arXiv:1005.2894v3 [cs.DC] 8 Dec 2018

Optimal Gradient Clock Synchronization in Dynamic Networks

Fabian Kuhn[1] Christoph Lenzen[2] Thomas Locher[3] Rotem Oshman[4]
[1]University of Freiburg, Germany
kuhn@cs.uni-freiburg.de
[2]MPI Saarbruecken, Germany
clenzen@mpi-inf.mpg.de
[3]ABB Research, Switzerland
thomas.locher@ch.abb.com
[4]Tel Aviv University, Israel
roshman@tau.ac.il

Abstract

We study the problem of clock synchronization in highly dynamic networks, where communication links can appear or disappear at any time. The nodes in the network are equipped with hardware clocks, but the rate of the hardware clocks can vary arbitrarily within specific bounds, and the estimates that nodes can obtain about the clock values of other nodes are inherently inaccurate. Our goal in this setting is to output a logical clock at each node such that the logical clocks of any two nodes are not too far apart, and nodes that remain close to each other in the network for a long time are better synchronized than distant nodes. This property is called *gradient clock synchronization*.

Gradient clock synchronization has been widely studied in the static setting, where the network topology does not change. We show that the asymptotically optimal bounds obtained for the static case also apply to our highly dynamic setting: if two nodes remain at distance d from each other for sufficiently long, it is possible to upper bound the difference between their clock values by $\mathcal{O}(d \log(D/d))$, where D is the diameter of the network. This is known to be optimal even for static networks. Furthermore, we show that our algorithm has optimal *stabilization time*: when a path of length d appears between two nodes, the time required until the clock skew between the two nodes is reduced to $\mathcal{O}(d \log(D/d))$ is $\mathcal{O}(D)$, which we prove to be optimal. Finally, the techniques employed for the more intricate analysis of the algorithm for dynamic graphs provide additional insights that are also of interest for the static setting. In particular, we establish self-stabilization of the gradient property within $\mathcal{O}(D)$ time.

1 Introduction

A core algorithmic problem in distributed computing is to establish coordination among the participants of a distributed system, which is often achieved through a common notion of time. Typically, every node in a network has its own local hardware clock, which can be used for this purpose; however, hardware clocks of different nodes run at slightly different rates, and the rates can change over time. This *clock drift* causes clocks to drift out of synch, requiring periodic communication to restore synchronization. However, communication is typically subject to delay, and although an upper bound on the delay may be known, specific message delays are unpredictable. Consequently, estimates for the current local time at other nodes are inherently inaccurate.

A distributed clock synchronization algorithm computes at each node a *logical clock*, and the goal is to synchronize these clocks as tightly as possible. Traditionally, distributed clock synchronization algorithms focus on minimizing the *clock skew* between the logical clocks of any two nodes in the network. The clock skew between two clocks is simply the difference between the two clock values. The maximum clock skew that may occur in the worst case between any two nodes at any time is called the *global skew* of a clock synchronization algorithm. A well-known result states that no algorithm can guarantee a global skew better than $\Omega(D)$, where D denotes the diameter of the network [1]. However, in many cases it is more important to tightly synchronize the logical clocks of nearby nodes in the network than it is to minimize the global skew. For example, if a time division multiple access (TDMA) protocol is used to coordinate access to a shared communication medium in a wireless sensor network, it suffices to synchronize the clocks of nodes that interfere with each other when transmitting. The problem of providing better guarantees on the synchronization quality between nodes that are closer is called *gradient clock synchronization*. The problem was introduced in a seminal paper by Fan and Lynch [7], where the authors show that a clock skew of $\Omega(\log D/\log\log D)$ cannot be prevented between immediate neighbors in the network. The largest possible clock skew that may occur between the logical clocks of any two adjacent nodes at any time is called the *local skew* of a clock synchronization algorithm. For static networks, it has been proved that the best possible local skew that an algorithm can achieve is bounded by $\Theta(\log D)$ [15, 16].

While tight bounds have been shown for the static model, the dynamic case has not been as well understood. A dynamic network arises in many natural contexts: for example, when nodes are mobile, or when communication links are unreliable and may fail and recover. The dynamic network model we consider in this article is general: it allows communication links to appear and disappear arbitrarily, subject only to a global connectivity constraint (which is required to maintain a bounded global skew). Hence the model is suitable for modeling various types of dynamic networks which remain connected over time.

In a dynamic network the distances between nodes change over time as communication links appear and disappear. Consequently, we divide the synchronization guarantee into two parts: a *global skew guarantee* bounds the skew between any two nodes in the network at any time, and a *dynamic gradient skew guarantee* that bounds the skew between two nodes as a function of the distance between them and how long they remain at that distance.

In [11], three of the authors showed that a clock synchronization algorithm cannot react immediately to the formation of new links, and that a certain *stabilization time* is required before the clocks of newly-adjacent nodes can be brought into synch. The stabilization time is inversely related to the synchronization guarantee: the tighter the synchronization required in stable state, the longer the time to reach that state. Intuitively, this is because when strict synchronization guarantees are imposed, the algorithm cannot change clock values quickly without violating the guarantee, and hence it takes longer to react. The algorithm given in [11] achieves the optimal trade-off between skew bound and stabilization time; however, its local skew bound is $\mathcal{O}(\sqrt{D})$, which is far from optimal.

In this article, we propose an algorithm, referred to as $\mathcal{A}^{\mathrm{OPT}}$, that achieves the same asymptotically optimal skew bounds as in the static model: if two nodes remain at distance d for sufficiently long, the skew between them is reduced to $\mathcal{O}(d\log(D/d))$, where D is the dynamic diameter of the network (corresponding roughly to the time it takes for information to propagate from one end of the network to the other). The stabilization time of the algorithm, that is, the time to reach this guarantee, is $\mathcal{O}(D)$.

2 Related Work

The fundamental problem of synchronizing clocks in distributed systems has been studied extensively and many results have been published for various models over the course of the last approximately 30 years (see, e.g, [20, 22, 23, 24]). Until recently, the main focus has been on bounding the clock skew that may occur between any two nodes in the network. Using the well-known shifting argument [20], which exploits the variable message delays to construct indistinguishable executions, it has been shown that a clock skew of $D/2$ cannot be prevented on any graph of diameter D [1]. This lower bound holds even if clocks do not drift. Indistinguishable executions can also be constructed by exploiting variable clock rates [5], which can be used together with the shifting argument to prove a stronger lower bound of roughly D for algorithms that must ensure that all clock values are always within a linear envelope of real time [16]. In light of these results, the algorithm proposed by Srikanth and Toueg [24] is asymptotically optimal as it guarantees a skew of at most $\mathcal{O}(D)$ between any two clocks. The accuracy of their algorithm is also optimal in the sense that all clock values are within a linear envelope of real time, i.e., a better accuracy with respect to real time cannot be guaranteed. A crucial shortcoming of this algorithm is that a clock skew of $\Omega(D)$ may occur between neighboring nodes.

The problem of synchronizing clocks of nodes that are close-by as accurately as possible has been introduced by Fan and Lynch [7]. In their work, the authors show that a clock skew of $\Omega(\log D/\log\log D)$ between neighboring nodes cannot be avoided if the clock values must increase at a constant minimum progress rate. Subsequently, this result has been improved to $\Omega(\log D)$ [16]. If we take the minimum logical clock rate α, the maximum logical clock rate β, and the maximum clock drift rate ρ into account, the more general statement of the lower bound is that a clock skew of $\Omega(\log_b D)$, where $b := \min\{1/\rho, (\beta-\alpha)/(\alpha\rho)\}$ cannot be avoided. The first algorithm guaranteeing a sublinear bound on the worst-case clock skew between neighbors achieves a bound of $\mathcal{O}(\sqrt{\rho D})$ [17, 18]. Recently, this result has been improved to $\mathcal{O}(\log D)$ [15] (where the base of the logarithm is a constant) and subsequently to $\mathcal{O}(\log_b D)$ [16]. Thus, tight bounds have been achieved for static networks in which neither nodes nor edges fail.

The problem of synchronizing clocks in the presence of faults has also received considerable attention (see, e.g., [4, 9, 13, 19, 21]). Some of the proposed algorithms are able to handle not only simple crash failures but also Byzantine behavior, which is outside the scope of this article. However, while these algorithms can tolerate a broader range of failures, their network model is not fully dynamic as their results rely on the assumption that a large part of the network remains non-faulty and stable at all times. For the fully dynamic setting, it has been shown that there is an inherent trade-off between the clock skew $\mathcal{S}$ guaranteed between neighboring nodes that have been connected for a long time and the time it takes to guarantee a small clock skew over newly added edges. In particular, the time it takes to reduce the clock skew over new edges to $\mathcal{O}(\mathcal{S})$ is $\Omega(D/\mathcal{S})$, where n denotes the number of nodes in the network [11]. In the same work, it is shown that for $\mathcal{S} \in \Omega(\sqrt{\rho D})$, there is an algorithm that reduces the clock skew between any two nodes to $\mathcal{O}(\mathcal{S})$ in $\Theta(D/\mathcal{S})$ time. In this article, we show that $\mathcal{S}$ can be reduced to $\mathcal{O}(\log_b D)$, i.e., the same optimal bound as for static networks can be achieved, while still establishing this bound within $\Theta(D/\mathcal{S})$ time on newly formed edges.

Another notion of fault-tolerance is *self-stabilization* [3], i.e., the ability to recover correct operation after a period of arbitrary transient faults. Many clock synchronization algorithms are self-stabilizing simply because of their continuous strive for maintaining synchronization. However, a strong gradient property is a more involved requirement than just minimizing the global skew, hence self-stabilization is not immediate for our algorithm; the previous works on the static case do not yield this result. In contrast, in the dynamic setting, we exploit self-stabilization properties of the algorithm in order to safely establish the gradient property on recently appeared edges (without disrupting the guarantees for edges that have been present for a long time). Consequently, we obtain self-stabilization of the gradient property as a corollary of our analysis.

3 Preliminaries

In this section we introduce the dynamic clock synchronization problem and the model for dynamic networks that will be used in this paper. We begin by reviewing classical (static) clock synchronization.

Clock synchronization. In the clock synchronization problem, each node u is equipped with a continuous and differentiable *hardware clock* $H_u : \mathbb{R}_0^+ \to \mathbb{R}_0^+$, which is initialized to $H_u(0) := 0$. We use $h_u(t)$ to denote the rate $\frac{d}{dt}H_u(t)$ at which node u's hardware clock advances at time t.[1] The hardware clocks advance at roughly the rate of real time, but they suffer from clock drift bounded by $\rho \in (0, 1)$; formally, we assume that at all times t we have $h_u(t) \in [1 - \rho, 1 + \rho]$ for all nodes u. As a result, for any two times $t_1 \leq t_2$ we have

$$(1 - \rho)(t_2 - t_1) \leq H_u(t_2) - H_u(t_1) \leq (1 + \rho)(t_2 - t_1).$$

The objective of a clock synchronization algorithm (CSA) is to output a left-differentiable[2] *logical clock* $L_u : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ (also initialized to $L_u(0) := 0$), such that at all times, the logical clock values of different nodes are close to each other (we elaborate on this requirement below). We use $l_u(t)$ to denote the rate $\frac{d}{dt}L_u(t)$ of u's hardware clock at time t. The logical clocks are also required to have bounded drift: there must exist constants $\alpha, \beta > 0$, such that for all t we have $l_u(t) \in [\alpha, \beta]$.

In the algorithm we present in this paper, nodes always increase their logical clocks at either the rate of their hardware clock $h_u(t)$, or at a rate of $(1 + \mu) \cdot h_u(t)$, where $\mu \in \mathcal{O}(1)$ is a parameter of the algorithm. Thus, the algorithm bounds the drift of the logical clocks, and we have that $\alpha := 1 - \rho$ and $\beta := (1 + \rho)(1 + \mu)$.

3.1 The Dynamic Graph Model

The estimate graph. In [12] two of the authors introduced an abstraction called *the estimate layer*, which simplifies reasoning about CSAs. Synchronization typically involves periodic exchanges of clock values between nodes, either through direct communication, or by other means (e.g., reference broadcast synchronization [6]). The estimate layer encapsulates all means by which nodes can estimate the clock values of other nodes, and eliminates the need to reason explicitly about delay bounds and other parameters of the system.

The estimate layer provides an *estimate graph*, where each edge $\{u, v\}$ represents the fact that node u has some means of estimating v's current clock value and vice versa. The edges of the estimate graph are not necessarily direct communication links between nodes (see [6] for examples).

[1] Unless otherwise specified, times are always in $\mathbb{R}_0^+$.

[2] This requirement can be dropped. It is introduced to simplify the presentation. The same results can be derived even for discontinuous (in particular discrete) clocks by approximating the true clocks by left-differentiable functions and accounting for the difference in the uncertainty of estimates.

Node u is provided with a *local estimate* $\tilde{L}_u^v$ of L_v, whose accuracy is guaranteed by the estimate layer:

$$\forall t\ \forall u \in V, v \in N_u(t) : |L_v(t) - \tilde{L}_u^v(t)| \le \epsilon_{\{u,v\}}, \tag{1}$$

where $\epsilon_{\{u,v\}} \in \mathbb{R}^+$ is called the *uncertainty*, or the *weight*, of the edge $\{u, v\}$, and $N_u(t)$ is the set of neighbors of u at time t, which will be formally introduced shortly. The uncertainty of a path $p = (u_0, u_1, \dots, u_k)$, ϵ_p is defined as

$$\epsilon_p := \sum_{i=1}^{k} \epsilon_{\{u_{i-1},u_i\}}.$$

In the sequel, we refer to estimate edges of the sort described above simply as *edges*; similarly, when we say "the graph" we mean the estimate graph. We do not reason explicitly about the communication graph, as the salient aspects of communication are encapsulated by the estimate layer.

Dynamic networks. We consider dynamic networks over a fixed set of nodes V of size $n := |V|$. Edge insertions and removals are modeled as discrete events controlled by a worst-case adversary. In keeping with the abstract representation from [12], we say that there is an *estimate edge* $\{u, v\}$ between two nodes $u, v \in V$ at time $t \ge 0$ iff u and v have a means of obtaining clock value estimates about each other at time t. As explained above, this does not necessarily mean that there is a direct communication link between u and v at time t.

We do not assume that nodes detect the formation or failure of a communication link between them at the same time, which introduces some asymmetry into the model. Hence, we model the network as a *directed dynamic graph* $G = (V, E)$, where $E : \mathbb{R}_0^+ \to 2^{(V\times V)}$ maps non-negative times t to a set of directed estimate edges $E(t)$ that exist at time t. If $(u, v) \in E(t)$, then at time t node u has an estimate for node v's logical clock, but not necessarily vice-versa. Formally, the set of node u's neighbors at time t is defined as $N_u(t) := \{v \mid (u, v) \in E(t)\}$. We assume that any asymmetry in the graph corresponds to the delay in nodes finding out about link status changes and is only temporary; this is explained below.

In the following, we frequently refer to *undirected* edges $\{u, v\}$; when we write $\{u, v\} \in E(t)$, we mean that both $(u, v) \in E(t)$ and $(v, u) \in E(t)$. We say that edge $\{u, v\}$ exists throughout a time interval $[t_1, t_2]$ if for all $t \in [t_1, t_2]$ we have $\{u, v\} \in E(t)$. By extension, a path p is said to exist throughout $[t_1, t_2]$ if all its edges exist throughout the interval.

Each undirected estimate edge $\{u, v\}$ is associated with three parameters:

- The *estimate uncertainty* $\epsilon_{\{u,v\}}$, as explained above.
- The *detection delay* $\tau_{\{u,v\}}$. We assume that u and v detect if the edge disappears "at" the respective other node within $\tau_{\{u,v\}} \in \mathbb{R}^+$ time. Formally,
  (a) if $(u, v) \notin E(t)$, then there is some time $t' \in [t - \tau_{\{u,v\}}, t + \tau_{\{u,v\}}]$ so that $(v, u) \notin E(t')$; and, symmetrically,
  (b) if $(v, u) \notin E(t)$, then there is some time $t' \in [t - \tau_{\{u,v\}}, t + \tau_{\{u,v\}}]$ so that $(u, v) \notin E(t')$.
- We assume that u and v can exchange messages with *message delay* $\mathcal{T}_{\{u,v\}}$. More precisely, nodes that share an estimate edge can actively exchange information if required (possibly

through other nodes, if there is no direct communication link between them), and $\mathcal{T}_{\{u,v\}}$ bounds how long such communication might be delayed. Formally, if u sends a message at time t and $u \in N_v(t')$ for all $t' \in [t, t + \mathcal{T}_{\{u,v\}}]$, then v will receive this message at some time $t'' \in [t, t + \mathcal{T}_{\{u,v\}}]$.[3] If u is not during this entire interval in N_v, the message may or may not be delivered; if it is delivered, however, it is guaranteed to arrive within the specified interval.

We remark that our algorithm will use explicit communication by messages as above only upon formation of an edge, to perform a simple handshake.

Causality and the dynamic estimate diameter. While local message exchange may be infrequent, flooding techniques may ensure quick dissemination of timing information on a global level without necessitating a large (amortized) number of messages per time unit. We therefore characterize how information propagates through the dynamic graph without imposing a particular communication structure. In this context, we are interested in the global skew. Our algorithm will ensure that any node whose logical clock attains the current maximum clock value will run at the speed of the hardware clock, i.e., no faster than at rate $1+\rho$. Further, the logical clock of any node always runs at least at rate $1-\rho$. For estimating the global skew, the maximum logical clock speed $(1+\rho)(1+\mu)$ is of no significance. More generally, this is true for any algorithm that satisfies an optimal envelope condition, i.e., that guarantees the best approximation of real-time offered by the hardware clocks.

For a synchronization message M sent from u at time t that is received by v at time $t' > t$ let $U(M)$ denote the uncertainty in its delay, i.e., in particular the receiver v knows that M was in transit for at least $t' - t - U(M)$ time units (clearly, $U(M) \leq \mathcal{T}_{\{u,v\}}$ but potentially it is much smaller).

We define the family of relations $\overset{\eta}{\rightsquigarrow}$, $\eta \in \mathbb{R}_0^+$, on $V \times \mathbb{R}$ as specified below. Intuitively, for node u and v, times t and $t' \geq t$, and a value $\eta \geq 0$, $(u,t) \overset{\eta}{\rightsquigarrow} (v,t')$ can be interpreted as follows. At time t', node v can lower bound u's clock value at time t (hardware or logical) with an error of at most η. Specifically,

- $\forall u \in V$, $\forall t : (u,t) \overset{0}{\rightsquigarrow} (u,t)$ (u knows its own clock perfectly).
- $\forall u, v \in V$, $\forall t'' \geq t' \geq t$, $\forall \eta \in \mathbb{R}_0^+ : (u,t) \overset{\eta}{\rightsquigarrow} (v,t') \Rightarrow (u,t) \overset{\eta'}{\rightsquigarrow} (v,t'')$, where $\eta' := \eta + \frac{4\rho}{1+\rho}(t''-t')$ (v knows that u's clock runs at least at $\frac{1-\rho}{1+\rho}$ times the rate of v's hardware clock. The maximum error is obtained if u's clock runs at rate $1+\rho$ and v's hardware clock runs at rate $1-\rho$).
- If M is a message sent by v at time t' and received by w at time $t'' \geq t'$, then $\forall u \in V$, $\forall t \leq t'$, $\forall \eta \in \mathbb{R}_0^+ : (u,t) \overset{\eta}{\rightsquigarrow} (v,t') \Rightarrow (u,t) \overset{\eta'}{\rightsquigarrow} (w,t'')$, where $\eta' := \eta + (1-\rho)U(M) + 2\rho(t''-t')$ (u's hardware clock progresses by at most $(1+\rho)(t''-t')$ during the transit time, but w can safely add $(1-\rho)(t''-t'-U(M))$ to the estimate).

A fundamental lower bound [2] shows that the performance of a CSA in a static network depends on the diameter of the network. In dynamic networks there is no immediate equivalent to

[3]Note that the neighbor relation seems to be "reversed" here in the prerequisite for the reception of a message. This definition reflects that the estimate edge must exist for the node receiving the message. However, this detail is irrelevant to the functionality of the algorithm.

a diameter. Informally, the diameter corresponds to the time it takes (at most) for information to spread from one end of the network to the other. The above relation integrates this information with the amount of uncertainty that is attached to this communication; this is crucial in our scenario comprising heterogeneous edges since, for instance, a communication path that is slower in terms of the time it takes to traverse it might yield much more accurate estimates of clock values.

Definition 3.1 (Dynamic Estimate Radius and Diameter)**.** *Given a dynamic graph G, we say that node $v \in V$ has a* dynamic estimate radius *of $R_v(t)$ at time t if for every $u \in V$, there is some $t' \leq t$ so that $(u,t') \overset{R_v(t)}{\rightsquigarrow} (v,t)$, where $R_v(t)$ is minimal with this property. Moreover, G has a* dynamic estimate diameter $D(t) := \max_{v \in V}\{R_v(t)\}$ *(or simply "diameter" for short).*

Because this definition refers to the actual communication, (some) dynamic estimate radii might be much smaller than the dynamic diameter at the same instant of time. Moreover, both values strongly depend on the structure of message exchange. However, the lower bounds from the static case apply in the sense that the dynamic estimate diameter is lower bounded in terms of the maximum over all pairs of nodes v, w of the minimal sum of uncertainties on *any* possible communication path from v to w. Hence, if the communication layer provides an asymptotically optimal dynamic diameter, a global skew bound that behaves roughly as $\mathcal{O}(D(t))$ (neglecting disturbances due to large fluctuations of $D(t)$) is asymptotically optimal.

As the primary focus of this work is not on the global skew, we refrain from further discussing these points except for the following remark. We can make an arbitrary node u_0 artificially faster (by multiplying its hardware clock rate by $(1+\rho)/(1-\rho)$) so that it is always the node with the maximal hardware clock value in the network. This can be seen as replacing its hardware clock by one of drift $\tilde{\rho} \leq (1+\rho)^2/(1-\rho) - 1 \approx 3\rho$. A CSA can easily guarantee that this node also has the largest logical clock value in the network at all times. All our statements then apply if we replace the drift bound ρ by $\tilde{\rho}$ and $D(t)$ by $R_{u_0}(t)$, which might be beneficial in networks with a large discrepancy between $D(t)$ and $R_{u_0}(t)$.

3.2 Dynamic Clock Synchronization

Throughout the paper, we frequently refer to *the skew on a path* $p = (u_0, \ldots, u_k)$ at time t, by which we mean $|L_{u_0}(t) - L_{u_k}(t)|$. The goal of a CSA is to minimize the skew on all paths.

To measure the quality of a CSA we consider two kinds of requirements: a *global skew constraint* which gives a bound on the difference between any two logical clock values in the system, and a *gradient skew constraint*, which becomes stronger the closer two nodes u, v are in the subgraph induced by the edges that have been present for a sufficiently long time to stabilize. In particular, for nodes that remain neighbors for a long time, the gradient skew constraint imposes a much smaller permissible clock skew than the global skew constraint.

Definition 3.2 (Global Skew)**.** *For any time t, a CSA guarantees a* global skew *of $\mathcal{G}(t)$, if for any two nodes $u, v \in V$ it holds that $L_u(t) - L_v(t) \leq \mathcal{G}(t)$.*

Definition 3.3 (Stable Gradient Skew)**.** *Given a non-decreasing function $\mathcal{S} : \mathbb{R}_0^+ \to \mathbb{R}_0^+$, we say that a CSA $\mathcal{A}$ guarantees a* stable gradient skew *of $\mathcal{S}$ with* stabilization time $\mathcal{T}_S$ *if for each time t and each path $p = (u_0, \ldots, u_k)$ that exists throughout $[t - \mathcal{T}_S, t]$, we have that*

$$L_{u_0}(t) - L_{u_k}(t) \leq \mathcal{S}(\epsilon_p).$$

More generally, one can express the skew bound as a function of the length of the time interval during which the path p existed (cf. [10]). The literature on gradient clock synchronization (e.g., [8, 11, 16, 18]) is typically concerned with the *local skew* of a CSA, which bounds the skew on any single edge. The local skew can be considered equivalent to the stable gradient skew $\mathcal{S}(1)$, provided that all edges are of uniform weight 1.

The stable gradient skew and the stabilization time are functions of D, a bound on the dynamic estimate diameter of the network that held for sufficient time (and are thus inherently dependent on t as well), and potentially other parameters such as the bound on the clock drift ρ or the minimum edge weight. We usually omit these dependencies to simplify the notation.

4 An Optimal Dynamic Gradient CSA

In this section we describe a CSA $\mathcal{A}^{\text{OPT}}$ which achieves the optimal stable gradient skew, and reaches this stable skew in the optimal stabilization time, in light of the trade-off presented in Section 8. We begin in Section 4.1 by introducing the overall strategy used to achieve a stable skew of $\Theta(d \log(D/d))$ in static graphs; this strategy also underlies the design of the dynamic algorithm. In Section 4.2, we give an informal overview of the algorithm, and the technical details follow in Section 4.3. We remark that both the description of the algorithm and in particular its analysis given in Section 5 is complicated by a number of technical details that need to be resolved, but may obfuscate the key ideas behind the reasoning. We refer the reader to [14] for a simplified presentation in a less involved (but unrealistic) model focusing on the key aspects of the problem and its analysis.

4.1 Achieving a Stable Skew of $\Theta(d \log(D/d))$

The optimal static algorithm [12, 16] and the algorithm we present here share the same high-level structure. Both achieve a (static or stable) gradient skew of $\Theta(d \log_\sigma(D/d))$ on paths of length (or weight) d, where the base σ of the logarithm is a function of the parameter μ and the drift ρ. In this section we introduce several notions that underlie the design of both algorithms. For simplicity, we ignore here the dynamic behavior of the graph, and present the *static-graph* version of the definitions (as used in [12, 16]), assuming that all edge weights are 1. This version is simpler than the weighted dynamic-graph version and is helpful in understanding the dynamic algorithm. In Section 5 we give the full dynamic versions of these notions and use them to analyze the dynamic skew of the algorithm.

The static algorithm is based on a discretized version of the gradient skew requirement. Let $C = \{C_s\}_{s \in \mathbb{N}}$ be the non-increasing sequence defined by $C_s := D/\sigma^s$. The algorithm guarantees the following condition (up to constants we neglect here): for any path $p = (u_0, \ldots, u_k)$ and any integer $s \in \mathbb{N}$, if the path p has length $d_p \geq C_s$, then at all times t we have

$$L_{u_0}(t) - L_{u_k}(t) \leq s \cdot d_p.$$

This discretized condition is equivalent to the standard $\Theta(d \log_\sigma(D/d))$-gradient skew requirement: if p is a path of length d_p, then for $s = \lceil \log_\sigma(D/d_p) \rceil$ we have

$$C_s = \frac{D}{\sigma^{\lceil \log_\sigma(D/d_p) \rceil}} \leq d_p,$$

and therefore the discretized condition asserts that the skew on p is no greater than $s \cdot d_p = \lceil \log_\sigma(D/d_p) \rceil \cdot d_p \in \Theta(d_p \log_\sigma(D/d_p))$.

From the algorithm's point of view, the discretized condition divides the paths into *levels*, where paths of level s are of length $d \approx D/\sigma^s$ and the skew on such paths is upper bounded by $s \cdot d$. If we evenly distribute the permissible skew over the edges of the path, we see that each of the d edges should only contribute a skew of roughly s to the total. And indeed, this is exactly what each node executing the algorithm tries to accomplish: it tries to ensure that for all $s \in \mathbb{N}$, none of its edges exhibit a skew of more than s. Similarly, in the weighted version of the static algorithm [12], each node tries to ensure that no adjacent edge of weight w_e carries a skew of more than $s \cdot w_e$, so that when we sum over all the edges of a path p of weight w_p the total skew will be no more than $s \cdot w_p$. The overall gradient skew is then $\mathcal{O}(w_p \cdot \log_\sigma(D/w_p))$, a direct generalization of the unweighted case.

The description above is informal but we will see that tests of the form "is there some neighbor whose clock is more than $s \cdot w_e$ ahead or behind?" make up the basis of the algorithm. Essentially, through such tests nodes check if their adjacent edges contribute more than their fair share of the skew on some path.

If a node finds that the skew over some of its edges is too large, it can adjust the speed of its logical clock to compensate. The algorithm uses only two rates, a *slow rate* and a *fast rate*. When a node uses the slow rate we say that it is in *slow mode*, and when it uses the fast rate we say that it is in *fast mode*. At the heart of the algorithm are the rules for deciding which mode to use; we proceed to describe these rules, which are based on the static rules from [12, 16] but also take into account the dynamic behavior of the graph.

4.2 Overview of the Algorithm

When an edge first appears, the algorithm is first concerned with reducing the skew on *long* paths that contain the edge. Once this is accomplished, it allows the skew on shorter paths to also be reduced, and then on even shorter paths, until eventually the skew on individual edges is reduced to its stable value. In some sense, the algorithm takes the global skew $\mathcal{G}$, which cannot be avoided, and *redistributes* it throughout the network until the gradient property is satisfied. Notice that longer paths have a larger (that is, weaker) gradient skew bound, so they are in some sense easier to deal with. In particular, for the longest paths in the network, the gradient skew bound is the same as the global skew bound. Since the global skew bound holds for any two nodes in the network, it can never be violated by adding new edges, so these longest paths immediately satisfy their gradient skew requirement as soon as they appear.

Neighbor sets. Throughout the algorithm, each node partitions its neighbors according to the amount of time it has had an edge to each neighbor. More precisely, each node u maintains an ordered list $N_u^0, N_u^1, \ldots$ of neighbor sets, where $N_u^0 \supseteq N_u^1 \supseteq \ldots$. To simplify the presentation we initially assume an infinite list of sets; we will later see that nodes only need to store a finite prefix of the list, but we defer this discussion to a later point. Moreover, as the neighbor sets change at discrete times, we need a convention what $N_u^s(t)$ means if the set is modified at time t. We define that if node v is added to N_u^s at time t^- and removed at time t^+ (without intermediately leaving the set) then $v \in N_u^s(t)$ for all $t \in [t^-, t^+]$. We assume that the neighbor sets change only finitely often in finite time, implying that for all u, v, and s, the set $\{t \,|\, v \in N_u^s(t)\}$ is closed.[4]

Informally, if $v \in N_u^s$ at time t, then at time t node u is concerned with maintaining a good skew on paths of level s containing edge $\{u, v\}$. In contrast, if $v \notin N_u^s$, then node u is "not worried" about level s paths containing $\{u, v\}$. Accordingly, when an edge $\{u, v\}$ is discovered, node u first adds v to N_u^0, then after some time it adds v to N_u^1, and so on. Recall from Section 4.1 that the index s of a level decreases as the length of the path in the level increases, so adding edges in the order $N_u^0, N_u^1, \dots$ corresponds to dealing first with longer paths and then with shorter ones.

Specifically, when node u first discovers edge $\{u, v\}$, it *immediately* adds v to N_u^0; hence $N_u = N_u^0$, because this is the set of all neighbors that node u has discovered. Each of the remaining sets is updated within time $\Theta(\mathcal{G}/\mu)$. The sets are updated in a loosely synchronized manner. Both nodes u and v coordinate adding the edge $\{u, v\}$ to their respective sets. In a time interval during which nodes add edges to their level s neighbor set N_u^s, we can only show non-trivial gradient skew guarantees for levels different from s. In order to always have non-trivial guarantees for the skew on paths of all lengths, we need to loosely synchronize the insertions of different edges such that insertions of different edges on different levels are sufficiently separated from each other. The details appear in Section 4.3.

Whenever a node u discovers that one of its edges $\{u, v\}$ has disappeared, it immediately removes node v from all neighbor sets $N_u^0, N_u^1, \dots$ Finally, for simplicity, we assume that at time 0, $N_u(0)$ contains all edges that are present at time 0 and all neighbor sets N_u^s are initialized to $N_u(0)$, i.e., for all $s \geq 0$, $N_u^s(0) = N_u(0)$ as there is no violation at time 0.

The fast and slow conditions. Each edge e is associated with a weight κ_e, which roughly corresponds to the uncertainty ϵ_e of the edge. The algorithm is designed to guarantee the following conditions governing when a node is in fast or in slow mode. These conditions are not the actual rules used by nodes to determine when to enter fast or slow mode, but we will see in Section 4.3 that the rules are quite similar; the conditions we give here refer to the clock values of neighbors, which a node cannot estimate exactly, and the actual triggers for entering fast or slow mode have to compensate for this inaccuracy. We will see in Section 5 that the fast and slow mode triggers (given in Section 4.3) implement the fast and slow mode conditions (given below).

The first condition, **FC**, specifies when a node u must be in fast mode. It states that some neighbor in N_u^s is "too far ahead of u", and no other neighbor in N_u^s is "too far behind", where "too far" here roughly corresponds to s times the weight of the edge (as outlined in Section 4.1).

Definition 4.1 (**FC**: The Fast Mode Condition)**.** *For all $s \in \mathbb{N}, u \in V$, and times t, if*

- *For some $w \in N_u^s(t)$ we have $L_w(t) - L_u(t) \geq s \cdot \kappa_{\{u,w\}}$, and*
- *For all $v \in N_u^s(t)$ we have $L_u(t) - L_v(t) \leq s \cdot \kappa_{\{u,v\}} + 2\mu\tau_{\{u,v\}}$,*

then node u is in fast mode at time t.

The term $2\mu\tau_{\{u,v\}}$ in the second requirement compensates for the drift that can accumulate on an edge while only one of its endpoints is aware of its existence (recall that the length of this period is bounded by $\tau_{\{u,v\}}$).

[4]This convention simplifies the notation in our proofs. However, since clocks are continuous functions, this convention does not bear any implication for the behavior of the algorithm.

