CS599: Structure and Dynamics of Networked Information (Spring 2005)
04/25/2005: Analysis of Gossip-based Averaging
Scribes: David Kempe

1 Averaging using Gossip

Last lecture, we started looking at the averaging problem: each node i has a number x_i, and we want to compute the average $\bar{x} = \frac{1}{n}\sum_i x_i$ of these numbers. We looked at the following simple algorithm Push-Sum [3]: Each node i has a *sum* s_i (initialized to $s_i := x_i$), and a *weight* w_i (initialized to $w_i := 1$). In each round, each node executes the following protocol:

Algorithm 1 Push-Sum

1: Send the pair $(s_i/2, w_i/2)$ to yourself and a uniformly randomly chosen other node.
2: Let J_i be the set of all nodes that have sent a message to i in this round.
3: The new values are $s'_i := s_i/2 + \sum_{j \in J_i} s_j/2$ and $w'_i := w_i/2 + \sum_{j \in J_i} w_j/2$.
4: Keep track of s_i/w_i as approximation of the average.

We will prove the following theorem, showing that the estimates converge exponentially fast:

Theorem 1 *If all x_i are non-negative, then within $O(\log n + \log \frac{1}{\delta} + \log \frac{1}{\epsilon})$ rounds, all estimates s_i/w_i are within $(1 \pm \epsilon)$ of the true average $\bar{x} = \frac{1}{n}\sum_i x_i$, with probability at least $1 - \delta$.*

In order to analyze this protocol, we will track what "share" of each node j's number x_j is currently contributing to each node's sum s_i. So we look at a vector $\vec{v}_i$, in which the j^{th} component $v_{i,j}$ denotes the fraction of node j's value that currently is part of node i's sum. This means that initially, we have $v_{i,i} = 1$ for all i, and $v_{i,j} = 0$ for all $i \neq j$. In this view, our protocol can be expressed as follows:

Algorithm 2 Push-Vector

1: Send the vector $\frac{1}{2}\vec{v}_i$ to yourself and a uniformly randomly chosen other node.
2: Let J_i be the set of all nodes that have sent a message to i in this round.
3: The new vector is $\vec{v}'_i := \frac{1}{2}\vec{v}_i + \sum_{j \in J_i} \frac{1}{2}\vec{v}_j$.

This new version of the protocol traces the old version in the following sense (as is easy to observe, and can be proved by induction on the time steps of the protocol):

Fact 2 *At any time during the execution of the protocol, and for any node i, we have that $s_i = \sum_j v_{i,j} x_j$, and $w_i = \sum_j v_{i,j}$.*

Thus, the estimate that node i has is $\frac{\sum_j v_{i,j} x_j}{\sum_j v_{i,j}}$, and we want to show that this quantity converges exponentially fast to $\bar{x}$. One way that we could guarantee convergence would be if all $v_{i,j}$ were equal, and thus equal to $1/n$. In that case, we would have exactly the true average. However, this is clearly too much to hope for, as even if they were all equal at some point, one node would be likely to receive many calls (up to $\Omega(\log n)$), and others no call at all, resulting in new values $\Omega(\frac{\log n}{n})$ vs. $\frac{1}{2n}$. However, upon closer inspection, we notice that we do not need quite such a strong condition. It would be enough if, for a fixed i, all $v_{i,j}$ were the same. So we do not need a node's value to be equally distributed among all other nodes; all we need is that a node has equally sized shares of everyone else's values.

This motivates studying how fast the vectors $\vec{v}_i$ converge to multiples of the all-ones vector $\vec{1}$. In order to talk about this convergence, we use $v_{t,i,j}, \vec{v}_{t,i}, s_{t,i}, w_{t,i}$ etc. to denote the values after t iterations of Push-Vector. For ease of notation, we will use the fact that $w_{t,i} = \sum_j v_{t,i,j}$. We then measure the convergence in terms of the error $\Delta_{t,i} = \max_j |\frac{v_{t,i,j}}{w_{t,i}} - \frac{1}{n}|$. We will prove the following two parts, giving the theorem together:

Lemma 3 *1. The $\Delta_{t,i}$ converge to 0 exponentially fast.*

2. When the $\Delta_{t,i}$ are small, the estimate of the average is good.

Proof. We first prove the (easier) second part of the lemma. Assume that at some point in time t, the errors $\Delta_{t,i}$ for all i are at most ϵ/n. Then, for each i,

$$\begin{aligned} \frac{|\frac{\sum_j v_{t,i,j} x_j}{w_{t,i}} - \bar{x}|}{|\bar{x}|} &= n \cdot \frac{|\sum_j (\frac{v_{t,i,j}}{w_{t,i}} - \frac{1}{n}) x_j|}{|\sum_j x_j|} \\ &\leq \frac{n}{|\sum_j x_j|} \cdot (\max_j |\frac{v_{t,i,j}}{w_{t,i}} - \frac{1}{n}|) \cdot \sum_j |x_j| \\ &\leq \epsilon, \end{aligned}$$

where we used the Triangle Inequality in the numerator for the second step, and the bound on $\Delta_{t,i}$ for the expression in parentheses in the third step. (We also were allowed to cancel the sums over x_j values because all x_j were assumed to be non-negative.) Notice that once we prove exponential convergence below, the time it takes to converge to error ϵ/n is only by an additive $O(\log n)$ larger than to converge to ϵ, so we were free to choose a value of ϵ/n here.

