

A Combined Push-Pull Access Framework for Digital Twin Alignment and Anomaly Reporting

Federico Chiariotti, Fabio Saggese, Andrea Munari, Leonardo Badia, and Petar Popovski

Abstract—A digital twin (DT) contains a set of virtual models of real systems that are synchronized to their physical counterparts. This enables quick experimentation, simulating the consequences of decisions in real time. However, the DT’s accuracy depends on timely updates that maintain alignment with the real system. We can distinguish between: (i) *pull*-updates, which follow a request from the DT to the sensors, to decrease its drift from the physical state; (ii) *push*-updates, which contain anomalies and are sent proactively by the sensors. In this work, we devise a push-pull scheduler (PPS) to integrate the two types of updates and dynamically allocate resources. Our scheme strikes a balance in the trade-off between DT alignment in normal conditions and anomaly reporting, reducing model drift by over 20% with respect to state-of-the-art solutions, while maintaining the same anomaly detection guarantees, as well as reducing the worst-case anomaly detection age of incorrect information (AoII) from 70 ms to 30 ms under the same drift constraint.

Index Terms—Digital Twin, Medium Access Control, Age of Incorrect Information, data relevance

I. INTRODUCTION

The digital twin (DT) paradigm has emerged as a transformative approach to bridge the gap between physical systems and their virtual representations [1]. A DT comprises a set of virtual models that mirror real systems and processes, maintaining synchronization with their physical counterparts through sensor data streams. It enables real-time monitoring, prediction, and control across diverse industrial applications [2], along with accelerated experimentation.

The utility of DTs relies on maintaining precise alignment between virtual models and the evolving physical reality they represent, balancing *proactive* system monitoring of gradual changes with *reactive* responses to critical anomalies [3]. Although both are essential for DT operation, they impose conflicting demands on communication systems. Proactive monitoring typically involves integrating data from multiple

sensors to detect system drift [4] that may be invisible to individual sensors. In contrast, reactive alerts address urgent events that require immediate attention [3], such as equipment faults or safety violations. Here, individual sensors can detect and report events, guaranteeing reliable worst-case responsiveness.

These conflicting requirements play to the strengths of pull- and push-based communication. The former approach involves a central node, such as a base station (BS), requesting data from sensors as dictated by the DT. This paradigm is particularly well-suited for periodic updates, goal-oriented control tasks, and scenarios where the central system can optimally schedule transmissions based on global knowledge [5]. On the other hand, push-based approaches enable sensors to autonomously transmit data when they detect anomalous conditions. Based on random access protocols [6], they can effectively report rare anomalies requiring immediate attention [7], [8]. While these approaches have been viewed as alternative [9], recent works suggest that their combination can yield a significant *coexistence gain* [10].

Consider an example with $M = 3$ machines, each monitored by 4 sensors over communication frames with 4 slots. Not all sensors are required to identify model drift: the detection probability is 90% if 3 out of 4 sensors respond. A pull-only strategy could cycle through sensor groups, yielding an average detection delay of $(M - 1)/2 = 1$ frame, with the same delay for anomalies. A better solution pulls data from 3 sensors, reserving the fourth slot as a “push opportunity” for anomaly reporting. This increases drift detection delay to $(M - 1)/2 + (1 - 0.9)M = 1.3$ frames, but enables immediate anomaly reporting. Moreover, some anomalies may be reported by the pulled sensors, enabling prompt detection. Conversely, if an anomaly appears correlated with a drift event, the BS can proactively pull the remaining sensors from the affected machine. In short, all incoming messages, whether push or pull, contribute to updating the BS’s knowledge.

Despite the coexistence gain, several technical challenges have hindered the development of unified frameworks [11], from the design of frame structures supporting both approaches to the optimization of the joint resource allocation under partial observations and interdependent information sources. To address these challenges, we propose a *push-pull scheduler* (PPS) that dynamically allocates resources using both paradigms. Our approach models the DT as a hidden Markov model (HMM), considering a communication frame structure compatible with 3rd Generation Partnership Project (3GPP) standards. Our key contribution is the development

F. Chiariotti and L. Badia ({federico.chiariotti,leonardo.badia}@unipd.it) are with the Dept. of Information Engineering, University of Padova, Italy. F. Saggese (fabio.saggese@ing.unipi.it) is with the Dept. of Information Engineering, University of Pisa, Italy. A. Munari (andrea.munari@dlr.de) is with the Inst. of Communications and Navigation, German Aerospace Center (DLR), Weßling, Germany. P. Popovski (petarp@es.aau.dk) is with the Dept. of Electronic Systems, Aalborg University, Denmark. This work was supported, in part, by the Velux Foundation, Denmark, through the Villum Investigator Grant WATER, nr. 37793, and partly by the Horizon Europe SNS “6G-GOALS” project with grant 101139232. F. Saggese’s work is funded by Horizon Europe MSCA Postdoctoral Fellowships with Grant 101204088. A. Munari would like to thank the Federal Ministry of Research, Technology, and Space (BMFTR) for supporting the xG-RIC project as part of the research program Communication Systems “Souverän. Digital. Vernetzt.” (grant number 16KIS2429K).

of the PPS joint belief update mechanism and transmission scheduling strategy, which jointly optimize DT alignment. Its performance is verified through comprehensive simulations: PPS reduces the worst-case anomaly age of incorrect information (AoII) by 60% under the same DT drift constraint.

II. SYSTEM MODEL

Consider a wireless sensor network (WSN) in which a set $\mathcal{N}$ of sensors, with $|\mathcal{N}| = N$, observe a spatially distributed process and communicate with a BS. Our dual objective is to maintain the DT of the system aligned with physical reality, as well as detecting anomalous conditions as quickly as possible.

