Brief Announcement: Deterministic Consensus and Checkpointing with Crashes: Time and Communication Efficiency

Bogdan S. Chlebus
School of Computer and Cyber Sciences
Augusta University
Augusta, Georgia, USA

Dariusz R. Kowalski
School of Computer and Cyber Sciences
Augusta University
Augusta, Georgia, USA

Jan Olkowski
Department of Computer Science
University of Maryland
College Park, Maryland, USA

ABSTRACT

We study consensus and checkpointing in synchronous distributed systems. There are n nodes that communicate by sending messages, and any two nodes can communicate directly. The nodes are prone to crashing, with an upper bound t on the number of crashes. Algorithms use overlay networks of choice to save on the amount of communication. We explore using Ramanujan graphs as such overlay networks. We demonstrate that Ramanujan graphs have topological properties conducive to fault-tolerance and time/communication efficiency of distributed algorithms. Our consensus algorithm assumes binary input values, runs in $O(t)$ time and sends $O(n+t\log t)$ bits. The algorithm sends the optimum number $O(n)$ of bits for $t = O(n/\log n)$, thus for this range of t it improves on the algorithm by Galil, Mayer and Yung [FOCS 1995] that also sends $O(n)$ bits but works in exponential time. The consensus algorithm can be implemented such that a node sends a message to at most one node at a round while maintaining the asymptotic time and communication performance bounds. Our checkpointing algorithm runs in linear time $O(n)$ and with $O(n\log^7 n)$ messages. It improves on the most communication-efficient and time-optimal algorithm by Galil, Mayer and Yung [FOCS 1995], which may have $O(n^{1+\epsilon})$ messages sent, for any chosen constant $\epsilon > 0$.

KEYWORDS

distributed algorithm, message passing, synchrony, node crash, consensus, checkpointing, Ramanujan graph, runtime performance, communication performance

ACM Reference Format:
Bogdan S. Chlebus, Dariusz R. Kowalski, and Jan Olkowski. 2022. Brief Announcement: Deterministic Consensus and Checkpointing with Crashes: Time and Communication Efficiency. In *Proceedings of the 2022 ACM Symposium on Principles of Distributed Computing (PODC '22), July 25–29, 2022, Salerno, Italy*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3519270.3538471

PODC '22, July 25–29, 2022, Salerno, Italy.

ACM ISBN 978-1-4503-9262-4/22/07.
https://doi.org/10.1145/3519270.3538471

1 INTRODUCTION

We develop distributed algorithms for binary consensus and checkpointing. A distributed system consists of nodes that communicate by sending messages. Any two nodes can communicate directly. The communication is synchronous, in that executions of algorithms are partitioned into rounds such that a message is delivered in the round in which it is transmitted. Algorithms use overlay networks of choice to save on the amount of communication.

Each node has a unique integer name in the set $\{1,\ldots,n\}$. The number n is known to all the nodes, in that it may be a part of code of an algorithm. Nodes are prone to crashing. An upper bound on the number of crashes in an execution is an integer denoted by t. We assume that the number t is known, in that it can be used in a code of an algorithm.

If a node can multicast and receive messages to/from any set of recipients/senders in a round then this is the *multi-port model*. If a node can send/receive a message to/from at most one sender/recipient in a round then we call this the *single-port model*.

We explore using Ramanujan graphs as overlay networks in designing distributed algorithms. We demonstrate that Ramanujan graphs have topological properties conducive to fault-tolerance and time/communication efficiency of distributed algorithms.

Agreement problems. In each of the two considered agreement problems, eventually every node needs to decide on a value. Deciding can be made at any time before halting in the course of an execution. A decision is irrevocable once made. The three constraints of validity, agreement, and termination need to be satisfied in each execution.

Agreement means that no two nodes decide on different values. Termination means that each node eventually decides, unless it crashes. Validity if formulated differently for each of the problems we consider. In the problem of *consensus*, each node starts with an initial input value, which is either 0 or 1. Consensus validity means that each node decides on the initial input value of some node. In the problem of *checkpointing*, nodes work to collect knowledge about these nodes that have already crashed. Checkpointing validity means that each node decides on a set of nodes E such that a node that crashed at the start does not belong to E and every nonfaulty node does belong to E.