The condition **SC** for being in slow mode is roughly symmetric to the fast mode condition: it states that some node in N_u^s is "too far behind u", and no other node in N_u^s is "too far ahead". The condition uses a value $\delta > 0$ that corresponds to the smallest uncertainty in the network. An exact value will be defined in Lemma 5.2; the algorithm is oblivious of δ and is correct if there exists *any* such $\delta > 0$ under which the slow mode condition is satisfied, and Lemma 5.2 shows that such a value exists.

Definition 4.2 (**SC**: The Slow Mode Condition)**.** *For all $s \in \mathbb{N}, u \in V$, and times t, if*

- *For some $w \in N_u^s(t)$ we have $L_u(t) - L_w(t) \geq \left(s + \frac{1}{2}\right) \cdot \kappa_{\{u,w\}} - \delta$, and*
- *For all $v \in N_u^s(t)$ we have $L_v(t) - L_u(t) \leq \left(s + \frac{1}{2}\right) \cdot \kappa_{\{u,v\}} + \delta + \mu(1+\rho)\tau_{\{u,v\}}$,*

then node u is in slow mode at time t.

The slow mode condition uses a slightly different value for "too far" from the fast mode condition. There are two immediate reasons for this: first, the conditions for being in fast mode and in slow mode must be mutually exclusive, otherwise a node might be required to be in both modes at the same time; hence the term $(s + 1/2)$ instead of s. And second, the slack δ is necessary to smooth out the discontinuities that occur when a neighbor is removed from N_u^s, by providing a small region around $(s + 1/2) \cdot \kappa_e$ in which a neighbor of u that is behind u can still keep node u in slow mode. Lemma 6.5 below captures this intuition formally and shows how the slack δ is used. Moreover, for technical reasons in **SC** a smaller term of $(1 + \rho)\mu\tau_{\{u,v\}}$ is sufficient to address the issue that skew may accumulate while only one endpoint of an edge is aware of the edge.

Note that the fast and slow mode conditions are disjoint, and their union does not cover the entire state space. In such cases nodes choose their mode according to their estimate of the maximum logical clock in the network, as described below.

Max estimates. As in [11, 16, 17, 18], each node maintains a local estimate M_u of the maximum logical clock value in the network. Max estimates are computed by flooding: each node always adds its current estimate M_u to each message it sends, and updates the estimate conservatively so that it cannot exceed the actual maximum logical clock value in the system. When a node receives a larger max estimate from some neighbor, it updates its own max estimate to match. The max estimates are computed such that the following constraints can be guaranteed.

Condition 4.3. *If the dynamic graph has a dynamic estimate diameter of $D(t)$, then for all $t \geq 0$ and for all nodes u we have*

$$M_u(t) \leq \max_{v \in V} \{L_v(t)\}, \tag{2}$$

$$M_u(t) \geq \max_{v \in V} \{L_v(t)\} - D(t), \tag{3}$$

$$M_u(t) \geq L_u(t), \tag{4}$$

Specifically, node u updates its max estimate M_u as follows. Whenever $M_u = L_u$, node u increases M_u at the rate of its logical clock. If $M_u > L_u$, node u has to make sure that it only increases M_u at a rate such that M_u remains upper bounded by the largest logical clock L_v in the network. As the largest logical clock progresses at rate at least $1 - \rho$ and node u's hardware clock progresses at rate at most $1 + \rho$, this can be achieved if u increases M_u at rate $\frac{1-\rho}{1+\rho}$ times the rate of its hardware clock. These rules suffice to guarantee (2) and (4). In order to also guarantee (3), nodes piggy-back their current max estimate to each message sent. Whenever a node u receives a message from a node v, u increases its max estimate to the largest possible value such that M_u is guaranteed to remain upper bounded by M_v (or by the existing max estimate M_u if that is larger). Condition (3) now follows directly from the definition of $D(t)$.

Note that as a result of Condition 4.3, the max estimate of any node is always accurate up to the diameter $D(t)$. In addition, (4) asserts that nodes cannot set their logical clock ahead of their max estimate. The max estimate $M_u(t)$ is used to determine the mode of node u when neither **FC** nor **SC** are satisfied.

Definition 4.4 (**MC**: The Max Estimate Condition)**.** *For all $u \in V$ and times t:*

- *If $L_u(t) = M_u(t)$ and for all $v \in N_u(t)$, we have $L_u(t) \geq L_v(t)$, then node u is in slow mode at time t.*
- *If $L_u(t) \leq M_u(t) - \iota$ and for all $v \in N_u(t)$, we have $L_u(t) \leq L_v(t)$, then node u is in fast mode at time t,*

where $\iota > 0$ is some small constant used to separate the two conditions.[5]

Global skew estimates. At all times $t \geq 0$, the algorithm requires each node u to have an estimate $\tilde{\mathcal{G}}_u(t)$ of the global skew $\mathcal{G}(t)$. We require that

$$\text{For all nodes } u \in V \text{ and for all times } t \geq 0,\ \tilde{\mathcal{G}}_u(t) \geq \mathcal{G}(t). \tag{5}$$

It turns out that a lack of guarantees on the accuracy of these estimates and/or their speed of change over time and across the network significantly complicates edge insertion. For the sake of a more accessible presentation, we thus assume a static (i.e., neither time- nor node-dependent) global skew estimate

$$\text{For all times } t \geq 0,\ \tilde{\mathcal{G}} \geq \mathcal{G}(t). \tag{6}$$

for now. Note that $\tilde{\mathcal{G}}$ must be chosen conservatively, as it must bound the global skew for all times, and thus relying on it may result in unnecessarily slow edge insertions. We will discuss how to adapt edge insertion to the much weaker condition (5) in Section 7, alongside a proof of the resulting (time-dependent) gradient property.

4.3 Detailed Description of the Algorithm

We describe the parameters and constants used to define the algorithm, the local variables maintained at each node, and finally the continuous and discrete transitions that modify these variables.

4.3.1 Parameters and Constants

ρ: As specified in Section 3, the constant $\rho \in (0, 1)$ specifies an upper bound on the drift of the hardware clocks.

[5] The analysis of the algorithm goes through even if we change the condition for entering fast mode to $L_u(t) < M_u(t)$. However, such a requirement cannot be realized, because there is no "first point in time" when $L_u(t) < M_u(t)$. To ensure that the algorithm is realizable we make sure that when we require a node to be in a certain mode, the conditions of the requirement form a closed region, and are strictly separated from any other requirement.

μ: This parameter governs the fastest possible logical clock rate. In slow mode, the logical clock is increased at the same rate as the hardware clock, and in fast mode the rate of the hardware clock is multiplied by $1+\mu$. The value of μ is bounded from below as a function of the drift ρ, because we must ensure that a node in fast mode is *always* faster than a node in slow mode, even when the hardware clock progresses slowly for the node in fast mode and quickly for the node in fast mode. To ensure that for any $u, v \in V$ we always have $(1+\mu)h_u(t) > h_v(t)$ it is sufficient to require $(1+\mu)(1-\rho) > 1+\rho$, which is equivalent to $\sigma > 1$ (see below). For technical reasons, we require that

$$\mu \leq \frac{1}{10}. \tag{7}$$

σ: The base of the logarithm in the desired gradient skew function, which is $\Theta(d \log_\sigma(D/d))$. To obtain the best asymptotic gradient skew bound, we set

$$\sigma := \frac{(1-\rho)\mu}{2\rho} > 1, \tag{8}$$

and control the base of the logarithm by setting the value of μ appropriately. Clearly, we must require that $\sigma > 1$, which imposes the constraint that $\mu > 2\rho/(1-\rho)$.

$\kappa_{\{u,v\}}$: Each edge $\{u,v\} \in \binom{V}{2}$ is associated with a *weight* $\kappa_{\{u,v\}}$, corresponding roughly to the uncertainty $\varepsilon_{\{u,v\}}$. The weights must satisfy

$$\kappa_{\{u,v\}} > 4(\epsilon_{\{u,v\}} + \mu\tau_{\{u,v\}}). \tag{9}$$

The term $4\mu\tau_{\{u,v\}}$ compensates for the time during which the edge $\{u,v\}$ only exists for one of its nodes u and v. Otherwise, the asymmetric behavior could result in one of the nodes erroneously being in fast mode (or, similarly, slow mode). Therefore, the time uncertainty of $\tau_{\{u,v\}}$ with respect to the symmetric existence of edges is reflected in κ with a prefactor of μ; we remark that μ also occurs as a factor in a term contributing to $\epsilon_{\{u,v\}}$ (in any system), as there must be a non-zero delay for propagating information about L_u to v.

4.3.2 Local Variables

Each node u maintains the following local variables throughout the execution of the algorithm.

L_u: the logical clock of node u.

$mult_u$: the current rate-multiplier for node u's logical clock. It can take only two values, 1 or $1+\mu$; when u is in slow mode we have $mult_u = 1$, and when u is in fast mode we have $mult_u = 1+\mu$.

M_u: node u's current estimate for the maximum logical clock in the network.

$N_u = N_u^0$: the set of all neighbors node u is aware of.

$N_u^1, N_u^2, \ldots$: the neighbor sets of node u for each of the levels it maintains.

***$\tilde{\mathcal{G}}_u$*:** the nodes' current global skew estimate. We assume that at all times t, $\tilde{\mathcal{G}}_u(t)$ is an upper bound on the actual global skew at time t. Prior to Section 7, we assume that simply $\tilde{\mathcal{G}}_u(t) = \tilde{\mathcal{G}}$ for all nodes and times.

$T_0{}^{\{u,v\}} < T_1{}^{\{u,v\}} < T_2{}^{\{u,v\}} < \ldots$: the logical times for adding edge $\{u,v\}$. For each edge $\{u,v\}$ and each $s = 1, 2, \ldots$, nodes u and v decide on logical times $T_1^{\{u,v\}}, T_2^{\{u,v\}}, \ldots$ when they add the respective neighbor to their respective level-s neighbor set. That is, node u adds v to N_u^s when its logical clock reaches $L_u(t) = T_s^{\{u,v\}}$. For convenience, each node also maintains a logical time $T_0^{\{u,v\}}$ that is used to define the times $T_s^{\{u,v\}}$ for $s \geq 1$. Note that the nodes u and v use the same values for $T_0^{\{u,v\}}, T_1^{\{u,v\}}, \ldots$, but because their logical clocks are not perfectly synchronized, they may update their neighbor sets at different times. For each edge $\{u,v\}$, the times $T_1^{\{u,v\}}, T_2^{\{u,v\}}, \ldots$ define a converging sequence, so that edges can be added on all infinite levels in finite time. In fact, in the analysis we assume that all nodes update all sets, that is, they use infinite levels. Note however, that only a $\mathcal{O}(\log \tilde{\mathcal{G}})$ levels are needed in the algorithm. Further, the times $T_s^{\{u,v\}}$ only depend on $T_0^{\{u,v\}}$ and on $\tilde{\mathcal{G}}_{\{u,v\}}$. All neighbor sets $N_u^s(t)$ are therefore implicitly given by $L_u(t)$ and some bounded additional information for each edge $\{u,v\}$, so that the sets $N_u^s(t)$ could be maintained by only managing a constant number of values per edge.

4.3.3 Rules for Updating the Local Variables

The algorithm makes three kinds of discrete transitions: the first kind occurs when a node discovers the formation or failure of a communication link. The second kind occurs when a node u's logical clock reaches an update time $T_s^{\{u,v\}}$ for some $s \in \mathbb{N}$ and an incident new edge. The responses to these events are given in Listing 1.

The third and final kind of transition is triggered when the slow mode trigger, the fast mode trigger, or the max estimate triggers, which correspond to **SC**, **FC**, and **MC** and will be stated shortly, require the node to change its mode; the logic governing a node's mode is shown in Listing 3. When no trigger holds and **MC** does not hold, the node is free to choose its mode nondeterministically; for example, it can stay in its current mode until it is required to switch modes.[6]

Between discrete transitions the value of each node u's logical clock increases at a rate of $l_u = mult_u \cdot h_u(t)$. In the remainder of the section we describe the algorithm's discrete transitions.

Coordinating with new neighbors and calculating the insertion times. When a new edge is formed, the two nodes start a simple protocol during which they agree on the logical times for adding each other to the respective neighbor sets. For simplicity, we assume that for each potential edge $\{u,v\}$, one of the two nodes u and v is the leader of the edge. This can for example be determined by assuming that nodes u and v have unique identifiers.[7] Assume that node u is the leader of an edge $\{u,v\}$. As soon as node u discovers the edge $\{u,v\}$, it starts the protocol for adding $\{u,v\}$. In order to make sure that also node v has discovered the edge, node u first waits for

[6]For simplicity it is assumed that the code in Listing 3 is evaluated *continuously*, so that, for example, as soon as the fast mode trigger holds for some node, that node is *already* in fast mode. An implementation of the algorithm can achieve this by adding a small "guard region" to the conditions, and changing mode *before* the triggers hold. All the triggers are strictly separated from each other, so such regions can be added to each trigger.

[7]If we drop the assumption that we can predefine a leader for each potential edge, it would be possible to use a more complicated handshake protocol to coordinate between the two nodes of an edge.

Listing 1: Responses to other events at node u

$\Delta := \frac{(1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}}+\tau_{\{u,v\}})}{1-\rho} + \tau_{\{u,v\}}$

when formation of an edge $\{u, v\}$ to node v is discovered:

  $N_u^0 := N_u^0 \cup \{v\}$

  if *u is the leader of the edge $\{u, v\}$* **then**

    wait for at least Δ time

    if *$v \in N_u^0(t')$ for all t' with $L_u(t') \in [L_u(t) - (1+\rho)(1+\mu)\Delta, L_u(t)]$* **then**

      $\tilde{\mathcal{G}}_{\{u,v\}} := \tilde{\mathcal{G}}_u$ // we assume $\tilde{\mathcal{G}}_u = \tilde{\mathcal{G}}$, except in Section 7

      $L_{ins} := L_u + \tilde{\mathcal{G}}_{\{u,v\}} + (1+\rho)(1+\mu)\mathcal{T}_{\{u,v\}}$

      send insertedge $\left(\{u, v\}, L_{ins}, \tilde{\mathcal{G}}_{\{u,v\}}\right)$ to v

      call computeInsertionTimes $\left(\{u, v\}, L_{ins}, \tilde{\mathcal{G}}_{\{u,v\}}\right)$

when receiving message insertedge $\left(\{u, v\}, L_{ins}, \tilde{\mathcal{G}}\right)$ from node v:

  wait for at least $\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}}$, but at most $\Delta - \tau_{\{u,v\}}$ time

  if *$v \in N_u^0(t')$ for all t' with $L_u(t') \in [L_u(t) - (1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}}), L_u(t)]$* **then**

    call computeInsertionTimes $\left(\{u, v\}, L_{ins}, \tilde{\mathcal{G}}\right)$

when failure of an edge to node v is discovered:

  foreach $s \in \{0, 1, \ldots\}$ **do**

    $N_u^s := N_u^s \setminus \{v\}$

    $T_s^{\{u,v\}} := \perp$

when $L_u = T_s^{\{u,v\}}$ (for some s and v)

  $N_u^s := N_u^s \cup \{v\}$

at least $\tau_{\{u,v\}}$ time units (w.r.t. real time). If the edge exists throughout that waiting period, node u decides on a global skew estimate $\tilde{\mathcal{G}}_{\{u,v\}}$ for the edge insertion (which is just node u's current global skew estimate $\tilde{\mathcal{G}}_u$) and a logical time to start adding the edge. Node u sends this information to node v. If node v sees the edge when receiving the information, it computes the edge insertion times based on the received information. The protocol guarantees that a) either both nodes insert the edge or they do not start inserting or cancel the insertion within $\tau_{\{u,v\}}$ time units of each other, and b) if both nodes insert the edge, they use the same insertion times and global skew estimate for the insertions. For further details and a formal argument, we refer to Lemma 5.5. Pseudo-code of the coordination protocol is given in Listing 1. The computation of the insertion times based on a logical time for start inserting and a given global skew estimate is given in Listing 2. When inserting an edge $\{u, v\}$, u and v compute a time interval of length $\mathcal{I}_{\{u,v\}}$ during which the edge $\{u, v\}$ is inserted on all levels. The duration $\mathcal{I}_{\{u,v\}}$ depends on the global skeq estimate $\tilde{\mathcal{G}}_{\{u,v\}}$ of the edge and it is computed differently depending on whether we work with a fixed, static global skew estimate $\tilde{\mathcal{G}}$ or whether the global skew estimate is allowed to be dynamically adapted. Outside Section 7, we assume the global skew estimate to be a fixed value $\tilde{\mathcal{G}}$. The insertion duration $\mathcal{I}_{\{u,v\}}$

Listing 2: Calculating insertion times

procedure computeInsertionTimes $\left(\{u,v\}, L, \tilde{\mathcal{G}}\right)$ **:**
  Compute $\mathcal{I}_{\{u,v\}} := \mathcal{I}(\tilde{\mathcal{G}}_{\{u,v\}})$ according to (10) or (11)
  $T_0^{\{u,v\}} := \min\left\{T \geq L \,:\, \frac{T}{\mathcal{I}_{\{u,v\}}(\tilde{\mathcal{G}})} \in \mathbb{Z}\right\}$
  for $s \in \{1, 2, \ldots\}$ **do**
    $T_s^{\{u,v\}} := T_0^{\{u,v\}} + \left(1 - \frac{1}{2^{s-1}}\right)\mathcal{I}_{\{u,v\}}(\tilde{\mathcal{G}})$

of an edge $\{u, v\}$ is then computed as

$$\mathcal{I}_{\{u,v\}} := \mathcal{I}(\tilde{\mathcal{G}}) := \left(\frac{20(1+\mu)}{(1-\rho)} + 56\mu + \frac{8+56\mu}{\sigma}\right) \cdot \frac{\tilde{\mathcal{G}}}{\mu}. \tag{10}$$

In Section 7, we show how our clock synchronization algorithm can adapt to a changing global skew. In this case, the time for inserting an edge has to be chosen larger mainly because we need to make sure that the time is chosen such that it is based on a global skew estimate that holds during the complete insertion process. The insertion time is further increased because the insertions of different edges might use different global skew estimates and thus, the times of inserting the edges on different levels are harder to coordinate (and separate) properly. For details, we refer to Section 7. In the case of a dynamic global skew, the insertion duration $\mathcal{I}_{\{u,v\}}$ of an edge $\{u, v\}$ is computed as

$$\mathcal{I}_{\{u,v\}} := \mathcal{I}(\tilde{\mathcal{G}}_{\{u,v\}}) := 2^{\lceil \log_2 \ell_{\{u,v\}} \rceil}, \tag{11}$$
$$\text{where } \ell_{\{u,v\}} := (1+\rho)(1+\mu)(\delta_{\{u,v\}} + 2\tau_{\{u,v\}}) + 8\mathcal{B} \cdot \frac{\tilde{\mathcal{G}}_{\{u,v\}}}{\mu}.$$

The parameter $\mathcal{B}$ is a constant that is introduced for convenience and which has to satisfy the following conditions:

$$\frac{\mu}{2\rho} \geq \mathcal{B} \geq \frac{320 \cdot 2^7}{(1-\rho)^2}. \tag{12}$$

We note that together with (7), the above inequality directly implies that for the dynamic global skew analysis in Section 7, we can assume that

$$\frac{\rho}{(1-\rho)^2} \leq \frac{1}{6400 \cdot 2^7}. \tag{13}$$

In the technical analysis, we sometimes use $\mathcal{I}$ for $\mathcal{I}(\tilde{\mathcal{G}})$, if $\tilde{\mathcal{G}}$ is clear from the context. Note that the sequence $T_1^{\{u,v\}}, T_2^{\{u,v\}}, \ldots$ converges to

$$T_\infty^{\{u,v\}} := T_0^{\{u,v\}} + \mathcal{I}_{\{u,v\}}.$$

Also note that although the sequence is infinite, it (and also the sets N_u^s) can be implicitly stored using only bounded information. Further, if an edge $\{u, v\}$ with leader u appears at time t, the total time to insert $\{u, v\}$ on all levels is in the order of

$$\Theta\left(\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}} + \mathcal{I}_{\{u,v\}}\right) \subseteq \mathcal{O}\left(\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}} + \frac{\tilde{\mathcal{G}}_u(t)}{\mu}\right).$$

Listing 3: Setting the rate of node u's logical clock

if *the slow mode trigger is satisfied* **then**
  $mult_u := 1$
else if *the fast mode trigger is satisfied* **then**
  $mult_u := 1 + \mu$
else `// Neither slow nor fast mode trigger are satisfied; check for max estimate triggers`
  if $L_u = M_u$ **then**
    $mult_u := 1$
  else if $L_u \leq M_u - \iota$ **then**
    $mult_u := 1 + \mu$

For convenience, for a given execution and a level $s \geq 1$, we define $\mathbb{T}_s$ to be the set of all level s insertion times $T_s^{\{u,v\}}$ used for any possible edge $\{u, v\}$ at any time. Further, we define $\mathbb{T} := \bigcup_{s \geq 1} \mathbb{T}_s$ to be the set of all edge insertion times of a given execution.

The fast and slow mode triggers. The rules for deciding when to enter the fast mode or the slow mode correspond to the conditions from Section 4.2, but they compensate for the uncertainty of the clock estimates to ensure that the conditions are satisfied. The triggers for switching modes are as follows.

Definition 4.5 (Fast Mode Trigger). *Node u satisfies the* fast mode trigger *at time t if there exists an integer $s \in \mathbb{N}$ such that*

- *For some $w \in N_u^s(t)$ we have $\tilde{L}_u^w(t) - L_u(t) \geq s \cdot \kappa_{\{u,w\}} - \epsilon_{\{u,w\}}$, and*
- *For all $v \in N_u^s(t)$ we have $L_u(t) - \tilde{L}_u^v(t) \leq s \cdot \kappa_{\{u,v\}} + 2\mu\tau_{\{u,v\}} + \epsilon_{\{u,v\}}$.*

The slow mode trigger incorporates some slack, which we also encountered in Definition 4.2; we now define it as a parameter δ_e for each edge e, and require

$$\delta_e \in \left(0, \frac{\kappa_e}{2} - 2\epsilon_e - 2\mu\tau_e\right).$$

This constraint ensures that the fast mode and the slow mode triggers are mutually exclusive (see Lemma 5.3). We note that $\frac{\kappa_e}{2} - 2\epsilon_e - 2\mu\tau_e > 0$ due to (9), which constrains the choice of κ_e.

Definition 4.6 (Slow Mode Trigger). *Node u satisfies the* slow mode trigger *at time t if there exists an integer $s \in \mathbb{N}$ such that*

- *For some $w \in N_u^s(t)$ we have $L_u(t) - \tilde{L}_u^w(t) \geq \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}} - \delta_{\{u,w\}} - \epsilon_{\{u,w\}}$, and*
- *For all $v \in N_u^s(t)$ we have $\tilde{L}_u^v(t) - L_u(t) \leq \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}} + \delta_{\{u,v\}} + \epsilon_{\{u,v\}} + \mu(1+\rho)\tau_{\{u,v\}}$.*

The fast and slow mode triggers are disjoint (as we will prove later), and since both are closed regions, they are strictly separated from each other: there are some states that satisfy neither condition. In these in-between regions, nodes choose their mode based on the max-estimate trigger, which ensures that **MC** is satisfied.

Definition 4.7 (Max Estimate Triggers)**.** *Node u satisfies the* fast max estimate trigger *at time t if the slow mode trigger is not satisfied and $L_u(t) \leq M_u(t) - \iota$. It satisfies the* slow max estimate trigger *at time t if the fast mode trigger is not satisfied and $L_u(t) = M_u(t)$.*

The code implementing these triggers is shown in Listing 3.

5 Analysis

In this section, we analyze the algorithm described in Section 4 and bound its worst-case global and dynamic gradient skew.

5.1 Basic Properties

We begin with some basic properties which were stated informally in Section 4. Essentially, in this subsection we show that the algorithm behaves "as intended," which is the foundation for our subsequent reasoning about skews. The first property states that the neighbor set N_u^s is a subset of N_u^{s-1} for all $s \geq 1$ at all times.

Lemma 5.1. *For all $u \in V$, at all times $t \geq 0$ we have $N_u(t) = N_u^0(t)$ and $N_u^s(t) \subseteq N_u^{s-1}(t)$ for all $s \geq 1$.*

Proof. At time 0, the neighbor sets are initialized to $N_u^s(0) = N_u(0)$ for all $s \geq 0$. Further, for every edge $e = \{u, v\}$ of node u, the update times T_s^e are reached in order $s = 1, 2, \ldots$, and at each such time, we only add to N_u^s nodes that already belong to $N_u^1, \ldots, N_u^{s-1}$. Therefore node additions preserve the property.

Nodes are only removed from neighbor sets in Line 17 of Listing 1. As a node is removed from all the neighbor sets, also node removals preserve the property claimed by the lemma. Formally, the claim of the lemma therefore follows by induction on node u's discrete transitions. □

In Section 4.2 we introduced the fast and slow mode conditions (**FC** and **SC**), and claimed that the algorithm implements these conditions; now we prove this claim. Our goal is to show that when the inaccuracy of the estimates is taken into account, the fast and the slow mode triggers hold whenever the fast and the slow mode conditions apply, respectively. Likewise, the max estimate condition **MC** is satisfied by the algorithm.

Lemma 5.2. *Algorithm $\mathcal{A}^{\mathrm{OPT}}$ satisfies the fast and slow mode conditions, as well as the max estimate condition.*

Proof. Let us start with **FC**. Suppose that the antecedent of **FC** holds at node u: that is, there is some $s \in \mathbb{N}$ such that for some $w \in N_u^s(t)$ we have

$$L_w(t) - L_u(t) \geq s \cdot \kappa_{\{u,w\}}, \tag{14}$$

and for all $v \in N_u^s(t)$ we have

$$L_u(t) - L_v(t) \leq s \cdot \kappa_{\{u,v\}} + 2\mu\tau_{\{u,v\}}. \tag{15}$$

The estimate $\tilde{L}_u^w(t)$ that node u has for node w satisfies $L_w(t) \leq \tilde{L}_u^w(t) + \epsilon_{\{u,w\}}$, and combined with (14) we obtain

$$\tilde{L}_u^w(t) - L_u(t) \geq s \cdot \kappa_{\{u,w\}} - \epsilon_{\{u,w\}}.$$

Similarly, for all $v \in N_u^s(t)$ we have $L_v(t) \leq \tilde{L}_u^v(t) + \epsilon_{\{u,v\}}$, so from (15),

$$L_u(t) - \tilde{L}_u^v(t) \leq s \cdot \kappa_{\{u,v\}} + 2\mu\tau_{\{u,v\}} + \epsilon_{\{u,v\}}.$$

Hence the fast mode trigger is satisfied and node u is in fast mode.

Now consider **SC**. Define $\delta := \min_{e \in E}\{\delta_e\} > 0$, and suppose that for this value of δ the antecedent of **SC** holds at node u: there is some $s \in \mathbb{N}$ such that for some $w \in N_u^s(t)$ we have

$$L_u(t) - L_w(t) \geq \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}} - \delta, \tag{16}$$

and for all $v \in N_u^s(t)$ we have

$$L_v(t) - L_u(t) \leq \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}} + \delta + \mu(1+\rho)\tau_{\{u,v\}}. \tag{17}$$

Now we use the other direction of the estimate accuracy guarantee: for all $w \in N_u^s(t)$ we have $L_w(t) \geq \tilde{L}_u^w(t) - \epsilon_{\{u,w\}}$. In particular, since $\delta \leq \delta_{\{u,w\}}$ for any $w \in N_u^s(t)$, from (16) we obtain

$$L_u(t) - \tilde{L}_u^w(t) \geq \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}} - \delta - \epsilon_{\{u,w\}} \geq \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}} - \delta_{\{u,w\}} - \epsilon_{\{u,w\}}.$$

Moreover, when replacing w with v we get that $L_v(t) \geq \tilde{L}_u^v(t) - \epsilon_{\{u,v\}}$, which together with (17) yields

$$\begin{aligned}\tilde{L}_u^v(t) - L_u(t) &\leq \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}} + \delta + \epsilon_{\{u,v\}} + \mu(1+\rho)\tau_{\{u,v\}} \\ &\leq \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}} + \delta_{\{u,v\}} + \epsilon_{\{u,v\}} + \mu(1+\rho)\tau_{\{u,v\}}.\end{aligned}$$

Therefore the slow mode trigger is satisfied, and node u is in slow mode.

It remains to show that the algorithm also satisfies the max estimate condition **MC**. Suppose first that **MC** requires the node to be in slow mode. Then $M_u(t) = L_u(t)$ and the fast mode trigger cannot be satisfied, as there is no neighbor $v \in N_u$ with $L_v > L_u$. Thus, u is in slow mode either because the slow mode trigger applies or because neither the slow nor fast mode trigger applies and $L_u(t) = M_u(t)$, cf. Listing 3.

Similarly, if **MC** requires the node to be in fast mode, $M_u(t) \geq L_u(t) - \iota$ and there is no neighbor $v \in N_u$ with $L_v > L_u$. Hence, the slow mode trigger is not satisfied and u will be in fast mode. Hence, the max estimate condition **MC** is satisfied. $\square$

Next we show that the slow and fast mode triggers are never in conflict. While this statement is not needed for deriving the guarantees of the algorithm, we could not actually *implement* the algorithm if it did not hold.

Lemma 5.3. *For all $u \in V$, the slow and fast mode triggers are never satisfied at the same time.*

Proof. Suppose for the sake of contradiction that for some node $u \in V$ and time t, the fast mode trigger is satisfied for an integer s and the slow mode trigger is satisfied for an integer s'. We consider two cases.

I. $s \leq s'$. Due to Lemma 5.1, we have that $N_u^{s'}(t) \subseteq N_u^s(t)$ in this case. Because the slow mode trigger is satisfied for s', there is a node $w \in N_u^{s'}$ such that

$$L_u(t) - \tilde{L}_u^w(t) \geq \left(s' + \frac{1}{2}\right) \kappa_{\{u,w\}} - \delta_{\{u,w\}} - \epsilon_{\{u,w\}}. \tag{18}$$

However, since $w \in N_u^{s'}(t) \subseteq N_u^s(t)$, the second part of the fast mode condition applies to w, and it states that

$$L_u(t) - \tilde{L}_u^w(t) \leq s \cdot \kappa_{\{u,w\}} + \epsilon_{\{u,w\}} + 2\mu\tau_{\{u,w\}}. \tag{19}$$

Combining (18) and (19) yields

$$\begin{aligned}\left(s' + \frac{1}{2}\right) \kappa_{\{u,w\}} - \delta_{\{u,w\}} - \epsilon_{\{u,w\}} &\leq s \cdot \kappa + \epsilon_{\{u,w\}} + 2\mu\tau_{\{u,w\}} \\ &\leq s' \cdot \kappa + \epsilon_{\{u,w\}} + 2\mu\tau_{\{u,w\}}.\end{aligned}$$

By re-arranging the terms, we obtain

$$\kappa_{\{u,w\}} \leq 4\epsilon_{\{u,w\}} + 4\mu\tau_{\{u,w\}} + 2\delta_{\{u,w\}}. \tag{20}$$

However, recall that δ_e is chosen in the range $(0, \kappa_e/2 - 2\epsilon_e - 2\mu\tau_e)$ for each edge e. Therefore $4\epsilon_{\{u,w\}} + 4\mu\tau_{\{u,w\}} + 2\delta_{\{u,w\}} < \kappa_{\{u,w\}}$, contradicting (20).

II. $s > s'$, that is, $s \geq s' + 1$. In this case Lemma 5.1 states that $N_u^s(t) \subseteq N_u^{s'}(t)$. Because the fast mode trigger is satisfied for s, there is some node $w \in N_u^s(t)$ such that

$$\tilde{L}_u^w(t) - L_u(t) \geq s \cdot \kappa_{\{u,w\}} - \epsilon_{\{u,w\}}. \tag{21}$$

Because $w \in N_u^s(t) \subseteq N_u^{s'}(t)$, the second part of the slow mode trigger also applies to w:

$$\tilde{L}_u^w(t) - L_u(t) \leq \left(s' + \frac{1}{2}\right) \kappa_{\{u,w\}} + \delta_{\{u,w\}} - \epsilon_{\{u,w\}} + \mu(1+\rho)\tau_{\{u,w\}}. \tag{22}$$

As before, we combine (21) and (22) and obtain

$$\begin{aligned}\left(s' + \frac{1}{2}\right) \kappa_{\{u,w\}} + \delta_{\{u,w\}} + \epsilon_{\{u,w\}} + \mu(1+\rho)\tau_{\{u,w\}} &\geq s \cdot \kappa_{\{u,w\}} - \epsilon_{\{u,w\}} \\ &\geq (s'+1)\kappa_{\{u,w\}} - \epsilon_{\{u,w\}}.\end{aligned}$$

Re-arranging the terms yields

$$\kappa_{\{u,w\}} \leq 4\epsilon_{\{u,w\}} + 2\delta_{\{u,w\}} + 2\mu(1+\rho)\tau_{\{u,w\}}. \tag{23}$$

However, since $\rho < 1$, we have $2\mu(1+\rho)\tau_{\{u,w\}} < 4\mu\tau_{\{u,w\}}$, and since $\delta_{\{u,w\}} < \kappa_{\{u,w\}}/2 - 2\epsilon_{\{u,w\}} - 2\mu\tau_{\{u,w\}}$, we have $4\epsilon_{\{u,w\}} + 2\delta_{\{u,w\}} + 2\mu(1+\rho)\tau_{\{u,w\}} < \kappa_{\{u,w\}}$, contradicting (23).