A. Drift and Anomaly Model

We denote the set of clusters of sensors that are mapped to different virtual models within the DT as $\mathcal{D} = \{\mathcal{C}^{(1)}, \dots, \mathcal{C}^{(D)}\}$. We assume that $\cup_{i=1}^D \mathcal{C}^{(i)} = \mathcal{N}_d \subseteq \mathcal{N}$ and $\mathcal{C}^{(i)} \cap \mathcal{C}^{(j)} = \emptyset \forall i \neq j$, i.e., only a subset of the sensors might be part of a cluster, and clusters are mutually exclusive. We then model the evolution of each cluster as an HMM, whose hidden state represents the drift between the physical system and its DT. Individual sensors make observations, which are then used to estimate the hidden state for cluster i at frame k , $y^{(i)}(k) \in \mathcal{Y}^{(i)}$, where $\mathcal{Y}^{(i)}$ denotes the set of possible states. We then define $\mathcal{Y}_d^{(i)} \subset \mathcal{Y}^{(i)}$ as the set of *drift states*, i.e., states in which the error of the DT exceeds a tolerance threshold.

Time is divided in *frames* of equal duration. The transition probability function $\mathcal{F}^{(i)} : \mathcal{Y}^{(i)} \times \mathcal{Y}^{(i)} \rightarrow [0, 1]$ then defines the state evolution of each cluster. If the BS becomes aware of the drift with a given confidence ν_{reset} , it re-aligns the DT with reality. Individual sensors are unable to detect the drift event, as they do not have a full view of the environment.

This model is highly general, and can represent a wide array of filters and DT models, including complex predictive algorithms. The AoII $\Psi^{(i)}(k)$ of cluster i after frame k is defined as the time since the DT started drifting [12]:

$$\Psi^{(i)}(k) = k - \sup \left\{ \ell \in \{0, \dots, k\} : z^{(i)}(\ell) = 0 \right\}, \quad (1)$$

where $z^{(i)}(\ell)$ is an indicator variable which is equal to 1 if cluster i is in a drift state after frame ℓ , i.e., if $y^{(i)}(\ell) \in \mathcal{Y}_d^{(i)}$.

On the other hand, anomalies involve a single sensor, i.e., appear as abnormal measurements over a specific element of the process, can be considered as system states whose evolution is not captured by the DT model, and may be potentially harmful. We denote the subset of sensors that may observe anomalies as $\mathcal{N}_a$, with $N_a = |\mathcal{N}_a|$. We consider $\mathcal{N}_a \cup \mathcal{N}_d = \mathcal{N}$, but arbitrary subsets of nodes may be part of a DT model, observe anomalies, or both. The anomaly state is the stochastic process $\mathbf{x}(k) \in \{0, 1\}^{N_a}$; $x_n(k) = 1$ indicates that sensor n observes an anomaly in frame k . We model anomaly generation at sensor n as a time-varying Markov chain, whose transition matrix $\mathbf{M}_n(k)$ is

$$\mathbf{M}_n(k) = \begin{pmatrix} 1 - \lambda_n & \lambda_n \\ \mu_n + \xi_n(k) - \mu_n \xi_n(k) & (1 - \xi_n(k))(1 - \mu_n) \end{pmatrix}, \quad (2)$$

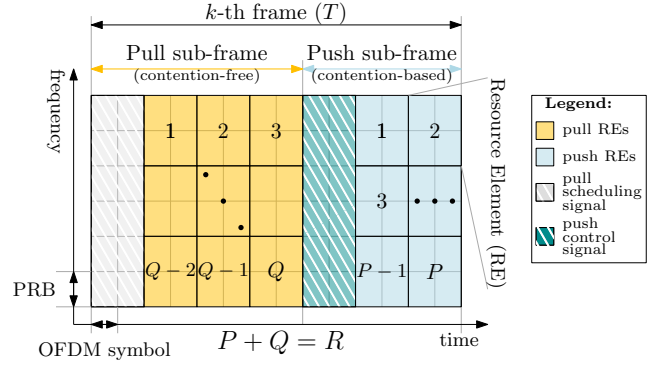


Fig. 1. Visualization of the MAC frame translated to a OFDM PHY layer, where each pull and push RE is made of 2 PRB and 2 OFDM symbols.

where λ_n is the anomaly rate in each frame and μ_n is the spontaneous resolution rate in each frame, i.e., the probability that the anomaly disappears without the need for external intervention. Finally, $\xi_n(k)$ is an indicator variable equal to 1 if sensor n successfully reports the anomaly during frame k , so that the system operator can solve the issue. Vectors $\boldsymbol{\lambda} \in \mathbb{R}_+^{N_a}$ and $\boldsymbol{\mu} \in \mathbb{R}_+^{N_a}$ collect each sensor's anomaly and resolution rates λ_n and μ_n , which are known to the BS.

Anomalies are directly identifiable by the involved sensors. We define the AoII of sensor n after frame k , $\Theta_n(k)$, as the time that node n has spent in state 1:

$$\Theta_n(k) = k - \sup \{ \ell \in \{0, \dots, k\} : x_n(\ell) = 0 \}. \quad (3)$$

B. Communication Model

The frame needs to be designed to accommodate both push- and pull-based communication modes. Accordingly, we employ an instance of the contention-free pull and contention-push (CFC-push/pull) frame [13], shown in Fig. 1. Each frame has a fixed duration of T seconds, comprises a total of R uplink (UL) REs, and is divided into pull and push sub-frames.

The pull sub-frame is divided in two: first, the BS transmits a downlink (DL) *pull scheduling signal*, composed of a synchronization signal (SS) and the (multi-cast) schedule for specific node transmissions. The scheduled nodes then transmit orthogonally over the $Q \in \{0, \dots, R\}$ pull REs [13]. In the remainder of the paper, the set of nodes scheduled in the k -th pull sub-frame is denoted as $\mathcal{S}_k \subseteq \mathcal{N}_d$, being $|\mathcal{S}_k| = Q$.

Similarly, the push sub-frame begins with a DL *push control signal*, which contains another SS and a threshold value $\theta^*(k)$: nodes whose AoII is below the threshold will remain silent to avoid collisions with higher-priority traffic [14]. Finally, the control signal includes the number $P = R - Q$ of available push REs. Accordingly, nodes that observe an anomaly decode the control signal to recover slot boundaries and the frame structure. The nodes whose AoII is over the threshold, i.e., $\Theta_n(k) \geq \theta^*(k)$, then transmit in a grant-free manner over the P push REs following a *framed slotted ALOHA (FSA)* scheme [15]: every device randomly selects one of the P REs.

