Termination of amnesiac flooding

Walter Hussak[1] · Amitabh Trehan[2]

Abstract

We consider a stateless 'amnesiac' variant of the stateful distributed network flooding algorithm, expanding on our conference papers [PODC'19, STACS'20]. Flooding begins with a set of source 'initial' nodes I seeking to broadcast a message M in rounds, in a network represented by an undirected graph (G, E) with set of nodes G and edges E. In every round, nodes with M send M to all neighbours from which they did not receive M in the previous round. Nodes do not remember earlier rounds, raising the possibility that M circulates indefinitely. Stateful flooding overcomes this by nodes recording every message circulated and ignoring M if received again. This overhead was assumed to be necessary. We show that almost optimal broadcast can still be achieved without this overhead. We prove that amnesiac flooding terminates on every finite graph and derive sharp bounds for termination times. Define (G, E) to be *I-bipartite* if the quotient graph, contracting all nodes in I to a single node, is bipartite. We prove that, if d is the diameter and $e(I)$ the eccentricity of the set I, flooding terminates in $e(I)$ rounds if (G, E) is I-bipartite and j rounds with $e(I) < j \leq e(I) + d + 1 \leq 2d + 1$ if (G, E) is non I-bipartite. The separation in the termination times can be used for distributed discovery of topologies/distances in graphs. Termination is guaranteed if edges are lost during flooding but not, in general, if there is a delay at an edge. However, the cases of single-edge fixed delays of duration τ rounds in single-source bipartite graphs terminate by round $2d + \tau - 1$, and all cases of multiple-edge fixed delays in multiple-source cycles terminate.

Keywords Flooding algorithm · Network algorithms · Graph bipartiteness · Process termination · Communication · Broadcast

1 Introduction

Flooding is one of the most fundamental of all graph/network algorithms - for example, to achieve *broadcast* in networks [2, 14, 23, 25, 27, 29, 30]. The algorithm that is usually implemented is: (i) sources send a message to all their neighbours, (ii) on first receipt, receiving nodes send the message to every neighbour from which they did not receive and (iii) on subsequent receipts, the receiving node does not send the message to any neighbour. This algorithm checks if the message has been received before, to ensure termination. It requires each node records a previous event in its state, making it a *stateful* protocol. We consider a stateless version of flooding, i.e.

✉ Walter Hussak
   w.hussak@lboro.ac.uk

   Amitabh Trehan
   amitabh.trehan@durham.ac.uk

1   Computer Science, Loughborough University, Epinal Way,
    Loughborough, Leics LE11 3TU, UK

2   Computer Science, Durham University, Stockton Road,
    Durham DH1 3LE, UK

algorithms that are 'amnesiac' in the sense that they hold no record or memory of any event beyond the immediate round. The algorithm is as follows.

Definition 1 Amnesiac flooding algorithm. Let (G, E) be an undirected graph, with vertices G and edges E (representing a network where the vertices represent the nodes of the network and edges represent the connections between the nodes). Computation proceeds in synchronous 'rounds' where each round consists of nodes receiving messages being sent from neighbours. A receiving node sends messages to some neighbours in the next round. No messages are lost in transit. The algorithm is defined by the following conditions.

(i) All nodes from a subset of sources or *initial nodes* $I \subseteq G$ send a message M to all of their neighbours in round 1.

(ii) In subsequent rounds, every node that received M from a neighbour in the previous round, sends M to all, and only those, nodes from which it did not receive M. Flooding *terminates* when M is no longer sent to any node in the network.

Hitherto, it was not known whether a stateless flooding algorithm exists that terminates on every graph. In this paper, we show that amnesiac flooding is itself such an algorithm. If e and d are the eccentricity of g_0 and diameter of (G, E) respectively, the termination times for a single source g_0 are e rounds if (G, E) is bipartite and j rounds with $e < j \leq e + d + 1$ otherwise. These times are asymptotically time optimal. The difference in times between bipartite and non-bipartite graphs means that a stateless algorithm can approximate the topology or diameter/eccentricities of a network in certain cases.

Gopal, Gopal and Kutten [14] highlighted the memory requirement inherent in stateful flooding and its overhead on fast networks. For example, when multiple messages are being flooded, nodes need to keep a record of the messages they have seen and search through their data structure whenever a message arrives before deciding if it can be further flooded. This creates a bottleneck, in particular when the transmission links are fast. Either specialised high speed hardware has to be used for the search, or alternate flooding protocols that limit the memory overhead (as they do in their paper) have to be used. This is also a limitation for low memory devices, e.g. sensor networks. Since amnesiac flooding is stateless using no memory beyond the present round, it does not incur this memory overhead and bottleneck. One advantage of the stateful variant of flooding is that it can be used to construct spanning structures, such as spanning trees, which can be subsequently re-used for efficient communication [2, 22, 27, 31]. It is not easy to maintain these structures in dynamic conditions, i.e. when edges/nodes are being added or removed. We show that even when edges are being arbitrarily or adversarially removed from the graph, amnesiac flooding will still terminate. However, termination is not guaranteed in asynchronous models where the delivery time of messages on edges can vary - we show this with a non-terminating delivery schedule for the simplest of cycles. Neither is it guaranteed for general graphs even if there is a fixed delivery delay at only one edge.

This paper is structured as follows. The rest of this section looks at related work. In Sect. 2, we prove that if a message is flooded from a set I of initial nodes in round 1, all nodes receive the message at most twice and hence amnesiac flooding terminates (Theorem 2.1). In Sect. 3, this result is used to derive sharp time bounds for the number of rounds it takes for flooding to terminate. As mentioned above, in the case of a single initial node g_0, flooding terminates in e rounds if (G, E) is bipartite and j rounds with $e < j \leq e + d + 1$ if (G, E) is not bipartite. The difference in the time bounds rests on the absence of neighbouring nodes (called 'ec-nodes' here) equidistant from the initial node g_0 in bipartite graphs. For general sets of initial nodes I, we define *I-bipartite* graphs as those which do not have any neighbouring nodes equidistant from the set of initial nodes I and show

that the difference in termination times between bipartite and non bipartite graphs for single initial nodes g_0 carries over to a similar difference in termination times between I-bipartite and non I-bipartite graphs for general I. For I-bipartite graphs termination occurs in round $e(I)$ (Theorem 3.4), where $e(I)$ is the eccentricity of I, i.e. the furthest distance of a node from the set I. The main theorem of the paper is for non I-bipartite graphs, where termination is shown to occur in a number of rounds which lies in the range between $e(I) + 1$ and $e(I) + d + 1$:

(Theorem 3.7) *Let (G, E) be a graph that is not I-bipartite and let d be the diameter. Then, flooding terminates in round j where j is in the range*

$$e(I) < j \leq e(I) + d + 1.$$

In the case where there is more than one initial node in I, termination of non I-bipartite graphs is achieved at the $e(I) + 1$ lower bound if all nodes in I have neighbouring nodes in I (Theorem 3.8). In Sect. 4 we provide some extensions of synchronous amnesiac flooding to cases where message transit along an edge may take more than one round. We show that termination of amnesiac flooding is resilient to loss of edges or vertices during the flooding process (Theorem 4.1), but not, in general, to delays even at a single edge. However, if the flooding is from a single source then the termination result for bipartite graphs extends to the case where there is a fixed delay of message transit of τ rounds at a single edge, with the termination times more like the the non-bipartite graph termination times with no delays:

(Theorem 4.2) *Let (G, E) be a bipartite graph, $g_0 \in G$ be an initial node, $e = \{x, y\} \in E$ be an edge, and τ be an integer greater than 1. Then flooding initiated from g_0, in which a message in transit along e takes τ rounds and 1 round along any other edge, visits every node at most twice and completes in not more than $2d + \tau - 1$ rounds where d is the diameter of (G, E).*

Amnesiac flooding from multiple sources in even or odd cycles with any number of possibly different fixed delays at edges, terminates:

(Theorem 4.3) *Let (G, E) be an edge-weighted cycle in which (possibly different) positive integer edge weights represent duration of message transit in rounds along edges and $I \subseteq G$ be a set of initial nodes. Then, flooding initiated from initial nodes in I terminates in not more than the weight σ of (G, E) number of rounds, where the weight σ of (G, E) is the sum of the weights of the edges.*

However, if the duration of message transit along an edge is variable, as in the case of an asynchronous model of amnesiac flooding, then flooding is not guaranteed to terminate even on a cycle graph. Section 5 is the concluding section and contains a number of open problems.

Related work

The proof of termination in Sect. 2 of this paper appeared as a conference brief announcement by Hussak and Trehan in PODC 2019 [15]. The proof of separate time bounds for termination in bipartite and non-bipartite graphs in the case of a single initial node appeared in STACS 2020 [17]. This paper is an extended version of the two conference papers, generalizing the previous results to the case of a set I of multiple initial nodes, with a structural characterization in terms of certain quotient 'I-bipartite' graphs of where the corresponding separation of time bounds occurs. We also show sharpness of the bounds for the time to termination. Further extensions and also negative results are given in this paper on termination of amnesiac flooding in which there are delays of message transit along edges. Another study by Hussak and Trehan [16] considers amnesiac flooding with different messages being simultaneously flooded under various different forwarding rules when messages are received at nodes.

Flooding-based algorithms (or flooding protocols) appear in areas ranging from GPUs, high performance, shared memory and parallel computing, to mobile ad-hoc networks (MANETs), Mesh Networks, Complex Networks etc [30]. Rahman et al [28] show that flooding can even be adopted as a reliable and efficient routing scheme, comparable to sophisticated point-to-point forwarding schemes, in some ad-hoc wireless mobile network applications.