□

Finally, we characterize the behavior of the neighbor coordination mechanism introduced in Section 4.3 to ensure that nodes add each other as neighbors in a roughly symmetric manner. Intuitively, node u is trying to insert (or has inserted) edge $\{u,v\}$ at time t if and only if its variables $T_s^{\{v,w\}} \neq \perp$ at time t. If the edge disappears in the view of one of the nodes, at the latest $\tau_{\{v,w\}}$ time later this happens also for the other, and both stop considering the edge for evaluating their mode until it is reinserted. However, for the sake of the analysis, we want to discard the initial time period in which it is possible that not both u and v agree on the values $T_s^{\{v,w\}}$; this time period is irrelevant, since neither node will actually add the edge to one of its neighbor sets for $s \in \mathbb{N}$ before this agreement is established. This is captured by the following definition and lemma. We state them for the general case that $\tilde{\mathcal{G}}_u(t)$ is not constant, as this does not affect the proof of the lemma and will be of use in Section 7.

Definition 5.4. *For any node u and any time t, we define the set $\mathbf{T}_u^{\{u,v\}}(t) := \left\{T_0^{\{u,v\}}, \dots, T_\infty^{\{u,v\}}\right\}$, where $T_s^{\{u,v\}}$, $s \in \mathbb{N}_0$ refers to the state of u's respective variable at time t. We simply write $\mathbf{T}_u^{\{u,v\}}(t) = \perp$ if u never computed these variables or set them to $\perp$ according to Algorithm 1.*

If $\mathbf{T}_u^{\{u,v\}}(t) \neq \perp$, denote by $T_{u,0}^{\{u,v\}}(t)$ the time $T_0^{\{u,v\}} \in \mathbf{T}_u^{\{u,v\}}(t)$; otherwise, $T_{u,0}^{\{u,v\}}(t) := \infty$. We define the Boolean variable

$$\mathcal{A}_u^{\{u,v\}}(t) := \left(\mathbf{T}_u^{\{u,v\}}(t) \neq \perp \wedge t \geq \min\left\{T_{u,0}^{\{u,v\}}(t), T_{v,0}^{\{u,v\}}(t)\right\}\right).$$

Lemma 5.5. *For every (potential) edge $\{u,v\}$ and for all $t \geq 0$, the following three statements are true:*

$$\text{(I)} \qquad \left(\mathcal{A}_u^{\{u,v\}}(t) \wedge \mathcal{A}_v^{\{u,v\}}(t)\right) \Longrightarrow \mathbf{T}_u^{\{u,v\}}(t) = \mathbf{T}_v^{\{u,v\}}(t),$$

$$\text{(II)} \qquad \left(\mathcal{A}_u^{\{u,v\}}(t) \wedge \neg\mathcal{A}_v^{\{u,v\}}(t)\right) \Longrightarrow \left(\neg\mathcal{A}_u^{\{u,v\}}(t+\tau_{\{u,v\}}) \wedge \neg\mathcal{A}_v^{\{u,v\}}(t+\tau_{\{u,v\}})\right),$$

$$\text{(III)} \qquad \left(\neg\mathcal{A}_u^{\{u,v\}}(t) \wedge \mathcal{A}_v^{\{u,v\}}(t)\right) \Longrightarrow \left(\neg\mathcal{A}_u^{\{u,v\}}(t+\tau_{\{u,v\}}) \wedge \neg\mathcal{A}_v^{\{u,v\}}(t+\tau_{\{u,v\}})\right).$$

Proof. Without loss of generality, assume that node u is the leader of edge $\{u,v\}$. If node u has never discovered neighbor v before time t, we have $\mathcal{A}_u^{\{u,v\}}(t) =$ **false** and because v can only start inserting the edge after receiving a message from u, we also have $\mathcal{A}_v^{\{u,v\}}(t) =$ **false**. Hence, we can assume that there has been a time $t' \leq t$ when u discovered neighbor v. Let $\underline{t}$ be the last such time before t.

Case 1: $\mathcal{A}_u^{\{u,v\}}(t) =$ true. Thus, $v \in N_u(t')$ for all $t' \in [\underline{t}, t]$. Consequently, $u \in N_v(t')$ for all $t' \in [\underline{t} + \tau_{\{u,v\}}, t - \tau_{\{u,v\}}]$. Abbreviate

$$\Delta := \frac{(1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}})}{1-\rho} + \tau_{\{u,v\}}.$$

At the logical time t_S when $L_u(t_S) - L_u(\underline{t}) = (1+\rho)(1+\mu)\Delta$, u sent a message to v, informing it about the times $\mathbf{T}_u^{\{u,v\}}(t)$, by communicating the global skew estimate $\tilde{\mathcal{G}}_{\{u,v\}} = \tilde{\mathcal{G}}_u(t_S)$ and $L_{ins} = L_u(t_S) + \tilde{\mathcal{G}}_u(t_S) + (1+\rho)(1+\mu)\mathcal{T}_{\{u,v\}}$. Based on these parameters, in the call to computeInsertionTimes, the logical time $T_{u,0}^{\{u,v\}}$ is set to a value of at least L_{ins}. Let t_0 be the time such that $\min\{L_u(t_0), L_v(t_0)\} = T_{u,0}^{\{u,v\}}$. We claim that $t_0 \geq t_S + \mathcal{T}_{\{u,v\}}$. To see this, consider any node $w \in \{u,v\}$ and bound

$$\begin{aligned} L_w(t_S + \mathcal{T}_{\{u,v\}}) &\le L_w(t_S) + (1+\rho)(1+\mu)\mathcal{T}_{\{u,v\}} \\ &\le L_v(t_S) + \mathcal{G}(t_S) + (1+\rho)(1+\mu)\mathcal{T}_{\{u,v\}} \\ &\le L_v(t_S) + \tilde{\mathcal{G}}_u(t_S) + (1+\rho)(1+\mu)\mathcal{T}_{\{u,v\}} \\ &= L_{ins} \\ &\le T_{u,0}^{\{u,v\}}. \end{aligned}$$

In particular, the definition of $\underline{t}$ implies that $t \ge t_0 \ge t_S + \mathcal{T}_{\{v,w\}}$.

Case 1a: $\mathcal{A}_v^{\{u,v\}}(t) = \mathbf{true}$. In this case we need to show that $\mathbf{T}_u^{\{u,v\}}(t) = \mathbf{T}_v^{\{u,v\}}(t)$ in order to establish (I). Denote by t_R the maximal time in $[0,t]$ when v received an insertedge$(\{u,v\},\cdot,\cdot)$ message from u; such a time must exist, as otherwise $\mathcal{A}_v^{\{u,v\}}(t) = \mathbf{false}$. Because u waits for at least Δ time after an edge has formed (from u's perspective) before sending a message, v cannot receive any other insertedge$(\{u,v\},\cdot,\cdot)$ message from u during $[t_S-\Delta+\mathcal{T}_{\{u,v\}}, t_R] \supseteq [t_R-\mathcal{T}_{\{u,v\}}-\tau_{\{u,v\}}, t_R]$. We claim that $u \in N_v(t')$ for all $t' \in [t_R - \mathcal{T}_{\{u,v\}} - \tau_{\{u,v\}}, t]$. Otherwise, v would satisfy $T_s^{\{u,v\}} = \perp$ for all $s \in \{0,1,\ldots\}$ at some time $t' \in [t_R - \mathcal{T}_{\{u,v\}} - \tau_{\{u,v\}}, t]$, and if there is only such a $t' < t_R$, it would have ignored the message received at time t_R because

$$L_v(t_R - \mathcal{T}_{\{u,v\}} - \tau_{\{u,v\}}) \ge L_v(t_R) - (1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}}).$$

We conclude that the message sent by u at time t_S is received by v; therefore, $t_R \in [t_S, t_S + \mathcal{T}_{\{u,v\}}]$ is actually the time when this message is received. (I) now follows because $t_S + \mathcal{T}_{\{u,v\}} \le t_0 \le t$, v computes $\mathbf{T}_v^{\{u,v\}}(t_R) = \mathbf{T}_u^{\{u,v\}}(t_S)$ upon reception of the message, and v does not change these variables again until time t.

Case 1b: $\mathcal{A}_v^{\{u,v\}}(t) = \mathbf{false}$. In this case we need to show that $\left(\neg\mathcal{A}_u^{\{u,v\}}(t+\tau_{\{u,v\}}) \wedge \neg\mathcal{A}_v^{\{u,v\}}(t+\tau_{\{u,v\}})\right)$ in order to establish (II). We claim that there is a time $t' \in [t_S - \Delta + \tau_{\{u,v\}}, t]$ so that $u \notin N_v(t')$. Otherwise, we had $u \in N_v$ throughout $[t_S - \Delta + \tau_{\{u,v\}}, t]$ and v would receive and not discard the message by u, as

$$L_v(t_R) \ge L_v(t_S) \ge L_v\left(t_S - \Delta + \tau_{\{u,v\}}\right) + (1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}}).$$

However, this would entail that $\mathcal{A}_v^{\{u,v\}}(t) = \mathbf{true}$, so indeed such a time t' must exist.

Denote by $t'' \in [t' - \tau_{\{u,v\}}, t' + \tau_{\{u,v\}}] \subseteq [t_S - \Delta, t + \tau_{\{u,v\}}]$ a time so that $v \notin N_u(t'')$; by the communication model, such a time exists. In fact, we have that $t'' > t_S$, as the edge must be continuously present from the perspective of u for $(1+\rho)(1+\mu)\Delta$ local time, i.e., since at least Δ real time, before it sends a message. Therefore, u will set $T_s^{\{u,v\}} := \perp$ for all $s \in \{0,1,\ldots\}$ at time t''. While the definition of $\underline{t}$ admits that u may observe the reappearance of the edge at a time larger than t, u will not recompute the values $T_s^{\{u,v\}}$ or send another message to v by time $t + \tau_{\{u,v\}}$. This implies that also v does not recompute its values $T_s^{\{u,v\}}$ during $[t'', t + \tau_{\{u,v\}}]$, and it follows that $\left(\neg\mathcal{A}_u^{\{u,v\}}(t+\tau_{\{u,v\}}) \wedge \neg\mathcal{A}_v^{\{u,v\}}(t+\tau_{\{u,v\}})\right)$, as claimed.

Case 2: $\mathcal{A}_u^{\{u,v\}}(t) = \mathbf{false} \wedge \mathcal{A}_v^{\{u,v\}}(t) = \mathbf{true}$. In this case we need to show that $\left(\neg\mathcal{A}_u^{\{u,v\}}(t+\tau_{\{u,v\}}) \wedge \neg\mathcal{A}_v^{\{u,v\}}(t+\tau_{\{u,v\}})\right)$ in order to establish (III). Denote by t_R the latest time before t when v received an insertedge$(\{u,v\},\cdot,\cdot)$ message from u; as $\mathcal{A}_v^{\{u,v\}}(t) = \mathbf{true}$, such a time must exist.

Denote by $t_S \in [t_R - \mathcal{T}_{\{u,v\}}, t_R]$ the time when it was sent. Note that $\mathcal{A}_v^{\{u,v\}}(t) = \textbf{true}$ also implies that v neither discarded the message, i.e., it recomputed the times $T_s^{\{u,v\}}$ at time t_R, nor did it set $T_s^{\{u,v\}} := \perp$ for any s during $[t_R, t]$. Hence, $u \in N_v(t')$ for all $t' \in [t_R - \mathcal{T}_{\{u,v\}} - \tau_{\{u,v\}}, t]$. We infer that $v \in N_u(t')$ for all $t' \in [t_R - \mathcal{T}_{\{u,v\}}, t - \tau_{\{u,v\}}] \subseteq [t_S, t - \tau_{\{u,v\}}]$. This implies that u cannot detect the reappearance of the edge during this interval and thus will not call computeInsertionTimes during $(t_S, t + \tau_{\{u,v\}}]$; it follows that $\mathcal{A}_u^{\{u,v\}}(t + \tau_{\{u,v\}}) = \textbf{false}$.

To see that also $\mathcal{A}_v^{\{u,v\}}(t + \tau_{\{u,v\}}) = \textbf{false}$, note that since $\mathcal{A}_u^{\{u,v\}}(t) = \textbf{false}$, it must hold that $v \notin N_u(t')$ for some $t' \in [t_S, t]$. Therefore, $u \notin N_v(t'')$ for some $t'' \in [t_S - \tau_{\{u,v\}}, t + \tau_{\{u,v\}}] \subseteq [t_R - \mathcal{T}_{\{u,v\}} - \tau_{\{u,v\}}, t + \tau_{\{u,v\}}]$. As $u \in N_v(t')$ for all $t' \in [t_R - \mathcal{T}_{\{u,v\}} - \tau_{\{u,v\}}, t]$, we obtain that $t'' \in (t, t + \tau_{\{u,v\}}] \subseteq (t_R, t + \tau_{\{u,v\}}]$. Consequently, v resets $T_s^{\{u,v\}} := \perp$ for all $s \in \{0, 1, \ldots\}$ at time $t'' \in (t_R, t + \tau_{\{u,v\}}]$, yielding that $\mathcal{A}_v^{\{u,v\}}(t + \tau_{\{u,v\}}) = \textbf{false}$. This proves Statement (III), concluding the proof. □

5.2 The Global Skew

Like its predecessors in [12, 16], our algorithm achieves an asymptotically optimal global skew. In static networks of diameter D where each message has an uncertainty U in its transit time, the best possible global skew guarantee is $\Theta((\rho + U)D)$ [12, 16], and the dynamic estimate diameter satisfies $\mathcal{D}(t) \in \Omega((\rho + U)D)$ at all times. Let $\mathcal{D} := \max_t \mathcal{D}(t)$ be the maximal network uncertainty of a given execution of our algorithm. In the following, we show that the algorithm always guarantees a global skew of $\mathcal{O}(\mathcal{D})$. In fact, we show the following stronger statement:

Theorem 5.6. *Let ι be defined as in Definition 4.4.*

I. On any dynamic graph executing $\mathcal{A}^{\mathrm{OPT}}$, at any time the global skew increases at rate at most 2ρ.

II. On any dynamic graph executing $\mathcal{A}^{\mathrm{OPT}}$, at any time t when the global skew exceeds $\mathcal{D}(t) + \iota$, it decreases at rate at least $\mu(1 - \rho) - 2\rho > 0$.

Proof. It suffices to show that (i) any node with the largest clock value throughout the network is in slow mode and (ii) whenever the global skew exceeds $\mathcal{D}(t) + \iota$, any node with the smallest clock value in the network is in fast mode. The theorem then follows because any clock in slow mode is at most 2ρ faster than any other clock and

$$(1 + \mu)(1 - \rho) - (1 + \rho) = \mu(1 - \rho) - 2\rho \overset{(8)}{>} 0.$$

Let u and v be nodes with the largest and smallest logical clock values at an arbitrary time t, respectively. By Inequalities (2) and (4), we have $M_u(t) = L_u(t)$. Provided that $\mathcal{G}(t) = L_u(t) - L_v(t) > \mathcal{D}(t) + \iota$, from Inequality (3) it follows that $M_v(t) \geq L_u(t) - \mathcal{D}(t) > L_v(t) - \iota$. Therefore, due to the max estimate condition **MC**, we conclude that Statements (i) and (ii) are true, yielding the claims of the theorem. □

Note that Statement *II.* of the theorem implies that the global skew is self-stabilizing in the sense that it is reduced at an asymptotically optimal rate of $\mu(1 - \rho) - 2\rho \in \Omega(\mu)$ when it exceeds the best possible guarantee. One could add a simple consistency check mechanism to the algorithm that forces logical clocks to be instantaneously set to close values (i.e., violate the progress bound of $(1+\rho)(1+\mu)$) whenever a global skew exceeding a certain multiple of $D(t)$ is detected. While (together with self-stabilizing implementations of the algorithms rules and neighbor sets) this would ensure quick stabilization from arbitrarily corrupted states, such behavior might be undesired if $D(t)$ decreases rapidly in a fault-free execution, as clock values would change too rapidly; this could also cause violations of the gradient skew bound.

5.3 Analysis of the Gradient Skew

5.3.1 Preliminary Definitions and Statements

As described in Section 4.1, the gradient skew requirement states that paths of certain lengths cannot have an average skew that exceeds a certain bound. In particular, for every positive integer s, there is a length C_s such that paths p of length $\kappa_p \geq C_s$ have an average skew of at most $\mathcal{O}(s\kappa_e)$ per edge e, where the values C_s are exponentially decreasing in s. Since in the process of the analysis we will have to use different such sequences C_s, we do not explicitly define the values here, but will work with an abstract *gradient sequence* as defined in the following Definition 5.7 for most of the analysis.

Definition 5.7 (Gradient Sequences)**.** *A* gradient sequence *is a non-increasing sequence of values* $C = \{C_s\}_{s\in\mathbb{N}}$.

The specific sequences that will later be used to prove the gradient skew properties of the algorithm roughly look as follows. At all times t, we have $C_1 \geq 2\mathcal{G}(t)$. As a consequence, the gradient skew property for level 1 will follow directly from the bound on the global skew. Further, for most levels s, we have $C_{s+1} = C_s/\sigma$ such that also for short paths, we obtain a sufficiently strong requirement on the skew.

We have seen that the different values of s correspond to different average skew bounds. Throughout the proof, we will mostly make arguments for a particular such level s. In the following, we define the sets of edges and paths used when arguing about level s.

Definition 5.8 (Level-s Edge Set)**.** *For all* $s \in \mathbb{N}$*, we define*

$$E^s(t) := \left\{ \{u,v\} \in \binom{V}{2} \,\middle|\, v \in N_u^s(t) \wedge u \in N_v^s(t) \right\}.$$

Definition 5.9 (Level-s Paths)**.** *We define for all* $u_0 \in V$*,* $s \in \mathbb{N}$*, and all times* t *the set of* level-s paths starting at node u_0 at time t *to be*

$$P_{u_0}^s(t) := \{p = (u_0,\dots,u_k) \,|\, \forall i \in \{0,\dots,k-1\}:\ \{u_i,u_{i+1}\} \in E^s(t)\}.$$

For convenience of notation, we also define reversed and concatenated paths.

Definition 5.10 (Path Reversal and Concatenation)**.** *Given the path* $p = (u_0,\dots,u_k)$*, the corresponding* reversed path *is* $\bar{p} := (u_k,\dots,u_0)$*. Given two paths* $p = (u_0,\dots,u_k)$ *and* $q = (u_k,\dots,u_\ell)$*, their* concatenation *is* $p \circ q := (u_0,\dots,u_k,\dots,u_\ell)$*.*

Note that for all levels s and all times t, reversal and concatenation of level-s paths again yield level-s paths.

In addition, we define two notions of "weighted skew", essentially capturing how far away certain paths are from the level-s skew bound. The multiplicative factors in the two conditions correspond to the factors in the fast and slow mode condition **FC** and **SC**.

Definition 5.11. *For all paths $p = (u, \dots, v)$, all $s \in \mathbb{N}$, and all times t we define*

$$\xi_p^s(t) := L_u(t) - L_v(t) - s\kappa_p.$$

We further define for all $u \in V$ that

$$\Xi_u^s(t) := \max_{p \in P_u^s(t)} \{\xi_p^s(t)\}.$$

Definition 5.12. *For all paths $p = (u, \dots, v)$, all $s \in \mathbb{N}$, and all times t we define*

$$\psi_p^s(t) := L_v(t) - L_u(t) - \left(s + \frac{1}{2}\right)\kappa_p.$$

Furthermore, we set for all $u \in V$

$$\Psi_u^s(t) := \max_{p \in P_u^s(t)} \{\psi_p^s(t)\}.$$

A gradient sequence C defines what clock skew is allowed on different paths. Rather than directly defining a gradient skew requirement as described in Section 4.1, we define a condition that is based on the value introduced in Definition 5.12. If this requirement is satisfied, we say that the system is *legal*. Legality is formally defined as follows.

Definition 5.13 (Legality)**.** *Given a weighted, dynamic graph G and a gradient sequence C, for each $s \in \mathbb{N}$ the system is (C, s)*-legal at time t and node $u \in V$, *if and only if it holds that*

$$\Psi_u^s(t) < \frac{C_s}{2}$$

The system is C-legal at t and u *if it is (C, s)-legal for all $s \in \mathbb{N}$ at node u and time t.*

The legality definition can be used to derive an upper bound on the clock skew between any two nodes u and v.

Lemma 5.14. *Assume that for some $s \in \mathbb{N}$ and a path $p = (u, \dots, v) \in P_u^s(t)$, the system is (C, s)-legal at nodes u and v at time t. Then, $|L_v(t) - L_u(t)| < (s + 1/2)\kappa_p + C_s/2$.*

Proof. Since the system is (C, s)-legal at u at time t, we have that

$$L_v(t) - L_u(t) - \left(s + \frac{1}{2}\right)\kappa_p = \psi_p^s(t) \le \Psi_u^s(t) < \frac{C_s}{2}.$$

We therefore get that $L_v(t) - L_u(t) < (s + 1/2)\kappa_p + C_s/2$. From legality at v and because $p \in P_u^s(t)$ implies that the reversed path is in $P_v^s(t)$, we get the same upper bound on $L_u(t) - L_v(t)$. □

We now show a few simple properties that follow from the various definitions and the basic structure of the algorithm.

Lemma 5.15. *The following statements hold at all times t and all nodes $u \in V$.*

(i) $\forall s, s' \in \mathbb{N}$, $s' \le s$: $P_u^s(t) \subseteq P_u^{s'}(t)$.

(ii) $\forall s, s' \in \mathbb{N}$, $s' \le s$: $\Psi_u^s(t) \le \Psi_u^{s'}(t)$.

(iii) If for some $s \in \mathbb{N}$ the system is (C, s)-legal and $C_s = C_{s+1}$, then the system is also $(C, s+1)$-legal.

Proof. Statement (i) is a direct consequence of Lemma 5.1 and Definition 5.9. Statement (ii) follows from Definition 5.12 together with Statement (i). Finally, Statement (iii) follows from Statement (ii) together with Definition 5.13. □

5.3.2 Stabilization Condition and Convergence to Small Skews

In order to prove the claimed bound on the stabilization time, we require a *stabilization condition*, which depends on the s-legality of the system for a certain $s \in \mathbb{N}$. For ease of presentation, for every $s \in \mathbb{N}$, we define a parameter that will be used in the definition of the stabilization condition, as well as throughout the remainder of the proof.

$$\forall s \geq 2 : \Theta_s := \frac{C_{s-1}}{(1+\rho)\mu}. \tag{24}$$

Definition 5.16 (Stabilization Condition)**.** *For a node $u \in V$, a gradient sequence C, an integer $s > 1$, and a time t, we say that u satisfies the (C, s)-*stabilization condition *at time t if and only if*

(S0) *For all $t' \in [t - \Theta_2, t]$, we have $C_1 \geq 2\mathcal{G}(t')$.*

(S1) *For all $s' \in \{2, \ldots, s-1\}$ and all nodes $v \in V$ for which $|L_u(t) - L_v(t)| \leq s'C_{s'-1} + \big(2\rho + \mu(1+\rho)\big)\Theta_{s'}$, the system is (C, s')-legal at node v at all times in $[t - \Theta_{s'}, t]$.*

(S2) *We have:*

$$\forall T_s \in \mathbb{T}_s : |L_u(t) - T_s| \geq (1+\mu)(1+\rho)\Theta_s + sC_{s-1}$$

A simple observation is that the stabilization condition directly implies level-1 legality.

Lemma 5.17. *Let $t \geq 0$ be a time and assume that condition (S0) of the stabilization condition holds at some node $u \in V$ at some time $t' \in [t, t + \Theta_2]$. Then the system is $(C, 1)$-legal at all nodes $v \in V$ at time t.*

Proof. Consider arbitrary nodes $v, w \in V$. For any path $p = (v, \ldots, w)$, we have

$$\psi_p^1(t) = L_w(t) - L_v(t) - \frac{3}{2}\kappa_p < |L_w(t) - L_v(t)| \leq \mathcal{G}(t) \overset{(S0)}{\leq} \frac{C_1}{2}.$$

Given Definition 5.13, this implies that v is $(C, 1)$-legal at time t. ◻

This can be seen as an induction anchor, starting from which increasingly stronger bounds can be established for higher levels s. The core theorem of our analysis, stated next, provides the matching induction step.

Theorem 5.18. *Fix a level $s > 1$, a node $u \in V$ and an interval $[t^-, t^+]$. Let $\Lambda_s := C_{s-1}/(2(1-\rho)\mu)$, and suppose that for each $t \in [t^-, t^+]$ and for each path $(u, \ldots, v) \in P_u^s(t)$, if $\kappa_{(u,\ldots,v)} \leq C_{s-1}$, then the endpoint v satisfies the (C, s)-stabilization condition at time t. In this case, for all*

$$t \in \left[t^- + \Lambda_s + 2\Theta_s, t^+\right]$$

we have

$$\Psi_u^s(t) < 2\rho\Lambda_s = \frac{\rho C_{s-1}}{(1-\rho)\mu} = \frac{C_{s-1}}{2\sigma}.$$

Proving this theorem is technically challenging, but self-contained. Therefore, we postpone its proof to Section 6, in order to show how it is used to establish the desired gradient skew first.

5.3.3 Derivation of Skew Bounds

We now have all the necessary technical tools to prove the gradient skew bound of our algorithm. As pointed out earlier, our algorithm has some self-stabilization [3] properties in the following sense: Even if we start the algorithm from a configuration in which no non-trivial gradient skew bound holds, the system adapts and converges to a state in which the desired gradient skew bound holds. Edge insertion exploits this by adding edges level by level, every time waiting until the respective level and all higher levels have stabilized to small skews again. This entails that, at any given time, for level-s paths all but at most one level contributes a reduction by factor σ to the skew bound given by level-s legality. Taking into account the connection between logical clock values and real time, this motivates the following definition.

Definition 5.19 (Gradient Sequences for Static Global Skew Estimate)**.** *Set*

$$\Delta_s := \left(1 - \frac{1}{2^{s-1}}\right)\mathcal{I}(\tilde{\mathcal{G}}) + \left(\frac{5(1+\mu)}{2(1-\rho)\mu} + 2s\right)C_{s-1}. \tag{25}$$

Fix a time t and let $L(t)$ be maximal satisfying that $L(t) \le L_u(t)$ for all $u \in V$ and that $L(t)/\mathcal{I}(\tilde{\mathcal{G}}) \in \mathbb{Z}$. Then, for $s \in \mathbb{N}$, $u \in V$, and a parameter $\hat{\mathcal{G}}$, define

$$C_s^{(t,u)} := \begin{cases} \frac{2\hat{\mathcal{G}}}{\sigma^{s-1}} & \text{if } L_u(t) \ge L(t) + \Delta_s \\ \frac{2\hat{\mathcal{G}}}{\sigma^{\max\{s-2,0\}}} & \text{else.} \end{cases}$$

The next lemma shows that the system is legal at all nodes and times with respect to the above gradient sequences, granted that $\hat{\mathcal{G}}$ is an upper bound on the global skew and there is some initial time period of length $\Lambda_2 + 3\Theta_2 \in O(\hat{\mathcal{G}})$ during which the system is legal. For simplicity, we also make the assumption that $\sigma \ge 3$. However, by modifying the insertion times computed in Algorithm 2 (using a base depending on σ), one can handle any $\sigma > 1$; the downside is that $\mathcal{I}(\tilde{\mathcal{G}})$ goes to infinity as σ approaches 1.

Lemma 5.20. *Assume that $\sigma \ge 3$ and that the system is $C^{(t,u)}$-legal at all times $t \in [t_0, t_0+\Lambda_2+3\Theta_2]$ and nodes u. If $\hat{\mathcal{G}} \ge \mathcal{G}(t)$ for all $t \ge t_0$, then the system is $C^{(t,u)}$-legal at all times $t \ge t_0$ and nodes u.*

Proof. Assume for contradiction that there is a node $u \in V$ and a minimal time $\bar{t} > t_0 + \Lambda_2 + 3\Theta_2$ violating $C^{(\bar{t},u)}$-legality.[8] Because $C_1^{(t,u)} \ge 2\hat{\mathcal{G}} \ge 2\mathcal{G}(t)$ for any $t \ge t_0$, (S0) is satisfied at time t at any node $v \in V$. In particular, this is true for node u and time $\bar{t}$, yielding that the system is $(C^{(\bar{t},u)}, 1)$-legal at node u and time $\bar{t}$ by Lemma 5.17. Let $\bar{s} > 1$ be the minimal level on which legality is violated at node u and time $\bar{t}$. By Lemma 5.15, this implies that $C_{\bar{s}}^{(\bar{t},u)} \ne C_{\bar{s}-1}^{(\bar{t},u)}$. Therefore, $L_u(\bar{t}) \notin [L(\bar{t}) + \Delta_{\bar{s}-1}, L(\bar{t}) + \Delta_{\bar{s}})$.

We show that the preconditions of Theorem 5.18 are satisfied at node u for level $\bar{s}$, times $t^+ = \bar{t}$ and $t^- = \bar{t} - \Lambda_s - 2\Theta_{\bar{s}}$, and the gradient sequence given by

$$C_s := \begin{cases} \frac{2\hat{\mathcal{G}}}{\sigma^{s-1}} & \text{if } C_{\bar{s}}^{(\bar{t},u)} = \frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-1}} \\ \frac{2\hat{\mathcal{G}}}{\sigma^{\max\{s-2,0\}}} & \text{if } C_{\bar{s}}^{(\bar{t},u)} = \frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-2}}. \end{cases}$$

[8] As logical clocks are continuous, the set of times when the system is not legal at some node is closed. Thus, in any execution in which the claim of the lemma does not hold, such a minimal time exists.

This leads to the desired contradiction, as then

$$\Psi_u^{\bar{s}}(\bar{t}) < \frac{C_{\bar{s}-1}}{2\sigma} = \frac{C_{\bar{s}}^{(\bar{t},u)}}{2}.$$

Hence, it remains to show that for any $t \in [\bar{t} - \Lambda_s - 2\Theta_{\bar{s}}, \bar{t}]$ and any path $(u, \dots, v) \in P_u^{\bar{s}}(t)$ with $\kappa_{(u,\dots,v)} \leq C_{\bar{s}-1}^{(\bar{t},u)}$, v satisfies the $(C^{(\bar{t},u)}, \bar{s})$-stabilization condition at time t. As $C_1 = C_1^{(\bar{t},u)} \geq 2\mathcal{G}(t)$, (S0) holds at all times.

For (S1) and (S2), we will make a case distinction. However, both cases will use the following observation. The system is $(C^{(t,v)}, \bar{s})$-legal at node v and $(C^{(t,u)}, \bar{s})$-legal at node u at any time $t \in [t_0, \bar{t})$ by the minimality of $\bar{t}$. As $C_{\bar{s}-1} \leq \min\{C_{\bar{s}}^{(t,u)}, C_{\bar{s}}^{(t,v)}\}$, this entails $(C, \bar{s}-1)$-legality at both nodes and we can apply Lemma 5.14 to bound

$$|L_v(t) - L_u(t)| \leq \left(\bar{s} - \frac{1}{2}\right) \kappa_{(u,\dots,v)} + \frac{C_{\bar{s}-1}}{2} \leq \bar{s} C_{\bar{s}-1}. \tag{26}$$

As clocks are continuous, this bound also applies for $t = \bar{t}$. We now proceed to the case distinction.

Case 1: $L(\bar{t}) \geq L(\bar{t}) + \Delta_{\bar{s}}$, i.e., $C_{\bar{s}}^{(\bar{t},u)} = \frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-1}}$. Then

$$\begin{aligned}
L_v(t) &\overset{(26)}{\geq} L_u(t) - \bar{s} C_{\bar{s}-1} \\
&\geq L_u(\bar{t}) - (1+\rho)(1+\mu)(\bar{t} - t) - \bar{s} C_{\bar{s}-1} \\
&\geq L_u(\bar{t}) - (1+\rho)(1+\mu)\left(\frac{1}{2(1-\rho)\mu} + \frac{2}{(1+\rho)\mu} + \bar{s}\right) C_{\bar{s}-1} \\
&= L_u(\bar{t}) - \left(\frac{(5-3\rho)(1+\mu)}{2(1-\rho)\mu} + \bar{s}\right) C_{\bar{s}-1} \\
&\geq L(\bar{t}) + \Delta_{\bar{s}} - \left(\frac{(5-3\rho)(1+\mu)}{2(1-\rho)\mu} + \bar{s}\right) C_{\bar{s}-1} \qquad (27) \\
&> L(\bar{t}) + \left(1 - \frac{1}{2^{\bar{s}-1}}\right) \mathcal{I}(\tilde{\mathcal{G}}) - \left(\frac{(1+\rho)(1+\mu)}{(1-\rho)\mu} + \bar{s}\right) C_{\bar{s}-1} \\
&= L(\bar{t}) + \left(1 - \frac{1}{2^{\bar{s}-1}}\right) \mathcal{I}(\tilde{\mathcal{G}}) + (1+\mu)(1+\rho)\Theta_{\bar{s}} + \bar{s} C_{\bar{s}-1}.
\end{aligned}$$

Hence, if (S2) is violated for some $T_{\bar{s}} \in \mathbb{T}_{\bar{s}}$, then $T_{\bar{s}} > L(\bar{t}) + \left(1 - \frac{1}{2^{\bar{s}-1}}\right) \mathcal{I}(\tilde{\mathcal{G}})$. The smallest possible such $T_{\bar{s}}$ is $L(\bar{t}) + \left(2 - \frac{1}{2^{\bar{s}-1}}\right) \mathcal{I}(\tilde{\mathcal{G}})$. However, by definition of $L(\bar{t})$,

$$\begin{aligned}
L_v(t) &\leq L_v(\bar{t}) \\
&\leq \min_{w \in V}\{L_w(\bar{t})\} + \mathcal{G}(\bar{t}) \\
&< L(\bar{t}) + \mathcal{I}(\tilde{\mathcal{G}}) + \tilde{\mathcal{G}} \\
&\leq L(\bar{t}) + \frac{3\mathcal{I}(\tilde{\mathcal{G}})}{2} - \left(\frac{1+\mu}{\mu} + 2\right) 2\hat{\mathcal{G}} \\
&= L(\bar{t}) + \frac{3\mathcal{I}(\tilde{\mathcal{G}})}{2} - (1+\mu)(1+\rho)\Theta_2 - 2C_1 \\
&\overset{\sigma \geq 3}{\leq} L(\bar{t}) + \left(2 - \frac{1}{2^{\bar{s}-1}}\right) \mathcal{I}(\tilde{\mathcal{G}}) - (1+\mu)(1+\rho)\Theta_{\bar{s}} - \bar{s} C_{\bar{s}-1},
\end{aligned}$$

as $\bar{s} \geq 2$. Therefore, (S2) is satisfied.

