Convergence Detection in Epidemic Aggregation

Pasu Poonpakdee[1], Neriman Gamze Orhon[2,*], and Giuseppe Di Fatta[1]

[1] The University of Reading, Whiteknights, Reading, Berkshire, RG6 6AY, UK
{p.poonpakdee,g.difatta}@reading.ac.uk
[2] Yaşar University, Üniversite Caddesi, Ağaçli Yol, Bornova, İzmir PK. 35100
gamzeorhon@gmail.com

Abstract. Emerging challenges in ubiquitous networks and computing include the ability to extract useful information from a vast amount of data which are intrinsically distributed. Epidemic protocols are a bio-inspired approach that provide a communication and computation paradigm for large and extreme-scale networked systems. These protocols are based on randomised communication, which provides robustness, scalability and probabilistic guarantees on convergence speed and accuracy. This work investigates the convergence detection problem in epidemic aggregation, which is critical to minimise the execution time for a given approximation error of the estimated aggregate. Global and local convergence criteria are presented and compared. The experimental analysis shows that a local convergence criterion can be adopted to minimise and adapt the number of cycles in epidemic aggregation protocols.

Keywords: epidemic protocols, gossip-based protocols, extreme-scale computing, decentralised algorithms.

In extreme-scale distributed systems, computing and spreading global information is a particularly challenging task. Centralized paradigms are not suitable for this task, as they introduce bottlenecks and failure intolerance; fully decentralised and fault-tolerant approached are needed.

The data aggregation problem refers to the computation of a global aggregation function in a network of nodes, where each node is holding a local value. Examples of global aggregation functions are the sum, the average, the maximum, the minimum, random samples, quantiles, etc. The goal of data aggregation in networks is the determination at each node of the exact value, or of a good approximation, of a global aggregation function over the distributed set of values.

Several distributed aggregation protocols have been proposed. They can be divided into two main classes: tree-based protocols and Epidemic (or Gossip-based) protocols. The former performs a tree-based communication throughout a tree overlay structure (e.g., [1]). Tree-based protocols support a minimum

* This work was carried out while N. G. Orhon was at the University of Reading, UK, for a placement of the Erasmus Training Programme (June-Sept. 2013).

D. an Mey et al. (Eds.): Euro-Par 2013 Workshops, LNCS 8374, pp. 292–300, 2014.

number of communications, but can be affected by single points of failure and require additional communication overhead to manage the overlay structure in dynamic environments. The second class is based on Epidemic protocols, which are robust and scalable, as they use a randomised communication paradigm.

Recently, it has been shown that epidemic protocols can be adopted for the formulation of data mining algorithms [2, 3], even under node churn and network failures [4].

Epidemic protocols have communication costs usually greater than tree-based protocols. In order to achieve a given approximation error, epidemic aggregation protocols have to be run for a sufficient number of rounds and their convergence speed depends on several factors. Limiting the number of protocol rounds is an important aspect for their adoption in practical applications. This work investigates adaptive criteria that detect the convergence to the desired approximation error and minimise the overall duration of the epidemic aggregation.

The rest of the paper is organised as follows. Epidemic protocols are presented in section 1. Section 2 introduces the convergence detection criteria for epidemic aggregation protocols. Section 3 presents the results of simulations and a comparative analysis. Finally, section 4 provides some conclusive remarks.

1 Epidemic Aggregation

Let us consider a networked system where each node i ($1 \leq i \leq N$) holds some local value x_i and needs to compute a global aggregation function $f(x_1, \ldots, x_N)$. Local approximations of the global aggregate function can be obtained with an epidemic aggregation protocol.

Epidemic protocols are typically described as periodic and synchronous with a cycle length Δ_T and are executed for a number of cycles T_{max}. At all discrete times t ($0 < t \leq T_{max}$), each node independently exchanges information with a peer, which is ideally selected uniformly at random among all nodes in the system. This selection operation is provided by a peer sampling service, the membership protocol, which is also implemented with an epidemic approach. Nodes periodically exchange their local state and the reception of a remote state triggers the update of the local state at a node. The update produces a reduction of the variance in the estimates in the system. The definitions of local state, type of messages and update operation depend on the particular aggregation protocol and the target global function. A good approximation of the global aggregate function can be obtained at every node within a number of protocol cycles.

Membership protocols and aggregation protocols are briefly reviewed in the following sections.

1.1 Membership Protocols

The node sampling service is considered a fundamental abstraction in distributed systems [5]. In large-scale systems nodes cannot build and maintain a complete directory of memberships. A membership protocol builds and maintains a partial

view of the system, which is used to provide the random node selection service. The distributed set of views implicitly defines an overlay topology. A membership protocol periodically and randomly changes the local views, thus generating a sequence of random overlay topologies.

