Oscar: A Data-Oriented Overlay For Heterogeneous Environments*

Sarunas Girdzijauskas‡, Anwitaman Datta†, Karl Aberer‡

‡Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

†Nanyang Technological University (NTU), Singapore

Abstract

Quite a few data-oriented overlay networks have been designed in recent years. These designs often (implicitly) assume various homogeneity which seriously limit their usability in real world. In this paper we present some performance results of the Oscar overlay, which simultaneously deals with heterogeneity as observed in the internet (capacity of computers, bandwidth) as well as non-uniformity observed in data-oriented applications.

1 Introduction

Practical scalable P2P systems need to take into account heterogeneity explicitly in the system's design. For data-oriented overlays (internet-scale peer-to-peer index structures) heterogeneity is encountered both because of the peculiarities of the environment as well as the application characteristics. Measurement studies of deployed P2P systems show heterogeneity arising because of either diverse availability of resources like storage, bandwidth, computation and content at peers, or variation in individual willingness to contribute resources to the system, as well as software artifacts like default configurations. Data-oriented applications are characterized by non-uniform distribution of keys over the key-space as well as skewed query or access patterns. A number of research efforts address these issues in various ways. P2P systems like CAN [5], Mercury [4], P-Grid [1], skip graphs [3, 9] and derivatives [2, 6] look into into some sub-problems of these issues, like addressing load-balance under non-uniform key distributions. However, these approaches usually take advantage of uniformity assumptions on peers' capacity in terms of bandwidth consumption and storage capacity, which limits the practicality for realistic peer-to-peer environments.

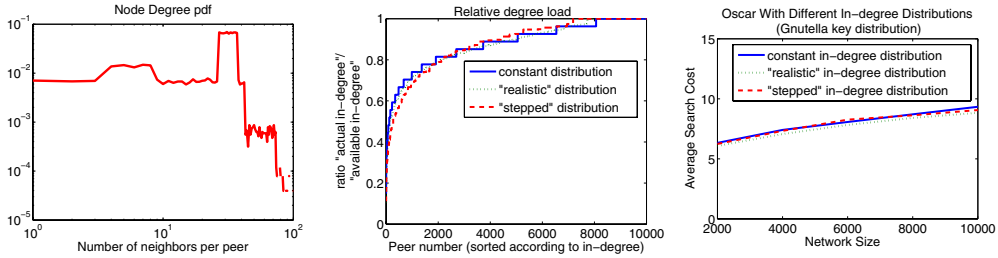
In the experiments presented in this paper we show that our recently proposed system Oscar [8] successfully deals with both the heterogeneity observed in the internet, particularly bandwidth and storage resource heterogeneity at peers, as well as non-uniformity observed in data-oriented applications, particularly skewed key distributions as well as skewed access loads which in turn uses disproportionate storage and bandwidth respectively. The Oscar overlay enjoys all the benefits of systems like P-Grid or Mercury which support complex non-uniform key distribution and hence non-exact queries (e.g. range or similarity queries) but does not suffer from node in-degree imbalance, while exhibiting good lookup performance.

Oscar is based on small-world construction principles [10] which do not restrain a peer from determining the size of the key-space based on both storage and bandwidth constraints, or having limited amount of routing links. The in- and out-degrees of a peer can vary depending peers' local decision, still providing guarantees of efficient search globally. Because the links are chosen randomly the in-degree of a peer can also be easily adjusted, where individual peers refuse further connections based on a local decision. Such features of small-world approaches enables accommodating and exploiting peer heterogeneity - storage as well as bandwidth. Peers are free to choose the maximum amount of outgoing and incoming links locally, depending on their bandwidth budget to maintain the links as well as cater to the query traffic, based on their locally perceived bandwidth or other constraints. Similarly, peers are free to choose the key-space to be responsible for based on their storage capacity and bandwidth constraint to answer the corresponding queries.

2 Oscar Overlay

It is known that to build a routing efficient network with skewed key spaces one needs to know a probability density function of peer identifiers over the identifier space [7]. One of the simplest way of doing that is to randomly sample the network and get an approximation of the key distribution, e.g. Mercury [4]. However the real-world distributions can be totally arbitrary and the only sufficient approxima-

*The work presented in this paper was (partly) carried out in the framework of the EPFL Center for Global Computing and supported by the Swiss National Funding Agency OFES as part of the European project Evergrow No 001935. The work presented in this paper was supported (in part) by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322.