Next, consider any $s \in \{2, \dots, \bar{s}-1\}$ and node $w \in V$ with $|L_v(t) - L_w(t)| \leq sC_{s-1} + (2\rho + \mu(1+\rho))\Theta_s)$. We have that

$$\begin{aligned}
L_w(t) &\geq L_v(t) - sC_{s-1} - (2\rho + \mu(1+\rho))\Theta_s \\
&\overset{(28)}{\geq} L(\bar{t}) + \Delta_{\bar{s}} - \left(\frac{(5-3\rho)(1+\mu)}{2(1-\rho)\mu} + \bar{s}\right) C_{\bar{s}-1} - \left(\frac{2\rho}{(1+\rho)\mu} + s + 1\right) C_{s-1} \\
&> L(\bar{t}) + \Delta_{\bar{s}} - \left(\frac{(5-3\rho)(1+\mu)}{2(1-\rho)\mu} + \bar{s}\right) C_{\bar{s}-1} - \left(\frac{1}{\sigma} + s + 1\right) C_{s-1} \\
&= L(\bar{t}) + \Delta_{\bar{s}} - \left(\frac{(5-3\rho)(1+\mu)}{2(1-\rho)\mu} + \bar{s} + 1\right) C_{\bar{s}-1} - (s+1)C_{s-1} \\
&\overset{\sigma \geq 3}{\geq} L(\bar{t}) + \Delta_{s+1} - \left(\frac{(5-3\rho)(1+\mu)}{2(1-\rho)\mu} + s + 2\right) C_s - (s+1)C_{s-1} \\
&> L(\bar{t}) + \left(1 - \frac{1}{2^s}\right) \mathcal{I}(\tilde{\mathcal{G}}) - (s+1)C_{s-1} \\
&= L(\bar{t}) + \Delta_s + \frac{\mathcal{I}(\tilde{\mathcal{G}})}{2^s} - \left(\frac{5(1+\mu)}{2(1-\rho)\mu} + 3s + 1\right) \frac{2\hat{\mathcal{G}}}{\sigma^{s-2}} \\
&\overset{\sigma \geq 3}{\geq} L(\bar{t}) + \Delta_s + \frac{1+\mu}{\mu} \cdot C_{s-1} \\
&= L(\bar{t}) + \Delta_s + (1+\mu)(1+\rho)\Theta_s.
\end{aligned}$$

For any time $t' \in [t - \Theta_s]$, this yields that

$$L_w(t') \geq L_w(t) - (1+\mu)(1+\rho)\Theta_s \geq L(\bar{t}) + \Delta_s.$$

We conclude that $C_s^{(t',w)} = C_s$, which by minimality of $\bar{t}$ and $\bar{s}$ implies that the system is (C, s)-legal at node w and time t', i.e., (S1) is satisfied. Therefore, all all preconditions of Theorem 5.18 are satisfied and Case 1 leads to a contradiction.

Case 2: $L_u(\bar{t}) < L(\bar{t}) + \Delta_{\bar{s}-1}$, i.e., $C_{\bar{s}}^{(\bar{t},u)} = \frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-2}}$. Then, for any $s \in \{2, \dots, \bar{s}-1\}$, time t, and node $w \in V$, we have that

$$C_s = \frac{2\hat{\mathcal{G}}}{\sigma^{\max\{s-2,0\}}} \leq C_s^{(t,w)}.$$

Thus, as $\bar{t} \geq t_0 + \Lambda_s + 3\Theta_2 \geq t_0 + \Lambda_s + 2\Theta_{\bar{s}} + \Theta_s$, by minimality of $\bar{t}$ (S1) is satisfied for all $t \in [\bar{t} - \Lambda_s - 2\Theta_{\bar{s}}, \bar{t}]$.

Concerning (S2), note that

$$\begin{aligned}
\frac{\mathcal{I}(\tilde{\mathcal{G}})}{2^{\bar{s}-1}} &\overset{\sigma \geq 3}{>} \left(\frac{5(1+\mu)}{2(1-\rho)\mu} + 2(\bar{s}-1)\right) \frac{2\hat{\mathcal{G}}}{\sigma^{\max\{\bar{s}-3,0\}}} + \left(\frac{1+\mu}{\mu} + 2\bar{s}\right) \frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-2}} \\
&= \left(\frac{5(1+\mu)}{2(1-\rho)\mu} + 2(\bar{s}-1)\right) C_{\bar{s}-2} + (1+\mu)(1+\rho)\Theta_{\bar{s}} + 2\bar{s}C_{\bar{s}-1},
\end{aligned}$$

where we used that the ratio between left- and right-hand side is minimized for $\bar{s} = 3$ (as opposed to minimal $\bar{s} = 2$ like in other places).

Together with (26), this yields that

$$\begin{aligned} L_v(t) &\overset{(26)}{\leq} L_u(t) + \bar{s}C_{\bar{s}-1} \\ &< L(\bar{t}) + \Delta_{\bar{s}-1} + \bar{s}C_{\bar{s}-1} \\ &= L(\bar{t}) + \left(1 - \frac{1}{2^{\bar{s}-2}}\right)\mathcal{I}(\tilde{\mathcal{G}}) + \left(\frac{5(1+\mu)}{2(1-\rho)\mu} + 2(\bar{s}-1)\right)C_{\bar{s}-2} + \bar{s}C_{\bar{s}-1} \\ &\leq L(\bar{t}) + \left(1 - \frac{1}{2^{\bar{s}-1}}\right)\mathcal{I}(\tilde{\mathcal{G}}) - (1+\mu)(1+\rho)\Theta_{\bar{s}} - \bar{s}C_{\bar{s}-1}. \end{aligned}$$

As $L(\bar{t}) + \left(1 - \frac{1}{2^{\bar{s}-1}}\right)\mathcal{I}(\tilde{\mathcal{G}})$ is the smallest possible logical time that is at least $L(\bar{t})$ and in $\mathbb{T}_{\bar{s}}$, (S2) cannot be violated for any $T_{\bar{s}} \geq L(\bar{t})$. On the other hand,

$$\begin{aligned} L_v(t) &\geq L_v(\bar{t}) - (1+\mu)(1+\rho)(\bar{t} - t) \\ &\geq L(\bar{t}) - (1+\mu)(1+\rho)(\Lambda_s + 2\Theta_{\bar{s}}) \\ &> L(\bar{t}) - \frac{5(1+\mu)}{2(1-\rho)\mu} \cdot \frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-2}} \\ &\overset{\sigma\geq 3}{\geq} L(\bar{t}) - \frac{\mathcal{I}(\tilde{\mathcal{G}})}{2^{\bar{s}-1}} + \left(\frac{1+\mu}{\mu} + \bar{s}\right)\frac{2\hat{\mathcal{G}}}{\sigma^{\bar{s}-2}} \\ &= L(\bar{t}) - \frac{\mathcal{I}(\tilde{\mathcal{G}})}{2^{\bar{s}-1}} + (1+\mu)(1+\rho)\Theta_{\bar{s}} + \bar{s}C_{\bar{s}-1}. \end{aligned}$$

Hence, (S2) can also not be violated with respect to any $T_{\bar{s}} < L(\bar{t})$. Thus, again all preconditions of Theorem 5.18 are met and the proof is complete. □

In order to obtain a bound on the gradient skew, it remains to show that there is an interval $[t_0, t_0 + \Lambda_2 + 3\Theta_2]$ during which the system is $C^{(t,u)}$-legal at all times $t \geq t_0$ and nodes u. If $\hat{\mathcal{G}} \approx \tilde{\mathcal{G}}$, the above reasoning is sufficient for this purpose.

Corollary 5.21. *If $\hat{\mathcal{G}} \geq \mathcal{G}(t)$ at all times t and $\sigma \geq 3$, the system is $C^{(t,u)}$-legal at all times $t \geq \frac{2\mathcal{I}(\tilde{\mathcal{G}})}{1-\rho}$ and nodes u.*

Proof. Observe that within $\frac{\mathcal{I}}{1-\rho}$ time, it must occur that $L(t) = \min_{u\in V}\{L_u(t)\}$. Shifting the time axis, it is hence sufficient to show the claim for all times $t \geq \frac{\mathcal{I}(\tilde{\mathcal{G}})}{1-\rho}$ under the assumption that $L(0) = \min_{u\in V}\{L_u(0)\}$.

We modify $C^{(t,u)}$ to "switch on" its guarantees level by level. That is, we consider the gradient sequence

$$\bar{C}_s^{(t,u)} := \begin{cases} C_1^{(t,u)} & \text{if } s = 1 \\ C_s^{(t,u)} & \text{if } s > 1 \text{ and } L_u(t) \geq L(0) + \Delta_s \\ \bar{C}_{s-1}^{(t,u)} & \text{if } s > 1 \text{ and } L_u(t) < L(0) + \Delta_s. \end{cases}$$

The proof is now analogous to the one of Lemma 5.20, with the exception that if $L_u(\bar{t}) < L(0) + \Delta_{\bar{s}}$, then $C_{\bar{s}}^{(t,u)} = C_{\bar{s}-1}^{(t,u)}$ and Lemma 5.15 immediately yields a contradiction. As the case $L_u(\bar{t}) \geq L(0) + \Delta_{\bar{s}}$ only uses bounds for levels $s \in \{2, \ldots, \bar{s}-1\}$ at logical times of at least $L(0) + \Delta_s$ (and on level 1 at times $t \geq 0$), the weaker guarantees offered by $\bar{C}$ are sufficient. Because during the time interval $[0, 0 + \Lambda_2 + 3\Theta_2]$ we have that $\bar{C}_s^{(}t, u) = \bar{C}_1^{(t,u)} = 2\hat{\mathcal{G}}$, the prerequisite that the system is legal at all nodes during this interval is satisfied.

Finally, observe that $L\left(\frac{\mathcal{I}}{1-\rho}\right) \geq L(0) + \mathcal{I}$, implying that $\bar{C}^{(t,u)} = C^{(t,u)}$ at times $t \geq \frac{\mathcal{I}}{1-\rho}$. □

We can now infer that the system achieves a gradient skew based on $\hat{\mathcal{G}}$, an upper bound on the global skew that holds at all times, and all edges that have been present for $O\left(\frac{\tilde{\mathcal{G}}}{\mu}\right)$ time, where $\tilde{\mathcal{G}}$ is the a priori upper bound on the global skew that is known to the algorithm.

Theorem 5.22. *Suppose that $\sigma \geq 3$ and $\hat{\mathcal{G}} \geq \mathcal{G}(t)$ for all times t. Denote by $G_{\mathcal{I}}(t)$ the graph on nodes V with all edges $\{u, v\}$ that have been continuously present for at least $\frac{2\mathcal{I}+\tilde{\mathcal{G}}+(1+\rho)(1+\mu)\mathcal{T}_{\{u,v\}}}{1-\rho} \in O(\frac{\tilde{\mathcal{G}}}{\mu})$ time. If path $p = (u, \dots, v)$ exists in $G_{\mathcal{I}}$ at time t, it holds that*

$$|L_v(t) - L_w(t)| = \left(\log_\sigma \frac{\hat{\mathcal{G}}}{\kappa_p} + O(1)\right) \kappa_p.$$

Proof. As $\hat{\mathcal{G}} \geq \mathcal{G}(t)$ for all t, we can apply Corollary 5.21. This shows that for all times $t \geq \frac{2\mathcal{I}}{1-\rho}$, the system is $C^{(t,u)}$-legal at each node u. In particular, it is legal w.r.t. the gradient sequence given by $C_s = \frac{2\hat{\mathcal{G}}}{\sigma^{\max\{s-2,0\}}}$. At smaller times, $G_{\mathcal{I}}$ contains no edges. Hence, for any path $p = (u, \dots, v)$ that exists in $G_{\mathcal{I}}$ at such a time t and any $s \in \mathbb{N}$, Lemma 5.14 yields that

$$|L_v(t) - L_w(t)| \leq \left(s + \frac{1}{2}\right) \kappa_p + \frac{C_s}{2}.$$

Choosing $s = 2 + \left\lceil \log_\sigma \frac{\hat{\mathcal{G}}}{\kappa_p} \right\rceil$, we have that $C_s \leq \kappa_s$ and the statement of the theorem follows. □

This theorem has two shortcomings. First, even if $\hat{\mathcal{G}} \ll \tilde{\mathcal{G}}$, it does not provide any guarantee at times $t \ll \frac{\tilde{\mathcal{G}}}{\mu}$. Second, if the global skew is large and then decreases later, the gradient skew bound does not adapt. In the next section, we address these points.

5.4 Fast Stabilization in Case of Small Global Skew

We will again rely on Lemma 5.20, but need to handle the case that $\hat{\mathcal{G}} \ll \tilde{\mathcal{G}}$ differently than in Corollary 5.21. First, we prove the precondition of the lemma under the assumption that for $O(\frac{\hat{\mathcal{G}}}{\mu})$ time no edge is inserted, and then we show that within $O(\frac{\hat{\mathcal{G}}}{\mu})$ time, such a "silent" period must occur.

Lemma 5.23. *Set $\Gamma := \frac{(29+8\mu)\hat{\mathcal{G}}}{2(1-\rho)\mu} \in O(\frac{\hat{\mathcal{G}}}{\mu})$. Suppose that $\sigma \geq 3$, that $\mathcal{G}(t) \leq \hat{\mathcal{G}}$ for all $t \in [t', t' + \Gamma]$, and that no node inserts an edge during this interval. Then the system is $C^{(t,u)}$-legal at all times $t \in [t' + \Gamma - \Lambda_2 - 3\Theta_2, t' + \Gamma]$ and nodes u.*

Proof. Set $C_s := \frac{2\hat{\mathcal{G}}}{\sigma^{s-1}}$, $t_2 := t' + \Lambda_2 + \left(2 + \frac{(1+\mu)(1+\rho)}{1-\rho}\right)\Theta_2 + \frac{C_1}{1-\rho}$ and $t_s := t_{s-1} + \Theta_{s-1} + \Lambda_s + 2\Theta_s$ for $3 \leq s \in \mathbb{N}$. Consider the gradient sequence

$$C_s^{(t)} := \begin{cases} 2\hat{\mathcal{G}} & \text{if } s = 1 \\ C_{s-1}^{(t)} & \text{if } t < t_s \\ C_s & \text{if } t \geq t_s. \end{cases}$$

We claim that the system is legal with respect to this sequence at all nodes and times $t \in [t', t'+\Gamma]$, which we show by induction on the level s. For $s = 1$, the claim is trivial. For step from $s-1$ to $s \geq 2$, note that, by Lemma 5.15, the claim is immediate from $(s-1)$-legality for all times $t < t_s$, so assume that $t \geq t_s$. We apply Theorem 5.18 for level s with $t^+ = t$, $t^- = t - \Lambda_s - 2\Theta_s$, and the gradient sequence C_s, which will complete the induction step.

It remains to establish the preconditions of the theorem. (S0) holds because $t^- \geq t' + \Theta_2$. (S1) holds because for any time $t'' \geq t - \Lambda_s - 2\Theta_s - \Theta_{s'} \geq t_{s-1}$ we have that $C_{s-1}^{(t'')} = C_{s-1}$. Concerning (S3), by assumption there are no edge insertions during $[t', t'+\Gamma]$. W.l.o.g., we may assume that there are no edge insertions after time $t'+\Gamma$: future events cannot influence past clock values, and we can always extend the execution such that no further edges are inserted (e.g. by removing all edges at time $t'+\Gamma$). Hence, the only remaining case is that there is a logical time $T_s \in \mathbb{T}_s$ and a node v such that $L_v(t') > T_s$, but $L_v(t^-) < T_s + (1+\mu)(1+\rho)\Theta_s + C_{s-1}$. However, $t^- \geq t_2 - \Lambda_2 - 2\Theta_2 \geq \left(\frac{(1+\mu)(1+\rho)}{1-\rho}\right)\Theta_2 + \frac{2C_1}{1-\rho}$ and thus

$$L_v(t^-) \geq L_v(t') + (1+\mu)(1+\rho)\Theta_2 + 2C_1 \overset{\sigma \geq 3/2}{>} T_s + (1+\mu)(1+\rho)\Theta_s + sC_{s-1},$$

showing (S2).

We conclude that the system is C-legal at all nodes and times $t \in [t_\infty, t'+\Gamma]$, where $t_\infty := \lim_{s\to\infty} t_s$. As

$$\begin{aligned}
t' + \Gamma - t_\infty &= \Gamma - \frac{(1+\mu)(1+\rho)}{1-\rho} \cdot \Theta_2 + \frac{C_1}{(1-\rho)} - \sum_{s=2}^{\infty} (\Lambda_s + 3\Theta_s) \\
&= \Gamma - \left(\frac{1+\mu}{(1-\rho)\mu} + 1\right) C_1 - \frac{\sigma}{\sigma-1} \cdot \left(\frac{C_1}{2(1-\rho)\mu} + \frac{C_1}{(1+\rho)\mu}\right) \\
&\overset{\sigma \geq 3}{>} \frac{5\hat{\mathcal{G}}}{(1-\rho)\mu} \\
&> \frac{C_1}{2(1-\rho)\mu} + \frac{C_1}{(1+\rho)\mu} \\
&= \Lambda_2 + 3\Theta_2.
\end{aligned}$$

As $C_s \leq C_s^{(t,u)}$ for all times t and nodes u, the claim of the lemma follows. ∎

Lemma 5.24. *Set $\Gamma' := \frac{(2+(1+\rho)\mu)(29+10\mu)\hat{\mathcal{G}}}{(1-\rho)^2\mu} \in O(\frac{\hat{\mathcal{G}}}{\mu})$ and suppose that $\mathcal{G}(t) \leq \hat{\mathcal{G}} \leq \frac{20\tilde{\mathcal{G}}}{(1+\rho)(29+8\mu)}$ for all $t \in [t', t'+\Gamma']$. Then there is some time $t'' \in [t', t'+\Gamma'-\Gamma]$ such that no edges are inserted during $t \in [t'', t''+\Gamma]$.*

Proof. If the statement does not hold for $t'' = t'$, let t be the minimal time when an edge insertion occurs during $[t', t'+\Gamma]$ at some node u. Denoting by s the level on which the edge is inserted, we have that $L_u(t) = T_s$ for some $T_s \in \mathbb{T}_s$. Observe that the next larger logical time at which some edge could be inserted is $T_s + \frac{\mathcal{I}}{2^s}$. For any node v, we have that $L_v\left(t + \frac{\hat{\mathcal{G}}}{1-\rho}\right) \geq L_v(t) + \hat{\mathcal{G}} \geq T_s$ and $L_v\left(t + \frac{\mathcal{I}}{2^s(1+\rho)(1+\mu)} + \frac{\hat{\mathcal{G}}}{(1+\rho)(1+\mu)}\right) \leq T_s + \frac{\mathcal{I}}{2^s}$. Hence, either the claim holds for $t'' = t + \frac{\hat{\mathcal{G}}}{1-\rho} + \varepsilon$ (for some sufficiently small $\varepsilon > 0$) or

$$\frac{\mathcal{I}}{2^s} \leq \frac{(2+(1+\rho)\mu)\hat{\mathcal{G}}}{1-\rho} + (1+\rho)(1+\mu)\Gamma. \tag{28}$$

In the latter case, we use that for any node v, we have that

$$L_v\left(t+\frac{\mathcal{I}}{(1-\rho)2^{s-1}}+\frac{\hat{\mathcal{G}}}{1-\rho}\right)\geq T_s+\frac{\mathcal{I}}{2^{s-1}}=z\mathcal{I}$$

for some $z\in\mathbb{Z}$. Setting $\bar{t}:=\max_{v\in V}\{t_v\,|\,L_v(t_v)=z\mathcal{I}\}$, we then have that

$$L_v\left(\bar{t}-\frac{\hat{\mathcal{G}}}{1-\rho}\right)\geq L_v(\bar{t})-\hat{\mathcal{G}}\geq z\mathcal{I}$$

and

$$L_v\left(\bar{t}+\frac{\mathcal{I}-2\hat{\mathcal{G}}}{2(1+\rho)(1+\mu)}\right)\leq L_v(\bar{t})+\frac{\mathcal{I}}{2}-\hat{\mathcal{G}}\leq\left(z+\frac{1}{2}\right)\mathcal{I},$$

which due to

$$\frac{\mathcal{I}-2\hat{\mathcal{G}}}{2(1+\rho)(1+\mu)}>\frac{20\tilde{\mathcal{G}}}{2(1+\rho)(1-\rho)\mu}\geq\frac{(29+8\mu)\hat{\mathcal{G}}}{2(1-\rho)\mu}=\Gamma$$

yields that no edges are inserted during $[t'',t''+\Gamma]$ if we set $t'':=\bar{t}+\varepsilon$ for sufficiently small $\varepsilon>0$. As

$$\begin{aligned}
\bar{t}+\Gamma &\leq t+\frac{\mathcal{I}}{(1-\rho)2^{s-1}}+\frac{\hat{\mathcal{G}}}{1-\rho}+\Gamma\\
&\leq t'+\frac{\mathcal{I}}{(1-\rho)2^{s-1}}+\frac{\hat{\mathcal{G}}}{1-\rho}+2\Gamma\\
&\overset{(28)}{\leq} t'+\frac{(2+(1+\rho)\mu)\hat{\mathcal{G}}}{(1-\rho)^2}+\frac{2(1+\rho)(1+\mu)\Gamma}{1-\rho}+\frac{\hat{\mathcal{G}}}{1-\rho}+2\Gamma\\
&< \left(\frac{4+2(1+\rho)\mu}{1-\rho}\right)\left(\frac{\hat{\mathcal{G}}}{1-\rho}+\Gamma\right)\\
&= t'+\Gamma',
\end{aligned}$$

this proves the claim of the lemma. □

Together, the above results yield the following theorem.

Theorem 5.25. *Suppose that $\sigma\geq 3$ and $\hat{\mathcal{G}}\geq\mathcal{G}(t)$ for all times t. Set $S:=\Gamma'$ if $\hat{\mathcal{G}}\leq\frac{20\tilde{\mathcal{G}}}{(1+\rho)(29+8\mu)}$ and $S:=\frac{2\mathcal{I}}{1-\rho}$ otherwise, and let $s\in\mathbb{N}$. At any time $t\geq S\in O(\frac{\hat{\mathcal{G}}}{\mu})$, any level-$s$ path $p=(u,\dots,v)$ satisfies that*

$$|L_v(t)-L_w(t)|=\left(s+\frac{1}{2}\right)\kappa_p+\frac{\hat{\mathcal{G}}}{\sigma^{\max\{s-2,0\}}}.$$

Proof. If $S\neq\Gamma'$, we can apply Corollary 5.21. Otherwise, we apply Lemma 5.24 to show that the preconditions of Lemma 5.23 are satisfied for a time $t'\leq t-\Gamma$. This enabls us to apply Lemma 5.20 for a time $t_0\leq t$. In both cases, we have shown $C^{(t,u)}$-legality at all nodes $u\in V$. As $C_s^{(t,u)}\leq\frac{\hat{\mathcal{G}}}{\sigma^{\max\{s-2,0\}}}$ for all t, u, and s, the claim now follows from Lemma 5.14. □

In particular, if the global skew is bounded by $\hat{\mathcal{G}}$ for $O(\frac{\hat{\mathcal{G}}}{\mu})$ time, on fully inserted edges we have a stable gradient skew that depends on $\hat{\mathcal{G}}$ only.

Corollary 5.26. *Denote by $G_\infty(t)$ the graph on nodes V and with all edges that have been inserted on all levels. Then for any path $(u, \ldots, v)$ in this graph we have that*

$$|L_v(t) - L_w(t)| = \left(\log_\sigma \frac{\hat{\mathcal{G}}}{\kappa_p} + O(1)\right)\kappa_p.$$

5.5 Discussion

Before proceeding to the remaining more technical sections, we briefly put the obtained results in context.

Optimality

Our algorithm is simultaneously optimal or asymptotically optimal in terms of several parameters:

Global Skew. The global skew bound given by Theorem 5.6 is optimal in the sense that there are executions with dynamic diameter of D in which a skew of D cannot be avoided [12, 16].

Clock Rates. The algorithm guarantees that no logical clock runs at rate smaller than $1-\rho$ and the largest clock value increases at most at rate $1+\rho$, which is clearly optimal. Moreover, the maximum logical clock rate is $(1+\mu)(1+\rho)$. The above proofs assumed that $\sigma \geq 3$ and thus $\mu \geq \frac{6\rho}{1-\rho}$. However, more careful reasoning would show that any $\mu > \frac{2\rho}{1-\rho}$ is feasible (at the expense of diverging insertion times as μ approaches this bound). This is optimal, as it is necessary that $(1-\rho)(1+\mu) > 1+\rho$ so that nodes with slow hardware clocks can catch up to those with fast hardware clocks.

Gradient Skew. Provided that the global skew has been bounded by $\hat{\mathcal{G}}$ for sufficiently long, the stable gradient skew between nodes u and v connected by a path p with κ_p is $\left(\log_\sigma \frac{\hat{\mathcal{G}}}{\kappa_p} + O(1)\right)\kappa_p$, where $\kappa_e = 4(\epsilon_e + \mu\tau_e)$. The first term is matched by a lower bound of $\epsilon_p \log_{\Theta(\sigma)} \frac{\hat{\mathcal{G}}}{\epsilon_p}$ for the static case [16], i.e., if $\mu\tau_e \ll \epsilon_e$, this bound is optimal up to factor $4 + o(1)$.

The additional slack of $\mu\tau_e$ compensates for the amount by which logical clock values can drift apart while only one endpoint is aware of the edge. It seems plausible that such a term is needed, but no matching lower bound is known. However, it is worth pointing out that $\epsilon_e \geq \mu d_e$, where d_e is the time it takes for information on clock values to propagate along the edge e. One can thus expect that in most systems $\tau_e \in O(\epsilon_e)$.

Stabilization Time. For a given global skew bound $\hat{\mathcal{G}}$, Theorem 5.25 shows that within $O(\frac{\hat{\mathcal{G}}}{\mu})$ time the gradient skew bound holds for all inserted edges. A lower bound of $\Omega(D)$ for establishing the gradient skew bound is given in Section 8, where D is the (current) diameter of the graph. As pointed out above, there are executions in which a global skew of D cannot be avoided; in fact a skew of $\Omega(D)$ can be hidden from the algorithm entirely. Therefore, for $\mu \in \Theta(1)$, this bound would be asymptotically optimal (recall that we assumed $\mu \in O(1)$; for larger values of μ, we would obtain a bound of $\Theta(\hat{\mathcal{G}})$ in Theorem 5.25). Note also that $O(\frac{\hat{\mathcal{G}}}{\mu})$ time is trivially necessary, as a newly inserted edge e may exhibit a skew of $\hat{\mathcal{G}}$, regardless of its uncertainty ϵ_e.

Unfortunately, our algorithm takes $\Theta(\frac{\tilde{\mathcal{G}}}{\mu} + \tau_e + \mathcal{T}_e)$ time to *insert* a newly discovered edge e, cf. Theorem 5.22. The additive terms of τ_e and $\mathcal{T}_e$ are necessary for guaranteeing *any* communication between its endpoints, and therefore must be present either explicitly or via the abstraction of the estimate layer (by making stronger assumptions on the interface it provides). However, it may be the case that $\tilde{\mathcal{G}} \gg \hat{\mathcal{G}}$, especially since we assume that $\tilde{\mathcal{G}}$ is an upper bound on the global skew that holds at all times. In Section 7, we discuss how to insert edges based on local, time-dependent global skew estimates $\tilde{\mathcal{G}}_u(t)$, overcoming this issue.

The leading constants in Theorem 5.22 and Theorem 5.25 are moderate. For example, if $\mu \leq \frac{1}{100}$ and thus $\rho \leq \frac{\mu}{100}$ ($\rho \leq 10^{-5}$ for a typical quartz oscillator), then

$$\frac{2\mathcal{I} + \tilde{\mathcal{G}}}{1-\rho} < \frac{43\tilde{\mathcal{G}}}{\mu}$$

and

$$\Gamma' < \frac{58\hat{\mathcal{G}}}{\mu}.$$

We remark that in the interest of a more streamlined presentation, we did not attempt to optimize constants. We conjecture that the leading constants can be reduced to less than 10 without introducing additional techniques.

Comparison to Simultaneous Insertion on all Levels

In [16], we presented a simpler insertion strategy and analysis that inserts edges on all levels right after discovering them. The idea is to initially give the edge a very large weight κ, so that the gradient skew bound is trivially satisfied due to the global skew bound, and then reduce the weight exponentially until the final value is reached. Adding some additional slack to the (final) κ, it is then shown that the gradient property holds on all existing edges w.r.t. the time-dependent values of κ.

Compared to the solution present so far, the gradient skew bound we achieve here is slightly better (as no additional slack is needed). More importantly, the insertion time is asymptotically optimal in case $\tilde{\mathcal{G}} = \hat{\mathcal{G}}$, in contrast to a multiplicative overhead of $\Theta(\frac{\tilde{\mathcal{G}}}{\min_e\{\epsilon_e\}})$ for the simpler strategy. Given that the simpler strategy has a leading constant of at least 24 in its insertion time bound, the above bounds compare favorably with this approach.

A big advantage of the simpler approach, however, is that it can readily use local and time-dependent estimates $\tilde{\mathcal{G}}_u(t)$ for edge insertion, thus performing better in case of a large gap between $\tilde{\mathcal{G}}$ and $\hat{\mathcal{G}}$. As mentioned earlier, we adapt our insertion strategy to account for such local estimates in Section 7. Unfortunately, this results in a very large leading constant, meaning that the simpler insertion strategy performs better in practice, as $\frac{\tilde{\mathcal{G}}}{\min_e\{\epsilon_e\}}$ is extremely unlikely to exceed 10^3. We thus prove that an asymptotically optimal insertion time can be achieved even for time- and node-dependent estimates of the global skew, but leave whether they can be used to obtain small insertion times in practice open.

6 Proving Convergence

On an abstract level, the proof of Theorem 5.18 follows the same strategy as in the static case. Consider the potentials $\Psi^s := \max_{u \in V}\{\Psi_u^s\}$ and $\Xi^s := \max_{u \in V}\{\Xi_u^s\}$, $s \in \mathbb{N}$. By the design of the algorithm, in the static case Ψ^s grows at any time at rate at most $2\rho = 1 + \rho - (1 - \rho)$ because *any* node v that is the endpoint with larger clock value of a path maximizing Ψ^s must be in slow mode. The reason is that **SC** not being satisfied at the node on level s implied the existence of a neighbor w of v that is more than $s\kappa$ ahead, yielding a path with even larger ψ^s-value. On the other hand, the algorithm is much more aggressive with respect to Ξ^s: Whenever $\Xi^s > 0$, it is shown that *any* node v that is the endpoint with the smaller clock value of a path maximizing Ξ^s will be in fast mode. This holds as **FC** not being satisfied for s at v would permit to extend the path by a neighbor w of v to a new path with larger ξ^s-value. These two statements play together to ensure that the maximal values of Ψ^s, i.e., the sequence C_s, decreases exponentially in s, which is the essence of the statement of Theorem 5.18.

6.1 Existence of Relevant Paths

Given the level-s stabilization condition, we can conclude that paths with sufficiently large ξ- or ψ-values to be of interest cannot have a large weight, and infer that the system must have been s'-legal for all $1 < s' < s$ at one of the path's endpoints for a long time.

Lemma 6.1. *Let $s > s' \geq 1$ be two integers and let $t \geq 0$ be a time. Assume that the (C, s)-stabilization condition holds at node u at some time in $[t, t + \Theta_{s'}]$ (if $s' > 1$) or $[t, t + \Theta_2]$ (if $s' = 1$), and let $p = (u, \ldots, v)$ be a path for which $p \in P_u^{s'}(t)$. Moreover, suppose that $\xi_p^s(t) \geq 0$ or $\psi_p^s(t) \geq -\kappa_p/2$. Then*

$$\kappa_p \leq C_{s'}$$

and the system is s'-legal at v and time t.

Proof. Because $\psi_p^s(t) = \xi_p^s(t) - \kappa_p/2$, we assume, w.l.o.g., that $\xi_p^s(t) \geq 0$. It holds that

$$0 \leq \xi_p^s(t) = L_u(t) - L_v(t) - s\kappa_p \leq \mathcal{G}(t) - \frac{\kappa_p}{2} \leq \frac{C_1 - \kappa_p}{2},$$

implying $\kappa_p \leq C_1$. The last inequality follows from (S0) because the stabilization condition at node u holds at some time $t' \in [t, t + \Theta_2]$. Together with Lemma 5.17, this implies the claim of the lemma for $s' = 1$ (and arbitrary $s > 1$).

For $s' \geq 2$, we can therefore define s'' to be the largest integer in $\{1, \ldots, s'\}$ for which both $\kappa_p \leq C_{s''}$ and v is (C, s'')-legal at time t. We want to show that $s'' = s'$, so assume for the sake of contradiction that $s'' < s'$. From the precondition that $p \in P_u^{s'}(t)$, we also get that $p \in P_u^{s''}(t)$ (Statement (i) of Lemma 5.15). Since we have (C, s'')-legality at nodes u and v at time t, we can use Lemma 5.14 to obtain an upper bound on the skew between u and v:

$$|L_u(t) - L_v(t)| \leq \left(s'' + \frac{1}{2}\right)\kappa_p + \frac{C_{s''}}{2} \leq (s'' + 1)C_{s''}.$$

We know that the (C, s)-stabilization condition holds at node u at some time $t' \in [t, t + \Theta_{s'}]$.