The required assumptions are that the physical network topology is a connected graph, a routing protocol is available and an initialisation mechanism for the overlay topology is provided.

Memberships protocols can be seen as a distributed implementation of multiple random walks. After sufficiently many cycles the entries in the local view are uniformly distributed. In regular connected graphs, random walks converge to uniform independent samples of the node set in a polynomial number of steps. In expander graphs, i.e. sparse graphs that are very well connected, random walks converge to the uniform distribution in $O(log(N))$ [6]. For this reason, the initial overlay topology is typically not a concern and simulations, including those presented in this work, often adopt a random graph.

The membership protocol adopted in this work is *CYCLON* [7], which has been chosen for its simplicity, effectiveness and low communication cost. In a preliminary analysis, no significant improvement was found in adopting more accurate and complex approaches (e.g.,[27]).

1.2 Aggregation Protocols

Several epidemic aggregation protocols have been proposed (e.g., [8, 9, 10, 11]). In the Push-Sum Protocol (PSP) [8], the local state of a node i is represented as a pair $\langle s_{i,t}, w_{i,t} \rangle$: the initial value $s_{i,0}$ and weight $w_{i,0}$ depend on the global aggregation function to be computed. For example, for computing the global average, $s_{i,0}$ is initialised with the local value x_i and the weight $w_{i,0}$ with 1.

At each cycle t, each node halves its local value and weight $\langle s_{i,t}, w_{i,t} \rangle = \langle \frac{1}{2} s_{i,t-1}, \frac{1}{2} w_{i,t-1} \rangle$ and sends the other halves to a randomly selected node. In each cycle N messages are sent in total. The global mass of the values $(\sum_{i=1}^{N} s_i)$ and of the weights $(\sum_{i=1}^{N} w_i,)$ is guaranteed to be preserved when a reliable communication protocol is used. This is known as the mass conservation invariant. At any cycle t, the local approximation of the global aggregate is given by the ratio $\frac{s_{i,t}}{w_{i,t}}$, which converges to the true target value under validity of the mass conservation invariant.

The *diffusion speed* is the minimum number of protocol cycles t^* required to achieve a good approximation of the global aggregate function with high probability: $Prob(e_{i,t} < \varepsilon) \geq 1 - \delta, \forall t \geq t^*$, where $e_{i,t}$ the approximation error at node i and ε and δ two arbitrary small positive constants, respectively the maximum approximation error and the maximum probability of greater error than ε. In PSP, the *diffusion speed* t^* has been shown to have a complexity $O(\log(N) + \log(\varepsilon^{-1}) + \log(\delta^{-1}))$ [8].

In [9, 10], two similar algorithms are proposed; they can be both referred to as Push-Pull Gossip protocol (PPG). PPG uses a push-pull scheme that improves the diffusion speed w.r.t. PSP by adopting a symmetric exchange of the local

state between communication peers. The node state is represented by a value v_i, which is the local approximation of the global average and is initialised with the local value x_i. At each cycle, a node i randomly chooses a node j to perform a pair-wise averaging operation: the two nodes exchange and update their local values with $\frac{v_i+v_j}{2}$. In PPG, $2 \times N$ messages are sent in total at each cycle.

In several works [12, 13, 11], it has been shown that in asynchronous communication networks PPG violates the mass conservation invariant and does not converge to the correct target value.

The Symmetric Push-Sum Protocol (SPSP) [11] combines the accuracy and simplicity of the push-based approach and the efficiency of the push-pull scheme. SPSP does not require synchronous communication with atomic operations; it achieves a convergence speed similar to the push-pull scheme, while keeping the accuracy of the push scheme. SPSP and the global average as target function have been used in the simulations presented in this work.

2 Convergence Detection

The target value of the aggregation protocol corresponds to the population mean $\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$. At each cycle t of the aggregation protocol, a local approximation $v_{i,t}$ of the target value is available. Under validity of the mass conservation invariant ($\sum_{i=1}^{N} v_{i,t} = \sum_{i=1}^{N} x_i, \forall t$), the approximations tend to the target value, i.e. $\lim_{t\to\infty} v_{i,t} = \mu_x, \forall i$. The standard deviation of the approximations in the system is defined as $\sigma_v(t) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_{i,t} - \mu_x)^2}$.

In the simulations, a *global convergence criterion* can be tested by computing the minimum cycle t^* at which the system reaches a small standard deviation: $\sigma_v(t) < \varepsilon, \forall t \geq t^*$ and where ε is a small positive constant.