To prove the first part of the lemma, we study a *potential function*, which measures how "close" to converged the system is. We have seen such functions before, for instance in the analysis of Berger's model for dynamic monopolies [1]. Here, our potential function will be the sum of variances of the vectors $\vec{v}_{t,i}$. Formally, we define

$$\Phi_t = \sum_{i,j} (v_{t,i,j} - \frac{w_{i,t}}{n})^2.$$

We will show that this potential function decreases exponentially fast, and that a small value for it implies good convergence.

Lemma 4 *The conditional expectation of Φ_t satisfies* $\mathrm{E}\left[\Phi_{t+1} \mid \Phi_t = \phi\right] = (\frac{1}{2} - \frac{1}{2n})\phi$.

Proof. Consider the values $v_{i,j}, w_i$, etc. at time t, and let $f(i)$ denote the random node called by node i in round t. Then, with all random choices known, node i's new vector $\vec{v}_i'$ and weight w_i' are

$$\begin{aligned} \vec{v}_i' &= \frac{1}{2}\vec{v}_i + \frac{1}{2}\sum_{k:f(k)=i} \vec{v}_j, \\ w_i' &= \frac{1}{2} w_i + \frac{1}{2}\sum_{k:f(k)=i} w_j. \end{aligned}$$

Plugging these values into the new potential Φ_{t+1}, we obtain that

$$\begin{aligned} \Phi_{t+1} &= \sum_{i,j} \left(\frac{1}{2}(v_{i,j} - \frac{w_i}{n}) + \frac{1}{2}\sum_{k:f(k)=i}(v_{k,j} - \frac{w_k}{n})\right)^2 \\ &= \frac{1}{4}\sum_{i,j}(v_{i,j} - \frac{w_i}{n})^2 + \frac{1}{4}\sum_{i,j}\sum_{k:f(k)=i}(v_{k,j} - \frac{k_i}{n})^2 + \frac{1}{2}\sum_{i,j}\sum_{k:f(k)=i}(v_{i,j} - \frac{w_i}{n})(v_{k,j} - \frac{w_k}{n}) \\ &\quad + \frac{1}{2}\sum_{i,j}\sum_{k,k':k\neq k',f(k)=f(k')=i}(v_{k,j} - \frac{w_k}{n})(v_{k',j} - \frac{w_k'}{n}). \end{aligned}$$

Noticing that in the second sum, the term $(v_{k,j} - \frac{k_i}{n})^2$ appears exactly once for each k (namely for the particular i with $f(k) = i$), we see that the first two sums are precisely equal to $\frac{1}{2}\Phi_t$. Similarly, we can simplify the fourth sum by noticing that each pair k, k' with $f(k) = f(k')$ will appear for exactly one i. So we simplify

$$\begin{aligned}
\Phi_{t+1} &= \frac{1}{2}\Phi_t + \frac{1}{2}\sum_{i,j,k}(v_{i,j} - \frac{w_i}{n})(v_{k,j} - \frac{w_k}{n}) \cdot [f(k) = i] \\
&\quad + \frac{1}{2}\sum_{j,k,k':k\neq k'}(v_{k,j} - \frac{w_k}{n})(v_{k',j} - \frac{w_k'}{n}) \cdot [f(k) = f(k')].
\end{aligned}$$

Here, we are using Iverson's convention: $[f(k) = i] := \begin{cases} 1 & \text{if } f(k) = i \\ 0 & \text{otherwise} \end{cases}$.

In this form, the expectation of Φ_{t+1} is not too difficult to evaluate: we can use linearity of expectation, and notice that the only terms actually depending on the random choices are $[f(k) = i]$ and $[f(k) = f(k')]$. As they are 0-1 random variables, their expectation is exactly equal to the probability of being 1, which can be easily seen to be $1/n$ for both. (For the first one, this is obvious; for the second one, notice that for any choice of $f(k)$, the probability that $f(k') = f(k)$ is $1/n$, so the same holds overall.) Substituting all of these, we obtain that