There are numerous applications of flooding as a distributed algorithm. It is often used to set up spanning trees [2, 22, 27, 31] and for solving various distributed computing problems such as *leader election*. For example, the time-optimal leader election of Peleg [26] builds on a variant of flooding which also does not check previously received messages to detect termination. The algorithm does continuous flooding of node IDs to determine the node in the network with the highest ID while using either the knowledge of the graph diameter or a condition on the node IDs to achieve and detect termination. Kutten et al. [20], on the other hand, use controlled stateful flooding and resulting spanning trees and structures for deterministic and randomised efficient time and message leader election.

Statelessness has always been a desirable property and stateless algorithms have been developed in various contexts. Adamek, Nesterenko, Robinson, Tixeuil [1] and Karp and Kung [18] offer stateless (flooding and other) solutions for routing. Adamek et al. provide the first known stateless geocast algorithms. One of these is based on flooding itself and also permits converting previous controlled flooding solutions to be stateless. Awerbuch and Khandekar developed stateless solutions for multi-commodity flows [3]. They also developed a framework for distributed and stateless solutions for packing and covering linear programs by multiple cooperating but uncoordinated agents [4] . In these, they list a number of desirable properties of statelessness (especially in this context): i) Self-stabilization: recovery from adversarial 'hard' crashes, ii) Incremental and Local adjustment: easily adjusts locally to changes, and iii) Allows asynchrony (in their context).

In [11], Dolev, Erdmann, Lutz, Schapira, and Zair, propose a model of stateless computation in which nodes have no internal state, receive external inputs and compute a global function purely on the basis of the inputs they receive from neighbours connected via directed links. They then discuss self-stabilization in this model and also the power of stateless computation proving that stateless computation is sufficiently powerful to solve a nontrivial range of computational manners in a resilient manner with small message lengths.

Note that amnesiac flooding uses rules that avoid forwarding to the most recently chosen nodes - similar ideas have been used before in distributed computing, e.g. in social networks [10] and broadcasting [12]. Processes such as random walks [8, 9, 13, 20, 21] and their deterministic variant Rotor-Router (or Propp) machine [7, 19] can also be seen as restricted variants of flooding. Finally, in some models such as population protocols, the low memory makes termination very difficult to achieve leading to research that tries to provide termination e.g. [24]. Like population protocols, amnesiac flooding is a very simple and natural algorithm - it would be interesting to investigate if there are processes in nature close to amnesiac flooding.

Since our results on amnesiac flooding [15–17], there have been a number of follow-up works by various researchers. In a series of works by Turau [32–35], proofs of termination for non-bipartite graphs work with the bipartite graph (called the 'auxiliary' graph) that is obtained by taking two copies of the given non-bipartite graph minus edges connecting ec-nodes, and creating edges between the different copies of pairs of ec-nodes. So, a message is received once by each copy of a node in the auxiliary graph rather than twice in the original graph. In [33], Turau provides a simple extension to reduce termination time in non-bipartite graphs by sending the initial broadcast message in two consecutive rounds (though this now requires the initiators to remember, in round 2, the message and their state of round 1). In [32, 33], the 'k-flooding problem' is posed which aims to find a set S of size k such that amnesiac flooding when started concurrently by all nodes of S terminates in the least number of rounds. This problem is shown to be NP-Hard and bounds on approximation are given. The possibility of modifying the message to achieve termination in asynchronous networks is also discussed.

Bayramzadeh, Kshemkalyani, Molla and Sharma [5, 6] consider the variants of simultaneous flooding of multiple different messages that we suggest in [16]. They propose a variant of amnesiac flooding that is 'weak' in the sense that the nodes have the knowledge of the diameter of the

network and use that knowledge to achieve termination in one of the variants we show to be non-terminating in our original amnesiac flooding with simultaneous different messages.

2 Termination

Throughout this paper, we assume that graphs are finite, undirected and connected. In keeping with the papers by Hussak and Trehan [15] and [17] we shall refer to graphs (G, E) rather than $G = (V, E)$ and talk of 'nodes' $g \in G$ rather than 'vertices' $v \in V$. Terminology regarding paths and cycles is standard. A *walk* differs from a path in that nodes (vertices) may be visited more than once. A *closed walk* is a walk whose start and end nodes are the same. By 'flooding' we will implicitly mean the amnesiac flooding of Definition 1. In this section we prove that (amnesiac) flooding initiated from any set of initial nodes $I \subseteq G$, all at once, terminates.

Definition 2 Let (G, E) be the graph and $I \subseteq G$ a set of initial nodes. The *round-sets* $R_0, R_1, \ldots$ are defined as:

R_0 is the set I of initial nodes,
R_i is the set of nodes receiving a message in round i
    $(i \geq 1)$.

Clearly, if $R_j = \emptyset$ for some $j \geq 0$, then $R_i = \emptyset$ for all $i \geq j$. Flooding *terminates* in round j if $R_j \neq \emptyset$ and $R_i = \emptyset$ for all $i > j$.

Theorem 2.1 *Any node $g \in G$ is contained in at most two distinct round-sets.*

Proof Define $\mathcal{R}$ to be the set of finite sequences of consecutive round-sets of the form:

$$\underline{R} = R_s, \ldots, R_{s+d}, \ where \ s \geq 0, \ d > 0$$
$$and \ R_s \cap R_{s+d} \neq \emptyset. \tag{1}$$

In (1), s is the *start-point* $s(\underline{R})$ and d is the *duration* $d(\underline{R})$ of $\underline{R}$. Note that, a node $g \in G$ belonging to R_s and R_{s+d} may also belong to other R_i in (1). If a node $g \in G$ occurs in three different round-sets R_{i_1}, R_{i_2} and R_{i_3}, then at least one of the durations between R_{i_1} and R_{i_2}, R_{i_2} and R_{i_3} or R_{i_1} and R_{i_3} will be even. Consider the subset $\mathcal{R}^e$ of $\mathcal{R}$ of sequences of the form (1) where d is even. To prove that no node is in three round-sets, it suffices to prove that $\mathcal{R}^e$ is empty.

We assume that $\mathcal{R}^e$ is non-empty and derive a contradiction. Let $\mathcal{R}^e_{min}$ be the subset of $\mathcal{R}^e$ comprising sequences of minimum (even) duration $\hat{d}$, i.e.

$$\mathcal{R}^e_{min} = \{\underline{R} \in \mathcal{R}^e : \forall \ \underline{R}' \in \mathcal{R}^e . d(\underline{R}') \geq d(\underline{R}) = \hat{d}\} \tag{2}$$

Clearly, if $\mathcal{R}^e$ is non-empty then so is $\mathcal{R}^e_{min}$. Let $\underline{R}^* \in \mathcal{R}^e_{min}$ be the sequence with earliest start-point $\hat{s}$, i.e.

$$\underline{R}^* = R_{\hat{s}}, \ldots, R_{\hat{s}+\hat{d}} \tag{3}$$

where

$$\forall \ \underline{R}' \in \mathcal{R}^e_{min} . s(\underline{R}') \geq s(\underline{R}^*) = \hat{s} \tag{4}$$

By (1), there exists $g \in R_{\hat{s}} \cap R_{\hat{s}+\hat{d}}$. Choose node g' which sends a message to g in round $\hat{s} + \hat{d}$. As g' is a neighbour of g, either g' sends a message to g in round $\hat{s}$ or g sends a message to g' in round $\hat{s} + 1$. We show that each of these cases leads to a contradiction.

Case (i) g' sends a message to g in round $\hat{s}$
In this case, either round $\hat{s} - 1$ is round 0 and g' is an initial node in I, or g' received a message in round $\hat{s} - 1$. Thus, the sequence

$$\underline{R}^{*'} = R_{\hat{s}-1}, R_{\hat{s}}, \ldots, R_{\hat{s}+\hat{d}-1},$$
$$where \ g' \in R_{\hat{s}-1} \cap R_{\hat{s}+\hat{d}-1}, \tag{5}$$

has $d(\underline{R}^{*'}) = (\hat{s} + \hat{d} - 1) - (\hat{s} - 1) = \hat{d}$ which is even and so $\underline{R}^{*'} \in \mathcal{R}^e_{min}$. As $\underline{R}^{*'} \in \mathcal{R}^e_{min}$, by (4)

$$s(\underline{R}^{*'}) \geq s(\underline{R}^*) \tag{6}$$

But, from (5), $s(\underline{R}^{*'}) = \hat{s} - 1$ and, from (4), $s(\underline{R}^*) = \hat{s}$. Thus, by (6),

$$\hat{s} - 1 = s(\underline{R}^{*'}) \geq s(\underline{R}^*) = \hat{s}$$

which is a contradiction.
Case (ii) g sends a message to g' in round $\hat{s} + 1$
By the definition of $\mathcal{R}^e$, the smallest possible value of $\hat{d}$ is 2. However, it is not possible to have $\hat{d} = 2$ in this case as then

$$\underline{R}^* = R_{\hat{s}}, R_{\hat{s}+1}, R_{\hat{s}+2}$$

and g sends a message to g' in round $\hat{s} + 1$. As we chose g' to be such that g' sends a message to g in round $\hat{s} + \hat{d} = \hat{s} + 2$ this cannot happen because g cannot send a message to g' and g' to g in consecutive rounds by the definition of flooding. So,

$$\underline{R}^* = R_{\hat{s}}, R_{\hat{s}+1}, \ldots, R_{\hat{s}+\hat{d}-1}, R_{\hat{s}+\hat{d}}$$

where $\hat{s} + 1 < \hat{s} + \hat{d} - 1$. Consider the sequence

$$\underline{R}^{*''} = R_{\hat{s}+1}, \ldots, R_{\hat{s}+\hat{d}-1} \tag{7}$$

As g' receives a message from g in round $\hat{s} + 1$ and g' sends a message to g in round $\hat{s} + \hat{d}$, it is clear that $g' \in R_{\hat{s}+1} \cap$