In practical applications, it is useful to determine a *local convergence criterion*, which can be used to terminate the protocol iterations. The global convergence can be estimated locally by monitoring the approximation error at the node. To this aim, each node can run multiple independent instances of the epidemic aggregation protocol, which are initialised with the same local value x_i. The sample average and the sample standard deviation over the local multiple approximations can be used in place of the population mean and population standard deviation. If m is the number of multiple instances of the protocol at each node, the local approximation at node i and time t is indicated as $v_{i,j,t}$ $(0 < j \leq m)$. The sample mean is given by $\bar{v}_{i,t} = \frac{1}{m}\sum_{j=1}^{m} v_{i,j,t}$ and the corrected sample standard deviation is $s_{i,t} = \sqrt{\frac{1}{m-1}\sum_{j=1}^{m}(v_{i,j,t} - \bar{v}_{i,t})^2}$.

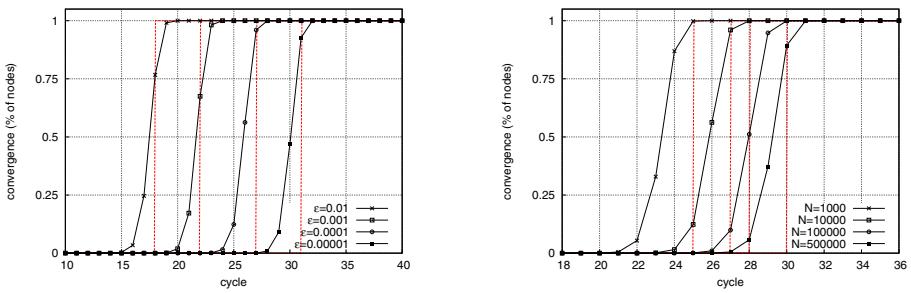
The first local convergence criterion is defined as: $s_{i,t} < \varepsilon$. In the remainder of this work, this local convergence criterion is indicate by $\langle m \rangle M$. For example, $6M$ indicates a local criterion based on six aggregation protocols. In this case, the local convergence criterion is determined at an additional communication cost, which is $(m-1)$ times the cost of a single aggregation protocol.

A second local approach based on a single local protocol can be determined by monitoring the local estimate of the node and the remote estimates of its

communication peers over time. A local buffer (FIFO queue) of size h is used to store the local estimate and the remote estimates received with $PUSH$ and $PULL$ messages. At the reception of a message from node j, a node i stores the local estimate v_i and the remote estimate v_j in the local buffer before updating the local estimate. For example, if the buffer has size $h = 4$, at the beginning of the cycle t, it contains a set of estimates $S_i = \{v_{i,t-1}, v_{j,t-1}, v_{i,t-2}, v_{u,t-2}\}$. An average squared error from the target value can be used to determine a local convergence criterion. In this case, the best estimate of the target value is given by the current local estimate $v_{i,t}$.

The second local convergence criterion is defined on the root-mean-square error, $e_{i,t} < \varepsilon$, where $e_{i,t} = \sqrt{\frac{1}{h}\sum_{v \in S_i}(v - v_{i,t})^2}$. This local convergence criterion is indicate by $\langle h \rangle H$. For example, $4H$ indicates a local criterion based on a history buffer of size $h = 4$.

A further and hybrid approach can be defined by combining the previous two local approaches: multiple local protocols are used in combination with a history buffer. This local convergence criterion is indicate by $\langle m \rangle M \langle h \rangle H$. For example, $2M8H$ indicates a local criterion based on a history buffer of size 8 with two local aggregation protocols. In this case, the best estimate of the target value is given by the average of the multiple local approximations, i.e. $\bar{v}_{i,t} = \frac{1}{m}\sum_{j=1}^{m} v_{i,j,t}$. The third local convergence criterion is defined on the root-mean-square error, $e_{i,t} < \varepsilon$, where $e_{i,t} = \sqrt{\frac{1}{h}\sum_{v \in S_i}(v - \bar{v}_{i,t})^2}$.



(a) varying ε ($N = 10^4$) (b) varying N ($\varepsilon = 10^{-4}$)

Fig. 1. Method with multiple protocols ($6M$) for different values of ε and network size N. Corresponding global transitions are shown in dashed lines.