Because logical clocks progress at a rate between $1-\rho$ and $(1+\mu)(1+\rho)$, we get

$$\begin{aligned}|L_u(t')-L_v(t')| &\le |L_u(t)-L_v(t)| + ((1+\mu)(1+\rho)-(1-\rho))\Theta_{s'}\\ &\le (s''+1)C_{s''} + (2\rho+\mu(1+\rho))\Theta_{s'}\\ &\le (s''+1)C_{s''} + (2\rho+\mu(1+\rho)))\Theta_{s''+1}.\end{aligned}$$

From Requirement (S1) of the (C,s)-stabilization condition at node u at time t' for level $s''+1\le s'<s$, we therefore get that the system is $(C,s''+1)$-legal at node v and time t.

As $p\in P_u^{s''}(t)$ also implies that $\bar{p}\in P_v^{s''}(t)$, $(s''+1)$-legality at node v and time t implies that

$$\frac{C_{s''+1}}{2} > \Psi_v^{s''+1}(t) \ge \psi_{\bar{p}}^{s''+1}(t) = \psi_{\bar{p}}^{s}(t) + (s-(s''+1))\kappa_p \ge \psi_{\bar{p}}^{s}(t)+\kappa_p = \xi_p^s(t) + \frac{\kappa_p}{2} \ge \frac{\kappa_p}{2},$$

i.e., $\kappa_p < C_{s''+1}$, contradicting the maximality of s''. We conclude that $s''=s'$, concluding the proof. □

Next, we show that the level-s stabilization condition guarantees that paths of relevant skew on level s must have existed for at least Θ_s time on level s. Moreover, if we append an edge e to such a path that exists only in one direction, we can still argue that the entire path has been a level-s path until at most τ_e time ago.

Lemma 6.2. *Assume that node $u\in V$ satisfies the (C,s)-stabilization condition for a gradient sequence C and an integer $s>1$ throughout a time interval $[t^-,t^+]$. Consider a path $p=(u,\dots,v)\in P_u^s(t^+)$. If $\xi_p^s(t^+)\ge 0$, $\psi_p^s(t^+)\ge -\kappa_p/2$, $\xi_{\bar{p}}^s(t^+)\ge 0$, or $\psi_{\bar{p}}^s(t^+)\ge -\kappa_p/2$, then $p\in P_u^s(t)$ for all*

$$t\in\left[t^- - \Theta_s, t^+\right].$$

Furthermore, let $p'=p\circ(v,w)=(u,\dots,v,w)$. If $w\in N_v^s(t^+)$ and $\xi_{p'}^s(t^+-\tau_{\{v,w\}})\ge 0$, $\psi_{p'}^s(t^+-\tau_{\{v,w\}})\ge -\kappa_{p'}/2$, $\xi_{\bar{p}'}^s(t^+-\tau_{\{v,w\}})\ge 0$, or $\psi_{\bar{p}'}^s(t^+-\tau_{\{v,w\}})\ge -\kappa_{p'}/2$, then $p'\in P_u^s(t)$ for all

$$t\in\left[t^- - \Theta_s, t^+ - \tau_{\{v,w\}}\right],$$

and this time interval is non-empty.

Proof. For all t, we have $\xi_{\bar{p}}^s(t)=\psi_p^s(t)+\kappa_p/2$ and $\psi_{\bar{p}}^s(t)=\xi_p^s-\kappa_p/2$. Therefore, assuming that $\xi_p^s(t^+)\ge 0$, $\psi_p^s(t^+)\ge -\kappa_p/2$, $\xi_{\bar{p}}^s(t^+)\ge 0$, or $\psi_{\bar{p}}^s(t^+)\ge -\kappa_p/2$ implies that either $\xi_p^s(t^+)\ge 0$ or $\psi_p^s(t^+)\ge -\kappa_p/2$. Because the system satisfies the (C,s)-stabilization condition at node u at time t^+, using Lemma 6.1, together with $p\in P_u^s(t^+)\supseteq P_u^{s-1}(t^+)$ both cases imply that $\kappa_p\le C_{s-1}$.

In the following, we define

$$T_s^+ := \min\left\{T_s\in\mathbb{T}_s : T_s\ge L_u(t^+)\right\} \quad\text{and}\quad \forall s'\le s : T_{s'}^- := \max\left\{T_{s'}\in\mathbb{T}_{s'} : T_{s'}\le L_u(t^-)\right\}.$$

For all $t\in[t^-,t^+]$, the stabilization condition yields that

$$T_s^- + (1+\rho)(1+\mu)\Theta_s - sC_{s-1} < L_u(t) < T_s^+ - (1+\rho)(1+\mu)\Theta_s + sC_{s-1}. \tag{29}$$

Let X be the set of nodes of the path p. In order to prove the first part of the lemma, we show that

$$\forall t\in[t^- - \Theta_s, t^+],\ \forall x\in X:\ x \text{ is } (C,s-1)\text{-legal at time } t \text{ and } N_x^s(t)\supseteq N_x^s(t^+). \tag{30}$$

Note that this directly implies the first part of the lemma. Before showing (30), we first show that

$$\forall t \in [t^- - \Theta_s, t^+] \text{ for which (30) holds}, \forall x \in X: \; T_s^- < L_x(t) < T_s^+. \tag{31}$$

As at time t (30) holds, the sub-path $(u, \ldots, x) \in P_u^s(t) \supseteq P_u^{s-1}(t)$ and the system is $(C, s-1)$-legal at both u and x at time t. Applying Lemma 5.14, we see that

$$|L_u(t) - L_x(t)| \leq s \cdot C_{s-1}$$

for all $x \in X$. Because $t > t^- - \Theta_s$, we have $L_u(t) > L_u(t^-) - (1+\mu)(1+\rho)\Theta_s$ and therefore

$$|L_u(t^-) - L_x(t)| < (1+\mu)(1+\rho)\Theta_s + s \cdot C_{s-1}.$$

Together with Inequality (29), this implies that $T_s^- < L_x(t) < T_s^+$ and thus (31) for all $x \in X$.

Let us now show that (30) holds. Trivially, it holds that $N_x^s(t^+) \supseteq N_x^s(t^+)$. In addition, due to Statement (i) of Lemma 5.15 we have that $p \in P_u^{s-1}(t^+)$ and Lemma 6.1 therefore establishes (30) for time $t = t^+$. Now suppose for the sake of contradiction that the non-empty maximal time interval $[t', t^+] \subseteq [t^- - \Theta_s, t^+]$ for which both statements of (30) are satisfied is not equal to $[t^- - \Theta_s, t^+]$, i.e., $t' > t^- - \Theta_s$. At time t', (30) still holds and therefore from (31), we get that $T_s^- < L_x(t') < T_s^+$. As nodes add level-s neighbors only at times $T_s \in \mathbb{T}_s$, it follows that there is some time $t'' \in [t^- - \Theta_s, t')$ such that $N_x^s(t) \supseteq N_x^s(t') \supseteq N_x^s(t^+)$ for all $t \in [t'', t']$. Therefore, $p \in P_u^{s-1}(t) \subseteq P_u^s(t)$ for all such t and because u satisfies the (C, s)-stabilization condition throughout interval $[t^-, t^+]$ and thus at some time in $[t, t+\Theta_s] \subseteq [t, t+\Theta_{s-1}]$, we can apply Lemma 6.1 to infer that the system is $(s-1)$-legal at x at time t. This is a contradiction to the maximality of the interval $[t', t^+] \subseteq [t^- - \Theta_s, t^+]$ (i.e., the minimality of t'). We conclude that indeed $t' = t^- - \Theta_s$, in particular showing the first statement of the lemma.

It therefore remains to prove the second part of the lemma. For the path p' used in the second part of the claim, we know that either

$$\xi_{p'}^s(t^+ - \tau_{\{v,w\}}) \geq 0 \quad \text{or} \quad \psi_{p'}^s(t^+ - \tau_{\{v,w\}}) \geq -\frac{\kappa_{p'}}{2}. \tag{32}$$

Let us first assume that $\tau_{\{v,w\}} < \Theta_s$ and that $v \in N_w^s(t^+ - \tau_{\{v,w\}})$. From $w \in N_v^s(t^+)$ and (30), this implies that $w \in N_v^s(t^+ - \tau_{\{v,w\}})$ and thus $p' \in P_u^s(t^+ - \tau_{\{v,w\}})$. Further, by the stabilization condition, the system is $(C, s-1)$-legal at u at time $t^+ - \tau_{\{v,w\}}$. Because by (32), either $\xi_{p'}^s(t^+ - \tau_{\{v,w\}}) \geq 0$ or $\psi_{p'}^s(t^+ - \tau_{\{v,w\}}) \geq -\kappa_{p'}/2$, Lemma 6.1 then implies that $\kappa_{p'} \leq C_{s-1}$. Therefore, in that case exactly the same argument as for the first part of the lemma also shows that for all $t \in [t^- - \Theta_s, t^+ - \tau_{\{v,w\}}]$ we have $p' \in P_u^s(t)$. To also prove the second claim, it therefore suffices to show that $\tau_{\{v,w\}} < \Theta_s$ and that $v \in N_w^s(t^+ - \tau_{\{v,w\}})$.

To this end, we will prove by induction that

(i) $\forall s' \in \{1, \ldots, s\} : p' \in P_u^{s'}(t^+ - \tau_{\{v,w\}})$, and

(ii) $\forall s' \in \{2, \ldots, s\} : \tau_{\{v,w\}} < \Theta_{s'}$.

For $s' = s$, these statements imply the above and hence complete the proof.

Before we continue with the induction, we make the following observation. In the first part of the lemma, we proved (31), implying that no node $x \in X$ adds a new neighbor to N_x^s during the interval $[t^- - \Theta_s, t^+]$. Thus, for all $x \in X$ and all $y \in N_x^s(t^+)$, y has been added to N_x^s at the latest at time $t < t^- - \Theta_s$ for which $L_x(t) = T_s^-$. As x adds y to set N_x^s after adding y to sets $N_x^{s'}$ for all $s' < s$, this also implies that for all $x \in X$, all $y \in N_x^s(t^+)$, and all $s' \leq s$, node x adds y to $N_x^{s'}$ at the latest at time t for which $L_x(t) = T_{s'}^-$. Note that this includes node w which is in the set $N_v^s(t^+)$. By Lemma 5.5, both nodes on an edge add the edge to the respective neighbor sets at the same logical times. We therefore also know that w adds v to $N_w^{s'}$ at the latest at time $t_w^{s'}$ such that $L_w(t_w^{s'}) = T_{s'}^-$. To prove (i) for a specific s', it is thus sufficient to prove for all $x \in X \cup \{w\}$ (note the inclusion of w!) that $L_x(t^+ - \tau_{\{v,w\}}) \geq T_{s'}^-$.

We now proceed with the induction. We anchor it at $s' = 2$. Let us first consider (ii) for $s' = 2$. From (S0) of the (C, s)-stabilization condition at node u at time t^+, we have $C_1 > (1 + \rho)\mu\tau_{\{v,w\}}$ and therefore

$$\tau_{\{v,w\}} < \frac{C_1}{(1+\rho)\mu} = \Theta_2. \tag{33}$$

Further, as node u satisfies the (C, s)-stabilization condition at time t^+ (for $s \geq 2$), from (S2), we have

$$\begin{aligned} L_u(t^+ - \tau_{\{v,w\}}) &\geq L_u(t^+) - (1+\mu)(1+\rho)\tau_{\{v,w\}} \\ &\overset{(S2)}{\geq} T_2^- + (1+\mu)(1+\rho)\Theta_2 + 2C_1 - (1+\mu)(1+\rho)\tau_{\{v,w\}} \\ &\overset{(33)}{>} T_2^- + 2C_1 \\ &\overset{(S0)}{\geq} T_2^- + \mathcal{G}(t^+ - \tau_{\{v,w\}}). \end{aligned}$$

The last inequality follows from $(S0)$ because we already know that $\tau_{\{v,w\}} < \Theta_2$. For every node $x \in V$ (and therefore in particular for every node $x \in X \cup \{w\}$), we thus have

$$L_x(t^+ - \tau_{\{v,w\}}) > T_2^-.$$

We have already seen that this implies also Statement (i) for $s' = 2$.

The induction step comes in two parts. First, we prove for $s' \in \{1, \ldots, s-1\}$ that Statement (i) for s' and Statement (ii) for s' imply Statement (ii) for $s' + 1$. From Statement (i) for s', we know that $p' \in P_u^{s'}(t)$ for $t^+ - \tau_{\{v,w\}}$. From Statement (ii), we also know that $t^+ - \tau_{\{v,w\}} > t^+ - \Theta_{s'}$. Since the (C, s)-stabilization condition holds at node u at time t^+, we therefore know that u is (C, s')-legal at time $t^+ - \tau_{\{v,w\}}$. Together with (32), Lemma 6.1 then implies that $\kappa_{p'} \leq C_{s'}$. We thus get that

$$\tau_{\{v,w\}} \overset{(9)}{<} \frac{\kappa_{\{v,w\}}}{(1+\rho)\mu} \overset{(\kappa_{p'} \leq C_{s'})}{\leq} \frac{C_{s'}}{(1+\rho)\mu} = \Theta_{s'+1},$$

To conclude the induction step, we now also show that Statement (i) for s' and Statement (ii) for $s' + 1$ imply Statement (i) for $s' + 1$. We lower bound $L_u(t^+ - \tau_{\{v,w\}})$ as follows:

$$\begin{aligned} L_u(t^+ - \tau_{\{v,w\}}) &\geq L_u(t^+) - (1+\mu)(1+\rho)\tau_{\{v,w\}} \\ &\overset{\text{I.H.}}{>} L_u(t^+) - (1+\mu)(1+\rho)\Theta_{s'+1} \\ &\overset{(S2)}{\geq} T_{s'+1}^- + (1+\mu)(1+\rho)\Theta_{s'+1} + (s'+1)C_{s'} - (1+\mu)(1+\rho)\Theta_{s'+1} \\ &= T_{s'+1}^- + (s'+1)C_{s'}. \end{aligned}$$

As $p' \in P_u^{s'}(t)$, for each $x \in X \cup \{w\}$ Lemma 6.1 yields that the system is s'-legal at x and time $t^+ - \tau_{\{v,w\}}$ as well as $\kappa_{p'} \leq C_{s'}$. By Lemma 5.14, it follows that $L_x(t^+ - \tau_{\{v,w\}}) > T^-_{s'+1}$. As already noted, this implies that $p' \in P_u^{s'+1}(t^+ - \tau_{\{v,w\}})$, as required. □

6.2 Properties of Ξ

The next lemma shows, roughly speaking, that when a node u is too far ahead on some level-s paths—that is, it has a positive Ξ_u^s value—then all endpoints of paths p satisfying $\xi_p^s = \Xi_u^s$ are in fast mode, trying to catch up to u. Their clocks increase at a rate of at least $(1-\rho)(1+\mu)$, the slowest possible fast rate. However, we do not know what node u itself does in this situation: because of the local nature of the algorithm, u does not necessarily realize that it has a large skew, and it can be in either slow mode or fast mode. In the latter case, the skew on the path might actually increase, due to hardware clock drift; thus we cannot necessarily show that the skew decreases. The lemma states that over an interval, the weighted skew Ξ_u^s increases by at most u's logical clock increase, *minus* the catching-up that nodes trailing behind u achieve at a rate of $(1-\rho)(1+\mu)$. Finally, it may be the case that an edge e is not present throughout the entire relevent time period, causing us to "jump" back in by τ_e time; in this case, we use some additional slack in **FC** to gain a "reserve term" accounting for the resulting time difference later on.

Lemma 6.3. *Assume that a node $u \in V$ satisfies the (C,s)-stabilization for $s > 1$ throughout a time interval $[t^-, t^+]$, and that for all $t \in (t^-, t^+)$ we have $\Xi_u^s(t) > 0$. Then there exists a time*

$$t' \in \left[t^- - \Theta_s, t^-\right]$$

such that

$$\Xi_u^s(t^+) - \Xi_u^s(t') \leq L_u(t^+) - L_u(t') - (1-\rho)(1+\mu)(t^+ - t') - (1+\rho)\mu(t^- - t').$$

Proof. Set $u_0 := u$, and consider an arbitrary time $t \in (t^-, t^+)$. Let $p = (u_0, \ldots, u_k) \in P_u^s(t)$ be any path such that $\Xi_u^s(t) = \xi_p^s(t)$ (that is, a path that maximizes the value of $\xi_p^s(t)$ at time t). By assumption, $\xi_p^s(t) > 0$. We will show that for the endpoint u_k, the first condition of **FC** is satisfied; specifically, the next node u_{k-1} on the path satisfies $L_{u_{k-1}}(t) - L_{u_k}(t) \geq s \cdot \kappa_{\{u_{k-1}, u_k\}}$. Note that since $\xi_p^s(t) = L_{u_0}(t) - L_{u_k}(t) - s \cdot \kappa_p > 0$, we cannot have $u_0 = u_k$, i.e., u_{k-1} must exist.

Consider the sub-path $(u_0, \ldots, u_{k-1})$. Because it is a sub-path of p, and $p \in P_u^s(t)$, we also have $(u_0, \ldots, u_{k-1}) \in P_u^s(t)$. Further, by choice of p we know that $\xi_p^s(t) = \Xi_u^s(t) \geq \xi^s_{(u_0, \ldots, u_{k-1})}(t)$; that is,

$$L_{u_0}(t) - L_{u_k}(t) - s \cdot \kappa_{(u_0, \ldots, u_k)} \geq L_{u_0}(t) - L_{u_{k-1}}(t) - s \cdot \kappa_{(u_0, \ldots, u_{k-1})},$$

which we can re-arrange to obtain

$$L_{u_{k-1}}(t) - L_{u_k}(t) \geq s \cdot \left(\kappa_{(u_0, \ldots, u_k)} - \kappa_{(u_0, \ldots, u_{k-1})}\right) = s \cdot \kappa_{\{u_{k-1}, u_k\}}.$$

This is the first condition of **FC** at node u. We cannot guarantee that the second condition holds, but if it does, that is, if at time t we also have

$$\forall v \in N_{u_k}^s(t): \quad L_{u_k}(t) - L_v(t) \leq s \cdot \kappa_{\{u_k, v\}} + 2\mu\tau_{\{u_k, v\}}, \tag{34}$$

then **FC** is satisfied at node u_k at time t, and u_k must be in fast mode. In that case we have $l_{u_k}(t) \geq (1-\rho)(1+\mu)$, which is the rate needed for the statement of the lemma. We proceed by considering the longest suffix of $[t^-, t^+)$ such that **FC** holds for all nodes that maximize Ξ^s_u during the interval. At the point where **FC** stops holding for some node u_k that maximizes Ξ^s_u, we can show that some other node is "to blame for this", and that node has an even larger skew to u.

Let $\theta \in [t^-, t^+]$ be the infimal time such that for all $t \in (\theta, t^+)$, condition (34) above holds for all paths p (where p is a path maximizing $\xi^s_p(t)$, as defined above), where $\theta := t^+$ if no such time exists.

By definition, $\Xi^s_u(t) = \max_{p \in P^s_u(t)} \{\xi^s_p(t)\}$. Each $\xi^s_p(t)$ is continuous and left-differentiable, since it is obtained by taking the difference of logical clocks, which are themselves continuous and left-differentiable. Therefore $\Xi^s_u(t)$ is also continuous and left-differentiable. By choice of θ, for all $t \in (\theta, t^+)$ we have

$$d/dt^- \xi^s_{p(t)}(t) \leq d/dt^- L_u(t) - (1-\rho)(1+\mu)(t^+ - \theta)$$

(where $p(t)$ is the path such that $\xi^s_{p(t)}(t) = \Xi^s_u(t)$), because the other endpoint of $p(t)$ is in fast mode and its logical clock increases at a rate of at least $(1-\rho)(1+\mu)$. Consequently also $d/dt^- \Xi^s_u(t) \leq d/dt^- L_u(t) - (1-\rho)(1+\mu)(t^+ - \theta)$. Using the mean-value theorem (which generalizes to the case where the function is only semi-differentiable), we see that over any interval $[\theta_1, \theta_2] \subseteq [\theta, t^+]$ where P^s_u does not change,

$$\Xi^s_u(\theta_2) - \Xi^s_u(\theta_1) \leq L_u(\theta_2) - L_u(\theta_1) - (1-\rho)(1+\mu)(\theta_2 - \theta_1).$$

Now consider points in time when P^s_u changes. If a path is removed from P^s_u at time t then the value of $\Xi^s_u(t)$ can only decrease. If a path q is added to P^s_u at time t, then Lemma 6.2 shows that $\xi^s_q(t) < 0$, (otherwise q must be in P^s_u throughout $[t^-, t]$). By the conditions of the current lemma we know that $\Xi^s_u(t) > 0$, so $\xi^s_q(t) < \Xi^s_u(t)$, and again $\Xi^s_u(t)$ is not increased by the addition of q. It follows that over the entire interval $[\theta, t^+]$,

$$\Xi^s_u(t^+) - \Xi^s_u(\theta) \leq L_u(t^+) - L_u(\theta) - (1-\rho)(1+\mu)(t^+ - \theta). \tag{35}$$

Therefore, if $\theta = t^-$, then we set $t' := t^-$ and we are done.

Suppose that $\theta > t^-$. Let $p = (u_0, \ldots, u_k)$ be some path such that $\Xi^s_u(\theta) = \xi^s_p(\theta)$ and Condition (34) does not hold for u_k (by choice of θ such a path exists); that is, there is some $v \in N^s_{u_k}(\theta)$ such that

$$L_{u_k}(\theta) - L_v(\theta) > s \cdot \kappa_{\{u_k,v\}} + 2\mu\tau_{\{u_k,v\}}. \tag{36}$$

Let $p' := p \circ (u_k, v) = (u_0, \ldots, u_k, v)$. We use Lemma 6.2 to "switch" from path p to path p' and go back in time to time $\theta - \tau_{\{u_k,v\}}$, *increasing the weighted skew* as we go back in time. We have

$$\begin{aligned}
\xi^s_{p'}(\theta - \tau_{\{u_k,v\}}) &= \xi^s_{p'}(\theta) - (L_u(\theta) - L_u(\theta - \tau_{\{u_k,v\}})) + L_v(\theta) - L_v(\theta - \tau_{\{u_k,v\}}) \\
&\geq \xi^s_p(\theta) - (L_u(\theta) - L_u(\theta - \tau_{\{u_k,v\}})) + (1-\rho)\tau_{\{u_k,v\}} \\
&\quad + (L_{u_k}(\theta) - L_v(\theta) - s \cdot \kappa_{\{u_k,v\}}) \\
&\overset{(36)}{>} \xi^s_p(\theta) - (L_u(\theta) - L_u(\theta - \tau_{\{u_k,v\}})) + (1-\rho+2\mu)\tau_{\{u_k,v\}} \\
&= \Xi^s_u(\theta) - (L_u(\theta) - L_u(\theta - \tau_{\{u_k,v\}})) + (1-\rho+2\mu)\tau_{\{u_k,v\}} \\
&\geq \Xi^s_u(\theta) - (1+\rho)(1+\mu)\tau_{\{u_k,v\}} + (1-\rho+2\mu)\tau_{\{u_k,v\}} \\
&> 0,
\end{aligned} \tag{37}$$

where in the last step we used the fact that $(1-\rho)\mu > 2\rho$. Hence, Lemma 6.2 shows that $p' \in P_u^s(\theta - \tau_{\{u_k,v\}})$, giving

$$\begin{aligned}\Xi_u^s(\theta) - \Xi_u^s(\theta - \tau_{\{u_k,v\}}) &\leq \Xi_u^s(\theta) - \xi_{p'}^s(\theta - \tau_{\{u_k,v\}}) \\ &\overset{(37)}{<} L_u(\theta) - L_u(\theta - \tau_{\{u_k,v\}}) - (1-\rho+2\mu)\tau_{\{u_k,v\}}.\end{aligned}$$

We conclude that

$$\Xi_u^s(t^+) - \Xi_u^s(\theta - \tau_{\{u_k,v\}}) \overset{(35)}{\leq} L_u(t^+) - L_u(\theta - \tau_{\{u_k,v\}}) - (1-\rho)(1+\mu)(t^+ - (\theta - \tau_{\{u_k,v\}})) - (1+\rho)\mu\tau_{\{u_k,v\}}. \tag{38}$$

From Lemma 6.2 we know that $\theta - \tau_{\{u,v\}} \geq t^- - \Theta_s$, that is, we did not go back too far in time. Thus, if $\theta - \tau_{\{u_k,v\}} \leq t^-$, the statement follows by setting $t' := \theta - \tau_{\{u_k,v\}}$: Recall that by definition, $\theta \geq t^-$, and hence $(1+\rho)\mu(t^- - (\theta - \tau_{\{u,v\}})) \leq (1+\rho)\mu\tau_{\{u,v\}}$; therefore (38) shows that

$$\begin{aligned}\Xi_u^s(t^+) - \Xi_u^s(t') &\leq L_u(t^+) - L_u(t') - (1-\rho)(1+\mu)(t^+ - t') - (1+\rho)\mu\tau_{\{u_k,v\}} \\ &\leq L_u(t^+) - L_u(t') - (1-\rho)(1+\mu)(t^+ - t') - (1+\rho)\mu(t^- - t').\end{aligned}$$

Otherwise, if $\theta - \tau_{\{u_k,v\}} > t^-$, we drop the term $-(1+\rho)\mu\tau_{\{u_k,v\}}$ from (38) to obtain

$$\Xi_u^s(t^+) - \Xi_u^s(\theta - \tau_{\{u_k,v\}}) \overset{(35)}{\leq} L_u(t^+) - L_u(\theta - \tau_{\{u_k,v\}}) - (1-\rho)(1+\mu)(t^+ - (\theta - \tau_{\{u_k,v\}})).$$

To prove the claim, it is sufficient to find a time $t' \in [t^- - \Theta_s, \theta - \tau_{\{u_k,v\}}]$ for which

$$\Xi_u^s(\theta - \tau_{\{u_k,v\}}) - \Xi_u^s(t') \leq L_u(\theta - \tau_{\{u_k,v\}}) - L_u(t') - (1-\rho)(1+\mu)((\theta - \tau_{\{u_k,v\}}) - t') - (1+\rho)\mu(t^- - t'). \tag{39}$$

In other words, we need to show the original statement of the lemma, but only for the sub-interval $[t^-, \theta - \tau_{\{u_k,v\}}] \subset [t^-, t^+]$. The lemma then follows by summing (38) and (39).

To this end, we continue inductively, applying the entire argument over again to the interval $[t^-, \theta - \tau_{\{u_k,v\}}]$. At each step we go back in time at least $\min_{x \neq y \in V} \tau_{\{x,y\}} > 0$, and we never go further back than $t^- - \Theta_s$; therefore the induction halts after a finite number of steps, at a time

$$t' \in \left[t^- - \Theta_s, t^-\right],$$

for which it holds that

$$\Xi_u(t^+) - \Xi_u(t') \leq L_u(t^+) - L_u(t') - (1-\rho)(1+\mu)(t^+ - t') - (1+\rho)\mu(t^- - t'),$$

as required. □

We will also need a technical helper lemma about Ξ_u^s that guarantees that Ξ_u^s remains positive under certain circumstances, enabling to apply Lemma 6.3.

Lemma 6.4. *Let $[t^-, t^+]$ be an interval such that node u satisfies the (C, s)-stabilization condition throughout $[t^-, t^+]$, where $s > 1$. If*

$$\Xi_u^s(t^+) \geq 2\rho(t^+ - t^-) \tag{40}$$

and

$$L_u(t^+) - L_u(t^-) \leq (1+\rho)(t^+ - t^-), \tag{41}$$

then for all $t \in (t^-, t^+]$ we have $\Xi_u^s(t) > 0$.

Proof. Let $p = (u, \dots, v)$ be a path such that $\xi_p^s(t^+) = \Xi_u^s(t^+) \geq 2\rho(t^+ - t^-)$ and let $t \in [t^-, t^+]$ Lemma 6.2 states that $p \in P_u^s(t)$ and hence $\Xi_u^s(t) \geq \xi_p^s(t)$. How much can ξ_p^s decrease when we go back from time t^+ to time t? We have

$$L_u(t^+) - L_u(t) = L_u(t^+) - L_u(t^-) - \big(L_u(t) - L_u(t^-)\big) \leq (1+\rho)(t^+ - t^-) - (1-\rho)(t - t^-).$$

Therefore,

$$\begin{aligned}
\xi_p^s(t) &= \xi_p^s(t^+) - \big(L_u(t^+) - L_u(t)\big) + \big(L_v(t^+) - L_v(t)\big) \\
&> 2\rho(t^+ - t^-) - (1+\rho)(t^+ - t^-) + (1-\rho)(t - t^-) + (1-\rho)(t^+ - t) \\
&= 2\rho(t^+ - t^-) - (1+\rho)(t^+ - t^-) + (1-\rho)(t^+ - t^-) \\
&= 2\rho(t^+ - t^-) - 2\rho(t^+ - t^-) = 0.
\end{aligned}$$

□

6.3 Decrease of Ψ and Stability

Next, we show a simple lemma that roughly states that if a node is in fast mode, then a condition slightly weaker than the slow mode condition **SC** is false. We already know that **SC** itself cannot hold when a node is in fast mode from Lemma 5.2. The weaker condition is almost the same as **SC**, except that the slack δ is removed.

Lemma 6.5. *Assume that for some node $u \in V$, a level $s \in \mathbb{N}$, and times $t^- < t^+$ we have $(L_u(t^-), L_u(t^+)] \cap \mathbb{T}_s = \emptyset$. If*

$$t^- < t_0 := \min\left\{t \in [t^-, t^+] \mid L_u(t^+) - L_u(t) \leq (1+\rho)(t^+ - t)\right\},$$

then

$$\exists w \in N_u^s(t_0): \; L_w(t_0) - L_u(t_0) > \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}} + \mu(1+\rho)\tau_{\{u,w\}} \tag{42}$$

or

$$\forall v \in N_u^s(t_0): \; L_u(t_0) - L_v(t_0) < \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}}. \tag{43}$$

Proof. Assuming the contrary, the logical negation of (42) ∨ (43) is

$$\begin{aligned}
&\forall v \in N_u^s(t_0): \; L_v(t_0) - L_u(t_0) \leq \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}} + \mu(1+\rho)\tau_{\{u,w\}} \\
\wedge \qquad &\exists w \in N_u^s(t_0): \; L_u(t_0) - L_w(t_0) \geq \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}}.
\end{aligned}$$

Because $(L_u(t^-), L_u(t^+)] \cap \mathbb{T}_s = \emptyset$, no new neighbors are added to N_u^s in the interval $(t^-, t^+]$. Neighbors can be removed during the interval, but there are only finitely many neighbors to remove ($N_u^s(t^-)$ is finite). Recall that a node is considered being in N_u^s at both the times of its insertion and removal. Hence there is some $\tilde{t} \in [t^-, t_0)$ such that $N_u^s(t) = N_u^s(t')$ for all $t, t' \in [\tilde{t}, t_0]$. Furthermore, since logical clocks are continuous and $\delta > 0$, there is a sub-interval $[t_0', t_0] \subseteq [\tilde{t}, t_0]$ such that $t_0' < t_0$ and for all $t \in [t_0', t_0]$ we have

$$\begin{aligned}
&\forall v \in N_u^s(t_0) = N_u^s(t): \; L_v(t) - L_u(t) \leq \left(s + \frac{1}{2}\right)\kappa_{\{u,v\}} + \mu(1+\rho)\tau_{\{u,w\}} + \delta \\
\wedge \qquad &\exists w \in N_u^s(t_0) = N_u^s(t): \; L_u(t) - L_w(t) \geq \left(s + \frac{1}{2}\right)\kappa_{\{u,w\}} - \delta.
\end{aligned}$$

Thus, **SC** applies at u for all $t \in [t'_0, t_0]$, and u must be in slow mode during this interval, so $L_u(t_0) - L_u(t'_0) \leq (1+\rho)(t_0 - t'_0)$. But we also know by choice of t_0 that $L_u(t^+) - L_u(t_0) \leq (1+\rho)(t^+ - t_0)$; therefore,

$$L_u(t^+) - L_u(t'_0) = L_u(t^+) - L_u(t_0) + L_u(t_0) - L_u(t'_0) \leq (1+\rho)(t^+ - t'_0),$$

contradicting the definition of t_0. □

We are now ready to prove our main theorem.

Proof of Theorem 5.18. Assume for the sake of contradiction that at some time $t_0 \in [t^- + \Lambda_s + 2\Theta_s, t^+]$ we have $\Psi^s_u(t_0) \geq 2\rho\Lambda_s$; that is, there is a path $(u = u_k, \ldots, u_0) \in P^s_u(t_0)$ such that

$$\psi^s_{(u_k,\ldots,u_0)}(t_0) = \Psi^s_u(t_0) \geq 2\rho\Lambda_s. \tag{44}$$

For the inverse path $(u_0, \ldots, u_k)$ we have

$$\xi^s_{(u_0,\ldots,u_k)}(t_0) = \psi^s_{(u_0,\ldots,u_k)}(t_0) + \frac{\kappa_{(u_0,\ldots,u_k)}}{2} \geq 2\rho\Lambda_s + \frac{\kappa_{(u_0,\ldots,u_k)}}{2} > 0. \tag{45}$$

Lemma 6.1 shows that

$$\kappa_{(u_0,\ldots,u_k)} \leq C_{s-1}. \tag{46}$$

By the conditions of the lemma, u satisfies the (C, s)-stabilization condition throughout $[t^-, t^+]$. Hence Lemma 6.2 shows that for all $t \in [t^-, t_0]$ we have $(u_0, \ldots, u_k) \in P^s_u(t)$. In particular, each sub-path $(u_0, \ldots, u_i)$ is also in P^s_u throughout $[t^-, t_0]$, and by the conditions of the lemma, this shows that each node u_i on the path satisfies the (C, s)-stabilization condition throughout $[t^-, t_0]$.