$R_{\hat{s}+\hat{d}-1}$. Thus, $\underline{R}^{*''} \in \mathcal{R}$. As $\hat{d}$ is even, so is $(\hat{s} + \hat{d} - 1) - (\hat{s} + 1) = \hat{d} - 2$ and therefore $\underline{R}^{*''} \in \mathcal{R}^e$. Now, $\underline{R}^* \in \mathcal{R}^e_{min}$ and so, as $\underline{R}^{*''} \in \mathcal{R}^e$, we have, by (2),

$$d(\underline{R}^{*''}) \geq d(\underline{R}^*) \tag{8}$$

As $d(\underline{R}^{*''}) = \hat{d} - 2$ from (7) and $d(\underline{R}^*) = \hat{d}$ from (3), we have, by (8),

$$\hat{d} - 2 = d(\underline{R}^{*''}) \geq d(\underline{R}^*) = \hat{d}$$

This contradiction completes the proof.   □

Definition 3 Given $g \in G$, we use a superscript 1 to indicate that g belongs to a round-set for the first time, and a superscript 2 to indicate that it belongs to a round-set for the second time, i.e.

$$g^1 \in R_j$$

means that

$$g \in R_j \quad and \quad g \notin R_i \quad for \ all \ i \ with \ 0 \leq i < j,$$

and

$$g^2 \in R_j$$

means that

$$g \in R_j \quad and \quad g \in R_i \quad for \ some \ i \ with \ 0 \leq i < j.$$

Theorem 2.1 implies that $R_i = \emptyset$ for $i \geq 2|G|$, where $|G|$ is the number of vertices of (G, E), and therefore flooding always terminates. In the next section we give sharp bounds for the number of rounds to termination, in terms of the eccentricity of the set of initial nodes and the diameter of (G, E).

3 Time to termination

The question of termination of flooding is non-trivial when cycles are present in (G, E). The simple cases when (G, E) is an even cycle and when (G, E) is an odd cycle, as in Fig. 1 (where an arrow indicates a message received by a node in the given round), have quite different termination behaviours. The even cycle terminates remotely from the initial node g_0, after round e where e is the eccentricity of g_0 in (G, E). On the other hand, flooding on the odd cycle returns a message to the initial node g_0 and terminates after round $2e + 1 = e + d + 1$ resulting in a longer flooding process than the even cycle despite having fewer nodes and a smaller value of e. In this section, we show that these observations can
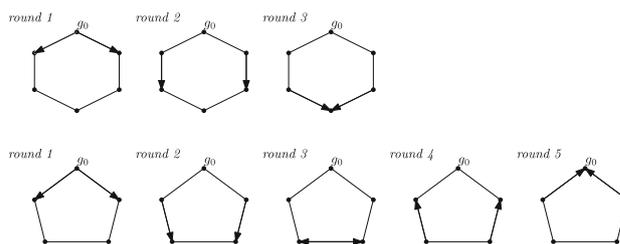

Fig. 1 Flooding in even and odd cycles

in a certain way be generalized to arbitrary graphs. In the previous section we showed that messages can be received at most twice. Observe that in the even cycle every node receives the message only once, whereas in the odd cycle every node receives the message twice. In general graphs, this difference is due to the absence or presence of neighbouring nodes that are equidistant from the set of initial nodes I, called 'ec-nodes' below. In this section, by considering the BFS layers around the initial set I, we show that in graphs that have no ec-nodes, the message just travels once forward through the BFS layers in $e(I)$ rounds, where $e(I)$ is the 'eccentricity' of I, i.e. the largest distance of a node from the set I. These are the bipartite graphs in the case of I being a single node (Lemma 3.3), and so we call them 'I-bipartite' graphs for general I. They terminate in round $e(I)$ and non I-bipartite graphs terminate no sooner than in round $e(I) + 1$ (Theorem 3.4). In graphs that have ec-nodes the message will have a one round bounce back through ec-nodes in the same BFS layer, and then the whole process will terminate within an additional diameter number of rounds, as nodes adjacent to a node receiving the message for the second time will receive the message for the second time in an adjacent round (Lemma 3.5). Hence the $e(I) + d + 1$ upper bound for graphs that are not I-bipartite (Theorem 3.7). The lower $e(I) + 1$ bound for termination of non I-bipartite graphs is sharp and is achieved if all nodes in I have neighbours in I (Theorem 3.8).

Definition 4 Let (G, E) be a graph with vertex set G and edge set E and let $I \subseteq G$ be a set of initial nodes. We will use the following definitions.

(i) For each $j \geq 0$, the distance set D_j will denote the set of points which are a distance j from I. i.e.

$$D_j = \{g \in G : \delta(I, g) = j\},$$

where, for any $g \in G$,

$$\delta(I, g) = min\{\delta(g_0, g) : g_0 \in I\}$$

and δ denotes the usual graph distance function for graph (G, E).

(ii) A node $g \in G$ is an *equidistantly-connected node*, abbreviated *ec-node*, iff it has a neighbour g' equidistant from I, i.e. iff there exists $g' \in G$ such that $\delta(I, g) = \delta(I, g')$ and $\{g, g'\} \in E$. The set of ec-nodes in G is denoted G_{ec}.

(iii) The *eccentricity* of I, denoted $e(I)$, is defined as

$$e(I) = max\{\delta(I, g) \; : \; g \in G\}$$

(iv) The graph (G, E) is *I-bipartite* iff (G, E) has no ec-nodes. Equivalently, (G, E) is *I-bipartite* iff the quotient graph of (G, E), in which the nodes of I are contracted to a single node, is bipartite.

Note that neighbouring initial nodes $g_0, g'_0 \in I$ are ec-nodes as they are both a distance 0 from I. We have the following basic properties of distance sets D_j and ec-nodes.

Lemma 3.1 *Let (G, E) be a graph and $I \subseteq G$ be a set of initial nodes.*

(i) *For all $j \geq 0$ and $i > j$, $D_j \subseteq R_j$ and $R_j \cap D_i = \emptyset$.*

(ii) *For all $j \geq 0$, $g \in D_j$ and $g' \in D_{j+1}$ such that g and g' are neighbours, g sends a message to g' in round $j + 1$, i.e. all nodes at a distance j from I send to all their neighbours which are a distance $j + 1$ in round $j + 1$.*

(iii) *If $j \geq 0$ and $g \in D_j$ is an ec-node, then $g^2 \in R_{j+1}$.*

Proof For (i), clearly every node at a distance j from I receives a message in round j and so $D_j \subseteq R_j$. Furthermore, every message received in round j will have travelled at most j edges away from I and so could not have reached a node which is at a distance $i > j$ from I. Thus, $R_j \cap D_i = \emptyset$.

For (ii), we note that the only circumstance in which a node g in D_j ($\subseteq R_j$ by (i)) does not send to a neighbour g' in D_{j+1} in round $j + 1$ is if g sent a message to g' in round j. This would need g to be in the round-set R_{j-1}, i.e. $g \in R_{j-1} \cap D_{j+1}$ which contradicts (i) which has $R_{j-1} \cap D_{j+1} = \emptyset$ as $j + 1 > j - 1$.

For (iii), if $j \geq 0$ and $g \in D_j$ is an ec-node, then by Definition 4(ii) there is an point g' equidistant from I, i.e. $g' \in D_j$ such that g and g' are neighbours. If $j = 0$ then $g, g' \in D_0 (= I \subseteq R_0)$ and send a message to each other in round 1 and so $g, g' \in R_1$. This means that $g^2 \in R_1$, as g is an initial node and $g^1 \in R_0$. If $j \geq 1$ then, by (i) of this lemma, $D_j \subseteq R_j$ and so both g and g' receive messages in round j. Also, by (i), neither sends a message in round j as $R_{j-1} \cap D_j = \emptyset$. Thus, g and g' send messages to each other in round $j + 1$. As this will be the second time they receive a message we have that $g^2 \in R_{j+1}$. $\square$

All nodes in a graph without ec-nodes, belong to at most one round-set.

Lemma 3.2 *Let (G, E) be a graph and $I \subseteq G$ be a set of initial nodes. Then (G, E) has an ec-node if and only if (G, E) has a node that is in two round-sets.*

Proof Suppose that (G, E) has no ec-nodes. Assume, on the contrary, that (G, E) has nodes that appear in two round-sets. Let R_j ($j \geq 1$) be the earliest round which contains a node g such that $g^2 \in R_j$ and $h \in R_{j-1}$ be a neighbour of g which sends to g in round j, so that $h^1 \in R_{j-1}$. Then, $h \in D_i$ for some $i \geq 0$ and $h^1 \in R_i$ by Lemma 3.1 (i). Thus, $i = j - 1$ and so $g^2 \in R_{i+1}$. As g is a neighbour of h, $g \in D_i$, D_{i+1}, or D_{i-1}. If $g \in D_i$ then g and h are ec-nodes contrary to our supposition that (G, E) has no ec-nodes. If $g \in D_{i+1}$ then $g^1 \in R_{i+1}$ by Lemma 3.1 (i), which is contrary to the assertion that $g^2 \in R_j = R_{i+1}$. If $g \in D_{i-1}$ then $g \in R_{i-1}$, by Lemma 3.1(i), and so $g^1 \in R_{i-1}$ as $g^2 \in R_{i+1}$. By Lemma 3.1 (ii), g sends to h in round $i = j - 1$. This is contrary to h sending to g in round j. Thus, our assumption that (G, E) has nodes that appear in two round-sets is false.

Conversely, suppose that (G, E) has an ec-node g, $g \in D_j$ ($\subseteq R_j$ by Lemma 3.1 (i)) say where $j \geq 1$. Then $g^2 \in R_{j+1}$ by Lemma 3.1 (iii). $\square$

In the case of a single initial node, bipartite graphs do not have any ec-nodes.