The previous and related work. The problem of checkpointing was introduced by De Prisco, Mayer, and Yung [14], who presented an algorithm of $O((t+1)n)$ message complexity. A checkpointing algorithm developed by Chlebus, Gasieniec, Kowalski, and Schwarzmann [5] sends $O(n(n-t))$ messages. Galil, Mayer, and Yung [17]

gave an algorithm solving checkpointing in time $O((t+1)8^{1/\epsilon})$ and with $O(n + tn^{\epsilon})$ messages, for any $\epsilon > 0$.

A consensus solution needs to send $\Omega(n)$ messages because each process is required to send at least one message. Galil, Mayer, and Yung [17] developed an algorithm that runs in $O(n^{1+\varepsilon})$ rounds, for any $0 < \varepsilon < 1$, and sends $O(n)$ messages. They also gave an algorithm for binary consensus sending $O(n)$ bits in messages, but the algorithm runs in exponential time. Chlebus and Kowalski [6] showed that consensus can be solved in $O(t + 1)$ time and with $O(n \log^2 t)$ messages under the assumption that the number $n - t$ of non-faulty nodes satisfies $n - t = \Omega(n)$. Chlebus, Kowalski, and Strojnowski [9] gave algorithms for binary consensus operating in time $O(t + 1)$ that send $O(n \log^2 n)$ bits for $t < \frac{n}{3}$ and $O(n \log^4 n)$ bits for any $t < n$.

A consensus algorithm is early-stopping if it runs in $O(f+1)$ time, where f is the number of failures actually occurring in an execution, while the algorithm may be designed for a known upper bound t on the number of crashes. Dolev, Reischuk, and Strong [16] gave an early-stopping solution for consensus with arbitrary process failures and a lower bound $\min\{t+1, f+2\}$ on the number of rounds. Coan [12] gave a consensus algorithm running in time $O(f + 1)$ that uses messages of size logarithmic in the size of the range of input values; see also Bar-Noy, Dolev, Dwork, and Strong [3] and Berman, Garay, and Perry [4]. Chlebus and Kowalski [7] developed an early stopping consensus algorithm sending $O(n \log^5 n)$ messages. Dolev and Lenzen [15] showed that any crash-resilient consensus algorithm deciding in exactly $f + 1$ rounds has $\Omega(n^2 f)$ worst-case message complexity.

Chor, Merritt, and Shmoys [11] gave a randomized algorithm for consensus that has $O(\log n)$ round complexity and $O(n^2 \log n)$ message complexity with high probability, while tolerating fewer than $\frac{n}{2}$ crashes. Bar-Joseph and Ben-Or [2] gave a randomized consensus algorithm against an adaptive adversary that controls crashes. The algorithm works in $O(\frac{\sqrt{n}}{\log n})$ expected time, which is provably optimal, while generating $O(\frac{n^{5/2}}{\log n})$ messages and communications bits. Kowalski and Mirek [21] demonstrated how to decrease the number of messages to $O(n^{3/2}$ polylog $n)$, while keeping the number of bits at $\Theta(n^{5/2}$ polylog $n)$ and slowing down the algorithm by a factor of $O(\log^2 n)$, by using deterministic fault-tolerant gossip from [7]. Chlebus and Kowalski [8] developed a randomized consensus algorithm that terminates in the expected $O(\log n)$ time and such that the expected number of bits sent and received by each process is $O(\log n)$ when the adversary is oblivious and such that a bound t on the number of crashes is a constant fraction of the number n of nodes. Gilbert and Kowalski [19] presented a randomized consensus algorithm that tolerates up to $\frac{n}{2}$ crashes and terminates in $O(\log n)$ time and sends $O(n)$ messages with high probability. Gilbert, Guerraoui, and Kowalski [18] developed an indulgent consensus algorithm, in that it solves consensus under eventual synchrony, while in synchronous executions it is early-stopping and achieves $O(n$ polylog $n)$ message complexity. Robinson, Scheideler, and Setzer [23] showed how to achieve an almost-everywhere consensus in $O(\log n)$ time with high probability against adversaries controlling crashes that are weaker than adaptive. Chlebus, Kowalski, and Strojnowski. [10] gave a quantum algorithm for binary consensus, executed by crash-prone quantum processes that operates in $O(\text{polylog } n)$ rounds while sending $O(n$ polylog $n)$ qubits against the adaptive adversary. Alistarh, Aspnes, King, and Saia [1] developed a randomized consensus algorithm for asynchronous message passing that sends the expected number of $O(nt + t^2 \log^2 t)$ messages.