We construct a sequence of non-increasing times $t_0 = t'_0 \geq t_1 \geq t'_1 \geq \ldots t_\ell \geq t_{\ell'}$, where $t_\ell \geq t_0 - \Lambda_s$ and $t_{\ell'} \in [t^-, t_0 - \Lambda_s]$, and where each pair t_i, t'_i for $i < \ell$ is associated with a path p_i of non-zero length ending at u_k. The construction maintains the following properties for all $0 \leq i \leq \ell$:

(1) For all $t \in [t^-, t'_i]$ we have $p_i \in P^s_{u_k}(t)$.

(2) We have

$$\xi^s_{p_i}(t'_i) \geq 2\rho\Lambda_s + \frac{\kappa_{p_i}}{2} - (1+\rho)(t_0 - t'_i) + L_u(t_0) - L_u(t'_i). \tag{47}$$

(3) If $i < \ell$ then we have

$$\xi^s_{p_i}(t_{i+1}) \geq 2\rho\Lambda_s + \frac{\kappa_{p_i}}{2} - (1+\rho)(t_0 - t_{i+1}) + L_u(t_0) - L_u(t_{i+1}). \tag{48}$$

Constructing the sequence. We construct the sequence as follows: first, we show that we can find an initial path p_0 satisfying Properties (1) and (2). Then we show that if we have already constructed the sequence up to i, such that Properties (1) and (2) hold for i, then we can extend the construction by one step, choosing a time t_{i+1} such that (3) holds at i as well, and selecting a new path p_{i+1} and time t'_{i+1} for which Properties (1) and (2) hold (until we finally reach some time $t_{\ell'} \in [t^-, t_0 - \Lambda_s]$ and the construction halts).

For the base of the construction we set $p_0 := (u_0, \dots, u_k)$ and $t'_0 := t_0$. For this path we have already seen that $(u_0, \dots, u_k) \in P^s_{u_k}(t)$ for all $t \in [t^-, t_0]$, so Property (1) is satisfied. Also, at time t_0 we have $\xi^s_{p_0}(t_0) \geq 2\rho\Lambda_s + \kappa_{p_0}/2$ by choice of p_0, and since $t'_0 = t_0$, this shows that Property (2) holds.

Suppose that we have already constructed the sequence up to time t'_i such that Properties (1) and (2) hold. In particular, at time t'_i we have

$$\xi^s_{p_i}(t'_i) \geq 2\rho\Lambda_s + \frac{\kappa_{p_i}}{2} - (1+\rho)(t_0 - t'_i) + L_u(t_0) - L_u(t'_i). \tag{49}$$

Suppose further that $t'_i > t_0 - \Lambda_s$ (otherwise the construction halts at i). Let $p_i = (w = w_0, \dots, w_m = u_k)$ be the path associated with the i-th step. We define

$$t_{i+1} := \min\left\{t \in [t_0 - \Lambda_s, t'_i] \mid L_w(t'_i) - L_w(t) \leq (1+\rho)(t'_i - t)\right\}. \tag{50}$$

The minimum is taken over a non-empty set because $L_w(t'_i) - L_w(t'_i) \leq (1+\rho)(t'_i - t'_i)$.

From (49), we get that

$$\begin{aligned}
\xi^s_{p_i}(t_{i+1}) &= \xi^s_{p_i}(t'_i) - \big(L_w(t'_i) - L_w(t_{i+1})\big) + \big(L_{u_k}(t'_i) - L_{u_k}(t_{i+1})\big) \\
&\overset{(49)}{\geq} 2\rho\Lambda_s + \frac{\kappa_{p_i}}{2} - (1+\rho)(t_0 - t'_i) + L_u(t_0) - L_u(t'_i) \qquad (51)\\
&\quad -(1+\rho)(t'_i - t_{i+1}) + L_{u_k}(t'_i) - L_{u_k}(t_{i+1}) \\
&= 2\rho\Lambda_s + \frac{\kappa_{p_i}}{2} - (1+\rho)(t_0 - t_{i+1}) + L_u(t_0) - L_u(t_{i+1}). \qquad (52)
\end{aligned}$$

This shows that Property (3) holds at i.

If $t_{i+1} = t_0 - \Lambda_s$, then we define $t'_{i+1} := t_{i+1}$ and $p_{i+1} := p_i$, and we are done. Thus, suppose that $t_{i+1} > t_0 - \Lambda_s$. In this case (52) shows that $\xi^s_{p_i}(t_{i+1}) > 0$. Consequently, since u_k satisfies the level s stabilization condition throughout $[t^-, t_0]$, we can use Lemma 6.1 to show that $\kappa_{p_i} \leq C_{s-1}$.

For the $(i+1)$-th step, we choose path p_{i+1} using Lemma 6.5, but first we must establish the conditions of the lemma. From I.H. (1), for all $t \in [t^-, t_{i+1}] \subseteq [t^-, t'_i]$ we have $p_i \in P^s_u(t)$. Hence, by the conditions of the lemma, node w satisfies the level s stabilization condition throughout $[t^-, t_{i+1}]$. In particular, this implies that

$$\mathbb{T}_s \cap [L_w(t^-), L_w(t_{i+1})] = \emptyset,$$

because the level s stabilization condition implies that node w's logical clock does not cross any update point T^e_s for any edge e at any point throughout the interval $[t^-, t_{i+1}]$.

This allows us to apply Lemma 6.5, which shows that

$$\exists w' \in N^s_w(t_{i+1}) \,:\, L_{w'}(t_{i+1}) - L_w(t_{i+1}) > \left(s + \frac{1}{2}\right)\kappa_{\{w,w'\}} + \mu(1+\rho)\tau_{\{w,w'\}}, \tag{53}$$

or

$$\forall v \in N^s_w(t_{i+1}) \,:\, L_w(t_{i+1}) - L_v(t_{i+1}) < \left(s + \frac{1}{2}\right)\kappa_{\{v,w\}}. \tag{54}$$

We consider each case separately.

First, suppose that (53) holds for some node w' and define $p_{i+1} := (w', w) \circ p_i = (w', w = w_0, \dots, w_m = u_k)$ and $t'_{i+1} = t_{i+1} - \tau_{\{w,w'\}}$. We call this a *forward step*, as the distance to u_k increases.

For the new path p_{i+1} we have

$$\begin{aligned}
\xi^s_{p_{i+1}}(t_{i+1}) &= \xi^s_{p_i}(t_{i+1}) + L_{w'}(t_{i+1}) - L_w(t_{i+1}) - s\cdot\kappa_{\{w,w'\}}\\
&\overset{(53)}{>} \xi^s_{p_i}(t_{i+1}) + \frac{\kappa_{\{w,w'\}}}{2} + \mu(1+\rho)\tau_{\{w,w'\}}\\
&\overset{(52)}{\geq} 2\rho\Lambda_s + \frac{\kappa_{p_{i+1}}}{2} - (1+\rho)(t_0 - t_{i+1}) + L_u(t_0) - L_u(t_{i+1}) + \mu(1+\rho)\tau_{\{w,w'\}}\\
&> 2\rho\Lambda_s - (1+\rho)(t_0 - t_{i+1}) + (1-\rho)(t_0 - t_{i+1})\\
&> 0.
\end{aligned} \tag{55}$$

By assumption we have $t_{i+1} > t_0 - \Lambda_s$, and hence $t_{i+1} > t^- + \Theta_s$. Because u_k satisfies the (C, s)-stabilization condition throughout $[t^-, t_{i+1}]$, we can apply Lemma 6.2 to the interval $[t^- + \Theta_s, t_{i+1}]$ to show that for all $t \in [t^-, t'_{i+1}]$ we have $p_{i+1} \in P^s_{u_k}(t)$, so Property (1) holds. The lemma also shows that $t'_{i+1} \geq t^-$.

Going back to time t'_{i+1}, we have

$$\begin{aligned}
\xi^s_{p_{i+1}}(t'_{i+1}) &= \xi^s_{p_{i+1}}(t_{i+1}) - \big(L_{w'}(t_{i+1}) - L_{w'}(t'_{i+1})\big) + \big(L_{u_k}(t_{i+1}) - L_{u_k}(t'_{i+1})\big)\\
&\geq \xi^s_{p_{i+1}}(t_{i+1}) - (1+\mu)(1+\rho)(t_{i+1} - t'_{i+1}) + L_{u_k}(t_{i+1}) - L_{u_k}(t'_{i+1})\\
&\overset{(55)}{>} 2\rho\Lambda_s + \frac{\kappa_{p_{i+1}}}{2} - (1+\rho)(t_0 - t_{i+1}) + L_u(t_0) - L_u(t_{i+1}) + \mu(1+\rho)\tau_{\{w,w'\}}\\
&\quad - (1+\mu)(1+\rho)\tau_{\{w,w'\}} + L_{u_k}(t_{i+1}) - L_{u_k}(t'_{i+1})\\
&= 2\rho\Lambda_s + \frac{\kappa_{p_{i+1}}}{2} - (1+\rho)(t_0 - t_{i+1} + \tau_{\{w,w'\}}) + L_u(t_0) - L_u(t'_{i+1})\\
&= 2\rho\Lambda_s + \frac{\kappa_{p_{i+1}}}{2} - (1+\rho)(t_0 - t'_{i+1}) + L_u(t_0) - L_u(t'_{i+1}).
\end{aligned}$$

This shows that Property (2) holds for $i+1$.

Now let us turn to the other case, in which (54) holds. From I.H. (1) we know that $p_i \in P^s_{u_k}(t_{i+1})$, and in particular, for the next node w_1 on the path $(w = w_0, w_1, \ldots, w_m = u_k)$, we have $w_1 \in N^s_w(t_{i+1})$. Thus, (54) shows that

$$L_w(t_{i+1}) - L_{w_1}(t_{i+1}) < \left(s + \frac{1}{2}\right)\kappa_{\{w,w_1\}}. \tag{56}$$

In this case, we define $p_{i+1} := (w_1, \ldots, w_m = u_k)$, that is, we remove w from the head of the path, and $t'_{i+1} := t_{i+1}$. We call this a *backward step*. Property (1) for $i+1$ follows immediately from I.H. (1) (for i). As for Property (2), we have

$$\begin{aligned}
\xi^s_{p_{i+1}}(t_{i+1}) &= \xi^s_{p_i}(t_{i+1}) - L_w(t_{i+1}) + L_{w_1}(t_{i+1}) + s\cdot\kappa_{\{w,w_1\}}\\
&\overset{(56)}{>} \xi^s_{p_i}(t_{i+1}) - \frac{\kappa_{\{w,w_1\}}}{2}\\
&\overset{(52)}{\geq} 2\rho\Lambda_s + \frac{\kappa_{p_i}}{2} - \frac{\kappa_{\{w,w_1\}}}{2} - (1+\rho)(t_0 - t_{i+1}) + L_u(t_0) - L_u(t_{i+1})\\
&= 2\rho\Lambda_s + \frac{\kappa_{p_{i+1}}}{2} - (1+\rho)(t_0 - t_{i+1}) + L_u(t_0) - L_u(t_{i+1}).
\end{aligned}$$

Since $t'_{i+1} = t_{i+1}$, Property (2) is satisfied for $i+1$. Note in particular that we have $\xi^s_{p_{i+1}}(t_{i+1}) > 0$, and hence $p_{i+1} \neq (u_k)$, because $\xi^s_{(u_k)}(t) = 0$ for all times t. This concludes the induction.

We note that the sequence we constructed is finite, that is, there is some $\ell \in \mathbb{N}$ such that $t'_\ell \in [t^-, t_0 - \Delta]$. This is because every time we make a forward step we have $t'_{i+1} \leq t_{i+1} - \min_{e \in \binom{V}{2}} \tau_e$, so after finitely many such steps we reach time $t_0 - \Delta$; as for backward steps, each such step shortens the path, so only finitely many backward steps can occur between two forward steps.

Properties of the chain construction. Let $v_0, \ldots, v_{\ell-1}$ denote the first node on each path $p_0, \ldots, p_{\ell-1}$ in the sequence above. Before proceeding, we establish the following additional properties of the chain $v_0, \ldots, v_{\ell-1}$.

(4) For all $i = 0, \ldots, \ell - 1$ we have $\kappa_{p_i} \leq C_{s-1}$.

(5) For all $i = 0, \ldots, \ell - 1$, node v_i satisfies the level s stabilization condition throughout $[t^-, t'_i]$.

(6) For all $i = 0, \ldots, \ell - 1$ we have $\Xi_{v_i}(t'_i) \geq 2\rho(t'_i - (t_0 - \Lambda_s))$ and for all $t \in (t_{i+1}, t'_i]$ we have $\Xi^s_{v_i}(t) > 0$.

(7) There is an index $m \in \{0, \ldots, \ell - 1\}$ such that in the construction all steps prior to index m are backward steps and all steps starting from index m are forward steps.

(8) For all $m \leq i < j \leq \ell$, we have

$$L_{v_i}(t_{i+1}) - L_{v_j}(t'_j) \leq (1 + \rho)(t_{i+1} - t'_j) - \left(s + \frac{1}{2}\right) \kappa_{(v_i, \ldots, v_j)}. \tag{57}$$

The proof of these properties follows.

Fix $i \leq \ell - 1$. We know that node u satisfies the level s stabilization condition throughout $[t^-, t^+]$. From Property (1) for index i we have $(v_i, \ldots, u_k) = p_i \in P^s_{u_k}(t)$ for all $t \in [t^-, t_i]$. Also, Property (2) states that

$$\begin{aligned} \xi^s_{p_i}(t'_i) &\geq 2\rho\Lambda_s - (1 + \rho)(t_0 - t'_i) + L_{u_k}(t_0) - L_{u_k}(t'_i) \\ &\geq 2\rho(t'_i - (t_0 - \Lambda_s)) \geq 0 \end{aligned}$$

Consequently, Lemma 6.1 shows that $\kappa_{p_i} \leq C_{s-1}$, so Property (4) holds, and Property (5) follows from the conditions of the current lemma.

Because $p_i \in P^s_{u_k}(t'_i)$ we have $\Xi^s_{v_i}(t'_i) \geq \xi^s_{p_i}(t'_i)$, and hence $\Xi^s_{v_i}(t'_i) \geq 2\rho(t'_i - (t_0 - \Lambda_s))$. Also, because $t'_i - (t_0 - \Lambda_s) \geq t'_i - t_{i+1}$, we can apply Lemma 6.4 to show that $\Xi^s_{v_i}(t) > 0$ throughout $(t_{i+1}, t'_i]$. This shows that Property (6) holds.

To show Property (7), we show that if a forward step occurs at index $i < \ell - 1$ of the construction, then at index $i + 1$ we also take a forward step. The property follows.

Suppose that this is not the case, that is, at some index $i < \ell - 1$ we have

$$L_{v_i}(t_i) - L_{v_{i-1}}(t_i) > \left(s + \frac{1}{2}\right) \kappa_{\{v_{i-1}, v_i\}} + \mu(1 + \rho)\tau_{\{v_{i-1}, v_i\}}$$

and at index $i + 1$ we have $v_{i+1} = v_{i-1}$ and

$$L_{v_i}(t_{i+1}) - L_{v_{i-1}}(t_{i+1}) < \left(s + \frac{1}{2}\right) \kappa_{\{v_{i-1}, v_i\}}.$$

We show that this implies that node v_{i-1}'s average rate over the interval $[t_{i+1}, t_i]$ was no greater than $1+\rho$, contradicting the choice of t_i as the *minimal* time such that v_{i-1}'s average rate over $[t_i, t'_{i-1}]$ did not exceed $1+\rho$.

Summing the two inequalities above yields

$$L_{v_{i-1}}(t_i) - L_{v_{i-1}}(t_{i+1}) < L_{v_i}(t_i) - L_{v_i}(t_{i+1}) - \mu(1+\rho)\tau_{\{v_{i-1},v_i\}}. \tag{58}$$

By definition of t_{i+1} we have $L_{v_i}(t'_i) - L_{v_i}(t_{i+1}) \le (1+\rho)(t'_i - t_{i+1})$, and since at index i we took a forward step, we defined $t'_i = t_i - \tau_{\{v_{i-1},v_i\}}$. Hence

$$\begin{aligned} L_{v_i}(t_i) - L_{v_i}(t_{i+1}) &= L_{v_i}(t_i) - L_{v_i}(t'_i) + L_{v_i}(t'_i) - L_{v_i}(t_{i+1}) \\ &\le (1+\rho)(1+\mu)\tau_{\{v_{i-1},v_i\}} + (1+\rho)(t_i - \tau_{\{v_{i-1},v_i\}} - t_{i+1}) \\ &= (1+\rho)(t_i - t_{i+1}) + \mu(1+\rho)\tau_{\{v_{i-1},v_i\}}. \end{aligned}$$

Combining with (58) yields

$$\begin{aligned} L_{v_{i-1}}(t_i) - L_{v_{i-1}}(t_{i+1}) &< (1+\rho)(t_i - t_{i+1}) + \mu(1+\rho)\tau_{\{v_{i-1},v_i\}} - \mu(1+\rho)\tau_{\{v_{i-1},v_i\}} \\ &\le (1+\rho)(t_i - t_{i+1}). \end{aligned}$$

This is a contradiction.

This shows that after the first forward step in the construction (if one occurs), no backward steps can occur. Thus, there is some index $m \in \{0, \ldots, \ell-1\}$ such that for all $i = m, \ldots, \ell-2$, node v_{i+1} is obtained from node v_i by a forward step at time t_{i+1}. (If no forward steps occur in the construction then we set $m := \ell - 1$.)

Finally we show Property (8). Fix i, j such that $m \le i < j \le \ell - 1$. All steps between index m and index $\ell - 1$ are forward steps, and hence for each $k = i, \ldots, j-1$ we have

$$L_{v_{k+1}}(t_{k+1}) - L_{v_k}(t_{k+1}) > \left(s + \frac{1}{2}\right)\kappa_{\{v_k,v_{k+1}\}} + \mu(1+\rho)\tau_{\{v_k,v_{k+1}\}}.$$

Also, because $t'_{k+1} = t_{k+1} - \tau_{\{v_k,v_{k+1}\}}$ we have

$$L_{v_{k+1}}(t_{k+1}) - L_{v_{k+1}}(t'_{k+1}) \le (1+\rho)(1+\mu)\tau_{\{v_k,v_{k+1}\}},$$

and by definition of t_{k+2},

$$L_{v_{k+1}}(t'_{k+1}) - L_{v_{k+1}}(t_{k+2}) \le (1+\rho)(t'_{k+1} - t_{k+2}).$$

Summing the three inequalities above yields, for each $k = i, \ldots, j-2$,

$$L_{v_k}(t_{k+1}) - L_{v_{k+1}}(t_{k+2}) < (1+\rho)(t_{k+1} - t_{k+2}) - \left(s + \frac{1}{2}\right)\kappa_{\{v_k,v_{k+1}\}}.$$

Summing over $k = i, \ldots, j-2$, we obtain

$$L_{v_i}(t_{i+1}) - L_{v_{j-1}}(t_j) \le (1+\rho)(t_i - t_j) - \left(s + \frac{1}{2}\right)\kappa_{(v_i,\ldots,v_{j-1})}.$$

For the final step, from $j-1$ at time t_j to j at time t'_j, we use only the first two inequalities, which show that

$$L_{v_{j-1}}(t_j) - L_{v_j}(t'_j) \le (1+\rho)\tau_{\{v_{j-1},v_j\}} - \left(s + \frac{1}{2}\right)\kappa_{\{v_{j-1},v_j\}}.$$

Since $t_j - t'_j = \tau_{\{v_{j-1},v_j\}}$, we combine the two inequalities above to obtain (57), as desired.

Bounding Ξ^s. The chain construction provides us with a sequence of sub-intervals $\{[t_0, t'_m]\} \cup \{[t_{i+1}, t'_i] \mid i = m, \ldots, \ell-2\} \cup \left\{[t_0 - \Lambda_s, t'_{\ell-1}]\right\}$, each associated with a node v_i that has a non-negative Ξ^s-value and a small average clock rate (at most $1+\rho$) over the entire sub-interval. Roughly speaking, over each sub-interval, Lemma 6.3 shows that Ξ^s decreases at an average rate of at least $(1-\rho)(1+\mu) - (1+\rho) = (1-\rho)\mu - 2\rho$; since at the end of the whole interval (time t_0) we had $\Xi^s_{v_0}(t_0) \geq 2\rho C_{s-1}/((1-\rho)\mu)$, at the beginning (time $t_0 - \Lambda_s = t_0 - C_{s-1}/((1-\rho)\mu)$) we will be able to show that for some node we have $\Xi^s(t_0 - \Lambda_s) \geq \Xi^s_{v_0}(t_0) + ((1-\rho)\mu - 2\rho) \cdot \Lambda_s > C_{s-1}$, which will result in a contradiction to $(s-1)$-legality at time $t_0 - \Lambda_s$. Thus, our strategy now is to apply Lemma 6.3 successively to the sub-intervals we obtained in the construction as we went back in time from t_0 to $t_0 - \Delta t$.

However, when we apply Lemma 6.3 to a sub-interval $[t_{i+1}, t'_i]$ there is some "overshoot": the lemma yields a time $t' \in [t_{i+1} - \Theta_s, t_{i+1}]$ such that over the interval $[t', t'_i]$, Ξ^s decreases quickly, but t' falls in some other sub-interval $[t_{j+1}, t_j]$. Before we can apply Lemma 6.3 again, we must deal with two concerns:

- The new interval $[t_{j+1}, t_j]$ is not necessarily contiguous to the interval $[t_{i+1}, t'_i]$ to which we applied Lemma 6.3; that is, we can have $j > i+1$. Thus we define a sequence of indices $i_0 \leq \ldots \leq i_h$ representing the indices of the sub-intervals in which we find ourselves after each application of Lemma 6.3.
- We have $t' \in [t_{j+1}, t_j]$, but we do not know whether $t' \in [t_{j+1}, t'_j]$ or $t' \in (t'_j, t_j]$. Our construction in the previous part only ensures that v_j has an average rate of at most $1+\rho$ over the interval $[t_{j+1}, t'_j]$; hence, if we have $t' \in (t'_j, t_j]$, we must first "go back in time" to t'_j before we can usefully apply Lemma 6.3 to v_j.

In other words, in the current part of the proof, we successively apply two kinds of steps: the first is an application of Lemma 6.3 to obtain a time t' for which we have a large average Ξ^s-increase rate; the second is a step back in time to the nearest preceding time $t'_j \leq t'$, in preparation for the next application of Lemma 6.3.

Accordingly, we define two sequences of times, $\theta_0 \geq \ldots \geq \theta_h$ and $\varphi_0 \geq \ldots \geq \varphi_h$, such that

$$t_{m+1} \geq \varphi_0 \geq \theta_1 \geq \varphi_1 \geq \ldots \geq \theta_h \geq t_0 - \Lambda_s \geq \varphi_h \geq t^-.$$

The first sequence $\varphi_0, \ldots, \varphi_h$ represents the times obtained by successively applying Lemma 6.3; the second sequence $\theta_1, \ldots, \theta_h$ represents the second step back in time. Each θ_j is chosen such that for some $k \leq \ell - 1$ we have $\theta_j \in [t_{k+1}, t'_k]$. Note that since at each index $j \geq m$ we take a forward step, we always have $t_{j+1} \leq t'_j < t_j$ (this is a property of the construction), and hence the index k is unique. We define a sequence of indices $i_0 \leq \ldots \leq i_h$ as follows:

$$i_j := \begin{cases} m & \text{if } j = 0, \\ \text{the unique index k such that } \theta_j \in [t_{k+1}, t'_k] & \text{if } j > 0. \end{cases} \tag{59}$$

Finally, each φ_j is chosen such that $\varphi_j \in [t_{i_j+1} - \Theta_s, t_{i_j+1}]$ (as φ_j is obtained by applying Lemma 6.3 to the interval $[t_{i_j+1}, \theta_j]$).

We maintain the following properties:

(i) For all $j = 0, \ldots, h$,

$$\Xi^s_{v_{i_j}}(\varphi_j) \geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{i_j+1}) + (1-\rho+2\mu)(t_{i_j+1} - \varphi_j) - \Big(L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_j}}(\varphi_j)\Big). \tag{60}$$

(ii) For all $j = 1, \dots, h$,

$$\Xi^s_{v_{i_j}}(\theta_j) \geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_j}) + (1-\rho)(1+\mu)(t'_{i_j} - \theta_j) - \Big(L_{v_{i_j}}(t'_{i_j}) - L_{v_{i_j}}(\theta_j)\Big). \quad (61)$$

Intuitively, at each point φ_j we claim an average increase rate of $((1-\rho)\mu - 2\rho)$ for Ξ^s as we go back in time over the interval $[\varphi_j, t_0]$: this follows from (60), because $L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_j}}(\varphi_j) \leq (1+\rho)(1+\mu)(t_{i_j+1} - \varphi_j)$. However, the statement of Property (i) is more precise, and keeps track of the exact clock increase of v_{i_j}. We need this additional information because the chain construction relates the clock values of the different nodes at points where we switch from one to the other (Property (8)), not at arbitrary times such as φ_j. Thus, we keep track of v_{i_j}'s increase from time φ_j to time t_{i_j+1}, at which the switch occurs. This leads us to show that because $\Xi^s_{v_{i_j}}(\varphi_j)$ is large, so is $\Xi^s_{v_{i_{j+1}}}(\varphi_j)$, and allows the induction to go through.

In Property (ii) we also keep track of v_{i_j}'s exact clock increase, but here the reason is different: we simply do not have a good enough bound on v_{i_j}'s clock increase over the interval $[\theta_j, t'_{i_j}]$. The choice of t_{i_j+1} yields an average rate of at most $1+\rho$ over $[t_{i_j+1}, t'_{i_j}]$, but tells us nothing about arbitrary points in the interval, such as θ_j. Therefore, we keep track of v_{i_j}'s clock increase over $[\theta_j, t'_{i_j}]$, and also of the decrease $(1-\rho)(1+\mu)(t_{i'_j} - \theta_j)$ provided by Lemma 6.3, which represents the fact that the other endpoints of paths maximizing Ξ^s are acting to catch up over $[\theta_j, t'_{i_j}]$. Later, when we go back in time to $\varphi_{j+1} \leq t_{i_j+1}$, we "complete" the sub-interval, and use the fact that v_{i_j}'s average rate over $[t_{i_j+1}, t'_{i_j}]$ is at most $1+\rho$ to obtain the average decrease rate of $((1-\rho)\mu - 2\rho)$ over all of $[\varphi_j, t_0]$.

The definition of the two sequences is mutually-recursive. We begin by showing how φ_j is chosen, assuming that if $j > 0$ then we have already chosen θ_j such that Property (ii) holds.

First, consider the base case, $j = 0$: we must find a time $\varphi_0 \in [t_{m+1} - \Theta_s, t_{m+1}]$ satisfying Property (i). This requires a bit more effort than the step, because we are claiming an average decrease rate of $((1-\rho)\mu - 2\rho)$ for Ξ^s over an interval $[\varphi_0, t_0] \supseteq [t_{m+1}, t_0]$; but since we skipped the prefix $v_0, \dots, v_{m-1}$ of the chain construction, the properties of the chain tell us nothing about v_m during the sub-interval $[t_m, t_0]$. Nevertheless, we show that we can apply Lemma 6.3 to the entire interval $[t_{m+1}, t_0]$.

Because only backward steps occur prior to index m, the path p_m is a sub-path of p_0; we know that $p_0 \in P^s_{u_k}(t)$ for all $t \in [t^-, t_0]$, and hence $p_m \in P^s_{u_k}(t)$ for all $t \in [t^-, t_0]$ as well. Also, from Property (3) of the construction,

$$\xi^s_{p_m}(t_{m+1}) \geq 2\rho\Lambda_s - (1+\rho)(t_0 - t_{m+1}) + L_{u_k}(t_0) - L_{u_k}(t_{m+1}), \quad (62)$$

so going forward to any time $t \in [t_{m+1}, t_0]$ we have

$$\begin{aligned}
\Xi^s_{v_m}(t) &\geq \xi^s_{p_m}(t) = \xi^s_{p_m}(t_{m+1}) + (L_{v_m}(t) - L_{v_m}(t_{m+1})) - (L_{u_k}(t) - L_{u_k}(t_{m+1})) \\
&\geq 2\rho\Lambda_s - (1+\rho)(t_0 - t_{m+1}) + (L_{v_m}(t) - L_{v_m}(t_{m+1})) + (L_{u_k}(t_0) - L_{u_k}(t)) \\
&\geq 2\rho\Lambda_s - (1+\rho)(t_0 - t_{m+1}) + (1-\rho)(t - t_{m+1}) + (1-\rho)(t_0 - t) \\
&\geq 2\rho(\Lambda_s - (t_0 - t_{m+1})),
\end{aligned}$$

which is strictly greater than 0 for $t \neq t_0$. This is sufficient to apply Lemma 6.3 to the interval $[t_{m+1}, t_0]$, yielding a time $\varphi_0 \in [t_{m+1} - \Theta_s, t_{m+1}]$ such that

$$\begin{aligned}
\Xi^s_{v_m}(\varphi_0) &\geq \Xi^s_{v_m}(t_0) + (1-\rho)(1+\mu)(t_0 - \varphi_0) - (L_{v_m}(t_0) - L_{v_m}(\varphi_0)) + \mu(1+\rho)(t_{m+1} - \varphi_0) \\
&\geq \xi^s_{p_m}(t_0) + (1-\rho)(1+\mu)(t_0 - \varphi_0) - (L_{v_m}(t_0) - L_{v_m}(\varphi_0)) + \mu(1+\rho)(t_{m+1} - \varphi_0) \\
&\geq \xi^s_{p_m}(t_{m+1}) + (L_{v_m}(t_0) - L_{v_m}(t_{m+1})) - (L_{u_k}(t_0) - L_{u_k}(t_{m+1})) \\
&\quad + (1-\rho)(1+\mu)(t_0 - \varphi_0) - (L_{v_m}(t_0) - L_{v_m}(\varphi_0)) + \mu(1+\rho)(t_{m+1} - \varphi_0) \\
&\overset{(62)}{\geq} 2\rho\Lambda_s - (1+\rho)(t_0 - t_{m+1}) + (1-\rho)(1+\mu)(t_0 - \varphi_0) - (L_{v_m}(t_{m+1}) - L_{v_m}(\varphi_0)) \\
&\quad + \mu(1+\rho)(t_{m+1} - \varphi_0) \\
&= 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{m+1}) + (1-\rho+2\mu)(t_{m+1} - \varphi_0) \\
&\quad - (L_{v_m}(t_{m+1}) - L_{v_m}(\varphi_0)) .
\end{aligned}$$

This shows that Property (i) holds for this choice of φ_0.

For the step, suppose that at $j \geq 1$ we have already chosen a time $\theta_j \geq t_0 - \Lambda_s$ such that

$$\Xi^s_{v_{i_j}}(\theta_j) \geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_j}) + (1-\rho)(1+\mu)(t'_{i_j} - \theta_j) - \left(L_{v_{i_j}}(t'_{i_j}) - L_{v_{i_j}}(\theta_j)\right).$$

Since $\theta_j \leq t'_{i_j}$ by choice of i_j, Properties (5) and (6) show that we have $\Xi^s_{v_{i_j}}(t) > 0$ for all $t \in (t_{i_j+1}, \theta_j]$ and that v_{i_j} satisfies the (C, s)-stabilization condition during $[t_{i_j+1}, \theta_j]$. Thus, we can apply Lemma 6.3 to obtain a time $\varphi_j \in [t_{i_j+1} - \Theta_s, t_{i_j+1}]$ such that

$$\Xi^s_{v_{i_j}}(\varphi_j) \geq \Xi^s_{v_{i_j}}(\theta_j) + (1-\rho)(1+\mu)(\theta_j - \varphi_j) - \left(L_{v_{i_j}}(\theta_j) - L_{v_{i_j}}(\varphi_j)\right) + \mu(1+\rho)(t_{i_j+1} - \varphi_j).$$

Together with the induction hypothesis, we obtain

$$\begin{aligned}
\Xi^s_{v_{i_j}}(\varphi_j) &\geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_j}) + (1-\rho)(1+\mu)(t'_{i_j} - \theta_j) + (1-\rho)(1+\mu)(\theta_j - \varphi_j) \\
&\quad - \left(L_{v_{i_j}}(t'_{i_j}) - L_{v_{i_j}}(\varphi_j)\right) + \mu(1+\rho)(t_{i_j+1} - \varphi_j) \\
&= 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_j}) + (1-\rho)(1+\mu)(t'_{i_j} - t_{i_j+1} + t_{i_j+1} - \varphi_j) \\
&\quad - \left(L_{v_{i_j}}(t'_{i_j}) - L_{v_{i_j}}(t_{i_j+1})\right) - \left(L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_j}}(\varphi_j)\right) + \mu(1+\rho)(t_{i_j+1} - \varphi_j).
\end{aligned}$$

Recall that by definition of t_{i_j+1} we have $L_{v_{i_j}}(t'_{i_j}) - L_{v_{i_j}}(t_{i_j+1}) \leq (1+\rho)(t'_{i_j} - t_{i_j+1})$. Thus, we have

$$\Xi^s_{v_{i_j}}(\varphi_j) \geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{i_j+1}) + (1-\rho+2\mu)(t_{i_j+1} - \varphi_j) - \left(L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_j}}(\varphi_j)\right).$$

Note that since $\varphi_j \geq t_{i_j+1} - \Theta_s \geq t_0 - \Lambda_s - \Theta_s$, we have $\varphi_j \geq t^-$, as desired. This completes the induction step for the sequence φ_j.