$$\begin{aligned}
\mathrm{E}\left[\Phi_{t+1} \mid \Phi_t = \phi\right] &= \frac{1}{2}\phi + \frac{1}{2}\sum_{i,j,k}(v_{i,j} - \frac{w_i}{n})(v_{k,j} - \frac{w_k}{n}) \cdot \mathrm{Prob}[f(k) = i] \\
&\quad + \frac{1}{2}\sum_{j,k,k':k\neq k'}(v_{k,j} - \frac{w_k}{n})(v_{k',j} - \frac{w_k'}{n}) \cdot \mathrm{Prob}[f(k) = f(k')] \\
&= \frac{1}{2}\phi + \frac{1}{2n}\sum_{i,j,k}(v_{i,j} - \frac{w_i}{n})(v_{k,j} - \frac{w_k}{n}) + \frac{1}{2n}\sum_{j,k,k':k\neq k'}(v_{k,j} - \frac{w_k}{n})(v_{k',j} - \frac{w_k'}{n}) \\
&= \frac{1}{2}\phi + \frac{1}{n}\sum_{i,j,k}(v_{i,j} - \frac{w_i}{n})(v_{k,j} - \frac{w_k}{n}) - \frac{1}{2n}\sum_{j,k}(v_{k,j} - \frac{w_k}{n})^2 \\
&= (\frac{1}{2} - \frac{1}{2n})\phi + \frac{1}{n}\sum_j(\sum_i v_{i,j} - \sum_i \frac{w_i}{n})(\sum_k v_{k,j} - \sum_k \frac{w_k}{n}) \\
&= (\frac{1}{2} - \frac{1}{2n})\phi.
\end{aligned}$$

In the last step, we used mass conservation, which implied that $\sum_i v_{i,j} = 1$, and $\sum_i w_i = n$, so that the second term actually became 0. Two steps earlier, we made the last sum run over all pairs k, k', and subtracted out the ones we had added in. We also notice that at that point, both sums are equal, so we added them up to form the first one. In summary, this proves the lemma about the conditional expectation. ■

By applying the lemma repeatedly, and using that $\Phi_0 \leq n$, we obtain that $\mathrm{E}[\Phi_t] \leq n \cdot 2^{-t}$. Thus, after $t = \log n + \log \frac{1}{\hat{\epsilon}}$ rounds, the expected potential is $\mathrm{E}[\Phi_t] \leq \hat{\epsilon}$. Markov's Inequality [4] states that for any non-negative random variable X and any value a, we have $\mathrm{Prob}[X \geq a] \leq \frac{\mathrm{E}[X]}{a}$. Applying it to Φ_t, and choosing $\hat{\epsilon} = \epsilon^2 \cdot \delta/2 \cdot 2^{-2\tau}$ thus guarantees that with probability at least $1 - \delta/2$, we have $\Phi_t \leq \epsilon^2 \cdot 2^{-2\tau}$. In particular, this bound applies to each term of the sum that constitutes Φ_t, so $|v_{t,i,j} - \frac{w_{t,i}}{n}| \leq \epsilon \cdot 2^{-\tau}$ for all i and j.

At this point, we have almost finished the proof; however, the quantity that we have just proven to be small is not quite the error measure $\Delta_{t,i}$ we are interested in. We still need to divide by $w_{t,i}$ to get exactly our error measure, and $w_{t,i}$ could potentially be quite small. This happens for instance when node i has not received a message from anyone in a while (unlikely, but possible), or when it did receive a message, it was from another node that had not received a message in a while. At this point, we can leverage the earlier analysis of the dissemination of a single message.

Look at all nodes at time $t - \tau$ (notice that our choice of t implies that $t \geq \tau$, so we are allowed to do that). At that point, at least one node $\hat{\imath}$ had weight at least 1. Consider the experiment in which this node has a "message", and look at when each node i receives the message. By our previous analysis, and, more specifically, a theorem by Frieze and Grimmett [2], after $\tau = 4 \log n + \log \frac{2}{\delta}$ steps, all nodes have received the message after τ steps with probability at least $1 - \delta/2$. Because the weight in a message is at worst halved in each round, and similarly while a node simply "holds" the message, we know that at time t, each node must have weight at least $2^{-\tau}$ with probability at least $1 - \delta/2$. Taking a Union Bound over both events considered, and dividing the earlier bound by $w_{i,t} \geq 2^{-\tau}$, we obtain that at time t, with probability at least $1 - \delta$, the errors are at most $\Delta_{i,t} = |\frac{v_{t,i,j}}{w_{t,i}} - \frac{1}{n}| \leq \epsilon$.

Finally, a simple inductive proof shows that once the error at all nodes drops below ϵ, it will stay below ϵ at all nodes deterministically, so the quality of the estimate never gets worse.

To complete the proof, we need to verify how large our t is exactly. Substituting the values of $\hat{\epsilon}$ and τ into t, we see that it is $O(\log n + \log \frac{1}{\epsilon} + \log \frac{1}{\delta})$. This completes the proof. ■

References

[1] E. Berger. Dynamic monopolies of constant size. *Journal of Combinatorial Theory Series B*, 83:191–200, 2001.

[2] A. Frieze and G. Grimmett. The shortest-path problem for graphs with random arc-lengths. *Discrete Applied Mathematics*, 10:57–77, 1985.

[3] D. Kempe, A. Dobra, and J. Gehrke. Computing aggregate information using gossip. In *Proc. 44th IEEE Symp. on Foundations of Computer Science*, pages 482–491, 2003.

[4] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1990.