For properties and construction of graphs with suitable expansion properties, see [13, 20, 22].

2 RAMANUJAN OVERLAY GRAPHS

Let $G = (V, E)$ denote a simple graph, where V is the set of vertices and E is the set of edges. For a set of vertices W, the notation $N_G^i(W)$ denotes the set of all vertices in V of distance at most i from some node in W in graph G. For two disjoint set of vertices W_1 and W_2, an edge $(v, w) \in E$ *connects* W_1 *with* W_2 if $v \in W_1$ and $w \in W_2$.

Next, we list properties of overlay graphs and their vertices relevant to efficiency of algorithms, following [9]. Let δ, γ and ℓ be positive integers and $0 < \varepsilon < 1$ be a real number.

Dense neighborhood: For a node $v \in V$, a set $S \subseteq N_G^{\gamma}(v)$ is said to be (γ, δ)-*dense-neighborhood for* v when every node in $S \cap N_G^{\gamma-1}(v)$ has at least δ neighbors in S.

Survival subset: For a set of vertices $B \subseteq V$, a subset $C \subseteq B$ is a δ-*survival subset for* B if every node's degree in the subgraph of G induced by C is at least δ.

Compactness: graph G is said to be $(\ell, \varepsilon, \delta)$-*compact* if, for any set $B \subseteq V$ of at least ℓ vertices, there is a subset $C \subseteq B$ of at least $\varepsilon\ell$ vertices that is a δ-survival subset for B.

The following property of overlay graphs is also relevant:

Expansion: graph G is ℓ-*expanding*, or is an ℓ-*expander*, if any two disjoint subsets of ℓ vertices each are connected by an edge.

For a constant d, let $G = G(n, d)$ denote a d-regular Ramanujan graph of n vertices. Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ be the eigenvalues of $G(n, d)$, and let $\lambda = \max(|\lambda_2|, |\lambda_n|)$. For a d-regular simple graph to be Ramanujan means $\lambda \leq 2\sqrt{d-1}$.

We also use the following notations:

$$\ell(n, d) = 4nd^{-1/8}, \quad \text{and} \quad \delta(d) = \frac{1}{2}\left(d^{7/8} - d^{5/8}\right).$$

THEOREM 2.1. *Every Ramanujan graph $G = G(n, d)$ is $\ell(n, d)$-expanding.*

THEOREM 2.2. *Every Ramanujan graph $G = G(n, d)$ has the property to be $(\ell(n, d), \frac{3}{4}, \delta(d))$-compact.*

THEOREM 2.3. *In a Ramanujan graph $G = G(n, d)$, a $(\gamma(n), \delta(d))$-dense-neighborhood of a vertex includes at least $\ell(n, d)$ vertices, for any $\gamma(n) \geq 2 \lg n$ and sufficiently large d.*

THEOREM 2.4. *Let $0 < \epsilon < 1$ be a fixed constant, and let A and B be two disjoint subsets of vertices of a Ramanujan graph $G = G(n, d)$. If $|A| = \epsilon \cdot n$ and $|B| > \frac{4n}{d\epsilon}$, then there exists an edge connecting A with B.*

3 CONSENSUS AND CHECKPOINTING

We give an algorithm for binary consensus that uses messages of size $O(1)$ bits. We assume for the analysis of the consensus algorithm that there is a constant α satisfying $0 < \alpha < 1$ such

that t is at most $\alpha \cdot n$. An execution takes $O(t + \log n)$ rounds and $O(n + t \log t)$ bits are transmitted in total. The efficiency of the algorithm is reflected by the property that one crash delays termination by $O(1)$ rounds, there are $O(1)$ bits transmitted per node, and $O(\log t)$ bits transmitted per crash. The optimal $O(n)$ number of bits/messages are sent as long as $t = O(\frac{n}{\log n})$. This partially answers the problem of bit-communication complexity of binary consensus posed by Galil, Mayer, and Yung [17], who showed that sending $O(n)$ bits in messages by a consensus algorithm is achievable for any bound on the number of crashes $t < n$, but their algorithm runs in a number of rounds exponential in n.

