8
[17] Mundt T and Vetterick J 2011 Network topology analysis in the cloud Proc. Int. Conf. on Internet Computing eds H Arabnia, V A Clincy and A M G Solo pp 326–32

doi:10.1088/1742-5468/2016/02/023402