3 Experimental Analysis

We carried out the experimental tests on PeerSim [14], a scalable network simulator based on discrete events. The overlay topology is initialised as a random graph with a constant out degree (30). The simulations are based on the aggregation protocol *SPSP* [11] and the membership protocol *CYCLON* [7] with

cache size 30 and shuffle length 15. We have adopted an asynchronous network setting with a uniform distribution of network latencies. The distributed values are initialised with a peak distribution: all nodes have initial value 0, but one that has the peak value N. In all simulations, the target value (average) is 1.0 regardless of the network size.

Figure 1 shows the effect of different values of the threshold ε (figure 1(a)) and of the network size N (figure 1(b)) on the global convergence criterion and the local one based on *six* multiple protocols ($6M$). The charts show the percentage of the nodes that have detected the convergence over time. Each curve shows the results of a single simulation and the global method is represented as a step function (dashed lines) and is used as reference.

In figure 1(a), the global convergence with a maximum error of 10^{-2}, 10^{-3}, 10^{-4} and 10^{-5}, is achieved respectively, at the cycles 18, 22, 27 and 31. The $6M$ curves associated to the different values of ε appear to be equally spaced. This clearly shows the *log* dependency of the diffusion speed on the inverse of the maximum approximation error ε.

In figure 1(b), the global convergence for the network sizes 10^3, 10^4, 10^5 and $5 \cdot 10^5$, is achieved, respectively, at cycles 25, 27, 28, and 30. The $6M$ curves similarly show the dependency of the diffusion speed on the network size.

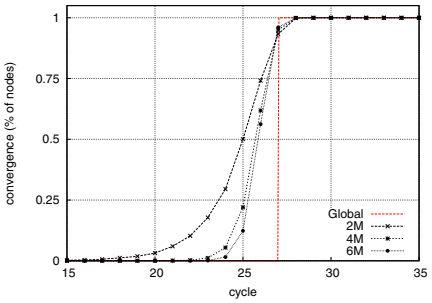
Figure 2 shows the convergence transition of the different methods with $\varepsilon = 10^{-4}$ and $N = 10^4$. Each curve shows the results of a single simulation and the global method is used as reference.

In figure 2(a), the behavior of the multiple protocols method is shown for different numbers of local protocols (2, 4 and 6). Obviously, the more protocols are used, larger the sample size and a better approximation of the global convergence is achieved. It is observed that the local convergence of most nodes is anticipated with respect to the global convergence.

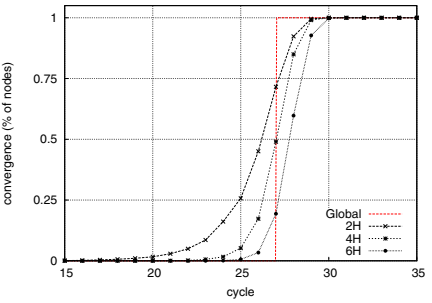
In figure 2(b), the behavior of the method based on a buffer of historical values is shown for different buffer sizes (2, 4 and 6). It is observed that the local convergence is delayed with respect to the approach $\langle m \rangle M$. This is due to the use of estimates from previous cycles in the $\langle h \rangle H$ convergence detection.

In figure 2(c), the behavior of the hybrid method is shown for a different number of local protocols (2, 4 and 6) and different buffer sizes (8, 16 and 24). The combinations of these two parameters have been chosen to fix the number of historical values from each local protocol to 2 (similarly to $2H$). For example, in $2M8H$ each node has two protocols which contribute with historical values from the past two cycles, for a subtotal of 4 local values. Each node also collects estimates from communication peers of previous cycles for a subtotal of 4 remote values.

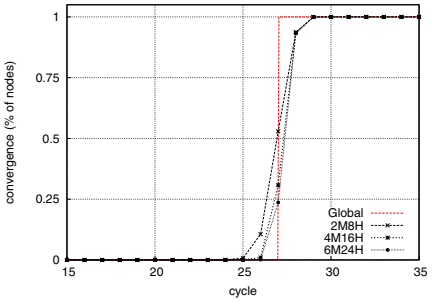
Figure 2(d) is used to provide an easy comparison of the different methods. It shows that the hybrid method performs better than the other two methods. The method $2M8H$ has only twice the communication cost of the approach based on a single protocol ($6H$) and is the most similar to the global criterion.



