Universidade do Minho
Escola de Engenharia

Semana da Escola de Engenharia
October 24 - 27, 2011

ROBUST DISTRIBUTED DATA AGGREGATION

Paulo Jesus, Carlos Baquero, and Paulo Sérgio Almeida
Departmento de Informática
Universidade do Minho
Campus de Gualtar, 4710-057 Braga
E-mail: {pcoj, cbm, psa}@di.uminho.pt

KEYWORDS
Distributed Algorithms, Data Aggregation, Fault-Tolerance, Dynamic Networks.

ABSTRACT

Data aggregation plays an important role in the design of scalable systems, allowing the determination of meaningful system-wide properties. Several aggregation algorithms have been proposed in the last few years, exhibiting different properties in terms of accuracy, speed and communication tradeoffs. However existing approaches exhibit many dependability issues when used in faulty and dynamic environments. In this work, we propose a novel fault-tolerant averaging based data aggregation algorithm: *Flow Updating*. The algorithm is based on manipulating flows (in the graph theoretical sense), that are updated using idempotent messages, providing it with unique robustness capabilities. Experimental results have shown that Flow Updating outperforms previous averaging algorithms in terms of time and message complexity, and unlike them it self adapts to churn, supporting node crashes and high levels of message loss.

INTRODUCTION

Aggregation plays an important role on distributed systems, in particular to provide meaningful global properties (e.g. network size; total storage capacity; average load; or majorities), and direct the execution of decentralized applications. For instance, several network statistics and administration information can be obtained from aggregation mechanisms, like: the amount of resources available, the average session time, or the average (maximum/minimum) network load.

In the particular case of Wireless Sensor Networks (WSN), aggregation techniques are essential to monitor and control the covered area, allowing the computation of diverse statistics, such as: the minimum/maximum temperature, the average humidity, measure the concentration of a toxic substance, the noise level, etc. Moreover, due to the specific constraint found in WSN, data collection is often only practicable if aggregation is performed (to restrain energy consumption).

Robbert Van Renesse defined aggregation as “the ability to summarize information”, stating that “it is the basis for scalability for many, if not all, large networking services” (Renesse, 2003). In a nutshell, it is considered a subset of information fusion, aiming at reducing the handled data volume (Nakamura et al., 2007).

Distributed data aggregation becomes particularly difficult to achieve when faults are taken into account, and especially if dynamic settings are considered. Few have approached the problem under these settings (Madden et al., 2002; Li et al., 2005; Ganesh et al., 2007; Kostoulas et al., 2005; Jelasity et al., 2005; Kennedy et al., 2009), proving to be hard to efficiently obtain accurate and reliable aggregation results.

A useful class of high accuracy aggregation algorithms is based on averaging techniques (Kempe et al., 2003; Jelasity et al., 2005; Chen et al., 2006; Wuhib et al., 2007). Such algorithms start from a set of input values spread across the network nodes, and iteratively average their values with neighbors. Eventually all nodes will converge to the same value and can estimate some useful metric. Different aggregation functions can derivated besides average (like counting and summing), according to the initial combinations of input values. These techniques are thought to be robust and accurate (converge over time), when compared to others, but in practice they exhibit relevant problems that have been overlooked, not supporting message loss nor node crashes, see (Jesus et al., 2009a) for more details.

We design a novel averaging based aggregation technique: Flow Updating. This new algorithm tolerates high levels of message loss and supports nodes arrival and departure/crash. Moreover, this technique achieves an improved convergence speed compared to previous approaches.

FLOW UPDATING

Flow Updating (FU) is an averaging based aggregation algo-rithm, that enables the decentralized computation of ag-gregation functions (e.g. average, count or sum). It works over any network topology and tolerates faults. Like existing gossip-based approaches, it averages values iteratively during the aggregation process towards converging to the global network average. It departs from current approaches, that keep the current “mass” value in a variable and send “mass” in messages; in these approaches message loss implies mass loss and consequent deviation from the correct estimate.

The key idea in FU is to use the flow concept from graph theory, and instead of storing in each node the current estimate in a variable, compute it from the input value and the contribution of the flows along edges to the neighbors.