Lemma 3.3 *Let (G, E) be a graph and suppose that I contains a single initial node g_0. Then, (G, E) is bipartite iff it has no ec-nodes.*

Proof It is easy to see that nodes equidistant from the initial node g_0 must belong to the same partite set. A graph is bipartite iff no edge connects two such nodes, and this is the case iff (G, E) has no ec-nodes by Definition 4 (ii). $\square$

From Lemma 3.2 we see that nodes in graphs without ec-nodes only receive a message once. For these graphs the time to termination is the number of rounds taken for the message to reach the most distant points from the set of initial nodes I. This is just the eccentricity of I, $e(I)$.

Theorem 3.4 *Let (G, E) be a graph and $I \subseteq G$ be a set of initial nodes with eccentricity $e(I)$. Then, flooding terminates in round $e(I)$ if and only if (G, E) is I-bipartite*

Proof

(G, E) is I-bipartite
implies (G, E) has no ec-nodes (by Definition 4(iv))
implies no node appears in 2 round-sets (by Lemma 3.2)
implies furthest nodes get message last
implies $R_{e(I)}$ is the last non-empty round-set
(by Lemma 3.1(i)).

The converse is also true. If flooding terminates in round $e(I)$, then the furthest nodes will receive the message for the first

and only time in that round. Thus, (G, E) has no ec-nodes by Corollary 3.6 (proved below independently of this theorem) and is I-bipartite. $\quad\square$

To find the time to termination in general graphs we need to find a bound on when nodes can belong to a round-set for the second time. As nodes can only belong to at most two round-sets, by Theorem 2.1, this will give a bound for termination of flooding in general graphs. The following lemma shows that nodes adjacent to a node receiving the message for the second time will receive the message for the second time in an adjacent round.

Lemma 3.5 *Let* (G, E) *be a graph and* $I \subseteq G$ *a set of initial nodes. If* $h \in G$ *and* $h^2 \in R_j$ *for some* $j \geq 1$*, and if* g *is a neighbour of* h*, then*

$$g^2 \in R_{j-1} \ \ or \ \ g^2 \in R_j \ \ or \ \ g^2 \in R_{j+1}$$

Proof Let i be the distance of h from I, i.e. $h \in D_i$. Then, as $h^2 \in R_j$, $j > i$ by Lemma 3.1 (i). As g is a neighbour of h, $g \in D_i$ or $g \in D_{i-1}$ or $g \in D_{i+1}$.
Case $g \in D_i$.
As $h, g \in D_i$ are neighbours they are both ec-nodes. Thus, by Lemma 3.1 (iii), $h^2 \in R_{i+1}$ and $g^2 \in R_{i+1}$. Therefore, $j = i + 1$ and $g^2 \in R_j$.
Case $g \in D_{i-1}$.
If $g \in R_j$ which is a later round-set than R_{i-1} as $j > i$ then, as $g^1 \in D_{i-1} \subseteq R_{i-1}$ by Lemma 3.1 (i) is in the earlier round-set R_{i-1}, it must be the case that g^2 rather than g^1 belongs to R_j. If $g \notin R_j$ and $g^2 \notin R_{j-1}$ then $g \notin R_{j-1}$ as we know that $g^1 \in R_{i-1}$ which is an earlier round-set than R_{j-1} as $j > i$. As $h \in R_j$, h sends to g in round $j + 1$ and so $g \in R_{j+1}$ which is a later round-set than R_{i-1} as $j > i$. As $g^1 \in R_{i-1}$, it must be the case that $g^2 \in R_{j+1}$.
Case $g \in D_{i+1}$, g *does not send to* h *in round* j.
In this case, as $h \in R_j$, h sends to g in round $j + 1$. Thus, $g \in R_{j+1}$ which is a later round-set than R_{i+1} as $j > i$ and therefore, as $g^1 \in D_{i+1} \subseteq R_{i+1}$ by Lemma 3.1 (i), it must be the case that $g^2 \in R_{j+1}$.
Case $g \in D_{i+1}$, g *sends to* h *in round* j.
In this case $g \in R_{j-1}$. We show that $g^1 \notin R_{j-1}$. Assume, on the contrary, that $g^1 \in R_{j-1}$. Then, by Lemma 3.1 (i), $g^1 \in D_{i+1} \subseteq R_{i+1}$ and thus $j - 1 = i + 1$. Hence, by Lemma 3.1 (i), $h^1 \in D_i \subseteq R_i = R_{j-2}$. Also, $g \notin R_{j-3}$ as $g^1 \in R_{j-1}$. To summarize:

$$g \notin R_{j-3}, \ \ h^1 \in R_{j-2}, \ \ g^1 \in R_{j-1}, \ \ h^2 \in R_j$$

So, h sends to g in round $j - 1$ and g sends to h in round j by the case assumption. This is a contradiction. Thus, the assumption that $g^1 \in R_{j-1}$ is false and, as $g \in R_{j-1}$, it follows that $g^2 \in R_{j-1}$. This completes the proof. $\quad\square$

Corollary 3.6 *Let* (G, E) *be a graph and* $I \subseteq G$ *be a set of initial nodes. Then* (G, E) *has an ec-node if and only if all nodes are in exactly two round-sets.*

Proof Follows from Lemmas 3.2 and 3.5. $\quad\square$

We can now give bounds for time to termination of flooding for graphs that have ec-nodes.

Theorem 3.7 *Let* (G, E) *be a graph that is not I-bipartite and let* d *be the diameter. Then, flooding terminates in round* j *where* j *is in the range*

$$e(I) < j \leq e(I) + d + 1. \tag{9}$$

Proof To show (9), we show that flooding terminates in round j where

$$e(I) < j \leq min\{\delta(I, g_{ec}) + d + 1 \ : \ g_{ec} \in G_{ec}\}. \tag{10}$$

If (G, E) is not I-bipartite it has an ec-node g_{ec}, by Definition 4 (iv). By Lemma 3.1 (iii), $g_{ec}^2 \in R_k$ where $k = \delta(I, g_{ec}) + 1$. Let h be an arbitrary node in G other than g_{ec}. Then, there is a path

$$h_0 = g_{ec} \longrightarrow h_1 \longrightarrow \ldots \longrightarrow h_l = h$$

where $l \leq d$. By repeated use of Lemma 3.5,

$$h_1^2 \in R_{j_1} \text{ where } k - 1 \leq j_1 \leq k + 1,$$
$$h_2^2 \in R_{j_2} \text{ where } j_1 - 1 \leq j_2 \leq j_1 + 1,$$
$$\ldots$$
$$h_l^2 \in R_{j_l} \text{ where } j_{l-1} - 1 \leq j_l \leq j_{l-1} + 1.$$

Thus,

$$h_l^2 \in R_{j_l} \quad \text{where} \quad k - l \leq j_l \leq k + l \tag{11}$$

Put $j = j_l$. From (11), as $k = \delta(I, g_{ec}) + 1$ and as $l \leq d$,

$$h_l^2 \in R_j \quad \text{where} \quad j \leq \delta(I, g_{ec}) + d + 1$$

and, as g_{ec} is any ec-node,

$$h_l^2 \in R_j \quad \text{where} \quad j \leq min\{\delta(I, g_{ec}) + d + 1\}$$

As (G, E) is not I-bipartite, $j > e(I)$ by Theorem 3.4 and so (10) holds. Then, (9) follows from (10) easily as $\delta(I, g_{ec}) \leq e(I)$ by Definition 4 (iii). $\quad\square$

Figure 2 shows that the bounds are sharp. Sharpness of the upper bound is shown by the graph on the left in which $I = \{g_0\}$, there are $k - 1$ nodes between g_0 and g_k, $k - 2$ nodes between g_1' and g_k' and between g_1'' and g_k''. The graph terminates in $3k + 1 = k + 2k + 1 = e(I) + d + 1$ rounds

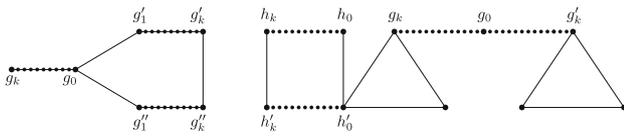**Fig. 2** Sharpness of bounds

Fig. 3 Conditions for round $e(I) + 1$ termination

for all $k \geq 1$. The slightly more complicated graph on the right in which $I = \{g_0\}$, there are $k - 1$ nodes between h_0 and h_k, h'_0 and h'_k, g_0 and g_k and between g_0 and g'_k, demonstrates sharpness of the lower bound and shows that flooding can terminate in a non-bipartite graph before round d. For all $k \geq 1$, $e(I) = 2k + 2$, $d = 3k + 3$ and flooding terminates in $2k + 3$ rounds and so $e(I) + 1 = 2k + 3 < d$.