We consider nodes to be able to perform power control, so that the communication channel is a pure *collision channel*.

Remark that no collision can occur during the pull sub-frame. The proposed structure is highly flexible: each RE may be comprised of an arbitrary number of OFDM symbols and PRBs depending on the PHY layer performance required, making the proposed system compatible with 3GPP standards.

III. NETWORK-AWARE DIGITAL TWIN UPDATE

DT drift tracking: As described in Sec. II-A, we consider the alignment of each DT's virtual model i as an independent HMM, over which the BS maintains the maximum a posteriori (MAP) probability distribution $\zeta^{(i)}(y; k)$. We assume the initial state of the nodes belonging to cluster i to be $y_0^{(i)}$, so the initial belief distribution is $\zeta^{(i)}(y; 0) = \mathcal{I}(y = y_0^{(i)})$, where $\mathcal{I}(\cdot)$ is the indicator function, equal to 1 if the argument is true and 0 otherwise. After each transition from frame $k-1$ to frame k , we update the *a priori* probability following the transition probability function $\mathcal{F}^{(i)}$, which is known to the BS:

$$\zeta^{(i),\text{pri}}(y; k) = \sum_{y' \in \mathcal{Y}^{(i)}} \zeta^{(i)}(y'; k-1) \mathcal{F}^{(i)}(y'; y). \quad (4)$$

Successful (pull and push) sensors transmissions are considered *observations* of the physical process. Observations from sensor n belong to set $\mathcal{O}_n$, while $\mathcal{O}^{(i)} = \prod_{n \in \mathcal{C}^{(i)}} \mathcal{O}_n$ for cluster i . The BS also knows the observation probability $\Omega: \mathcal{Y}^{(i)} \times \mathcal{O}^{(i)} \rightarrow [0, 1]$. However, sensor n may fail to report during frame k , an outcome denoted as χ , either because its packet collided or because it did not transmit; otherwise, it provides its observation $o_n \in \mathcal{O}_n$. The overall observation for sensor cluster i is then $\mathbf{o}^{(i)}(k) \in \tilde{\mathcal{O}}^{(i)} = \prod_{n \in \mathcal{C}^{(i)}} (\mathcal{O}_n \cup \chi)$. We define the set of observations compatible with the received packets as

$$\mathcal{L}(k) = \left\{ \mathbf{o} \in \mathcal{O}^{(i)} : o_n^{(i)}(k) \in \{o_n, \chi\} \forall n \in \mathcal{C}^{(i)} \right\}. \quad (5)$$

The HMM does not use the existence of a *single* transmission as a meaningful observation, as individual nodes do not know whether their DT is aligned. Instead, we provide the update rule for the MAP estimate (for discrete states and observations):

$$\zeta^{(i)}(y; k) = \frac{\zeta^{(i),\text{pri}}(y; k) \sum_{\mathbf{o} \in \mathcal{L}(k)} \Omega(y, \mathbf{o})}{\sum_{y' \in \mathcal{Y}^{(i)}} \zeta^{(i),\text{pri}}(y'; k) \sum_{\mathbf{o}' \in \mathcal{L}(k)} \Omega(y', \mathbf{o}')}, \quad (6)$$

and the drift belief is made by summing it for all the possible drifting states, i.e.,

$$\nu_k^{(i)} = \sum_{y \in \mathcal{Y}^{(i)}: z^{(i)}(k)=1} \zeta^{(i)}(y; k). \quad (7)$$

Anomaly tracking: Under a push-based threshold policy, in which nodes transmit if their AoII is higher than $\theta^*(k)$, the BS can maintain a belief $\varphi_n(\theta; k)$ over the AoII $\Theta_n(k)$ of node n after frame k . Since the system starts with no anomalies, the BS knows that the AoII is 0 for all nodes, i.e., $\varphi_n(\theta; 0) = \mathcal{I}(\theta = 0)$. The prior belief is updated after each pull sub-frame: if sensor n transmitted within it, i.e., $n \in \mathcal{S}_k$, the BS

reset its AoII to 0; otherwise, the prior is updated following eq. (2), yielding:

$$\varphi_n^{\text{pri}}(\theta; k) = \begin{cases} \mathcal{I}(\theta = 0), & n \in \mathcal{S}_k; \\ (1 - \lambda_n - \mu_n) \varphi_n(0; k-1) + \mu_n, & \theta = 0, n \notin \mathcal{S}_k; \\ \lambda_n \varphi_n(0; k-1), & \theta = 1, n \notin \mathcal{S}_k; \\ (1 - \mu_n) \varphi_n(\theta - 1; k-1), & \theta > 1, n \notin \mathcal{S}_k. \end{cases} \quad (8)$$

We then consider vector $\omega(k) \in (\mathcal{N} \cup \{-1, 0\})^S$, which contains the outcome of the push-based REs in the k -th push sub-frame, i.e., 0 if no node transmitted, -1 if a collision happened, or the index n of the node that successfully transmitted.

If node n successfully transmitted, i.e., if $n \in \omega(k)$, its AoII is reset to 0 with probability 1, and we have $\varphi_n(\theta; k+1) = \mathcal{I}(\theta = 0)$. On the other hand, if node n did not transmit, but there were no collisions in the frame, i.e., $\{-1, n\} \cap \omega(k) = \emptyset$, we know that $\Theta_n(k) \leq \theta^*(k)$, or node n would have transmitted. The belief is then updated as

$$\varphi_n(\theta; k+1) = \frac{\varphi_n^{\text{pri}}(\theta; k) \mathcal{I}(\theta \leq \theta^*(k))}{\sum_{j=0}^{\theta^*(k)} \varphi_n^{\text{pri}}(j; k)} \text{ if } \{-1, n\} \cap \omega(k) = \emptyset, \quad (9)$$

as values higher than $\theta^*(k)$ are not compatible with the outcome of the sub-frame. The belief update is more difficult if there is at least one collision, as the BS cannot know which nodes transmitted or even how many. In order to easily compute $\varphi_n(\theta; k+1)$ when $n \notin \omega(k) \wedge -1 \in \omega(k)$, i.e., if node n did not successfully transmit and there is at least one collision, we then need to make a series of approximations to apply Bayes' theorem and obtain the *a posteriori* belief for the outcome $\omega(k)$.