Theorem 3.1. *There exists a deterministic algorithm that solves consensus in $O(t + \log n)$ rounds sending a total of $O(n + t \log t)$ bits in messages.*

Using Ramanujan graphs of constant degrees as overlay graphs has the additional advantage that our consensus algorithm can be implemented in the single-port model with the same asymptotic time and communication performance bounds as in the multi-port model. This gives the first known consensus algorithm for the single-port model of a comparable time and message efficiency.

Theorem 3.2. *The exists an algorithm implemented in the single-port model that solves consensus in $O(t + \log n)$ rounds sending a total of $O(n + t \log t)$ bits in messages.*

We give a checkpointing algorithm working in linear time $O(n)$ and sending a nearly-optimal number of messages $O(n \log^7 n)$. This improves on the most message-efficient time-optimal solution previously known by Galil, Mayer, and Yung [17] by a polynomial factor.

Theorem 3.3. *There exists a deterministic algorithm that solves checkpointing in $O(n)$ rounds using $O(n \log^7 n)$ messages for any number of crashes $t < n$.*

4 DISCUSSION AND OPEN PROBLEMS

An immediately occurring question regarding consensus with crashes that follows from this work asks whether the component $\Theta(t \log t)$ in the communication bit complexity bound $O(n + t \log t)$ could be decreased. This question is open and applies to both multi-port and single-port settings. The message complexity of the time-optimal checkpointing algorithm given in this work may miss optimality by a poly-logarithmic factor in the multi-port model. The time performance of the message-optimal algorithm for checkpointing given by Galil, Mayer, and Yung [17] may miss time optimality by a polynomial factor in the multi-port model. Resolving simultaneous optimality with respect to time and message complexities of checkpointing is thus open. We conjecture that algorithms of comparable asymptotic performance as in the multi-port model could be implemented in the single-port model.

This work demonstrates that Ramanujan graphs can be used as overlay networks to structure communication for time and communication efficient algorithms solving consensus and checkpointing with nodes prone to crashes. We expect that the newly discovered properties of Ramanujan graphs could be applied to streamline algorithms for other problems in distributed computing and communication, including gossiping, counting, and majority consensus, with respect to time and communication efficiency.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 2131538.