Next we show how θ_{j+1} is chosen, assuming that we have already chosen a time φ_j satisfying Property (i). Assume also that $\varphi_j > t_0 - \Lambda_s$, as otherwise the construction halts. Recall that we must choose θ_{j+1} such that $\theta_{j+1} \in [t_{k+1}, t'_k]$ for some k. Thus, if $\varphi_j \in [t_{k+1}, t'_k]$ for some k, then we set $\theta_j := \varphi_j$; otherwise it must be that $\varphi_j \in (t'_k, t_k]$ for some unique k, and in this case we define $\theta_{j+1} := t'_k$.

The choice of θ_{j+1} induces an index i_{j+1} (which is the minimal index k from the definition of θ_{j+1}). The induction hypothesis (Property (i)) states that $\Xi^s_{v_{i_j}}(\varphi_j)$ is large, that is, there is some path $q \in P^s_{v_{i_j}}(\varphi_j)$ such that $\xi^s_q(\varphi_j)$ is large. To show Property (ii) at j, we first extend q into a path q' starting at $v_{i_{j+1}}$, and show that because $\xi^s_q(\varphi_j)$ is large, so is $\xi^s_{q'}(\varphi_j)$. Then we go back in time and show that $\xi^s_{q'}(\theta_{j+1})$ is large (that is, not much skew is lost as we step back from φ_j to θ_{j+1}), and finally we show that $q' \in P^s_{v_{i_{j+1}}}(\theta_{j+1})$, which implies that $\Xi^s_{v_{i_{j+1}}}(\theta_{j+1}) \geq \xi^s_{q'}(\theta_{j+1})$.

Formally, let $q = (v_{i_j}, \ldots, x) \in P^s_{v_{i_j}}(t)$ be a path such that $\xi^s_q(t) = \Xi^s_{v_{i_j}}(t)$. For the extended path $q' := (v_{i_{j+1}}, \ldots, v_{i_j}) \circ q$, we have

$$\begin{aligned}
\xi^s_{q'}(\varphi_j) &= \xi^s_q(\varphi_j) - L_{v_{i_j}}(\varphi_j) + L_{v_{i_{j+1}}}(\varphi_j) - s \cdot \kappa_{(v_{i_j},\ldots,v_{i_{j+1}})} \\
&= \Xi^s_{v_{i_j}}(\varphi_j) - L_{v_{i_j}}(\varphi_j) + L_{v_{i_{j+1}}}(\varphi_j) - s \cdot \kappa_{(v_{i_j},\ldots,v_{i_{j+1}})} \\
&\overset{(60)}{\geq} 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{i_j+1}) + (1-\rho)(1+\mu)(t_{i_j+1} - \varphi_j) \\
&\quad - \Big(L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_{j+1}}}(\varphi_j)\Big) - s \cdot \kappa_{(v_{i_j},\ldots,v_{i_{j+1}})}.
\end{aligned}$$

(Note that we omit the term $\mu(1+\rho)(t_{i_j+1} - \varphi_j)$ in (60), which is non-negative because $\varphi_j \leq t_{i_j+1}$.)

Next we deal with the gap $\Big(L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_{j+1}}}(t)\Big)$ using Property (8), which shows that

$$L_{v_{i_j}}(t_{i_j+1}) - L_{v_{i_{j+1}}}(t'_{i_{j+1}}) \leq (1+\rho)(t_{i_j+1} - t'_{i_{j+1}}) - \left(s + \frac{1}{2}\right)\kappa_{(v_{i_j},\ldots,v_{i_{j+1}})};$$

thus we have

$$\begin{aligned}
\xi^s_{q'}(\varphi_j) &\geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_{j+1}}) + (1-\rho)(1+\mu)(t'_{i_{j+1}} - \varphi_j) \\
&\quad - \Big(L_{v_{i_{j+1}}}(t'_{i_{j+1}}) - L_{v_{i_{j+1}}}(\varphi_j)\Big) + \frac{\kappa_{(v_{i_j},\ldots,v_{i_{j+1}})}}{2}.
\end{aligned}$$

Now let us go back in time to θ_{j+1}. Note that by definition of θ_{j+1} we have $\varphi_j - \theta_{j+1} \leq t_{i_{j+1}} - t'_{i_{j+1}} \leq \tau_{\left\{v_{i_{j+1}-1}, v_{i_{j+1}}\right\}}$. Therefore,

$$\begin{aligned}
\xi^s_{q'}(\theta_{j+1}) &= \xi^s_{q'}(\varphi_j) - \Big(L_{v_{i_{j+1}}}(\varphi_j) - L_{v_{i_{j+1}}}(\theta_{j+1})\Big) + (L_x(\varphi_j) - L_x(\theta_{j+1})) \\
&\geq 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_{j+1}}) + (1-\rho)(1+\mu)(t'_{i_{j+1}} - \varphi_j) \\
&\quad - \Big(L_{v_{i_{j+1}}}(t'_{i_{j+1}}) - L_{v_{i_{j+1}}}(\theta_{j+1})\Big) + \frac{\kappa_{(v_{i_j},\ldots,v_{i_{j+1}})}}{2} + (1-\rho)(\varphi_j - \theta_{j+1}) \\
&= 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_{j+1}}) + (1-\rho)(1+\mu)(t'_{i_{j+1}} - \theta_{j+1}) \\
&\quad - (1-\rho)(1+\mu)(\varphi_j - \theta_{j+1}) + (1-\rho)(\varphi_j - \theta_{j+1}) \\
&\quad - \Big(L_{v_{i_{j+1}}}(t'_{i_{j+1}}) - L_{v_{i_{j+1}}}(\theta_{j+1})\Big) + \frac{\kappa_{(v_{i_j},\ldots,v_{i_{j+1}})}}{2} \\
&> 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_{j+1}}) + (1-\rho)(1+\mu)(t'_{i_{j+1}} - \theta_{j+1}) \\
&\quad - \mu\tau_{\left\{v_{i_{j+1}-1}, v_{i_{j+1}}\right\}} - \Big(L_{v_{i_{j+1}}}(t'_{i_{j+1}}) - L_{v_{i_{j+1}}}(\theta_{j+1})\Big) + \frac{\kappa_{(v_{i_j},\ldots,v_{i_{j+1}})}}{2} \\
&\overset{(9)}{>} 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t'_{i_{j+1}}) + (1-\rho)(1+\mu)(t'_{i_{j+1}} - \theta_{j+1}) \\
&\quad - \Big(L_{v_{i_{j+1}}}(t'_{i_{j+1}}) - L_{v_{i_{j+1}}}(\theta_{j+1})\Big).
\end{aligned}$$

Finally, to show that $q' \in P^s_{v_{i_{j+1}}}(\theta_{j+1})$, recall that $q' = (v_{i_{j+1}}, \dots, v_{i_j}) \circ q$. Because $\theta_{j+1} \le t'_{i_{j+1}}$ (by definition of i_{j+1}), Property (1) of the chain shows that $p_{i_{j+1}} \in P^s_{v_{i_{j+1}}}(\theta_{j+1})$, and in particular $(v_{i_{j+1}}, \dots, v_{i_j}) \in P^s_{v_{i_{j+1}}}(\theta_{j+1})$, as this is a sub-path of $p_{i_{j+1}}$. Thus, to show that $q' \in P^s_{v_{i_{j+1}}}(\theta_{j+1})$, it remains to show that $q \in P^s_{v_{i_j}}(\theta_{j+1})$.

By definition of q we have $q \in P^s_{v_{i_j}}(\varphi_j)$, and from (60) it follows that

$$\begin{aligned}
\xi^s_q(\varphi_j) = \Xi^s_{v_{i_j}}(\varphi_j) &\ge 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{i_j+1}) + (1-\rho+2\mu)(t_{i_j+1} - \varphi_j) \\
&\quad - (1+\rho)(1+\mu)(t_{i_j+1} - \varphi_j) \\
&= 2\rho\Lambda_s + 2((1-\rho)\mu - 2\rho)(t_0 - \varphi_j) \overset{(8)}{>} 0.
\end{aligned}$$

Also, from Property (5), v_{i_j} satisfies the (C, s)-stabilization condition at time $\varphi_j \le t'_{i_j}$. Therefore, Lemma 6.2 shows that for all $t' \in [t^-, \varphi_j]$ we have $q \in P^s_{v_{i_j}}(t')$. In particular, then, $q \in P^s_{v_{i_j}}(\theta_{j+1})$. This shows that $\Xi^s_{v_{i_{j+1}}}(\theta_{j+1}) \ge \xi^s_{q'}(\theta_{j+1})$ and completes the induction.

The induction ends at a time $\varphi_h \in [t^-, t_0 - \Lambda_s]$ and a node v_{i_h} satisfying

$$\begin{aligned}
\Xi^s_{v_{i_h}}(\varphi_h) &\ge 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{i_h+1}) + (1-\rho)(1+\mu)(t_{i_h+1} - \varphi_h) \\
&\quad - \Big(L_{v_{i_h}}(t_{i_h+1}) - L_{v_{i_h}}(\varphi_h)\Big) + (1+\rho)\mu(t_{i_h+1} - \varphi_h) \\
&\ge 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - t_{i_h+1}) + (1-\rho)(1+\mu)(t_{i_h+1} - \varphi_h) \\
&\quad - (1+\rho)(1+\mu)(t_{i_h+1}) - \varphi_h) + (1+\rho)\mu(t_{i_h+1} - \varphi_h) \\
&= 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)(t_0 - \varphi_h) \\
&\ge 2\rho\Lambda_s + ((1-\rho)\mu - 2\rho)\Lambda_s \\
&= (1-\rho)\mu \cdot \frac{C_{s-1}}{2(1-\rho)\mu} = \frac{C_{s-1}}{2},
\end{aligned}$$

where in the second to last step we again used that $(1-\rho)\mu - 2\rho > 0$ due to Inequality (7). Let $p = (v_{i_h}, \dots, y) \in P^s_{v_{i_h}}(\varphi_h)$ be a path such that $\xi^s_p(\varphi_h) = \Xi^s_{v_{i_h}}(\varphi_h) \ge C_{s-1}/2$. Thus,

$$\Psi^{s-1}_y(\varphi_h) \ge \psi^{s-1}_{\bar{p}}(\varphi_h) = \xi^s_p(\varphi_h) + \frac{\kappa_p}{2} > \frac{C_{s-1}}{2}. \tag{63}$$

From Property (5), node v_{i_h} satisfies the (C, s)-stabilization condition at time φ_h. Hence, we can apply Lemma 6.1 to p, yielding that y is $(C, s-1)$-legal at time φ_h, contradicting (63). ☐

7 Dynamic and Local Global Skew Estimates

In this section, we extend the analysis to handling and adapting to dynamic global skew estimates. From here on, we therefore assume that when inserting an edge $\{u, v\}$, the insertion duration $\mathcal{I}_{\{u,v\}}$ is computed according to (11) in Algorithm 2. In the next lemma, we show that with this assumption the logical times of edge insertions on different levels are well separated, even if the insertions are for different edges and if different global skew estimates are used for inserting the different edges.

Lemma 7.1. *Let e and e' be two edges that are inserted with global skew estimates $\tilde{\mathcal{G}}_e$ and $\tilde{\mathcal{G}}_{e'}$, respectively. Further, let $s \geq 1$ and $s' \geq 1$ be two levels and consider the logical insertion times T_s^e and $T_{s'}^{e'}$. If $s \neq s'$, it holds that*

$$|T_s^e - T_{s'}^{e'}| \geq \frac{\min\{\mathcal{I}_e, \mathcal{I}_{e'}\}}{2^7 \cdot 4^{\min\{s,s'\}-2}}. \tag{64}$$

If $s = s'$, either (64) holds or $T_s^e = T_{s'}^{e'}$.

Proof. For convenience, we define $\ell_e := \left\lceil \log_2(\tilde{\mathcal{G}}_e/\mu + \mathcal{T}_e + \tau_e) \right\rceil$ and $\ell_{e'} := \left\lceil \log_2(\tilde{\mathcal{G}}_{e'}/\mu + \mathcal{T}_{e'} + \tau_{e'}) \right\rceil$. The lengths $\mathcal{I}_e$ and $\mathcal{I}_{e'}$ of the time insertion intervals are then $\mathcal{I}_e = \mathcal{B} \cdot 2^{3+\ell_e}$ and $\mathcal{I}_{e'} = \mathcal{B} \cdot 2^{3+\ell_{e'}}$, respectively. Without loss of generality, assume that $\ell_e \leq \ell_{e'}$ and thus also $\mathcal{I}_e \leq \mathcal{I}_{e'}$. Consider the insertion times T_s^e and $T_{s'}^{e'}$ of the edges e and e' on levels s and s', respectively. We have $T_s^e = T_0^e + \mathcal{I}_e\big(1 - \frac{1}{2^{s+1}-1}\big)$ and similarly $T_{s'}^{e'} = T_0^{e'} + \mathcal{I}_{e'}\big(1 - \frac{1}{2^{s'+1}-1}\big)$. Recall that T_0^e is chosen to be an integer multiple of $\mathcal{I}_e$ and $T_0^{e'}$ is chosen to be an integer multiple of $\mathcal{I}_e'$. Let $\Delta_\ell := \ell_e - \ell_{e'} \geq 0$ such that $\mathcal{I}_e = \mathcal{I}_{e'} \cdot 2^{\Delta_\ell}$. Defining $S_0^e := T_0^e/\mathcal{I}_{e'}$ and $S_0^{e'} := T_0^{e'}/\mathcal{I}_{e'}$, we can then write T_s^e and $T_{s'}^{e'}$ as

$$T_s^e = \mathcal{I}_{e'} \cdot \Bigg(\underbrace{S_0^e + 2^{\Delta_\ell}\left(1 - \frac{1}{2^{s+1}-1}\right)}_{=:x}\Bigg) \quad \text{and} \quad T_{s'}^{e'} = \mathcal{I}_{e'} \cdot \Bigg(\underbrace{S_0^{e'} + 1 - \frac{1}{2^{s'+1}-1}}_{=:y}\Bigg), \tag{65}$$

where S_0^e, $S_0^{e'}$, and Δ_ℓ are all non-negative integers. Consider x and y as defined in (65). We then have $|T_s^e - T_{s'}^{e'}| = \mathcal{I}_{e'} \cdot |x - y|$ and it therefore suffices to bound $|x-y|$. For any integer $a \geq 1$, it holds that $\frac{1}{2^a-1} = \sum_{i=1}^{\infty} \frac{1}{2^{a \cdot i}}$. As a consequence, we can write x and y as

$$x = \lceil x \rceil - \sum_{i=1}^{\infty} \frac{1}{2^{i(s+1)-k_\ell}} = \lceil x \rceil - \frac{2^{k_\ell}}{2^{s+1}-1} \quad \text{and} \quad y = \lceil y \rceil - \frac{1}{2^{(s'+1)}-1}, \tag{66}$$

where $k_\ell = \Delta_\ell \bmod (s+1)$ and thus $k_\ell \in \{0, \ldots, s\}$. From (66), we have

$$\lceil x \rceil - \frac{1}{2^{s-k_\ell+1}} > x = \lceil x \rceil - \frac{1}{2^{s-k_\ell+1}} \cdot \frac{2^{s+1}}{2^{s+1}-1} \overset{(s\geq 1)}{\geq} \lceil x \rceil - \frac{1}{2^{s-k_\ell+1}} \cdot \frac{4}{3} \overset{(s-k_\ell \geq 0)}{\geq} \lceil x \rceil - \frac{2}{3} \tag{67}$$

and

$$\lceil y \rceil - \frac{1}{2^{s'+1}} > y = \lceil y \rceil - \frac{1}{2^{s'+1}} \cdot \frac{2^{s'+1}}{2^{s'+1}-1} \overset{(s'\geq 1)}{\geq} \lceil y \rceil - \frac{1}{2^{s'+1}} \cdot \frac{4}{3}. \overset{(s'\geq 1)}{\geq} \lceil y \rceil - \frac{1}{3}. \tag{68}$$

Let us first consider the case where $\lceil x \rceil \neq \lceil y \rceil$. From the last inequalities of (67) and (68), we then get that

$$|x - y| > \frac{1}{3}. \tag{69}$$

Let us therefore come to the case where $\lceil x \rceil = \lceil y \rceil$. If $s - k_\ell < s'$, i.e., $s - k_\ell \leq s' - 1$, (67) and (68) imply that

$$y - x > \frac{1}{2^{s-k_\ell+1}} - \frac{1}{2^{s'+1}} \cdot \frac{4}{3} \overset{(s'-1 \geq s-k_\ell)}{\geq} \frac{1}{3 \cdot 2^{s-k_\ell+1}} \geq \frac{1}{6 \cdot 2^{\min\{s,s'\}}}. \tag{70}$$

Similarly, if $\lceil x \rceil = \lceil y \rceil$ and $s - k_\ell > s'$, i.e., $s - k_\ell \geq s' + 1$, we obtain

$$x - y > \frac{1}{2^{s'+1}} - \frac{1}{2^{s-k_\ell+1}} \cdot \frac{4}{3} \overset{(s-k_\ell \geq s'+1)}{\geq} \frac{1}{3 \cdot 2^{s'+1}} = \frac{1}{6 \cdot 2^{\min\{s,s'\}}}. \tag{71}$$

Finally, for $\lceil x \rceil = \lceil y \rceil$ and $s - k_\ell = s'$, we either have $k_\ell = 0$ and $s = s'$ in which case (67) and (68) imply that $x = y$ and therefore also $T^e_s = T^{e'}_{s'}$. Otherwise, assume that $k_\ell > 0$ and $s = s' + k_\ell$. We then get

$$x - y = \frac{1}{2^{s'+1}} \cdot \left(\frac{2^{s'+1}}{2^{s'+1} - 1} - \frac{2^{s+1}}{2^{s+1} - 1} \right) > \frac{1}{2^{s'+1}} \cdot \frac{2^{s+1} - 2^{s'+1}}{2^{s+s'+2}} \overset{(s>s')}{\geq} \frac{1}{8 \cdot 2^{2s'}} = \frac{1}{8 \cdot 4^{\min\{s,s'\}}}. \tag{72}$$

Combining (69), (70), (71), and (72), we either have $s = s'$ and $x = y$ or we get that

$$|x - y| \geq \frac{1}{8 \cdot 4^{\min\{s,s'\}}} = \frac{1}{2^7 \cdot 4^{\min\{s,s'\}-2}}.$$

Consequently, we either have $s = s'$ and $T^e_s = T^{e'}_{s'}$, or we obtain

$$|T^e_s - T^{e'}_{s'}| \geq \frac{\mathcal{I}_{e'}}{2^7 \cdot 4^{\min\{s,s'\}-2}} = \frac{\min\{\mathcal{I}_e, \mathcal{I}_{e'}\}}{2^7 \cdot 4^{\min\{s,s'\}-2}},$$

and thus the claim of the lemma follows. □

"The" gradient property here is actually a time-dependent notion, since the global skew varies over time. The algorithm will take some time to adapt to a smaller global skew, and this process is complicated by potential simultaneous edge insertions. To capture the former, we define for each time t a certain time $P(t)$ that lies sufficiently far in the past for the algorithm to have time to accomodate the corresponding global skew (at time $P(t)$). For each $t \in \mathbb{R}^+_0$ such that this value is defined, we set

$$P(t) := \max\{t' \in [0, t] \,|\, \mathcal{B} \cdot \mathcal{G}(t') = \mu(t - t')\}. \tag{73}$$

Note that $P(t)$ exists by continuity of $f(t') := \mathcal{B}\mathcal{G}(t') - \mu(t - t')$ for all $t \geq \mathcal{B}\mathcal{G}(0)/\mu$, since this implies $f(0) < 0$ and trivially we have that $f(t) \geq 0$. In the following, let $t_{\min}$ denote the minimal time so that $P(t_{\min})$ exists, i.e., $P : [t_{\min}, \infty) \to \mathbb{R}^+_0$ such that $P(t) \leq t$. Our goal will be to prove a non-trivial gradient property for times $t \geq t_{\min}$; for smaller times, the algorithm had insufficient time to converge to small skews (where the meaning of "small" depends on $\mathcal{G}(P(t))$, as clarified below).

Before proceeding with the definition of the gradient sequences we use, let us establish some basic properties of $P(t)$. In the following, we use the shorthand

$$\hat{\mathcal{G}} := 2\mathcal{G}(P(t)).$$

Lemma 7.2. *For each $t \geq t_{\min}$ and all $t' \in [P(t), t]$, it holds that*
(i) $\mathcal{G}(t') \leq \hat{\mathcal{G}}$ and
(ii) $\mathcal{G}(t') \geq (t - t')\mu/\mathcal{B}$.

Proof. Fix any $t \geq t_{\min}$. For $t' \in [P(t), t]$, by Theorem 5.6 it holds that

$$\mathcal{G}(t') \leq \mathcal{G}(P(t)) + 2\rho(t' - P(t)) \leq \mathcal{G}(P(t)) + 2\rho(t - P(t)) = \mathcal{G}(P(t)) + \frac{2\rho\mathcal{B}\mathcal{G}(P(t))}{\mu} \overset{(12)}{\leq} \hat{\mathcal{G}},$$

yielding Statement (i). Regarding Statement (ii), assume for the sake of contradiction that

$$f(t') := \mathcal{B}\mathcal{G}(t') - \mu(t - t') < 0.$$

Clearly, f is continuous and $f(t) \geq 0$. Hence, there must exist some $t'' \in (t', t]$ so that $f(t'') = 0$. This contradicts the maximality of $P(t) \leq t' < t'' \leq t$ among times smaller or equal to t with the property that $f(t) = 0$. We conclude that Statement (ii) must be true as well. □

The first property enables us to use $\hat{\mathcal{G}}$ as a global skew upper bound for defining gradient sequences pertinent for the entire interval $[P(t), t]$. As shown in the following lemma, the second property guarantees that, during $[(2P(t)+t)/3, t]$, no edge insertions happen on any level $s > 0$ that are based on a global skew estimate substantially smaller than $\mathcal{G}(P(t))$.

Lemma 7.3. *For each $t \geq t_{\min}$, all $t' \in [(2P(t)+t)/3, t]$, and any nodes u, v, if u adds v to N_u^s for any $s > 0$ at time t', then it holds that the corresponding call to **insertedge** (see Listing 2) computes $\mathcal{I}_{\{u,v\}}(\tilde{\mathcal{G}}) \geq (1-\rho)\mathcal{B}\mathcal{G}(P(t))/(10\mu)$, where $\tilde{\mathcal{G}}$ is the global skew estimate passed to **insertedge**.*

Proof. Fix t, t', u, and v, and suppose u adds v to N_u^s at time t'. Denote by t_0 the (most recent) time when u added v to N_u^0 and suppose that $\tilde{\mathcal{G}}$ was used in the call to **insertedge** at time t_0.

Assume that $w \in \{u, v\}$ is the leader of edge $\{u, v\}$. We distinguish two cases, the first being that $t_0 < P(t)$. For this case, note that the corresponding logical time for which the insertion is complete on all levels satisfies

$$\begin{aligned} T_\infty^{\{u,v\}} &\leq L_w(t_0) + (1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}} + 2\tau_{\{u,v\}}) + 2\mathcal{I}_{\{u,v\}} \\ &\overset{(11)}{=} L_w(t_0) + (1+\rho)(1+\mu)(\mathcal{T}_{\{u,v\}} + 2\tau_{\{u,v\}}) + 2\mathcal{B} \cdot 2^{3+\lceil \log(\tilde{\mathcal{G}}/\mu + \mathcal{T}_{\{u,v\}} + \tau_{\{u,v\}}) \rceil} \\ &\overset{(11)}{<} L_w(t_0) + 3\mathcal{I}_{\{u,v\}}. \end{aligned}$$

We conclude that the time t_∞ so that $L_w(t_\infty) = T_\infty^{\{u,v\}}$ is bounded from above by

$$t_\infty \leq t_0 + \frac{3\mathcal{I}_{\{u,v\}}}{1-\rho}.$$

Moreover, if $t' > t_\infty$ (which may happen if $w = v$), Statement (i) of Lemma 7.2 yields that

$$\begin{aligned} T_\infty^{\{u,v\}} &> L_u(t') \\ &\geq L_w(t') - \mathcal{G}(t') \\ &\geq L_w(t_\infty) + (1-\rho)(t' - t_\infty) - \hat{\mathcal{G}} \\ &= T_\infty^{\{u,v\}} + (1-\rho)(t' - t_\infty) - \hat{\mathcal{G}}. \end{aligned}$$

Therefore,

$$t' - t_0 = t' - t_\infty + t_\infty - t_0 \leq \frac{3\mathcal{I}_{\{u,v\}} + \hat{\mathcal{G}}}{1-\rho}.$$

As $t' \geq (2P(t)+t)/3$ is equivalent to $t - P(t) \leq 3(t' - P(t))$, this leads to

$$\mathcal{G}(P(t)) = \frac{\mu(t - P(t))}{\mathcal{B}} \leq \frac{3\mu(t' - P(t))}{\mathcal{B}} < \frac{3\mu(t' - t_0)}{\mathcal{B}} \leq \frac{9\mu\mathcal{I}_{\{u,v\}} + 3\mu\hat{\mathcal{G}}}{(1-\rho)\mathcal{B}} \overset{(7,12)}{<} \frac{9\mu\mathcal{I}_{\{u,v\}}}{(1-\rho)\mathcal{B}} + \frac{\mathcal{G}(P(t))}{10}.$$

Rearranging this inequality, we obtain that

$$\frac{(1-\rho)\mathcal{B}\mathcal{G}(P(t))}{10\mu} \leq \mathcal{I}_{\{u,v\}},$$

i.e., the claim of the lemma holds.

The second case is that $t_0 \geq P(t)$. Since trivially $t_0 < t' \leq t$, Statement (ii) of Lemma 7.2 implies that

$$\mathcal{I}_{\{u,v\}} > \frac{\mathcal{B} \cdot 2^3 \cdot \tilde{\mathcal{G}}}{\mu} \geq \frac{\mathcal{B} \cdot 2^3 \cdot \mathcal{G}(t_0)}{\mu} \overset{(7,12)}{>} 3\mathcal{G}(t_0) + \frac{4\mathcal{B}\mathcal{G}(t_0)}{\mu} \overset{(73)}{\geq} 3\mathcal{G}(t_0) + 4(t - t_0)$$

and hence

$$T_1^{\{u,v\}} > L_w(t_0) + \frac{2\mathcal{I}_{\{u,v\}}}{3} > L_u(t_0) + \frac{8(t-t_0)}{3} \overset{(7)}{>} L_u(t_0) + (1+\rho)(1+\mu)(t - t_0).$$

We conclude that

$$t' \geq t_0 + \frac{T_1^{\{u,v\}} - L_u(t_0)}{(1+\rho)(1+\mu)} > t,$$

contradicting the prerequisite that $t' \in [(2P(t) + t)/3, t]$. Therefore, the first case must always apply and the proof is complete. □

In summary, we have established that for any time $t \geq t_{\min}$ that (i) $2 \cdot \mathcal{G}(P(t))$ is a valid upper bound on the global skew throughout $[P(t), t]$ and (ii) that no edge e is inserted on any level $s > 0$ during $[(2P(t) + t)/3, t]$ with a value of $\mathcal{I}_e < (1 - \rho)\mathcal{B}\mathcal{G}(P(t))/(10\mu)$. The latter property ensures that the sets $\mathbb{T}_s$, $s > 0$, are sufficiently "sparse" (i.e., insertion times are sufficiently well-separated) for the algorithm to stabilize to small skews during $[(P(t)+2t)/3, t]$, based on the guaranteed upper bound of $2\mathcal{G}(P(t))$ on the global skew. Note that we restrict the time interval for which we show convergence to $[(P(t) + 2t)/3, t]$, so there is a buffer against interference from edge insertions with small values of $\mathcal{I}_e$ that may have occured before time $(2P(t) + t)/3$. However, we know little about $\mathbb{T}_s \cap (t, \infty)$, since the global skew might decrease quickly after time t; this is a technical issue that we will deal with by, essentially, ignoring insertions after time t.

Let us now formalize the above intuition by defining suitable time periods and (time- and node-dependent) gradient sequences so that the preconditions of Theorem 5.18 will be satisfied for any level $s > 1$, time, and node for which the respective gradient sequence C satisfies that $C_s < C_{s-1}$.

Definition 7.4 (Instability Periods). *For a level $s > 1$, a time $t \geq t_{\min}$, and a node u, we define the set of its s-*unstable times (with respect to time t) *as*

$$U_s(u,t) := \left\{ t' \in [(P(t) + 2t)/3, t] \,\middle|\, \exists\, T_s \in \mathbb{T}_s : |L_u(t') - T_s| \leq A_s(\hat{\mathcal{G}}) \right\},$$

where

$$A_s(\hat{\mathcal{G}}) := \left(\frac{(7+2\rho)(1+\mu)}{2\mu(1-\rho)} + 2s \right) \frac{2\hat{\mathcal{G}}}{\sigma^{s-2}}. \tag{74}$$

For a level $s' > 2$, a time $t \geq t_{\min}$, and a node u, the set of s'-recovery times is

$$R_{s'}(u,t) := \bigcup_{s<s'} \{t' \in [(P(t) + 2t)/3, t] \,|\, \exists\, T_s \in \mathbb{T}_s : B_{s,s'-1}(\hat{\mathcal{G}}) < |L_u(t') - T_s| - A_s(\hat{\mathcal{G}}) \leq B_{s,s'}(\hat{\mathcal{G}})\},$$

where

$$B_{s_1,s_2}(\hat{\mathcal{G}}) := \sum_{s=s_1}^{s_2-1} \beta_s(\hat{\mathcal{G}}) \quad \text{and} \quad \beta_s(\hat{\mathcal{G}}) := \left(\frac{(7+2\rho)(1+\mu)}{2\mu(1-\rho)} + s \right) \frac{2\hat{\mathcal{G}}}{\sigma^{s-2}}. \tag{75}$$

Here, the time intervals $U_s(u,t)$ provide a "buffer" around the (logical!) times from T_s, during which we will not make any non-trivial guarantees on level s at node u, i.e., the respective gradient sequence C will satisfy that $C_s = C_{s-1}$. The additional "buffers" provided by the sets $R_{s'}$ ensure that the gradient sequences at nodes with similar logical times do not differ in more than a single level. This is crucial for applying Theorem 5.18, since it requires the level-s stabilization condition to hold for all nodes with logical times from a certain range around the logical clock value of the node we examine.

Before defining suitable gradient sequences based on the above sets, we must show that the sets for different levels are pairwise disjoint. However, as mentioned earlier, we have no control over edge insertions at times larger than t. We overcome this by first considering a constrained set of executions for which there are no insertions after time t and only afterwards inferring skew bounds for arbitrary executions.

Definition 7.5 (Insertion-bounded Executions)**.** *For $t \in \mathbb{R}_0^+$, an execution is called* t-insertion-bounded *iff no edges are inserted on any level at times greater than t.*

For such executions, we can show that the sets specified in Definition 7.4 are disjoint.