(a) Synthetic spiky node degree distribution (b) relative degree load in Oscar peers (c) Search performance of Oscar given various in-degree distributions

Figure 1. Oscar's performance given various key and in-degree distributions

tion of the distribution would be gathering in a sample set the complete set of values which, of course, do not scale. In our previous work [8] we have shown that using such a technique Mercury fails to build routing efficient networks given arbitrary distribution functions. Moreover, we have also shown that it is not necessary to know the distribution function over the entire identifier space with uniform "resolution" – it is sufficient to "learn" well the distribution for only some regions of the identifier space while leaving other regions vaguely explored, making it the base idea of Oscar algorithms.

Oscar uses this intuition in order to build its routing network in a simple and efficient manner. An Oscar node u with an identifier u_{id} has to partition the identifier space into logarithmic partitions $A_1, A_2, ..A_{\log_a N}$. Each border between neighboring partitions is determined by a median value of the peer identifiers in the logarithmically decreasing peer populations, i.e. the border between A_1 and A_2 will be the median m_1 of the peer identifiers from the whole peer population $\mathcal{P}$, the border between A_2 and A_3 will be the median m_2 of the identifiers from the subpopulation $\mathcal{P} \setminus A_1$ etc. In general the border value between A_i and A_{i+1} will be the median m_i of peer identifiers from the subpopulation $\mathcal{P} \setminus B_i$, where $B_i = \cup_{j=1}^{i-1} A_j$. Ideally the first partition A_1 has to contain $\frac{1}{2}$ of the initial population, A_2 has to contain $\frac{1}{4}$ and so on. Since in practice it is not possible to exactly know the precise members of all the partitions, an Oscar node has to approximate the key range for each partition. For finding the median values an Oscar node has to uniformly sample each subpopulation B_i and determine the current median m_i from the acquired sample set. The random sampling technique proposed by Mercury is employed. To sample the subsets of the population B_i the Oscar nodes use random walkers which do not visit nodes with identifiers that do not belong to the current population B_i. Our simulation experiments show that such a technique yields very good results in practice even with very low sample sizes.

Applying the results from [7] we formulate the basis of Oscar's technique – the long-range link acquiring procedure:*each peer u first chooses uniformly at random one logarithmic partition A_i and then within that partition uniformly at random one peer v. This peer v will become a long-range neighbor of u.* Regardless the complexity of the distribution function it is sufficient to sample only $O(\log N)$ medians, hence the Oscar sampling technique is always scalable. The number of long-range links in Oscar is not restricted and can be assigned individually according to the needs of a particular peer, as long as there exist at least one such link per peer. It can be proven e.g. that in the worst case the search in Oscar network will be $O(\log^2 N)$.

3 Simulations

Here we show that the network built according to our proposed technique performs well and can adopt to various heterogeneous settings of the environment. We base our experiments on a simulation of the bootstrap of the Oscar network starting from scratch and simulating the network growth until it reaches 10000 peers. We have performed the simulations under various settings, namely varying key and in-degree distributions and performed the simulations under churn, i.e., simulating peer failures.

Each peer p in the network had values $\rho_{in}^{max}(p)$ and $\rho_{out}^{max}(p)$ which are respectively the maximal allowed in- and out-degree of the peer. During the network construction procedure each peer p was trying to establish $\rho_{out}^{max}(p)$ connections to other peers using long-range links. However only the peers which had less than ρ_{in}^{max} incoming links acknowledged to become peer p's neighbors. This ensures that each peer contributes at most what it is wiling to contribute to the system, which is also the norm in existing unstructured networks. Since Oscar is truly randomized approach we could employ the "power of two" technique [11] which allowed us to better load-balance the in-degree distribution among the peers subject to the constraint of individual peers willingness to contribute the resources. During the growth of the networks we were periodically rewiring long-range links of all the peers and measuring the performance

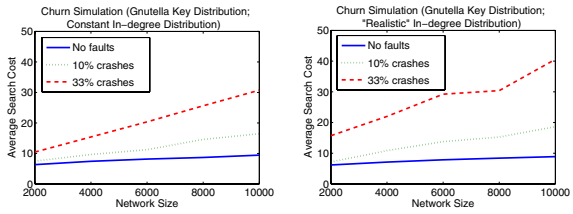
of a current network. As the performance metric we chose the average search cost which was induced by N random queries in the network.

Since our goal is to show the ability of our proposed technique to construct "routing efficient" networks (and dealing with churn is an orthogonal issue), for the first part of the tests we have simulated a fault-free environment, i.e. system without crashes. We will skip the presentation of Oscar's performance in homogeneous environments (since our previous work [8] shows that Oscar performs well under different key distributions given homogeneous peers and significantly outperforms Mercury) and will focus on peer heterogeneity issues.