REFERENCES

[1] Dan Alistarh, James Aspnes, Valerie King, and Jared Saia. 2018. Communication-efficient randomized consensus. *Distributed Computing* 31, 6 (2018), 489–501.
[2] Ziv Bar-Joseph and Michael Ben-Or. 1998. A tight lower bound for randomized synchronous consensus. In *Proceedings of the 17th ACM Symposium on Principles of Distributed Computing (PODC'98)*. 193–199.
[3] Amotz Bar-Noy, Danny Dolev, Cynthia Dwork, and H. Raymond Strong. 1992. Shifting gears: Changing algorithms on the fly to expedite Byzantine agreement. *Information and Computation* 97, 2 (1992), 205–233.
[4] Piotr Berman, Juan A. Garay, and Kenneth J. Perry. 1992. Optimal early stopping in distributed consensus (extended abstract). In *Proceedings of the 6th International Workshop on Distributed Algorithms (WDAG'92) (Lecture Notes in Computer Science, Vol. 647)*. Springer, 221–237.
[5] Bogdan S. Chlebus, Leszek Gasieniec, Dariusz R. Kowalski, and Alexander A. Schwarzmann. 2017. Doing-it-All with bounded work and communication. *Information and Computation* 254 (2017), 1–40.
[6] Bogdan S. Chlebus and Dariusz R. Kowalski. 2006. Robust gossiping with an application to consensus. *Journal of Computer System and Sciences* 72, 8 (2006), 1262–1281.
[7] Bogdan S. Chlebus and Dariusz R. Kowalski. 2006. Time and communication efficient consensus for crash failures. In *Proceedings of the 20th International Symposium on Distributed Computing (DISC'06) (Lecture Notes in Computer Science, Vol. 4167)*. Springer, 314–328.
[8] Bogdan S. Chlebus and Dariusz R. Kowalski. 2009. Locally scalable randomized consensus for synchronous crash failures. In *Proceedings of the 21st ACM Symposium on Parallelism in Algorithms and Architectures (SPAA'09)*. ACM, 290–299.
[9] Bogdan S. Chlebus, Dariusz R. Kowalski, and Michał Strojnowski. 2009. Fast scalable deterministic consensus for crash failures. In *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC'09)*. ACM, 111–120.
[10] Bogdan S. Chlebus, Dariusz R. Kowalski, and Michał Strojnowski. 2010. Scalable quantum consensus for crash failures. In *Proceedings of the 24th International Symposium on Distributed Computing (DISC'10) (Lecture Notes in Computer Science, Vol. 6343)*. Springer, 236–250.
[11] Benny Chor, Michael Merritt, and David B. Shmoys. 1989. Simple constant-time consensus protocols in realistic failure models. *J. ACM* 36, 3 (1989), 591–614.
[12] Brian A. Coan. 1993. Efficient agreement using fault diagnosis. *Distributed Computing* 7, 2 (1993), 87–98.
[13] Giuliana Davidoff, Peter Sarnak, and Alain Valette. 2003. *Elementary Number Theory, Group Theory, and Ramanujan Graphs*. Cambridge University Press.
[14] Roberto De Prisco, Alain J. Mayer, and Moti Yung. 1994. Time-optimal message-efficient work performance in the presence of faults. In *Proceedings of the 13th ACM Symposium on Principles of Distributed Computing (PODC'94)*. 161–172.
[15] Danny Dolev and Christoph Lenzen. 2013. Early-deciding consensus is expensive. In *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC'13)*. ACM, 270–279.
[16] Danny Dolev, Rüdiger Reischuk, and H. Raymond Strong. 1990. Early stopping in Byzantine agreement. *Journal of the ACM* 37, 4 (1990), 720–741.
[17] Zvi Galil, Alain J. Mayer, and Moti Yung. 1995. Resolving message complexity of Byzantine agreement and beyond. In *Proceedings of the 36th IEEE Symposium on Foundations of Computer Science (FOCS'95)*. IEEE, 724–733.
[18] Seth Gilbert, Rachid Guerraoui, and Dariusz R. Kowalski. 2007. On the message complexity of indulgent consensus. In *Proceedings of the 21st International Symposium on Distributed Computing (DISC'07) (Lecture Notes in Computer Science, Vol. 4731)*. Springer, 283–297.
[19] Seth Gilbert and Dariusz R. Kowalski. 2010. Distributed agreement with optimal communication complexity. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'10)*. SIAM, 965–977.
[20] Shlomo Hoory, Nathan Linial, and Avi Wigderson. 2006. Expander graphs and their applications. *Bull. Amer. Math. Soc.* 43, 4 (2006), 439–561.
[21] Dariusz R. Kowalski and Jaroslaw Mirek. 2019. On the complexity of fault-tolerant consensus. In *Revised Selected Papers from the 7th International Conference on Networked Systems (NETYS'19) (Lecture Notes in Computer Science, Vol. 11704)*. Springer, 19–31.
[22] Mike Krebs and Anthony Shaheen. 2011. *Expander Families and Cayley Graphs: a beginner's guide*. Oxford University Press.
[23] Peter Robinson, Christian Scheideler, and Alexander Setzer. 2018. Breaking the $\tilde{\Omega}(\sqrt{n})$ Barrier: Fast consensus under a late adversary. In *Proceedings of the 30th Symposium on Parallelism in Algorithms and Architectures (SPAA'18)*. ACM, 173–182.