Under threshold $\theta^*(k)$, the estimated transmission probability for node n is

$$\alpha_n(\theta^*(k); \varphi_n^{\text{pri}}(k)) = \sum_{j=\theta^*(k)+1}^{\infty} \varphi_n^{\text{pri}}(j; k). \quad (10)$$

We also define the set of potentially active nodes $\mathcal{A}(\theta^*(k))$, i.e., those with $\alpha_n(\theta^*(k); \varphi_n^{\text{pri}}(k)) > 0$, whose cardinality is $A(\theta^*(k))$. As there is a different belief distribution for each node, their estimated activation probabilities are also different; however, we will compute the belief update in case there are any collisions in the sub-frame by assuming that all nodes have the same activation probability:

$$\bar{\alpha}(k) = \frac{\sum_{n \in \mathcal{N}_a} \alpha_n(\theta^*(k); \varphi_n^{\text{pri}}(k))}{A(\theta^*(k))}. \quad (11)$$

This approximation is usually close to reality, as the nodes belief distributions will be rather similar to one another. The probability of there being a active nodes in a push sub-frame is then approximated as

$$\Pr(a|\theta^*(k)) \simeq \binom{A(\theta^*(k))}{a} (1 - \bar{\alpha}(k))^{A(\theta^*(k))-a} (\bar{\alpha}(k))^a, \quad (12)$$

where $\binom{n}{a}$ is the binomial coefficient. The outcome of the push sub-frame includes s successful transmissions, c collisions,

and $P - s - c$ empty REs. As mentioned above, we have no information on the identities of the nodes that collided. We further assume that no collision involves more than 3 nodes, neglecting the highly unlikely cases in which many nodes collide over the same RE, in order to limit the computational complexity of the belief update. We then compute the *a posteriori* probability of a nodes being active when observing s successes and c collisions:

$$\Pr(a|s, c; \theta^*(k)) \simeq \frac{\Pr(a|\theta^*(k)) \binom{a}{s, c, c} P^a 3^{a-s-2c}}{\sum_{a'=s+2c}^A \binom{a'}{s, c, c} P^{a'} 3^{a'-s-2c}}. \quad (13)$$

We can then compute the probability of event χ_n , which indicates that node n was among the colliding nodes:

$$\Pr(\chi_n|s, c; \theta^*(k)) \simeq \sum_{a=s+2c}^{A(\theta^*(k))} \frac{(a-s) \Pr(a|s, c; \theta^*(k))}{A(\theta^*(k)) - s}, \quad (14)$$

which is the same for each $n \in \mathcal{A}(\theta^*(k))$.

We make a further approximation, considering belief updates over individual nodes, rather than over the joint probability distribution. As we are generally dealing with rare events (only a few nodes are colliders, while most will not transmit), the error caused by the approximation is also usually negligible in practice. We can then provide the *a posteriori* belief over the AoII of each node by applying Bayes' theorem:

$$\varphi_n(\theta; k+1) = \begin{cases} \frac{\varphi_n^{\text{pri}}(\theta; k)(1 - \Pr(\chi_n|s, c; \theta^*(k)))}{\sum_{j=0}^{\theta^*(k)} \varphi_n^{\text{pri}}(j; k)}, & \theta \leq \theta^*(k); \\ \frac{\varphi_n^{\text{pri}}(\theta; k) \Pr(\chi_n|s, c; \theta^*(k))}{\sum_{j=\theta^*(k)+1}^{\infty} \varphi_n^{\text{pri}}(j; k)}, & \theta > \theta^*(k). \end{cases} \quad (15)$$

We can thus update all nodes that did not transmit successfully in a push sub-frame with at least one collision. The belief update operation is computationally light for the BS if implemented intelligently, as it involves $O(N\theta_{\max})$ operations, where $\theta_{\max}$ is the maximum AoII considered in the system.

IV. JOINT PUSH-PULL SCHEDULING

In the pull sub-frame, we aim at maximizing the information gain. In order to do this, we consider the set $\mathcal{S}^{(i)}$ of scheduled nodes for cluster i . The set of possible observations for $\mathcal{S}^{(i)}$ is then $\mathcal{M}(\mathcal{S}^{(i)}) = \{\mathbf{o}^{(i)}(k) \in \tilde{\mathcal{O}}^{(i)} : o_n^{(i)}(k) = \chi \forall n \notin \mathcal{S}^{(i)}\}$. We define the *a posteriori* entropy of the drift risk when getting observation $\mathbf{o}$ as

$$h(\mathbf{o}; k) = H \left(\sum_{y \in \mathcal{Y}^{(i)}(\mathbf{o}; z^{(i)}(k)=1)} \frac{\zeta^{(i), \text{pri}}(y; k)}{\eta(\mathbf{o}; k)} \right), \quad (16)$$

where $H(p)$ is the binary Shannon entropy and $\eta(\mathbf{o}; k) = \sum_{y \in \mathcal{Y}^{(i)}(\mathbf{o}; k)} \zeta^{(i), \text{pri}}(y; k)$ is the belief that the scheduled nodes will report $\mathbf{o}$. We then perform a look-ahead step, considering the information gain when scheduling node n :

$$G(n|\mathcal{S}^{(i)}) = \sum_{\mathbf{o} \in \mathcal{M}(\mathcal{S}^{(i)})} \eta(\mathbf{o}; k) h(\mathbf{o}; k) - \sum_{\mathbf{o}' \in \mathcal{M}(\mathcal{S}^{(i)} \cup \{n\})} \eta(\mathbf{o}'; k) h(\mathbf{o}'; k). \quad (17)$$

We can then sort nodes by their information gain, scheduling them iteratively: for each RE, we schedule the node with the highest information gain, then recompute $h(\mathbf{o}, k)$. We repeat this step until all Q REs in the pull sub-frame are assigned.