The essence of the algorithm is: each node i stores the flow $f(ij)$ to each neighbor j; i sends flow $f(ij)$ to j in a message; a node j receiving $f(ij)$ updates its own $f(ji)$ with $-f(ij)$. Messages simply update flows, being idempotent; the value in a subsequent message overwrites the previous one, it does not add to the previous value. If the skew symmetry of flows holds, the sum of the estimates for all nodes (global mass) will remain constant.

Enforcing the skew symmetry of flows along edges through idempotent messages is what confers FU its unique fault tolerance characteristics, that distinguish it from previous approaches. It tolerates message loss by design without requiring additional mechanisms to detect and recover mass from lost messages. It solves the mass conservation problem, not by instantaneous mass invariance (impossible to achieve in a failure-prone distributed system), but by having *mass convergence*.

FU copes with node departure/crash and node arrival, simply by maintaining a dynamic mapping of flows according to the current set of neighbors: re-moving the entries relative to leaving/crashing nodes, and adding entries for newly arrived nodes.

CONCLUSION

We designed new distributed data aggregation approach: *Flow Updating*. Like existing average-based algorithms it allows the accurate computation of aggregates at all nodes, converging to the exact result along time, and working independently from the network routing topology. Unlike previous approachs, it is robust against message loss, overcoming the “mass” loss problem verified on existing averaging algorithms. Moreover, it has shown to be able to adpat to changes of the network membership and supports node crashes.

Evaluation showed that FU clearly outperforms previous strategies, and unlike them it is fault tolerant and continuously adapts to changes without requiring protocol restarts (Jesus et al., 2009b; Jesus et al., 2010).

REFERENCES

Chen, J.-Y., Pandurangan, G. & Xu, D., 2006. Robust computation of aggregates in wireless sensor networks: Distributed randomized algorithms and analysis. *IEEE Trans. Parallel Distrib. Syst.*, 17(9), p.987–1000.

Ganesh, A., Kermarrec, A., Merrer, E.L. & Massoulie, L., 2007. Peer counting and sampling in overlay networks based on random walks. *Distributed Computing*, 20(4), p.267–278.

Jelasity, M., Montresor, A. & Babaoglu, O., 2005. Gossip-based aggregation in large dynamic networks. *ACM Transactions on Computer Systems*, 23(3), p.219–252.

Jesus, P., Baquero, C. & Almeida, P.S., 2009. Dependability in aggregation by averaging. In *Simpósio de Informática (INForum)*. Lisboa, 2009.

Jesus, P., Baquero, C. & Almeida, P.S., 2009. Fault-tolerant aggregation by flow updating. In *9th IFIP International Conference on Distributed Applications and Interoperable Systems*. Lisbon, 2009. Springer LNCS.

Jesus, P., Baquero, C. & Almeida, P.S., 2010. Fault-Tolerant Aggregation for Dynamic Networks. In *29th IEEE Int. Symposium on Reliable Distributed Systems*. New Delhi.

Kempe, D., Dobra, A. & Gehrke, J., 2003. Gossip-based computation of aggregate information. In *44th Annual IEEE Symposium on Foundations of Computer Science*.

Kennedy, O., Koch, C. & Demers, A., 2009. Dynamic approaches to in-network aggregation. In *25th IEEE International Conference on Data Engineering*.

Kostoulas, D. et al., 2005. Decentralized schemes for size estimation in large and dynamic groups. In *4th IEEE Int. Symposium on Network Computing and Applications*.

Li, J., Sollins, K. & Lim, K., 2005. Implementing aggregation and broadcast over distributed hash tables. *ACM Computer Communication Review*, 35(1), p.81–92.

Madden, S., Franklin, M., Hellerstein, J. & Hong, W., 2002. TAG: a Tiny AGgregation service for ad-hoc sensor networks. *ACM SIGOPS Operating Systems Review*, 36(SI), pp.131-46.

Nakamura, E., Loureiro, A. & Frey, A., 2007. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys*, 3(39).

Renesse, R.V., 2003. The importance of aggregation. *Future Directions in Distributed Computing*, (2584), p.87–92.

Wuhib, F., Dam, M., Stadler, R. & Clemm, A., 2007. Robust monitoring of network-wide aggregates through gossiping. In *10th IFIP/IEEE International Symposium on Integrated Network Management.*, 2007.