Theorem 3.8 *Let (G, E) be a graph and $I \subseteq G$ be a set of initial nodes such that every node in I has a neighbour in I. Then, flooding terminates in round $e(I) + 1$.*

Proof As all nodes in I have neighbours in I they are all ec-nodes. Let h be an arbitrary node in G. Then, by Definition 4 (iii), there is a path

$$h_0 = g_{ec} \longrightarrow h_1 \longrightarrow \ldots \longrightarrow h_l = h, \qquad (12)$$

where $0 \leq l \leq e(I)$, from some node $g_{ec} \in I$ to h. By Lemma 3.1 (iii), $g_{ec}^2 \in R_k$ where $k = \delta(I, g_{ec}) + 1 = 0 + 1 = 1$. By repeated use of Lemma 3.5, arguing as in Theorem 3.7 putting $k = 1$ in (11), we have that

$$h^2 = h_l^2 \in R_{j_l} \quad where \quad j_l \leq k + l \leq 1 + e(I). \qquad (13)$$

Thus, termination occurs in round $e(I) + 1$ at the latest. If h is a node that is a distance $e(I)$ from I, $h^1 \in R_{e(I)}$ and so we can infer from (13) that $h^2 \in R_{e(I)+1}$. It follows that termination occurs in round $e(I) + 1$ precisely. □

The condition that all nodes in I have neighbours in I, in Theorem 3.8, for termination in the lower bound (for non I-bipartite graphs) number of rounds $e(I) + 1$ is useful for identifying possible examples of interest such as when the subgraph induced by I is connected. Whilst sufficient, the condition is not necessary. In the graph on the left in Fig. 3, $I = \{g_0, g_{2k}, h_0\}$ and there are $2k - 1$ nodes between g_0 and g_{2k} and h_0 and h_{2k}. Clearly, g_{2k} does not have a neighbour in I yet flooding terminates in $2k + 1 = e(I) + 1$ rounds for all $k \geq 1$. However, relaxing the connectedness condition to I only having a pair of neighbouring nodes is not, in general, sufficient for round $e(I) + 1$ termination. The graph on the right in Fig. 3 has $I = \{h_0, h_1, g_0\}$ and $2k$ nodes between g_0 and g_{2k+1}. Flooding terminates in round $2k+3$, but $e(I)+1 = (k + 1) + 1 = k + 2$ for all $k \geq 0$.

The clear separation in the termination times of bipartite and non-bipartite graphs, in the case of a single initial node,

does not carry over directly to the general case of multiple initial nodes. For the general case, it is the presence or otherwise of ec-nodes that is the distinguishing property and we have defined I-bipartite graphs for this purpose. Unlike the single node case, having ec-nodes or not in the multiple initial node case cuts across the bipartite/non-bipartite boundary, i.e. I-bipartite graphs may not be bipartite and bipartite graphs may not be I-bipartite. The work on the k-flooding problem in [32], looking at finding a set of sources of size k that minimizes termination time, offers a different perspective on amnesiac flooding termination times.

4 Loss of edges and delays at edges

4.1 Loss of edges

In the case where edges and nodes are lost during flooding, a message transiting along an edge may never arrive at its target node. The termination result Theorems 2.1 assumes that the graph (G, E) is fixed. Termination still holds if the graph loses edges and nodes.

Theorem 4.1 *Suppose that*

$$(G_0, E_0), (G_1, E_1), \ldots, (G_i, E_i), (G_{i+1}, E_{i+1}), \ldots$$
$$where \;\; G_i \supseteq G_{i+1}, E_i \supseteq E_{i+1} \; for \; all \; i \geq 0. \qquad (14)$$

is a sequence of graphs where nodes or edges may be lost over time expressed in terms of rounds. If $I \subseteq G_0$ is a set of initial nodes from which flooding is initiated in round 0 and progresses in subsequent rounds along the operational nodes and edges, and round-sets are defined as in Definition 2, then any node $g \in G_0$ is contained in at most two distinct round-sets.

Proof The proof of Theorem 2.1 for fixed (G, E) assumes the contrary of the theorem, and infers a contradiction by examining a node in a round-set which had an occurrence in a previous round-set. The proof requires that the edges present in the later round were present in the earlier round. So, the proof of Theorem 2.1 is also valid for the case that edges are lost, i.e.

$$(G, E_0), (G, E_1), \ldots, (G, E_i), (G, E_{i+1}), \ldots$$
$$where \;\; E_0 \supseteq E_1 \supseteq \ldots \supseteq E_i \supseteq E_{i+1} \supseteq \ldots$$

Allowing loss of nodes follows easily. Given a graph (G, E_i) and a node $g \in G_i$, let E_i^g be the edges in E_i not incident with g. Then, $(G \setminus \{g\}, E_i^g)$ is the subgraph of (G, E_i) induced by the subset of nodes $G \setminus \{g\}$ and (G, E_i^g) is the subgraph of (G, E_i) in which g is isolated, i.e. with the same nodes as (G, E_i) but with edges incident with g removed. Clearly, flooding terminates on the sequence of graphs with monotonically decreasing sets of edges:

$$(G, E_0), (G, E_1), \dots, (G, E_i^g), (G, E_{i+1}^g), \dots$$

if and only if it terminates on the sequence of graphs where the isolated g is removed:

$$(G, E_0), (G, E_1), \dots, (G \setminus \{g\}, E_i^g), (G \setminus \{g\}, E_{i+1}^g), \dots$$

It follows, by inductively removing multiple nodes that become isolated, that flooding terminates on any sequence of graphs of the form (14). $\square$

It is worth noting that the proof of Theorem 4.1 does not require the graph to remain connected, so flooding terminates in all parts of the graph if edge loss results in a graph that becomes disconnected. Of course, in that case, some parts of the graph may not receive the message at all.

4.2 Fixed delays at edges

Here, we consider cases when messages take a fixed number of rounds greater than 1 to transit an edge. Flooding proceeds with respect to a global clock, the non-negative integer ticks of which define rounds. Flooding is initiated in round 1 when initial nodes send to all neighbours. Message transit time is given in terms of the number of ticks of the clock taken. Such flooding does not correspond to single-round message transit simply by adding nodes to an edge. Consider the example in Fig. 4 below. We have weighted graphs with edges labelled by positive integers representing a fixed message transit time in both directions. The graph on the left has message transit along edge $\{g_1, g_2\}$ taking 2 rounds. The graph on the right adds a node g_3 along $\{g_1, g_2\}$ so that all edges have message transit time of 1 round. The behaviour of flooding in the two graphs is different - the graph on the left terminates in 4 rounds whereas the graph on the right terminates in 2. A more substantial example below shows that message transit of more than one round along an edge may lead to non-termination of flooding. In Fig. 5 arrowheads at nodes indicate a message being received at a node. Arrowheads not at nodes (as in round 4 of Fig. 5) indicate messages in transit. The arrowheaded graph in a round, which we refer to as the 'flooding state', determines how flooding proceeds. As the flooding states in rounds 3 and 8 are identical, flooding does not terminate. The graph in Fig. 5 is not bipartite. Figure 6 gives an example of a bipartite graph with delays at
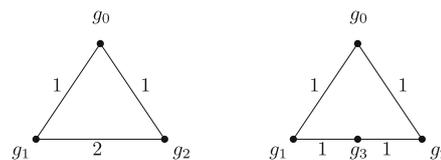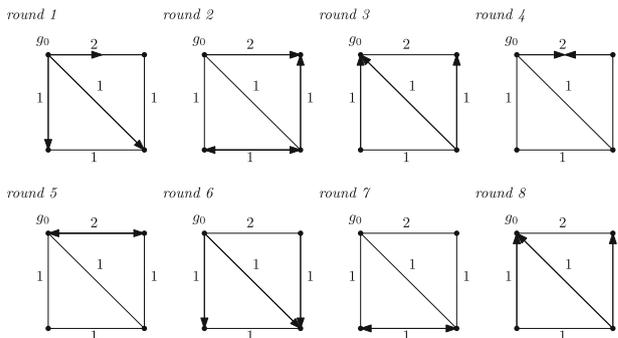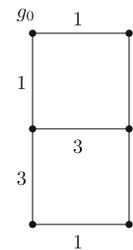

Fig. 4 Edge delay and additional nodes

Fig. 5 Non-termination of flooding with fixed edge delays

Fig. 6 Non-terminating fixed-delay bipartite example

edges, over which flooding does not terminate. In the example, which has two edges with delays, it can be checked that the flooding states for rounds 5 and 13 are identical and so flooding does not terminate. If there is only one initial node and a delay at a single edge in bipartite graphs, flooding does terminate.

Definition 5 Let (G, E) be an edge-weighted graph, $g_0 \in G$ be an initial node, $e = \{x, y\} \in E$ be an edge with weight τ where τ is an integer greater than 1 and let all other edges be of weight 1. Then, the graph (G^+, E^+) is obtained from (G, E) by adding $\tau - 1$ ticks along e as nodes, i.e.

$$G^+ = G \cup \{t_1, \dots, t_{\tau-1}\},$$
$$E^+ = (E \setminus \{x, y\}) \cup \{\{x, t^1\}, \{t^1, t^2\}, \dots, \{t^{\tau-1}, y\}\}.$$

So, (G^+, E^+) has all edges of duration 1. The distance function of (G^+, E^+) is denoted δ^+. If $g, g' \in G^+$, clearly $\delta^+(g, g') \leq \delta(g, g') + \tau - 1$, where δ is the distance function of the unweighted version of (G, E). Thus, the diameter d^+ of (G^+, E^+) is at most $d + \tau - 1$, where d is the diameter of unweighted (G, E).

Theorem 4.2 *Let (G, E) be a bipartite graph, $g_0 \in G$ be an initial node, $e = \{x, y\} \in E$ be an edge, and τ be an integer*
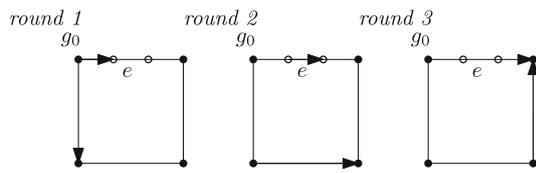
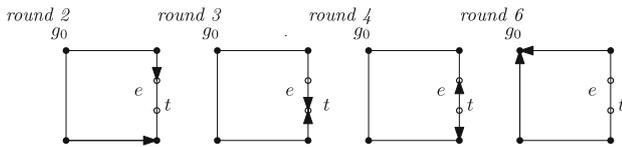**Fig. 7** Odd-delay edge e transited in one direction

Fig. 8 Odd-delay edge e transited in both directions

greater than 1. Then flooding initiated from g_0, in which a message in transit along e takes τ rounds and 1 round along any other edge, visits every node at most twice and completes in not more than $2d + \tau - 1$ rounds where d is the diameter of (G, E).