On the other hand, the aim for the push sub-frame is to allow nodes with a high AoII (relative to the rest of the network) to transmit as quickly as possible, while avoiding collisions. Collisions are more harmful than in standard FSA, as they have the additional effect of reducing the information that the BS can use to improve its estimate of nodes' AoII. Therefore, we set a maximum acceptable collision rate σ , while the highest possible AoII across the network that matches with the information available to the DT is defined as $\theta_{\max}(k) = \max_{n \in \mathcal{N}_a} \sup \{\theta : \varphi_n^{\text{pri}}(\theta; k) > 0\}$. We then compute the collision probability $\Pr(\chi|\theta^*)$ with threshold θ^* :

$$\Pr(\chi|\theta^*) = \sum_{a=2}^{A(\theta^*)} \left(1 - \frac{P!}{P^a(P-a)!} \right) \Pr(a|\theta^*). \quad (18)$$

Set $\mathcal{V}(k) = \{\theta^* \in \{0, \dots, \theta_{\max}(k) - 2\} : \Pr(\chi|\theta^*) \leq \sigma\}$ is the set of possible thresholds that have a collision probability below σ . The transmission threshold is then

$$\theta^*(k) = \begin{cases} \sup(\mathcal{V}(k)), & \text{if } \mathcal{V}(k) \neq \emptyset; \\ \max(0, \theta_{\max}(k) - 2), & \text{otherwise.} \end{cases} \quad (19)$$

V. NUMERICAL EVALUATION

In order to evaluate PPS, we performed Monte Carlo simulations,¹ running $N_{\text{ep}} = 100$ episodes with $K_{\text{ep}} = 1000$ frames each, considering the AoII as a metric for both DT drift and anomaly tracking. We report results for pull sub-frames as a function of the drift rate $\rho_d = \left(T \mathbb{E} \left[\inf \left(t : z^{(i)}(k) = 1 | y^{(i)}(0) = y_0^{(i)} \right) \right] \right)^{-1}$, corresponding to the frequency of drift events for each cluster. As drift is solved when $\nu^{(i)}(k) > \nu_{\text{reset}}$, this corresponds to the maximum possible rate of drift events, i.e., to the expected absorption time of the cluster Markov chain when starting from $y_0^{(i)}$, considering all drift states as absorbing. The frequency of anomalies at each node is upper bounded by the anomaly rate $\rho_a = (|\mathcal{N}_a|T)^{-1} \sum_{n \in \mathcal{N}_a} \lambda_n$.

For the sake of simplicity, we consider clusters of a fixed size, i.e., $|\mathcal{C}^{(i)}| = C \forall i$. We consider each sensor to have a binary state and transition independently, so the drift state of the system is the row vector $\mathbf{y}^{(i)}(k) \in \mathcal{Y}^{(i)} = \{0, 1\}^C$. We consider the DT to be drifting if a majority of nodes in the cluster is in state 1, i.e., $\mathcal{Y}_d^{(i)} = \{\mathbf{y} \in \{0, 1\}^C : \sum_{n \in \mathcal{C}^{(i)}} y_n \geq \frac{C}{2}\}$. Once the system is in a drift state, nodes in state 1 will remain in that state until the drift is reported. The transition probability is then defined by matrix $\mathbf{U}^{(i)}$, which can be computed from the individual nodes' transition probabilities. Additionally, since measurements are error-free and binary, the set of states that are compatible with observation $\mathbf{o}^{(i)}(k)$ becomes $\mathcal{L}(k) = \{\mathbf{y} \in \{0, 1\}^C : o_n(k) \in \{y_n, \chi\} \forall n \in \mathcal{C}^{(i)}\}$, and we have $\Omega(y, \mathbf{o}) = 1$ for all compatible observations.

¹The full simulation code used to generate the results is publicly available at <https://github.com/signetlabdei/push-pull-anomaly-tracking/>.

In the anomaly detection task, we consider anomalies not to resolve spontaneously, i.e., $\mu_n = 0 \forall n \in \mathcal{N}_a$. Additionally, we consider $N_a = 100$, with $\mathcal{N}_d \subset \mathcal{N}_a$, i.e., there are more nodes reporting anomalies than those involved in DT, but all DT sensors may also report anomalies. The relevant parameters for all simulations are reported in Table I.

A. Benchmark Algorithms

In the following, we consider two benchmark algorithms for DT drift: first, we use the common maximum age first (MAF) [16] benchmark, which simply selects the nodes with the highest age of information (AoI) in each frame. On the other hand, the Cluster Risk Aware (CRA) scheme attempts to minimize the risk of drift, exploiting $\nu_k^{(i)}(k)$. To ensure quick drift detection, CRA sorts clusters by their risk $\nu_k^{(i)}$ in decreasing order. All nodes in the highest-risk unscheduled cluster are then scheduled if at least C of the Q REs in the pull sub-frame are still free. Otherwise, the remaining REs are filled by random nodes from the next unscheduled cluster.

We also consider three state-of-the-art random access algorithms for push sub-frames: in addition to MAF, we select FSA and adaptive framed slotted ALOHA (AFSA). MAF is often used as a benchmark for random access policies, and can perform well for anomaly detection under heavy loads, as it is able to schedule packets orthogonally. In standard FSA [15], nodes with an anomaly transmit with probability p_{tx} until the anomaly is resolved. We tune the probability $p_{tx}(k) = (N_a \rho_a)^{-1} T P G_{FSA}$ to obtain a target load G_{FSA} under anomaly rate ρ_a . Instead, AFSA is a protocol inspired by the adaptive FSA literature [6] that follows the rate adaptation procedure $p_{tx}(k) = p_{tx}(k-1) + P^{-1} \gamma (N_\chi - N_{sil})$, where γ is a parameter controlling the rate variability, and N_χ and N_{sil} represent the number of REs in the past frame that resulted in a collision and no transmission, respectively. AFSA increases the rate when REs are unused and backs off after collisions. Finally, to avoid starvation, we limit its range by setting a minimum load G_{min} and a maximum load G_{max} .