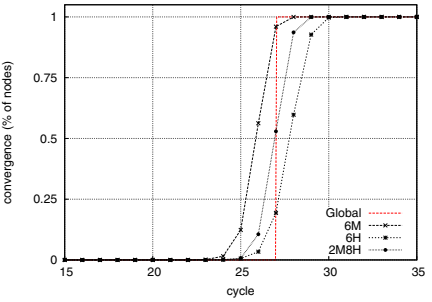
(a) multiple protocols (M)



(b) History buffer (H)



(c) Multiple protocols with buffer (MH)



(d) comparison

Fig. 2. Convergence transition of different methods ($\varepsilon = 10^{-4}$ and $N = 10^4$)

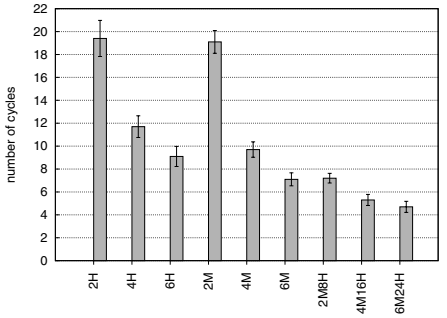


Fig. 3. Average number of cycles in the convergence transition ($\varepsilon = 10^{-4}$ and $N = 10^4$)

In order to validate the qualitative results of figure 2, for each case we have run multiple simulations with 10 different initialisation seeds. For each simulation setting, the number of cycles employed by each method to complete the convergence transition is recorded. The smaller this number, the better the transition approximates the step function of the global method. The average number of cycles and the standard deviation are shown in figure 3. The hybrid method confirms its shorter transition length to achieve a global convergence without a global communication.

4 Conclusions

This work has studied the convergence detection problem in decentralised aggregation based on epidemic protocols. This is important in efficient and practical applications, which require to achieve a given accuracy in the minimum number of cycles. This work has investigated local convergence criteria and has compared them with a global criterion. The experimental analysis has shown that local convergence detection can be employed to minimise and adapt the execution time of an epidemic aggregation protocol. This is an important step towards practical applications of epidemic protocols. Future work will be devoted to study the effect of node churn and network faults on the convergence detection.

References

[1] Dam, M., Stadler, R.: A generic protocol for network state aggregation. In: Proc. Radiovetenskap och Kommunikation (RVK), pp. 14–16 (2005)
[2] Di Fatta, G., Blasa, F., Cafiero, S., Fortino, G.: Epidemic k-means clustering. In: Proc. of the IEEE Int.l Conf. on Data Mining Workshops, pp. 151–158 (2011)
[3] Mashayekhi, H., Habibi, J., Voulgaris, S., van Steen, M.: GoSCAN: Decentralized scalable data clustering. Computing 95(9), 759–784 (2013)
[4] Di Fatta, G., Blasa, F., Cafiero, S., Fortino, G.: Fault tolerant decentralised k-means clustering for asynchronous large-scale networks. Journal of Parallel and Distributed Computing 73(3), 317–329 (2013)
[5] Jelasity, M., Voulgaris, S., Guerraoui, R., Kermarrec, A.M., van Steen, M.: Gossip-based peer sampling. ACM Trans. Comput. Syst. 25(3) (August 2007)
[6] Gillman, D.: A chernoff bound for random walks on expander graphs. SIAM Journal on Computing 27(4), 1203–1220 (1998)
[7] Voulgaris, S., Gavidia, D., Steen, M.: Cyclon: Inexpensive membership management for unstructured p2p overlays. Journal of Network and Systems Management 13(2), 197–217 (2005)
[8] Kempe, D., Dobra, A., Gehrke, J.: Gossip-based computation of aggregate information. In: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, pp. 482–491 (October 2003)
[9] Jelasity, M., Montresor, A., Babaoglu, O.: Gossip-based aggregation in large dynamic networks. ACM Trans. on Comp. Sys. 23(3), 219–252 (2005)
[10] Boyd, S., Ghosh, A., Prabhakar, B., Shah, D.: Randomized gossip algorithms. IEEE Transactions on Information Theory 52(6), 2508–2530 (2006)

[11] Blasa, F., Cafiero, S., Fortino, G., Di Fatta, G.: Symmetric push-sum protocol for decentralised aggregation. In: Proc. of the Int. l Conf. on Advances in P2P Systems, pp. 27–32 (2011)
[12] Jesus, P., Baquero, C., Almeida, P.: Dependability in aggregation by averaging. In: 1st Symposium on Informatics (INForum 2009), pp. 482–491 (September 2009)
[13] Rao, I., Harwood, A., Karunasekera, S.: Impacts of asynchrony on epidemic-style aggregation protocols. In: Proc. of the IEEE Int.l Conf. on Parallel and Distributed Systems, pp. 601–608 (2010)
[14] Montresor, A., Jelasity, M.: PeerSim: A scalable P2P simulator. In: Proc. of the 9th Int. Conference on Peer-to-Peer (P2P 2009), pp. 99–100 (September 2009)