Proof We distinguish two cases according to whether τ is odd or even. In the figures below hollow points signify tick nodes in (G^+, E^+) corresponding to ticks of the global clock as the message travels along e.

τ odd.

Firstly, suppose that the message arrives at the node at one end of e and travels along e for an odd number of rounds without encountering the message travelling in the opposite direction as depicted in Fig. 7. Flooding corresponds to normal flooding over the bipartite graph (G^+, E^+) obtained from (G, E) as in Definition 5, where every node is visited once. The only difference with flooding where messages take one round to transit e is that the message transit along e takes an extra $\tau - 1$ rounds and so the time to termination may increase from a maximum of d rounds for the bipartite (G, E) with normal transit at e to $d + \tau - 1$ which is less than the bound $2d + \tau - 1$ in the theorem statement. Secondly, suppose that messages pass both of the end nodes of e travelling in opposite directions and meet along e. The difference in the round numbers that the message arrives for the first time at x and at y to continue along e, is clearly odd as x and y are in different partite sets in the bipartite (G, E). Thus, the opposing messages arrive at a tick node t along e for the first time in the same round as in Fig. 8. In Fig. 8, the messages passes both of the end nodes of e travelling in opposite directions, 'meet' at a tick t in round 3 but then continue along their journeys in their respective directions. Thus, a tick node t behaves like an ordinary additional node, except when messages travelling in opposite directions meet at t. We call such a t in (G^+, E^+) a *sink-tick* node. If t was an ordinary node then, as it receives messages from both neighbours, the messages would go no further. If t is a sink-tick node we need it to reflood the mes-

sage. So, if τ is odd, we can simulate flooding over (G, E) as a flooding over (G^+, E^+) in which any sink-tick node t refloods the message to its neighbours on receiving the message. Then, if the first flooding emanating from a sink-tick node t does not interact with the flooding emanating directly from g_0, flooding over (G, E) corresponds to two floodings over (G^+, E^+) - that initiated from g_0 and that initiated from t. As (G^+, E^+) is bipartite each of these two floodings visits a node once and terminates, and so flooding over (G, E) delivers the message to each node twice. So, it suffices to show that the flooding over (G^+, E^+) initiated from g_0 does not interact with the reflooding from a sink-tick node t.

Let j be the round that the flooding from g_0 first reaches a sink-tick node t^i, where $1 \leq i \leq \tau - 1$, and assume, on the contrary, that $k > j$ is the first round that the reflooding from t_i interacts with the flooding from g_0. Then, there are two sequences of sends of the message from g_0 and t_i respectively

$$g_0 \rightarrow g_1 \rightarrow \ldots \rightarrow g_k \qquad (15)$$

and

$$g_0 \rightarrow h_1 \rightarrow \ldots \rightarrow h_{j-1} \rightarrow h_j \rightarrow h_{j+1} \rightarrow \ldots \rightarrow h_k \qquad (16)$$

where $g_k = h_k$, subscripts indicate the round a node receives a message and in (16) $h_j = t^i$ refloods. We note that in (15) none of the nodes $g_j, \ldots, g_k$ can be a tick node as flooding from g_0 of the bipartite (G^+, E^+) will only visit each node once (by Lemmas 3.2 and 3.3) and the tick nodes are visited earlier before the first visit of the sink-tick node. On the other hand, the reflooding from h_j in (16) visits tick nodes until it reaches either x or y, say x occurs after h_j. One of x or y must also occur before h_j. We can choose the sequence $g_0 \rightarrow \ldots \rightarrow h_{j-1}$ such that it has x, i.e.

$$g_0 \rightarrow \ldots \rightarrow h_{l_1} = x \rightarrow \ldots \rightarrow h_j = t^i \rightarrow \ldots$$
$$\rightarrow h_{l_2} = x \rightarrow \ldots \rightarrow h_k \quad (0 < l_1 < j < l_2 < k) \quad (17)$$

From (17), $\delta^+(g_0, g_k) = \delta^+(g_0, h_k) \leq k - (l_2 - l_1) < k$, where δ^+ is the graph distance function of (G^+, E^+). But, (G^+, E^+) is bipartite and flooding from g_0 only visits nodes once and so $\delta^+(g_0, g_k) = k$ by Lemma 3.1 (i) and (15). This contradiction means that the flooding from g_0 does not interact with the reflooding from t^i and so the net effect is two separate floodings of the bipartite (G^+, E^+) each of which visits every node once and terminates. The first flooding from g_0 on the bipartite (G^+, E^+) takes not more than the diameter of (G^+, E^+) number of rounds, $d - 1 + \tau \leq 2d + \tau - 1$. The reflooding from the sink-tick t^i starts after t^i is reached by the first flooding from
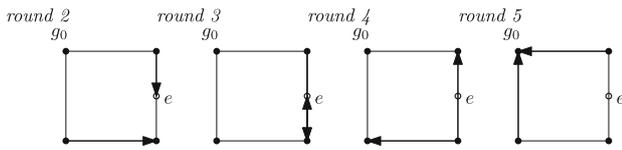
Fig. 9 Even-delay edge e transited in both directions

g_0, which is after at most $d + (\tau + 1)/2 - 1$ rounds, and then takes at most the same number of rounds to reach every node in G^+, thus terminating by round $2(d + (\tau + 1)/2 - 1) = 2d + \tau - 1$. Therefore, the theorem holds for odd τ.

τ even.

Here, if messages pass both of the end nodes of e travelling in opposite directions, as depicted in Fig. 9, they do not meet. Thus, tick nodes in (G^+, E^+) act like normal nodes and flooding terminates in the time it takes for (G^+, E^+) to terminate. If no cycle in (G, E) contains e, then (G, E) is two disjoint subgraphs induced by $G_x \subseteq G$ and $G_y \subseteq G$ of G, where $G = G_x \cup G_y$, $x \in G_x$, $y \in G_y$ and $G_x \cap G_y = \emptyset$, connected by e. The addition of $\tau - 1$ tick nodes along edge e gives a bipartite graph (G^+, E^+) where the partite sets of one of G_x or G_y are reversed. Flooding visits each node once in the bipartite (G^+, E^+) and terminates by round $d + \tau - 1$.

If a cycle in (G, E) does contain e, let g_t be any node in (G, E) and let t also denote the round that g_t receives the message for a second time. We show that $t \leq 2d + \tau - 1$. Let C be a cycle in (G, E) containing x and y that is of smallest length. First of all, we construct a cycle C_{g_t} which contains x, y, g_t and is of length at most $2d$ in (G, E). Suppose that the cycle C is oriented in a *clockwise* fashion thus:

$$C = c_0(= x) \to c_1(= y) \to \ldots \to c_m \to c_0$$

and that $\delta(x, g_t) \leq \delta(y, g_t)$. Let

$$p_1 = x \to u_1 \to \ldots \to u_k \to g_t \quad (k \geq 0)$$

be shortest path from x to g_t in (G, E). Consider, in turn, the shortest path from each of the nodes $c_1, c_2, \ldots$ to g_t. For some of these the shortest path to g_t may be *anticlockwise* along C back to $x = c_0$ and then along the shortest path p_1 from x to g_t. Choose the least i with $1 \leq i \leq m - 1$ which has its own shortest path (possibly clockwise along C) p_2 to g_t

$$p_2 = c_i \to v_1 \to \ldots \to v_l \to g_t \quad (l \geq 0, v_1 \neq c_{i-1})$$

The cycle C_{g_t} is then defined as the path p_1 backwards from g_t to x followed by the arc of C clockwise from $x = c_0$ to c_i and then followed by the path p_2 back to

g_0. (It is possible that the paths p_1 and p_2 converge to the same node earlier than g_t in which case C_{g_t} will be a closed walk, starting at g_t and returning to g_t after tracing all the paths, which is comprised of a cycle with a path attached from the cycle to g_t.) As the path from c_{i-1} to $c_0 = x$ anticlockwise along C and then along p_1 to g_t is a shortest path from c_{i-1} to g_t, it follows that this path is of length $\leq d$. The path p_2 from c_i to g_t is $\leq d$. The nodes c_{i-1} and c_i are neighbours. Thus, the length of C_{g_t} in (G, E) is at most $d + d + 1$. As (G, E) is bipartite the length of C_{g_t} is $\leq 2d$. So, C_{g_t} contains x, y, g_t and is of length at most $2d$ in (G, E). Therefore, in (G^+, E^+),

C_{g_t} *contains neighbouring ec-nodes g_{ec} and h_{ec}*
 and is of length $\leq 2d + \tau - 1$ (18)

By (18) we now construct a cycle C_{g_0} from two sequences of sent messages $\underline{s_1}$ and $\underline{s_2}$:

$$\underline{s_1} = w_0(= g_0) \to w_1 \to \ldots w_r(= g_{ec}) \to w_{r+1}(= h_{ec})$$
$$\underline{s_2} = z_0(= g_0) \to z_1 \to \ldots z_r(= h_{ec}) \to z_{r+1}(= g_{ec})$$

and define the direction along the cycle to be *clockwise* if g_{ec} is visited before h_{ec}. Let $w = w_j$ and $z = z_j$ be nodes in $\underline{s_1}$ and $\underline{s_2}$ respectively that belong to C_{g_t} such that all nodes after w in $\underline{s_1}$ and z in $\underline{s_2}$ belong to C_{g_t}. Clearly, C_{g_0} is a cycle (closed walk if $w = z$) of length at most $2d$ in (G, E) and $2d + \tau - 1$ in (G^+, E^+). Now, in order to show that there is a sequence of sends that deliver the message for a second time to g_t no later than round $2d + \tau - 1$, by Theorem 3.7 it suffices to find a path from g_0 to g_t that visits neighbouring ec-nodes and is of length less than or equal to $2d + \tau - 1$. There are two cases to consider:

(i) g_t *is nearer to g_{ec} than to h_{ec}.* The two left-hand figures in Fig. 10 depict two possible subcases where g_t is between z and w and where it is not. If g_t is between z and w, an anticlockwise path from g_0 along C_{g_0} past h_{ec} and g_{ec} up to w and then along C_{g_t} until g_t is reached is a path of length less than or equal to the maximum length of one revolution of either of the cycles C_{g_0} or C_{g_t} which is less than or equal to $2d + \tau - 1$. This is because the arc that is not traversed of the larger cycle results in a corresponding smaller arc of the smaller cycle to be traversed. (Fig. 10 suggests C_{g_0} is larger than C_{g_t} but it could be drawn the other way around.) If g_t is not between z and w then an anticlockwise path from g_0 along C_{g_0} past h_{ec} and g_{ec} up to g_t suffices.