B. Performance: DT Drift

We first analyze the performance with respect to the DT alignment objective. Performance depends not just on the overall drift rate ρ_d , but also on the transition matrix of each cluster. We consider $D = 20$ clusters with $C = 4$, and define a homogeneous and a heterogeneous scenario. In the former, all

nodes have the same probability $u_n(0, 1)$ of moving from state 0 to state 1, while in the latter, the probabilities for the sensors in a cluster are scaled according to vector $(1, 7, 7.25, 7.5)$, so, e.g., $u_4(0, 1) = 7.5u_1(0, 1)$. The probability of a sensor remaining in state 1 if the DT is still aligned is $u_n(1, 1) = 0.9$ for all clusters. Matrix $\mathbf{U}$ can be easily computed from these values. We then normalize $\mathbf{u}(0, 1)$ to match the target rate ρ_d .

We first consider performance in the heterogeneous scenario as a function of the drift rate ρ_d , with a fixed $Q = 10$: Fig. 2 shows the average and worst-case AoII. PPS's mean AoII gain increases with the load, as Fig. 2a shows: while CRA has no gain over MAF, PPS can select the nodes to poll more efficiently, reducing the average AoII by about 30%. Due to quantization, the worst-case performance gain is not constant, as shown by Fig. 2b: in this case, the gain over MAF is 10 ms, i.e., one frame, except for a few settings. In general, PPS never performs worse than the benchmark algorithms, and performs much better in several conditions.

Fig. 3 shows the performance of PPS in a purely pull-based system, i.e., $Q = R$, setting $\rho_d = 3$ Hz. Let us consider homogeneous clusters. In terms of the average drift AoII, shown in Fig. 3a, PPS outperforms MAF for smaller values of Q , aside from the extreme with $Q = 5$, while the gain is negligible for the other extreme, $Q = 15$. On the other hand, CRA tends to underperform for lower values of Q , but not for $Q \geq 10$. MAF and PPS have a similar performance if we consider the 99th percentile AoII, shown in Fig. 3b, while CRA is slightly worse. PPS has a larger advantage in the heterogeneous cluster scenario: its average AoII in Fig. 3a is almost 40% lower than MAF's and 50% lower than CRA's if Q is small. While the gain diminishes for larger values of Q , remaining over 20% for $Q = 15$. In this case, CRA performs as well as MAF for $Q \geq 10$. This general trend is confirmed for worst-case performance in Fig. 3b, in which PPS reduces the 99th percentile AoII by 10 ms over MAF and CRA.

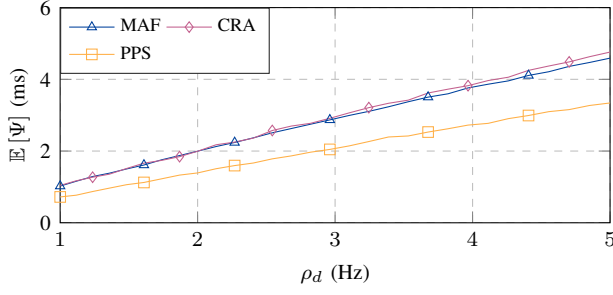
C. Performance: Anomaly Tracking

We then evaluate the anomaly tracking performance of PPS. Fig. 4 shows the performance of PPS as a function of the anomaly rate ρ_a with $P = 10$. If the load is very low, random access mechanisms such as FSA perform well, since collisions are extremely rare. We note that PPS performs as well as FSA in these conditions, for both the average and 99th percentile AoII, Θ_{99} . However, FSA is highly unstable at higher loads, while PPS degrades much more gracefully as the load increases, and while MAF remains the best solution for very high rates, PPS is better for rates up to 4 Hz.

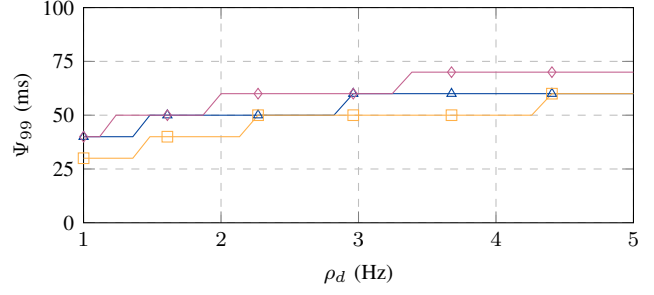
We can then consider the AoII for different values of P , shown in Fig. 5, when $\rho_a = 3$ Hz: a higher number of REs in the push sub-frame reduces the overall load, making the scenario easier for random access schemes. Fig. 5a shows the average AoII: PPS is the best-performing algorithm for $P \geq 7$, with a much lower AoII than the other random access algorithms when wireless resources are scarce. PPS's AoII is below 1 ms with $P \geq 10$, while FSA reaches that threshold only with $P \geq 12$, using 20% more resources. On the other

TABLE I
MAIN SIMULATION PARAMETERS.

Param.	Meaning	Value	Param.	Meaning	Value
R	REs per frame	20	T	Frame duration	10 ms
N_d	DT sensor nodes	80	N_a	Anomaly sensor nodes	100
$\mathcal{N}_a \cap \mathcal{N}_d$	Sensors	N_d	ν_{reset}	DT re-alignment thr.	0.95
ρ_d	DT drift rate	[1, 5] Hz	C	Nodes per DT cluster	4
ρ_a	Anomaly rate	[1, 5] Hz	μ	Anomaly resolution rate	0
σ	PPS collision thr.	0.2	G_{FSA}	FSA max load	0.9
γ	AFSA discount rate	0.1	G_{min}	AFSA min load	0.2
G_{max}	AFSA max load	1	N_{ep}	Monte Carlo episodes	100
K_{ep}	Episode duration	10 s			

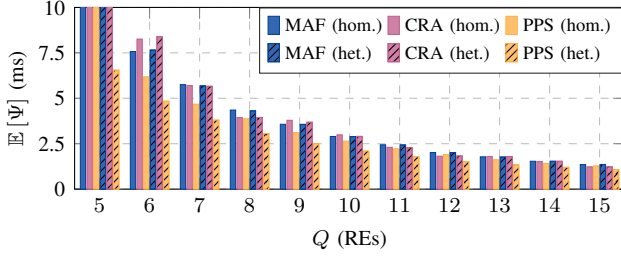


(a) Average drift AoII $\mathbb{E}[\Psi]$.