Lemma 7.6. *Consider a t-insertion-bounded execution for some $t \geq t_{\min}$. Then it holds for all nodes u and $1 < s' < s$ that $U_s(u,t) \cap U_{s'}(u,t) = \emptyset$ and $U_s(u,t) \cap R_{s'}(u,t) = \emptyset$.*

Proof. For a given $s > 1$ and $T_s \in \mathbb{T}_s$, abbreviate

$$U_s(T_s) := \left\{t' \in [(P(t)+2t)/3, t] \,\middle|\, |L_u(t') - T_s| \leq A_s(\hat{\mathcal{G}})\right\}$$

and, for $s' > s$,

$$R_{s'}(T_s) := \{t' \in [(P(t)+2t)/3, t] \,|\, B_{s,s'-1}(\hat{\mathcal{G}}) < |L_u(t') - T_s| - A_s(\hat{\mathcal{G}}) \leq B_{s,s'}(\hat{\mathcal{G}})\}.$$

This entails that

$$U_s(u,t) = \bigcup_{T_s \in \mathbb{T}_s} U_s(T_s) \quad \text{and} \quad R_{s'}(u,t) = \bigcup_{s<s'} \bigcup_{T_s \in \mathbb{T}_s} R_{s'}(T_s).$$

Moreover, we have

$$U_s(T_s) \cup \bigcup_{s'>s} R_{s'}(T_s) = \Big(T_s - A_s(\hat{\mathcal{G}}) - B_{s,\infty}(\hat{\mathcal{G}}), T_s + A_s(\hat{\mathcal{G}}) + B_{s,\infty}(\hat{\mathcal{G}})\Big), \tag{76}$$

where

$$B_{s,\infty}(\hat{\mathcal{G}}) := \lim_{s' \to \infty} B_{s,s'}(\hat{\mathcal{G}}) = \sum_{s'=s}^{\infty} \beta_{s'}(\hat{\mathcal{G}}).$$

Evaluating this limit is straightforward, yielding

$$\begin{aligned}
A_s(\hat{\mathcal{G}}) + B_{s,\infty}(\hat{\mathcal{G}}) \quad &= \quad \frac{2\hat{\mathcal{G}}}{\sigma^{s-2}}\left(\frac{(7+2\rho)(1+\mu)}{2\mu(1-\rho)}\left(1+\frac{\sigma}{\sigma-1}\right) + s\left(2+\frac{\sigma}{\sigma-1}\right) + \frac{\sigma}{(\sigma-1)^2}\right)\\
&\overset{\sigma\geq 101}{\leq} \quad \frac{2\hat{\mathcal{G}}}{4^{s-2}}\left(\left(\frac{1407}{200}+\frac{201\rho}{100}\right)\cdot\frac{1+\mu}{\mu(1-\rho)} + \frac{301s}{100\cdot 12^{s-2}} + \frac{101}{100^2}\right)\\
&\overset{s\geq 2}{\leq} \quad \frac{2\hat{\mathcal{G}}}{4^{s-2}}\left(\left(\frac{1407}{200}+\frac{201\rho}{100}\right)\cdot\frac{1+\mu}{\mu(1-\rho)} + \frac{301}{100} + \frac{101}{100^2}\right)\\
&\overset{(13,7)}{<} \quad \frac{16\hat{\mathcal{G}}}{4^{s-2}\mu(1-\rho)}.
\end{aligned} \tag{77}$$

In particular, this expression is maximized for $s = 2$, giving

$$\begin{aligned} A_s(\hat{\mathcal{G}}) + B_{s,\infty}(\hat{\mathcal{G}}) + \hat{\mathcal{G}} &\overset{(7,77)}{<} \frac{17\hat{\mathcal{G}}}{\mu(1-\rho)} \\ &= \frac{34\mathcal{G}(P(t))}{\mu(1-\rho)} \\ &\overset{(73)}{=} \frac{34 \cdot (t - P(t))}{(1-\rho)\mathcal{B}} \\ &\overset{(12)}{\leq} \frac{(1-\rho)(t-P(t))}{3}. \end{aligned}$$

Now consider any $T_s \in \mathbb{T}_s$ such that $L_v((2P(t)+t)/3) > T_s$ for some $v \in V$. From Statement (i) of Lemma 7.2 and the above inequality, we obtain that

$$\begin{aligned} L_u\left(\frac{P(t)+2t}{3}\right) - T_s &\geq L_u\left(\frac{2P(t)+t}{3}\right) + \frac{(1-\rho)(t-P(t))}{3} - T_s \\ &\geq L_v\left(\frac{2P(t)+t}{3}\right) - \mathcal{G}\left(\frac{2P(t)+t}{3}\right) + \frac{(1-\rho)(t-P(t))}{3} - T_s \\ &\geq \frac{(1-\rho)(t-P(t))}{3} - \hat{\mathcal{G}} \\ &> A_s(\hat{\mathcal{G}}) + B_{s,\infty}(\hat{\mathcal{G}}), \end{aligned}$$

and hence $U_s(T_s) = \emptyset$ and $R_{s'}(T_s) = \emptyset$ for all $s' > s$. Thus, any insertion time T_s lies sufficiently far in the past and is of no concern. As the execution is t-insertion-bounded, we conclude that it suffices to consider $s, s' \geq 2$, $T_s \in \mathbb{T}_s$, and $T_{s'} \in \mathbb{T}_{s'}$, so that there exists a node v inserting an edge $\{v, w\}$ on level s at the time $t_v \in [(2P(t)+t)/3, t]$ satisfying that $L_v(t_v) = T_s$ and a node v' inserting an edge $\{v', w'\}$ on level s' at the time $t_{v'} \in [(2P(t)+t)/3, t]$ satisfying that $L_{v'}(t_{v'}) = T_{s'}$.

Because we have $s \neq s'$, Lemma 7.1 states that

$$|T_s - T_{s'}| \geq \frac{\min\{\mathcal{I}_{\{v,w\}}(\tilde{\mathcal{G}}), \mathcal{I}_{\{v',w'\}}(\tilde{\mathcal{G}}')\}}{2^7 \cdot 4^{\min\{s,s'\}-2}},$$

where $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{G}}'$ are the estimates used in the computation of the logical insertion times T_s and $T_{s'}$, respectively (by the leaders of the inserted edges $\{u, v\}$ and $\{u', v'\}$). Applying Lemma 7.3, we see that

$$\begin{aligned} \frac{\min\{\mathcal{I}_{\{v,w\}}(\tilde{\mathcal{G}}), \mathcal{I}_{\{v',w'\}}(\tilde{\mathcal{G}}')\}}{2^7 \cdot 4^{\min\{s,s'\}-2}} &\geq \frac{(1-\rho)\mathcal{B}\mathcal{G}(P(t))}{10\mu \cdot 2^7 \cdot 4^{\min\{s,s'\}-2}} \\ &\overset{(12)}{\geq} \frac{32\mathcal{G}(P(t))}{4^{\min\{s,s'\}-2}\mu(1-\rho)} \\ &= \frac{16\hat{\mathcal{G}}}{4^{\min\{s,s'\}-2}\mu(1-\rho)} \\ &\overset{(77)}{>} A_s(\hat{\mathcal{G}}) + B_{s,\infty}(\hat{\mathcal{G}}) + A_{s'}(\hat{\mathcal{G}}) + B_{s',\infty}(\hat{\mathcal{G}}). \end{aligned}$$

Recalling (76), we conclude that

$$\left(U_s(T_s) \cup \bigcup_{s''>s} R_{s''}(T_s)\right) \cap \left(U_{s'}(T_{s'}) \cup \bigcup_{s''>s'} R_{s''}(T_{s'})\right) = \emptyset.$$

Thus, all cases are covered and the proof is complete. ∎

With this lemma and the definitions preceding it at hand, we can now specify gradient sequences suitable for deriving our skew bounds.

Definition 7.7 (Global and Local Gradient Sequences)**.** *Given times $t \geq t_{\min}$ and $t' \in [P(t), t]$, we define the* global gradient sequence $C^{(t,t')}$ *as follows. Set $\Delta_1(t) := t - P(t)$, and*

$$\Delta_s(t) := \sum_{s'=s}^{\infty} \frac{7\hat{\mathcal{G}}}{2(1-\rho^2)\mu\sigma^{\max\{s'-3,0\}}} \tag{78}$$

for $s > 1$. Denote by $s' \in \mathbb{N}$ the uniqe level such that $t' \in [t - \Delta_{s'}(t), t - \Delta_{s'+1}(t))$. Then

$$C_s^{(t,t')} := \begin{cases} \frac{2\hat{\mathcal{G}}}{\sigma^{s-1}} & \text{if } s \leq s' \\ \frac{2\hat{\mathcal{G}}}{\sigma^{s'-1}} & \text{else.} \end{cases}$$

For the above parameters and a node u, the local gradient sequence at u *is given by*

$$C_s^{(t,t',u)} := \begin{cases} C_{s-1}^{(t,t')} & \text{if } t' \in \bigcup_{s'' \leq s} U_{s''}(u,t) \cup R_{s''}(u,t) \\ C_s^{(t,t')} & \text{else.} \end{cases}$$

Since

$$\begin{aligned} \Delta_2(t) &= \sum_{s'=2}^{\infty} \frac{7\hat{\mathcal{G}}}{2(1-\rho^2)\mu\sigma^{\max\{s'-3,0\}}} \\ &= \left(1 + \frac{\sigma}{\sigma - 1}\right) \frac{7\hat{\mathcal{G}}}{2(1-\rho^2)\mu} \\ &= \left(1 + \frac{\sigma}{\sigma - 1}\right) \frac{7(t - P(t))}{2(1-\rho^2)\mathcal{B}} \\ &\overset{(12)}{\leq} \frac{t - P(t)}{3} \\ &< t - P(t), \end{aligned} \tag{79}$$

the global sequences are well-defined (i.e., decreasing in s), implying the same for the local sequences. We are now ready to prove our main result.

Lemma 7.8. *For any t-insertion-bounded execution with $t_{\min} \leq t$, it holds for all times $t' \in [P(t), t]$ and nodes $u \in V$ that u is $C^{(t,t',u)}$-legal at time t.*

Proof. Suppose that this statement is false: Let $\bar{t}$ be the smallest time such that there exists a node u and a time $t^+ \geq \bar{t}$ such that $\bar{t} \in [P(t^+), t^+]$ and u is not $C^{(t^+,\bar{t},u)}$-legal at time $\bar{t}$ for some level $\bar{s}$. W.l.o.g., assume that $\bar{s}$ is the smallest level for which u is not $C^{(t^+,\bar{t},u)}$-legal at time $\bar{t}$. To simplify the notation, in the following we use $\hat{\mathcal{G}}^+ := 2\mathcal{G}(P(t^+))$. Note that by statement (i) of Lemma 7.2, $\mathcal{G}(t) \leq \hat{\mathcal{G}}^+$ for all $t \in [P(t^+), t^+]$. This implies that for all $t \in \left[P(t^+) + \Theta_2^{(t^+,\bar{t},u)}, t^+\right]$, condition (S0) of the stabilization condition (cf. Def. 5.16) holds and therefore by Lemma 5.17, the system is $(C^{(t^+,\bar{t},u)},1)$-legal for all nodes $v \in V$ and all times $t' \in [P(t^+),t^+]$; hence, we in particular have $\bar{s} > 1$.

We will now define a gradient sequence $\bar{C}$ such that $\bar{C}_{\bar{s}} = C_{\bar{s}}^{(t^+,\bar{t},u)}$ and, for $s < \bar{s}$, $\bar{C}_s \geq C_s^{(t^+,\bar{t},u)}$. Note that, by minimality of $\bar{s}$, node u is $(\bar{C},\bar{s})$-legal if and only if it is $(C^{(t^+,\bar{t},u)},\bar{s})$-legal. It is thus sufficient to derive a contradiction to the assumption that node u is not $(\bar{C},\bar{s})$-legal at time $\bar{t}$. The sequence is defined as follows:

$$\bar{C}_s := \begin{cases} \frac{2\hat{\mathcal{G}}^+}{\sigma^{s-1}} & \text{if } C_{\bar{s}}^{(t^+,\bar{t},u)} = \frac{2\hat{\mathcal{G}}^+}{\sigma^{\bar{s}-1}} \\ \frac{2\hat{\mathcal{G}}^+}{\sigma^{\max\{s-2,0\}}} & \text{if } C_{\bar{s}}^{(t^+,\bar{t},u)} = \frac{2\hat{\mathcal{G}}^+}{\sigma^{\bar{s}-2}}. \end{cases}$$

Note that due to the minimality of $\bar{s}$, Statement (iii) of Lemma 5.15 shows that $C_{\bar{s}}^{(t^+,\bar{t},u)} < C_{\bar{s}-1}^{(t^+,\bar{t},u)}$ and therefore either $C_{\bar{s}}^{(t^+,\bar{t},u)} = \frac{2\hat{\mathcal{G}}^+}{\sigma^{\bar{s}-1}}$ or $C_{\bar{s}}^{(t^+,\bar{t},u)} = \frac{2\hat{\mathcal{G}}^+}{\sigma^{\bar{s}-2}}$. Hence, $\bar{C}$ is well-defined for all cases.

We will use Theorem 5.18 to prove $(\bar{C},\bar{s})$-legality of u at time $\bar{t}$, so our goal is to show that node u satisfies the preconditions to apply the lemma. We set the time

$$\underline{t} := \bar{t} - \frac{5}{2} \cdot \frac{\bar{C}_{\bar{s}-1}}{\mu(1-\rho^2)}. \tag{80}$$

Further, for a time $t \in [\underline{t},\bar{t}]$, we define

$$V_u(t) := \left\{ v \in V : \exists \text{ path } p = (u,\dots,v) \in P_u^{\bar{s}}(t) \text{ with } \kappa_p \leq \bar{C}_{\bar{s}-1} \right\}. \tag{81}$$

In order to apply Theorem 5.18, we show that for all times $t \in [\underline{t},\bar{t}]$, each node $v \in V_u(t)$ satisfies the $(\bar{C},\bar{s})$-stabilization condition at time t. Setting $t^- = \underline{t}$ and $t^+ = \bar{t}$, we can then apply the lemma. Since

$$\bar{t} - \underline{t} = \frac{5}{2} \cdot \frac{\bar{C}_{\bar{s}-1}}{\mu(1-\rho^2)} > \frac{\bar{C}_{\bar{s}-1}}{2(1-\rho)\mu} + 2\bar{\Theta}_{\bar{s}},$$

the lemma implies that at time $\bar{t}$,

$$\Psi_u^{\bar{s}}(\bar{t}) < \frac{\bar{C}_{\bar{s}-1}}{2\sigma} \leq \frac{\bar{C}_{\bar{s}}}{2}.$$

By Definition 5.13, this is a contradiction to the assumption that u is not $(\bar{C},\bar{s})$-legal at time $\bar{t}$.

We now show that the stabilization condition applies. Since $C_{\bar{s}}^{(t^+,\bar{t},u)} < C_{\bar{s}-1}^{(t^+,\bar{t},u)}$ (or Statement (iii) of Lemma 5.15 and the minimality of $\bar{s}$ yield a contradiction), we have that $\bar{t} \notin \bigcup_{s=0}^{\bar{s}-1}[t^+ - \Delta_s(t), t^+ - \Delta_{s+1}(t)]$. Therefore, $\bar{t} \geq t^+ - \Delta_{\bar{s}}$. In particular,

$$\begin{aligned}
\underline{t} &\geq \bar{t} - \frac{5}{2} \cdot \frac{\bar{C}_1}{\mu(1-\rho^2)} \\
&> t^+ - \Delta_2(t^+) - \frac{5\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)} \\
&\overset{(79)}{\geq} P(t^+) + \frac{2(t^+ - P(t^+))}{3} - \frac{5\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)} \\
&\overset{(73)}{=} P(t^+) + \frac{2\mathcal{B}\hat{\mathcal{G}}^+}{3\mu} - \frac{5\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)} \\
&\overset{(12)}{>} P(t^+) + \frac{14\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)} - \frac{5\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)} \\
&\overset{(24)}{>} P(t^+) + \bar{\Theta}_2.
\end{aligned}$$

In other words, $[\underline{t},\bar{t}] \subseteq [P(t^+)+\bar{\Theta}_2,t^+]$. As shown earlier, this entails that (S0) is satisfied at all nodes and times $t \in [\underline{t},\bar{t}]$.

Concerning (S1), from the previous observation that $\bar{t} \geq t^+ - \Delta_{\bar{s}}$, we have for $1<s<\bar{s}$ that

$$\begin{aligned}
\underline{t} &= \bar{t} - \frac{5}{2}\cdot\frac{\bar{C}_{\bar{s}-1}}{\mu(1-\rho^2)}\\
&\geq t^+ - \Delta_s(t^+) + \left(\Delta_s - \Delta_{s+1} - \frac{5}{2}\cdot\frac{\bar{C}_{\bar{s}-1}}{\mu(1-\rho^2)}\right)\\
&\overset{(78)}{\geq} t^+ - \Delta_s(t^+) + \left(\frac{7\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)\sigma^{\max\{s-3,0\}}} - \frac{5\hat{\mathcal{G}}^+}{2\mu(1-\rho^2)\sigma^{\max\{\bar{s}-3,0\}}}\right)\\
&> t^+ - \Delta_s(t^+) + \frac{\hat{\mathcal{G}}^+}{(1+\rho)\mu\sigma^{\max\{s-3,0\}}}\\
&\geq t^+ - \Delta_s(t^+) + \bar{\Theta}_s.
\end{aligned}$$

Thus, for $t \in [\underline{t}-\bar{\Theta}_s,\bar{t}]$ and $1<s<\bar{s}$, $C_s^{(t^+,t)} = 2\hat{\mathcal{G}}^+/\sigma^{s-1}$. We distinguish two cases according to which gradient sequence we use for $\bar{C}$.

Case $\bar{C}_s = 2\hat{\mathcal{G}}/\sigma^{\max\{s-2,0\}}$ for all s: Because for any node v and $s>1$, $C_s^{(t^+,t,v)} \leq \sigma C_s^{(t^+,t)} = \bar{C}_s$, the minimality of $\bar{t}$ and $\bar{s}$ imply that for all $t \in [\underline{t}-\bar{\Theta}_s,\bar{t}]$, $1<s<\bar{s}$, and $v \in V$, the system is $(\bar{C},s)$-legal at node v and time t. In particular, (S1) is satisfied at time t at all nodes $v \in V_u(t)$ w.r.t. $\bar{C}$ and $\bar{s}$.

Case $\bar{C}_s = 2\hat{\mathcal{G}}/\sigma^{s-1}$ for all s: Consider $1<s<\bar{s}$, a time $t\in[\underline{t}-\bar{\Theta}_s,\bar{t}]$, and a node $v \in V_u(t)$. We need to show that $C_s^{(t^+,t,v)} = C_s^{(t^+,t)} = \bar{C}_s$. Assume for the sake of contradiction that there is a minimal level $1<s<\bar{s}$ violating this claim for some t and $v\in V_u(t)$. We apply Lemma 5.14 to nodes u and v with respect to $(\bar{C},s-1)$-legality, where $\kappa_p \leq \bar{C}_{\bar{s}-1} \leq \bar{C}_{s-1}$. This yields

$$|L_v(t)-L_u(t)| < \left(s+\frac{1}{2}\right)\kappa_p + \frac{\bar{C}_{s-1}}{2} \leq s\bar{C}_{s-1},$$

and thus

$$\begin{aligned}
|L_v(t)-L_u(\bar{t})| &< s\bar{C}_{s-1} + (1+\rho)(1+\mu)(\bar{t}-(\underline{t}-\bar{\Theta}_s))\\
&= s\bar{C}_{s-1} + \frac{5(1+\mu)\bar{C}_{\bar{s}-1}}{2\mu(1-\rho)} + \frac{(1+\rho)(1+\mu)\bar{C}_{s-1}}{\mu(1-\rho)}\\
&\overset{(75)}{\leq} \beta_s(\hat{\mathcal{G}}^+).
\end{aligned} \tag{82}$$

Moreover, the fact that $C_s^{(t^+,t,v)} \neq C_s^{(t^+,t)}$ entails that

$$t \in \bigcup_{s'\leq s} U_{s'}(v,t^+) \cup R_{s'}(v,t^+)$$

Therefore, there exist $s'\leq s$ and $T_{s'} \in \mathbb{T}_{s'}$ so that

$$|L_v(t) - T_{s'}| \leq A_{s'}(\hat{\mathcal{G}}^+) + B_{s',s}(\hat{\mathcal{G}}^+).$$

Similarly, as we have that

$$\frac{2P(t^+)+t^+}{3} \overset{(79)}{\leq} t^+ - \Delta_2 \leq \bar{t} \leq t^+ \tag{83}$$

and $C_{\bar{s}}^{(t^+,\bar{t},u)} = 2\hat{\mathcal{G}}^+/\sigma^{\bar{s}-1} = C_{\bar{s}}^{(t^+,\bar{t})}$, it holds that

$$\bar{t} \notin \bigcup_{s''\leq \bar{s}} U_{s''}(v,t^+) \cup R_{s''}(v,t^+)$$

and hence

$$|L_u(\bar{t}) - T_{s'}| > A_{s'}(\hat{\mathcal{G}}^+) + B_{s',\bar{s}}(\hat{\mathcal{G}}^+).$$

Combining these two inequalities yields that

$$|L_u(\bar{t}) - L_v(t)| > B_{s',\bar{s}}(\hat{\mathcal{G}}^+) - B_{s',s}(\hat{\mathcal{G}}^+) = B_{s,\bar{s}}(\hat{\mathcal{G}}^+) \geq \beta_s(\hat{\mathcal{G}}^+),$$

contradicting (82).

It remains to show (S2) for each $v \in V_u(t)$ and all $t \in [\underline{t}, \bar{t}]$. Since we already established (S1), we can apply Lemma 5.14 to nodes u and v with respect to $(\bar{C}_{\bar{s}-1}, \bar{s}-1)$-legality, where $\kappa_p \leq \bar{C}_{\bar{s}-1}$. This yields

$$|L_v(t) - L_u(t)| < \left(\bar{s} - \frac{1}{2}\right)\kappa_p + \frac{\bar{C}_{\bar{s}-1}}{2} \leq \bar{s}\bar{C}_{\bar{s}-1}.$$

We claim that $\bar{t} \notin U_{\bar{s}}(u,t^+)$. For contradition, assume that $\bar{t} \in U_{\bar{s}}(u,t^+)$. We then have $C_{\bar{s}}^{(t^+,\bar{t},u)} = C_{\bar{s}-1}^{(t^+,\bar{t})}$. In order to have $C_{\bar{s}}^{(t^+,\bar{t},u)} \neq C_{\bar{s}-1}^{(t^+,\bar{t},u)}$, we thus need that $C_{\bar{s}-1}^{(t^+,\bar{t},u)} = C_{\bar{s}-2}^{(t^+,\bar{t})}$, which implies that $\bar{t} \in \bigcup_{s'\leq \bar{s}-1} U_{s'}(u,\bar{t}) \cup R_{s'}(u,\bar{t})$, which is a contradiction to the statement of Lemma 7.6. We can therefore conclude that $\bar{t} \notin U_{\bar{s}}(u,t^+)$. Together with (83), this entails for any $T_{\bar{s}} \in \mathbb{T}_{\bar{s}}$ that $|L_u(\bar{t}) - T_{\bar{s}}| \geq A_{\bar{s}}(\hat{\mathcal{G}}^+)$ and thus

$$\begin{aligned}
|L_v(t) - T_{\bar{s}}| &\geq |L_u(\bar{t}) - T_{\bar{s}}| - |L_u(\bar{t}) - L_u(t)| - |L_u(t) - L_v(t)| \\
&\geq A_{\bar{s}}(\hat{\mathcal{G}}^+) - (1+\rho)(1+\mu)(\bar{t} - \underline{t}) - \bar{s}\bar{C}_{\bar{s}-1} \\
&= A_{\bar{s}}(\hat{\mathcal{G}}^+) - \frac{5(1+\mu)\bar{C}_{\bar{s}-1}}{2\mu(1-\rho)} - \bar{s}\bar{C}_{\bar{s}-1} \\
&\overset{(74)}{=} \left(\frac{(1+\rho)(1+\mu)}{(1-\rho)\mu} + \bar{s}\right)\bar{C}_{\bar{s}-1} \\
&= (1+\rho)(1+\mu)\bar{\Theta}_{\bar{s}} + \bar{s}\bar{C}_{\bar{s}-1}.
\end{aligned}$$

We conclude that the preconditions for the application of Theorem 5.18 described earlier are met, yielding the stated contradiction to the original assumption that the claim of the lemma is wrong. □

Theorem 7.9. *At all times $t \geq t_{\min}$ and nodes $u \in V$, the system is $C^{(t)}$-legal, where $C_s^{(t)} = 4\mathcal{G}(P(t))/\sigma^{\max\{s-2,0\}}$.*

Proof. Consider any execution of the algorithm and fix a time $t \geq t_{\min}$. We create a t-insertion-bounded execution that is identical on $[0,t)$ as follows. We modify the given execution in that at time t all edges fail, i.e., $E(t') = \emptyset$ for all $t' \geq t$. Moreover, all nodes become aware of the non-existence of their incident edges at time t. Hence, all nodes clear their neighbor sets at time t

and, for all $s \in \mathbb{N}$, $\mathbb{T}_s \cap (t, \infty) = \emptyset$. Therefore, the resulting execution is t-insertion-bounded and identical to the original one on $[0, t)$.

We can apply Lemma 7.8 to the new execution, showing that at time t, each node u is $C^{(t,t,u)}$-legal. As $C_s^{(t,t,u)} \geq C_{s-1}^{(t,t)} = C_s^{(t)}$, each node is $C^{(t)}$-legal at time t in the modified execution. Since the modified execution is identical to the original execution during $[0, t)$ and logical clocks are continuous, the claim of the theorem follows. ☐

Corollary 7.10 (Gradient Property)**.** *For $t \geq t_{\min}$, set $E^\infty(t) := \cap_{s=1}^{\infty} E^s(t)$ and let p be a path connecting u and v in (V, E^∞) of minimal weight κ_p. For*

$$s(p) := \max\{2 + \lceil \log_\sigma(4\mathcal{G}(P(t))/\kappa_p \rceil, 1\},$$

it holds that

$$|L_u(t) - L_v(t)| \leq (s(p) + 1)\kappa_p.$$

Proof. Since p is a path in (V, E^∞) and $E^{s(p)} \subseteq E^\infty$, $p \in P_u^{s(p)}(t)$. By Theorem 7.9, the system is $C^{(t)}$-legal at u and v at time t. Applying Lemma 5.14 for level $s(p)$, we obtain that

$$|L_u(t) - L_v(t)| \leq \left(s(p) + \frac{1}{2}\right)\kappa_p + \frac{C_{s(p)}^{(t)}}{2} = \left(s(p) + \frac{1}{2}\right)\kappa_p + \frac{2\mathcal{G}(P(t))}{\sigma^{\max\{s(p)-2,0\}}} \leq (s(p) + 1)\kappa_p,$$

where the last inequality holds because $s(p) = 1$ implies that $\kappa_p > 4\mathcal{G}(P(t))$. ☐

8 Lower Bound on the Insertion Time

In this section, we strengthen the lower bound in [11] to match the stabilization time of $\mathcal{A}^{\text{OPT}}$. The original lower bound stated, roughly speaking, that the stabilization time of any $\mathcal{S}$-dynamic gradient CSA with a stable gradient skew of $\mathcal{S}^\infty$ cannot be better than $\Omega(D/\mathcal{S}^\infty(1))$ in graphs of diameter D. For CSAs with $\mathcal{O}(\log_{1/\rho} D)$-local skew, this bound implies that the stabilization time must be $\Omega(D/\log_{1/\rho} D)$. Algorithm $\mathcal{A}^{\text{OPT}}$ has a stabilization time of $\mathcal{O}(D)$, which does not match the bound in [11]; however, by refining the analysis in the lower bound, we can show that the algorithm is in fact asymptotically optimal in its stabilization time. In the stronger bound we reason about the *full* gradient property, which bounds the skew on paths of all distances, rather than just the local skew property, which bounds the skew on single edges.

Let us call a dynamic gradient CSA *non-trivial* if it has a stable gradient skew satisfying $\mathcal{S}^\infty(1) \in o(D)$. This essentially means that the algorithm guarantees a *local* skew (e.g., along single edges) that is better than the global skew.

Theorem 8.1. *Let $\mathcal{F} = \left\{f_D : \mathbb{R}_0^+ \to \mathbb{R}_0^+ \mid D \in \mathbb{R}\right\}$ be a family of functions, and let $c_1, c_2 \in (0, 1/16)$ be constants such that for all $f_D \in \mathcal{F}$ we have $f_D(c_1 D) \leq c_2 D$. Let $\mathcal{A}$ be a non-trivial stabilizing CSA guaranteeing a dynamic gradient skew of f_D in graphs of weighted diameter D. Then the stabilization time of $\mathcal{A}$ is at least $\Omega(D)$.*

Proof Sketch. We show that for sufficiently large diameters D, we can add a new edge and cause the skew on it to be larger than $\mathcal{S}$ after $\Omega(D)$ time. For simplicity we consider only line networks, where $D \in \Theta(n)$, but the proof can easily be modified to hold in general networks.

Consider a static line graph over $n+1$ nodes $v_0, \ldots, v_n$, where the estimate graph is the same as the communication graph and the weights of all edges are T. The diameter of the graph is $D = nT$. Let c_1, c_2 be the constants from the statement of the theorem, and let $u := v_{\lceil c_1 n \rceil}, v := v_{\lfloor n - c_1 n \rfloor}$. Finally, let $t_s \geq \mathcal{T}_S$ be some time after the stabilization time of the algorithm. By definition of $\mathcal{T}_S$, at any time after t_s, the skew on any path of weight d cannot exceed $2f_D(d)$.

The distance between v_0 and u and between v and v_n is at least $c_1 n$; thus, for all $t \geq t_s$ we have

$$\begin{aligned} L_{v_0}(t) - L_u(t) &\leq f_D(c_1 n) \leq c_2 n, \\ L_v(t) - L_{v_n}(t) &\leq f_D(c_1 n) \leq c_2 n. \end{aligned}$$

Also, $\operatorname{dist}(u, v) \geq n - c_1 n - 1 - (c_1 n + 1) = n - 2c_1 n - 2$.

In [11], we show that we can create an execution E in which

(a) There exists a time $t_2 \geq t_s$ such that $L_u(t_2) - L_v(t_2) \geq \frac{1}{4} \operatorname{dist}(u, v) \geq n - 2c_1 n - 2$, and

(b) The message delays on all edges between v_0 and u and between v and v_n are always at least $T/(1+\rho)$.

Next we create a new execution E', which is identical to E until time $t_1 := t_2 - c_1 n \cdot T/(1+\rho)$. At time t_1 in E', an edge between v_0 and v_n appears. Our goal is to maintain a large skew on $\{v_0, v_n\}$ until time t_2 to show that the algorithm has not stabilized by then.

Due to the large message delays, nodes u, v do not find out about the new edge until time t_2. Consequently, their skew in execution E' is the same as in E. The paths $v_0, v_1, \ldots, u$ and $v, \ldots, v_{n-1}, v_n$, which have weight at least $c_1 n$ by definition, are stable in both E and E'. Thus, the skew on each path cannot exceed $2f_D(c_1 n)$, that is, it cannot exceed $2c_2 n$. It follows that the skew between v_0 and v_n at time t_2 in E' is at least

$$\begin{aligned} L_{v_0}(t_2) - L_{v_n}(t_2) &= L_{v_0}(t_2) - L_u(t_2) + L_v(t_2) - L_{v_n}(t_2) + L_u(t_2) - L_v(t_2) \\ &\geq n - 2c_1 n - 2 - 4c_2 n > n/2 - 2. \end{aligned}$$

For sufficiently large n, this value exceeds $\mathcal{S}^\infty(1)$, since we assumed that $\mathcal{S}^\infty(1) \in o(D)$. Thus, we have showed that after $c_1 n \cdot T/(1+\rho) \in \Omega(D)$ time since edge $\{v_0, v_n\}$ appeared, the algorithm has not yet stabilized. $\square$

References

[1] S. Biaz and J. Lundelius Welch. Closed Form Bounds for Clock Synchronization Under Simple Uncertainty Assumptions. *Information Processing Letters*, 80(3):151–157, 2001.

[2] S. Biaz and J. Lundelius Welch. Closed Form Bounds for Clock Synchronization Under Simple Uncertainty Assumptions. *Information Processing Letters*, 80(3):151–157, 2001.

[3] E. W. Dijkstra. Self-stabilizing systems in spite of distributed control. *Communications of the ACM*, 17(11):643–644, 1974.

[4] D. Dolev, J. Halpern, B. Simons, and R. Strong. Dynamic Fault-tolerant Clock Synchronization. *Journal of the ACM (JACM)*, 42(1):143–185, 1995.

[5] D. Dolev, J. Halpern, and R. Strong. On the Possibility and Impossibility of Achieving Clock Synchronization. In *Proc. 16th ACM Symposium on Theory of Computing (STOC)*, pages 504–511, 1984.

[6] J. Elson, L. Girod, and D. Estrin. Fine-grained Network Time Synchronization Using Reference Broadcasts. *ACM SIGOPS Operating Systems Review*, 36:147–163, 2002.

[7] R. Fan and N. Lynch. Gradient Clock Synchronization. In *Proc. 23rd Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 320–327, 2004.

[8] R. Fan and N. Lynch. Gradient Clock Synchronization. In *Proc. 23rd ACM Symposium on Principles of Distributed Computing (PODC)*, pages 320–327, 2004.

[9] J. Halpern, B. Simons, R. Strong, and D. Dolev. Fault-tolerant Clock Synchronization. In *Proc. 3rd ACM Symposium on Principles of Distributed Computing (PODC)*, pages 89–102, 1984.

[10] F. Kuhn, C. Lenzen, T. Locher, and R. Oshman. Optimal Gradient Clock Synchronization in Dynamic Networks. In *Proc. 29th ACM Symposium on Principles of Distributed Computing (PODC)*, 2010.

[11] F. Kuhn, T. Locher, and R. Oshman. Gradient Clock Synchronization in Dynamic Networks. In *Proc. 21st ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 270–279, 2009.

[12] F. Kuhn and R. Oshman. Gradient Clock Synchronization Using Reference Broadcasts. In *Proc. 13th International Conference on Principles of Distributed Systems (OPODIS)*, pages 204–218, 2009.

[13] L. Lamport and P. Melliar-Smith. Synchronizing Clocks in the Presence of Faults. *Journal of the ACM (JACM)*, 32(1):52–78, 1985.

[14] C. Lenzen. *Synchronization and Symmetry Breaking in Distributed Systems*. PhD thesis, ETH Zurich, 2011.

[15] C. Lenzen, T. Locher, and R. Wattenhofer. Clock Synchronization with Bounded Global and Local Skew. In *49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, October 2008.

[16] C. Lenzen, T. Locher, and R. Wattenhofer. Tight Bounds for Clock Synchronization. *Journal of the ACM (JACM)*, 57(2), January 2010.

[17] T. Locher. *Foundations of Aggregation and Synchronization in Distributed Systems*. PhD thesis, ETH Zurich, February 2009. Diss. ETH No. 18249.

[18] T. Locher and R. Wattenhofer. Oblivious Gradient Clock Synchronization. In *Proc. 20th International Symposium on Distributed Computing (DISC)*, pages 520–533, 2006.

[19] J. Lundelius and N. Lynch. A New Fault-tolerant Algorithm for Clock Synchronization. In *Proc. 3rd ACM Symposium on Principles of Distributed Computing*, pages 75–88. ACM, 1984.

[20] J. Lundelius and N. Lynch. An Upper and Lower Bound for Clock Synchronization. *Information and Control*, 62(2/3):190–204, 1984.

[21] K. Marzullo and S. Owicki. Maintaining the Time in a Distributed System. In *Proc. 2nd ACM Symposium on Principles of Distributed Computing (PODC)*, pages 44–54, 1983.

[22] R. Ostrovsky and B. Patt-Shamir. Optimal and Efficient Clock Synchronization under Drifting Clocks. In *Proc. 18th ACM Symposium on Principles of Distributed Computing (PODC)*, pages 400–414, 1999.

[23] B. Patt-Shamir and S. Rajsbaum. A Theory of Clock Synchronization. In *Proc. 26th ACM Symposium on Theory of Computing (STOC)*, pages 810–819, 1994.

[24] T. K. Srikanth and S. Toueg. Optimal Clock Synchronization. *Journal of the ACM*, 34(3):626–645, 1987.