Different In-Degree Distributions. Heterogenous peers. Here we show by simulations that the Oscar network building technique results in routing efficient networks not only given homogeneous peers but also assuming node-degree heterogeneity. We have performed the simulations on Oscar given three different node degree distributions: constant, "realistic" and "stepped". In the constant distribution case, for all peers, ρ_{in}^{max} and ρ_{out}^{max} were set to 27 links. For the "realistic" node degree distribution we have used a synthetic spiky distribution (Figure 1(a)) to emulate the behavior of real P2P systems [12]. The mean in/out degree in the "realistic" case was also 27 links. In the "stepped" node degree distribution case for each peer ρ_{in}^{max} and ρ_{out}^{max} values were drawn uniformly at random from four possibilities: 19, 23, 27 and 39 links. Note that the average in/out degree of the node remained 27 links as in the constant and "realistic" distribution cases. The keys for the peers were drawn from Gnutella filename distribution.

After performing network construction simulations the results show that Oscar performed almost identically for all the in-degree distribution cases (Figure 1(c)). This shows that Oscar can easily adapt to any in-degree distribution without any loss in search performance. In Figure 1(b) we see that the in-degree distribution ratio was very similar in all three cases and exploited around 85% of available degree "volume" in the system of $10,000$ peers. We also observed in our experiments that in the Mercury network with the same setting and constant in-degree distribution only 61% of available degree "volume" were exploited.

Oscar under churn. To investigate robustness of the Oscar network we have performed simulations of our system in a faulty environment. We have performed querying on two cases of faulty network: in the 1st case 10% of the peer population was "killed" and in the 2nd 33% of the whole population was "killed". We assume that the ring structure was preserved by the devised self-stabilizing techniques (e.g. Chord ring maintenance algorithms). We have modified the greedy routing algorithm to meet the requirements of a faulty network, i.e. by introducing a backtracking mechanism in case the algorithm arrives to a peer with "dead" links. However, the possibility to backtrack incurs some "wasted" traffic. After performing the simulations on two faulty network cases we saw that Oscar can successfully cope with quite a high number of peer failures in the network. From Figure 2 we observe that Oscar remains navigable and that the search cost is fairly low given the high rate of failed peers.



(a) Search performance under churn of Oscar with constant in-degree distribution

(b) Search performance under churn of Oscar with "realistic" in-degree distribution

Figure 2. Churn in Oscar

References

[1] K. Aberer, A. Datta, M. Hauswirth, and R. Schmidt. Indexing data-oriented overlay networks. In *VLDB 2005*, 2005.

[2] J. Aspnes, J. Kirsch, and A. Krishnamurthy. Load balancing and locality in range-queriable data structures. In *PODC2004*, 2004.

[3] J. Aspnes and G. Shah. Skip graphs. In *SODA*, 2003.

[4] A. Bharambe, M. Agrawal, and S. Seshan. Mercury: Supporting scalable multi-attribute range queries. In *ACM SIGCOMM, Portland, USA*, 2004.

[5] P. Fraigniaud and P. Gauron. The content-addressable network d2b. Technical Report Technical Report LRI 1349, Univ. Paris-Sud, 2003.

[6] P. Ganesan, M. Bawa, and H. Garcia-Molina. Online balancing of range-partitioned data with applications to peer-to-peer systems. In *VLDB*, 2004.

[7] S. Girdzijauskas, A. Datta, and K. Aberer. On small world graphs in non-uniformly distributed key spaces. In *NetDB2005, Tokyo, Japan*, 2005.

[8] S. Girdzijauskas, A. Datta, and K. Aberer. Oscar: Small-world overlay for realistic key distributions. In *DBISP2P 2006,Seoul, Korea*, 2006.

[9] N. J. A. Harvey, Jones, M. B., S. Saroiu, M. Theimer, and A. Wolman. Skipnet: A scalable overlay network with practical locality properties. In *USITS'03, Seattle, WA*, March 2003.

[10] J. Kleinberg. The Small-World Phenomenon: An Algorithmic Perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, 2000.

[11] M. Mitzenmacher, A. Richa, and R. Sitaraman. The power of two random choices: A survey of techniques and results. In *Handbook of Randomized Computing, vol. 1*, 2000.

[12] D. Stutzbach, R. Rejaie, and S. Sen. Characterizing unstructured overlay topologies in modern p2p file-sharing systems. In *In the ACM SIGCOMM Internet Measurement Conference, Berkeley*, 2005.