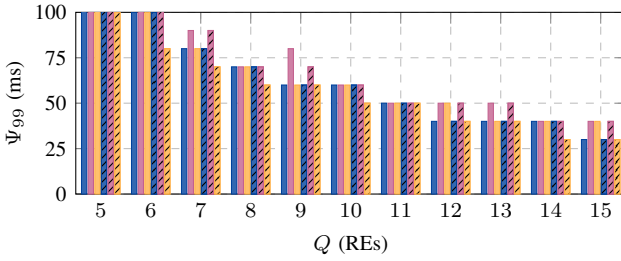


(b) 99th percentile drift AoII Ψ_{99} .

Fig. 2. Average and worst-case AoII for DT drift as a function of the drift rate ρ_d under a purely pull-based system with $Q = 10$.



(a) Average drift AoII $\mathbb{E}[\Psi]$.



(b) 99th percentile drift AoII Ψ_{99} .

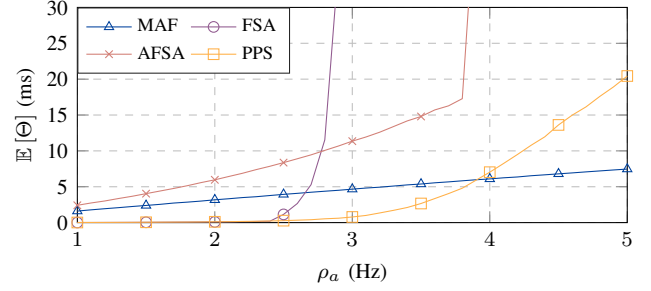
Fig. 3. Average and worst-case AoII for DT drift as a function of Q under a purely pull-based system with $\rho_d = 3$ Hz, considering homogeneous and heterogeneous clusters.

hand, MAF performs better than PPS when $P \leq 6$, but the gains from additional resources are very limited, as its pull-based nature cannot consider the actual observed anomalies.

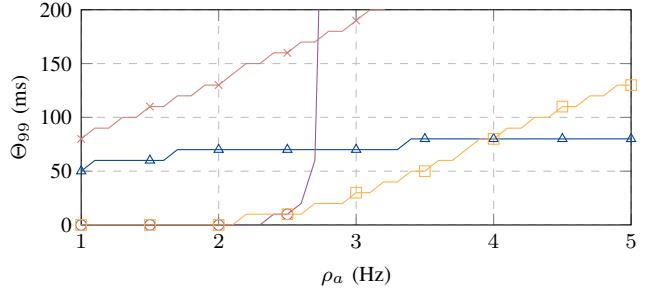
The limited gains are also noticeable in Fig. 5b, which shows the 99th percentile of the AoII of anomalies: while PPS manages to quickly reduce the worst-case AoII with more available resources, MAF has an unimpressive worst-case performance when there are plenty of transmission opportunities, while still outperforming. FSA and AFSA.

D. Performance: Joint DT Alignment and Anomaly Tracking

We then evaluate the joint performance of DT alignment and anomaly tracking by comparing PPS to the benchmarks when $\rho_d = \rho_a = 3$ Hz. Fig. 6 shows the performance of all combinations of schemes for the two sub-frames, using the best values of P and Q obtained by exhaustive search. Label MAF-MAF indicates that MAF was used in both sub-frames, MAF-FSA means MAF for pull and FSA for push, and so on.



(a) Average anomaly AoII $\mathbb{E}[\Theta]$.

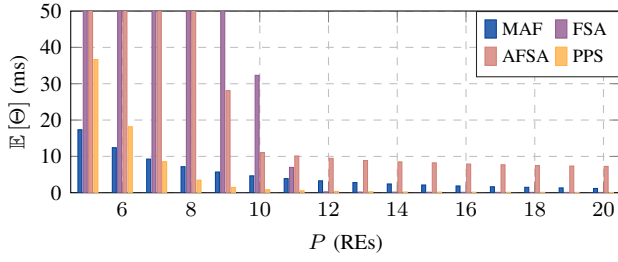


(b) 99th percentile anomaly AoII Θ_{99} .

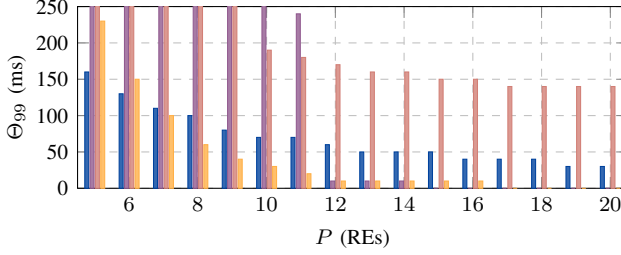
Fig. 4. Average and worst-case anomaly AoII as a function of ρ_a under a purely push-based system, with $P = 10$.

We evaluate the dual optimization of DT drift and anomaly tracking by setting a threshold for one as a constraint and evaluating the other. Fig. 6a shows the average DT drift AoII when setting the 99th percentile of the anomaly AoII to be at most 30 ms. PPS-based schemes are the only ones to achieve $\mathbb{E}[\Psi] \leq 1$ ms, gaining approximately 20% over CRA-MAF, the best combination of state-of-the-art schemes.

We then set $\mathbb{E}[\Psi] \leq 1$ ms as a system constraint, evaluating the anomaly AoII: in this case, PPS is even better, as it achieves $\Theta_{99} = 30$ ms, while state-of-the-art combinations all have at least 70 ms. CRA-PPS and PPS-MAF are closer than existing schemes, so even using PPS for just one of the two sub-frames may lead to significant gains.