(ii) g_t *is nearer to h_{ec} than to g_{ec}.* This case is a mirror image of (i) with clockwise instead of anticlockwise paths being used. The right-hand figures in Fig. 10 illustrate this. □
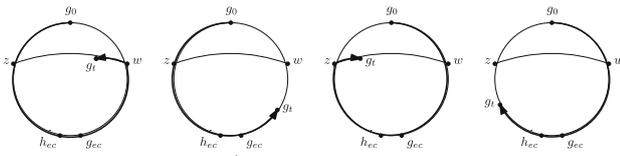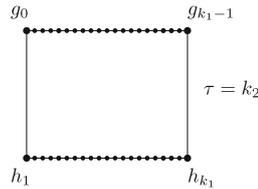
Fig. 10 Even-delay edge e termination time bound

Fig. 11 Sharpness of Theorem 4.2 bound

The $2d + \tau - 1$ bound in Theorem 4.2 is sharp for general bipartite graphs, as the example in Fig. 11 has, for all $k_1, k_2 \geq 2$, k_2 even, $d = k_1$, $\tau = k_2$ and termination occurs in round $2d + \tau - 1 = 2k_1 + k_2 - 1$.

Theorem 4.2 restricts flooding to one initial node and one delaying edge. However, if (G, E) is a cycle (not necessarily bipartite), then flooding terminates with any number of initial nodes and delaying edges.

Theorem 4.3 *Let (G, E) be an edge-weighted cycle in which (possibly different) positive integer edge weights represent duration of message transit in rounds along edges and $I \subseteq G$ be a set of initial nodes. Then, flooding initiated from initial nodes in I terminates in not more than the weight σ of (G, E) number of rounds, where the weight σ of (G, E) is the sum of the weights of the edges.*

Proof Let (G^+, E^+) be the cycle obtained from (G, E) by adding tick nodes along all edges of greater weight than 1 in a similar way to Definition 5. Denote successive vertices of G^+, starting at an arbitrary vertex in I, by the integers $\{0, \ldots, \sigma - 1\}$, where σ is the weight of the the cycle (G, E), so that $G^+ = G \cup T$ where T is the set of tick nodes. We will assume throughout that arithmetic is modulo σ and say that a message sent from $v \in G^+$ to $v + 1 \in G^+$ is *clockwise*, and one sent from $v \in G^+$ to $v - 1 \in G^+$ is *anticlockwise*. Each initial node $v \in I$ will generate a sequence of sends of the message in clockwise and anticlockwise directions:

$$v \rightarrow v + 1 \rightarrow \ldots, \quad v \rightarrow v - 1 \rightarrow \ldots$$

either of which may terminate if they encounter a sequence in the opposite direction at a node in G in some round. Any sequence of sent messages is defined by an initial node and a direction. For $v \in I$, we define positively and negatively decorated versions, v^+ and v^- respectively, of v and, correspondingly, positively and negatively decorated versions of subsets $S \subseteq I$:

$$S^+ = \{v^+ : v \in S\}, \quad S^- = \{v^- : v \in S\}$$

Define a relation $\sim$ on I by:

$$v \sim w \text{ iff } \exists m. \, v + m = w - m \, \wedge \, v + m \in G \quad (19)$$

for $v, w \in I$. Intuitively, $v \sim w$ means that the clockwise sequence of sends from v and the anticlockwise sequence from w can terminate each other at some node in G if neither has been terminated earlier. Note that $\sim$ is symmetric because if $v \sim w$ then $v + m = w - m$ as in (19) and thus $w + (\sigma - m) = w - m = v + m (\in G) = v - (\sigma - m)$ and so $w \sim v$. It is clearly reflexive as $v + m = w - m$ implies $w + (-m) = v - (-m)$. Also, if $v \sim w$ and $w \sim z$ then there exist m', m'' such that $v + m' = w - m'$ and $w + m'' = z - m''$. Thus, $v + (m' + m'') = z - (m' + m'')$ and so $v \sim z$ and therefore $\sim$ is transitive. As such, $\sim$ defines an equivalence relation on I whose set of equivalence classes we denote by $\mathcal{I}$.

Next, we define Q_i for $i \geq 0$:

$$Q_i = \{v^+ : v \in I \, \wedge \, v \rightarrow v + 1 \rightarrow \ldots \rightarrow v + i\}$$
$$\cup \, \{v^- : v \in I \, \wedge \, v \rightarrow v - 1 \rightarrow \ldots \rightarrow v - i\}$$

to be the set of the positive (versions of) initial nodes from which the clockwise sequence of sent messages hasn't terminated before round i, along with the negative (versions of) initial nodes from which the anticlockwise sequence of sent messages has not terminated. Clearly, $Q_{i+1} \subseteq Q_i$. Note that, Q_i may have neither, one or both versions of a particular initial node depending on whether the corresponding oriented sequences of sent messages has terminated by round i. However, as clockwise and anticlockwise sequences are always terminated in pairs and correspond to initial nodes that may be different but do belong to the same equivalence class $S \in \mathcal{I}$, the number of positive initial nodes in Q_i that belongs to S is equal to the number of negative initial nodes, i.e.

$$\forall i \geq 0 \, \forall S \in \mathcal{I}. \, |S^+ \cap Q_i| = |S^- \cap Q_i| \quad (20)$$

We now prove that flooding terminates by assuming that flooding does not terminate and show that this leads to a contradiction. If flooding does not terminate, flooding yields a, clockwise say, sequence of sends:

$$v \rightarrow v + 1 \rightarrow \ldots \rightarrow v + \sigma (= v) \quad (21)$$

that is not terminated in round σ so that $v^+ \in Q_\sigma$. By (20), there must be a $w^- \in Q_\sigma$ such that v and w are in the same equivalence class S, i.e. $v \sim w$. It follows, by the definition of $\sim$ in (19), that flooding also yields sequences of sends:

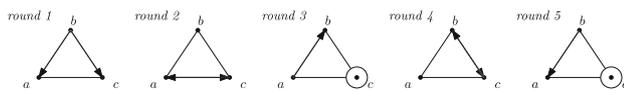$$v \rightarrow v + 1 \rightarrow \ldots \rightarrow v + m, \quad w \rightarrow w - 1 \rightarrow \ldots \rightarrow w - m, \quad (22)$$

Fig. 12 Varying delays at edges non-termination

where $m \leq \sigma$ and $v + m = w - m$ and $v + m \in G$. Clearly, the m in (22) must equal σ, else if $m < \sigma$ the clockwise sequence from v is terminated by the anticlockwise sequence from w in round m which is earlier than round σ and so flooding would not have yielded sequence (21). On the other hand, if $m = \sigma$, then, by (22), clockwise flooding from v is terminated in round σ, contrary to our assumption. This contradiction proves termination occurs by round σ, where σ is the weight of the graph (G, E). □

4.3 Varying delays at edges

In a general asynchronous model, where the delay of message transit along an edge may vary during the flooding process, termination is not guaranteed even on the smallest cycle. Our example is of non-termination over the triangle graph where an adaptive adversary at a node produces an infinite sequence of delays at the right times so as to prevent termination. The adaptive adversary is used just as an example and non-termination over cycles can arise for any number of different reasons. Consider the triangle graph in Fig. 12, in which computation proceeds with respect to a global clock whose integer ticks are the rounds and the adversary can decide the delay of message delivery on any edge in any round. In round 0, b has the message as the initial node which is then flooded. Figure 12 illustrates rounds 1-5. Both a and c send the message to each other in round 2. In round 3, a sends the message to b but the adversary makes c hold the message for one round, indicated by a circled node. After that, in round 4, b and c send the message to each other. In any given round, the corresponding arrowheaded graph determines in which ways flooding can proceed. As the arrowheaded graphs in rounds 2 and 4 are equivalent (by a relation- and adversary-preserving automorphism), flooding can proceed from round 4 in a similar way as from round 2 with the edges b and a interchanged and the adversary acting on node c as before. The repeated adversarial intervention prevents flooding from terminating.

5 Conclusion

It was previously thought that the stateless version of the terminating stateful synchronous flooding algorithm does not terminate in general. We have proved that it does terminate on all graphs and in optimal time. Thus, this amnesiac flooding achieves broadcast and can be used to set up spanning trees like stateful flooding in almost the same time without the additional overhead of storing every message required to achieve termination explicitly. Our work has shown that there is a clear separation of the times to termination, corresponding to whether the message is received once or twice by all nodes, in turn depending on whether the (quotient with respect to the set of sources) graph is bipartite or non-bipartite. This raises the question of whether this separation can be exploited to devise distributed procedures to detect the topology of a graph given distance measures or vice versa. If an upper bound D on the diameter d is known, a simple distributed algorithm at nodes to determine whether the network graph is bipartite is for the issuing source node to wait 2D+1 rounds to see if the message returns. It might also seem that, conversely, if the graph is known to be bipartite, a distributed algorithm for determining a bound for the diameter d is possible given that amnesiac flooding terminates in 2d+1 bounds - a node might measure the time it takes for the message to return. However, there are a number of problems to be overcome. If the graph is bipartite then, for the message to be returned to the source, the message has to be reflooded from 'sink' nodes which receive the message on all edges in a single round. This would result in the message possibly being returned to the issuing source node any number of times. Even if the graph is non-bipartite and a single return of the message is guaranteed, the issuing node may receive the message very early on, because of the proximity of an ec-node, and thereby underestimate d.

When the overhead of storing messages is not a problem, the main use of amnesiac flooding may be as a process that is known to terminate in 2d+1 rounds which can be used with state in distributed algorithms to obtain the diameter.

Open Problem 1. Can the diameter of a graph be obtained/ approximated by a distributed algorithm that uses amnesiac flooding along with a limited amount of state recording certain events?

The basic termination result for amnesiac flooding is significant for its applicability to arbitrary graphs with unknown diameters and network topologies. For common known bipartite topologies such as hypercubes and star networks termination is clear and the result has less significance, unless messages take different times to transit along different links. In the presence of delays even stateful flooding is problematic as, although the diameter is known, it is not clear how long messages should be stored. To this end, we have looked at the termination problem when message transit along an edge takes more than one round and found that if there is a delay at only one edge in bipartite graphs then amnesiac flooding terminates but not in non-bipartite graphs. This means that if delays at edges do not occur often, and it is unlikely that there are delays at two edges during a flooding, then termination will follow in bipartite graphs. Our result is for a single source. The multi-source case is open.

Open Problem 2. Does amnesiac flooding from multiple sources on a bipartite graph, where there is a fixed time of message transit greater than one round along a single edge, terminate?

What if there can be delays at more than one edge but that delays are short-lived? The example in Fig. 5 terminates if the message transit time along an edge with a delay returns to normal once the edge becomes clear of messages for the first time.

Open Problem 3. For which class of bipartite graphs, with fixed-delays at multiple edges, does amnesiac flooding from a single source terminate if the message transit time along an edge with a delay returns to normal once the edge becomes clear of messages for the first time?

Our example with delays at edges have assumed that the delayed transit time is in both directions. Even in the non-bipartite example of Fig. 5, if the delay is in only one direction, amnesiac flooding terminates.

Open Problem 4. For which class of graphs with directional fixed-delays at edges does amnesiac flooding from a single source terminate?

The only general multiple-source and multiple-edge fixed-delay termination result we have is for cycles. Can this be extended to cylinder grid graphs, i.e. products of cycles with line graphs, or any class of tori?

Open Problem 5. Suppose that (G_1, E_1) is a cycle and (G_2, E_2) is a line graph. Is multiple-source amnesiac flooding on the product graph $(G_1, E_1) \times (G_2, E_2)$, where there are fixed delays at multiple edges, guaranteed to terminate? What if (G_2, E_2) is an acyclic graph or a cycle graph?

As a final note, one can see processes such as random walks, coalescing random walks and diffusion as probabilistic extremal variants of flooding. Are there any implications or connections of our results on these or intermediate probabilistic models? What about randomised variants of amnesiac flooding?

Declarations

References

1. Adamek, J., Nesterenko, M., Robinson, J.S., Tixeuil, S.: Stateless reliable geocasting. In: 36th IEEE Symposium on Reliable Distributed Systems, SRDS 2017, Hong Kong, Hong Kong, p. 44–53, IEEE Computer Society (2017)

2. Attiya, H., Welch, J.: Distributed Computing: Fundamentals. Simulations and Advanced Topics. Wiley, Hoboken (2004)

3. Awerbuch, B., Khandekar, R.: Greedy distributed optimization of multi-commodity flows. In: Proceedings of the Twenty-Sixth Annual ACM Symposium on Principles of Distributed Computing (PODC '07). Association for Computing Machinery, New York, NY, USA, p. 274–283. https://doi.org/10.1145/1281100.1281140

4. Awerbuch, B., Khandekar, R.: Stateless distributed gradient descent for positive linear programs. In: Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing (STOC '08). Association for Computing Machinery, New York, NY, USA, p. 691–700. https://doi.org/10.1145/1374376.1374476

5. Bayramzadeh, Z., Kshemkalyani, A.D., Molla, A.R., Sharma, G.: Weak amnesiac flooding. In: 20th International Symposium on Parallel and Distributed Computing ISPDC, p. 122-129 (2021). https://doi.org/10.1109/ISPDC52870.2021.9521629

6. Bayramzadeh, Z., Kshemkalyani, A.D., Molla, A.R., Sharma, G.: Weak amnesiac flooding of multiple messages. In: Echihabi, K., Meyer, R., (eds.) Networked Systems. NETYS (2021). Lecture Notes in Computer Science, vol. 12754. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-91014-3_6, https://doi.org/10.1145/3369740.3369786

7. Cooper, J.N., Spencer, J.: Simulating a random walk with constant error. Comb. Probab. Comput. **15**(6), 815–822 (2006)

8. Das Sarma A., Danupon N., Gopal P.: Fast distributed random walks. In: PODC, p. 161–170 (2009)

9. Das Sarma A., Danupon N., Gopal P., Prasad T.: Efficient distributed random walks with applications. In: PODC (2010)

10. Doerr, B., Fouz, M., Friedrich, T.: Social networks spread rumors in sublogarithmic time. Electron. Notes Discret. Math. **38**, 303–308 (2011)

11. Dolev, D., Erdmann, M., Lutz, N., Schapira, M., Zair, A.: Stateless computation. In: Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC '17), p. 419–421, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3087801.3087854

12. Elsässer, R., Sauerwald, T.: The power of memory in randomized broadcasting. In: Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '08, p. 218–227, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2008)

13. Gkantsidis, C., Mihail, M., Saberi, A.: Random walks in peer-to-peer networks: algorithms and evaluation. Perform Eval **63**(3), 241–263 (2006)

14. Gopal, A.S., Gopal, I.S., Kutten, S.: Fast broadcast in high-speed networks. IEEE/ACM Trans. Netw. **7**(2), 262–275 (1999)

15. Hussak, W., Trehan, A.: On termination of a flooding process. In: Robinson, P., Ellen, F. (eds.) Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing, PODC 2019, Toronto, ON, Canada, p. 153–155. ACM (2019)

16. Hussak, W., Trehan, A.: Terninating cases of flooding. CoRR (2020). arXiv:2009.05776

17. Hussak, W., Trehan, A.: On the termination of flooding. In: Paul, C., Bläser, M. (eds.) 37th International Symposium on Theoretical Aspects of Computer Science (STACS 2020), Leibniz International Proceedings in Informatics (LIPIcs), vol. 154, p. 17:1–17:13, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2020)

18. Karp, B., Kung, H.T.: Gpsr: greedy perimeter stateless routing for wireless networks. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, p. 243–254, New York, NY, USA (2000)

19. Kosowski, A., Pajak, D.: Does adding more agents make a difference? A case study of cover time for the rotor-router. J. Comput. Syst. Sci. **106**, 80–93 (2019)

20. Kutten, S., Pandurangan, G., Peleg, D., Robinson, P., Trehan, A.: On the complexity of universal leader election. J. ACM **62**(1), 1–27 (2015)

21. Kutten, S., Pandurangan, G., Peleg, D., Robinson, P., Trehan, A.: Sublinear bounds for randomized leader election. Theor. Comput. Sci. **561**, 134–143 (2015)

22. Lynch, N.: Distributed Algorithms. Morgan Kaufmann Publishers, San Mateo, CA (1996)

23. McQuillan, J., Richer, I., Rosen, E.: The new routing algorithm for ARPANET. IEEE Trans. Commun. **28**, 711–719 (1980)

24. Michail, O., Spirakis, P.G.: Terminating population protocols via some minimal global knowledge assumptions. J. Parallel Distrib. Comput. **81–82**, 1–10 (2015)

25. Parulkar, G.M.: NOANET: An Experimental Flood Local Area Network PhD Dissertation, pp. 88–103. University of Delaware, Delaware (1987)

26. Peleg, D.: Time-optimal leader election in general networks. J. Parallel Distrib. Comput. **8**(1), 96–99 (1990)

27. Peleg, D.: Distributed Computing: A Locality Sensitive Approach. SIAM, Bangkok (2000)

28. Rahman, A., lesinski, W., Gburzynski, P.: Controlled flooding in wireless ad-hoc networks, In: Proceeding IWWAN'04, p. 73-78 (2004)

29. Segall, A.: Distributed network protocols. IEEE Trans. Inf. Theory **29**, 23–35 (1983)

30. Tanenbaum, A.: Computer Networks. Pearson Prentice Hall, Hoboken (2011)

31. Tel, G.: Introduction to Distributed Algorithms. Cambridge University Press, New York, NY (1994)

32. Turau, V.: Analysis of amnesiac flooding. (2020). arXiv:2002.10752

33. Turau, V.: Stateless information dissemination algorithms. In: Structural Information and Communication Complexity - 27th International Colloquium, SIROCCO, p. 183–199, Springer (2020)

34. Turau, V.: Amnesiac flooding: synchronous stateless information sissemination. In: 47th International Conference on Current Trends in Theory and Practice of Computer Science. SOFSEM 2021, pp. 59–73. Springer, LNTCS (2021)

35. Turau, V.: Synchronous concurrent broadcasts for intermittent channels with bounded capacities. In: Structural Information and Communication Complexity - 28th International Colloquium, SIROCCO, p. 296–312, Springer (2021)