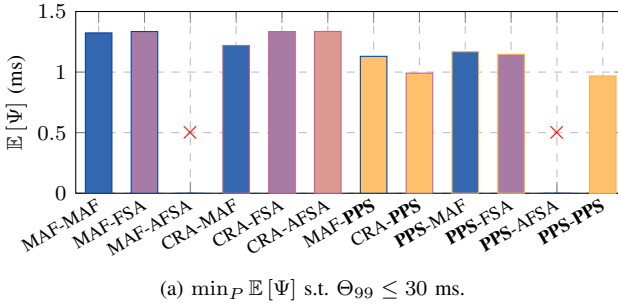


(a) Average anomaly AoII $E[\Theta]$.

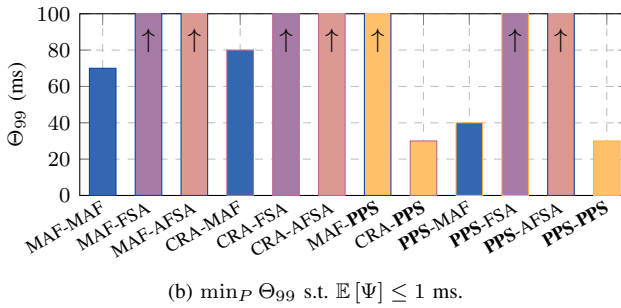


(b) 99th percentile anomaly AoII Θ_{99} .

Fig. 5. Average and worst-case anomaly AoII as a function of P under a purely push-based system.



(a) $\min_P E[\Psi]$ s.t. $\Theta_{99} \leq 30$ ms.



(b) $\min_P \Theta_{99}$ s.t. $E[\Psi] \leq 1$ ms.

Fig. 6. Joint objective performance, under the optimal P and Q and $\rho_d = \rho_a = 3$ Hz. Symbol $\times$ means that no value meets the constraints.

VI. CONCLUSIONS

In this work, we presented PPS, a scheme that can exploit frame subdivision to accommodate both push- and pull-based traffic and improve the performance of DTs. We targeted a dual objective system in which the DT alignment task is complicated by the presence of anomalies, presenting a solution that can significantly improve performance: a reduction of the mean DT drift of 20% under the same anomaly detection guarantees, or a reduction from 70 ms to 20 ms of worst-case

anomaly detection AoII under the same DT drift constraint.

This is a first step towards the push/pull integration, and future extensions will need to consider more realistic settings, as well as dynamic allocation of pull and push sub-frames resources, to meet this challenge.

REFERENCES

- [1] L. Wright and S. Davidson, "How to tell the difference between a model and a digital twin," *Adv. Modeling Sim. in Eng. Sci.*, vol. 7, no. 1, p. 13, 2020.
- [2] B. Qin, H. Pan, Y. Dai *et al.*, "Machine and deep learning for digital twin networks: A survey," *IEEE Internet Things J.*, vol. 11, no. 21, pp. 34 694–34 716, 2024.
- [3] A. De Benedictis, F. Flammini, N. Mazzocca *et al.*, "Digital twins for anomaly detection in the industrial internet of things: Conceptual architecture and proof-of-concept," *IEEE Trans. Ind. Informat.*, vol. 19, no. 12, pp. 11 553–11 563, 2023.
- [4] L. F. Rivera, M. Jiménez, G. Tamura, N. M. Villegas, and H. A. Müller, "Designing run-time evolution for dependable and resilient cyber-physical systems using digital twins," *J. Integr. Des. Process Sci.*, vol. 25, no. 2, pp. 48–79, 2022.
- [5] M. E. Ildiz, O. T. Yavascan, E. Uysal, and O. T. Kartal, "Pull or wait: How to optimize query age of information," *IEEE J. Sel. Areas Inf. Theory*, vol. 4, pp. 794–807, 2023.
- [6] J. Wang, R. Zhang, J. Yu *et al.*, "Age-efficient random access with load adaptation," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 15 210–15 223, 2024.
- [7] F. Chiariotti, A. Munari, L. Badia, and P. Popovski, "Goal-oriented medium access with distributed belief processing," *IEEE Transactions on Networking*, vol. 34, pp. 1643–1658, 2026.
- [8] I. Cosandal, N. Akar, and S. Ulukus, "Multi-threshold AoII-optimum sampling policies for continuous-time Markov chain information sources," *IEEE Trans. Inf. Theory*, vol. 71, no. 9, pp. 6968–6988, 2025.
- [9] P. Talli, E. D. Santi, F. Chiariotti *et al.*, "Pragmatic communication for remote control of finite-state markov processes," *IEEE J. Sel. Areas Commun.*, vol. 43, no. 7, pp. 2589–2603, 2025.
- [10] P. Agheli, N. Pappas, P. Popovski, and M. Kountouris, "Integrated push-and-pull update model for goal-oriented effective communication," *IEEE Trans. Commun.*, vol. 73, no. 11, pp. 10 914–10 928, 2025.
- [11] S. Cavallero, F. Saggese, J. Shiraishi *et al.*, "Coexistence of pull and push communication in wireless access for IoT devices," in *Proc. 25th Int. Wkshp. Signal Proc. Adv. Wireless Commun. (SPAWC)*. IEEE, 2024, pp. 841–845.
- [12] A. Maatouk, M. Assaad, and A. Ephremides, "The age of incorrect information: An enabler of semantics-empowered communication," *IEEE Trans. Wireless Commun.*, vol. 22, no. 4, pp. 2621–2635, 2023.
- [13] S. R. Pandey, F. Saggese, J. Shiraishi *et al.*, "Medium access for push-pull data transmission in 6G wireless systems," *accepted in IEEE Commun. Mag.*, *arXiv Preprint*, no. 2504.02637, 2025.
- [14] S. Kriouile and M. Assaad, "Minimizing the age of incorrect information for real-time tracking of Markov remote sources," in *Proc. Int. Symp. Inf. Theory (ISIT)*. IEEE, 2021, pp. 2978–2983.
- [15] Z. Yue, H. H. Yang, M. Zhang, and N. Pappas, "Age of information under frame slotted ALOHA-based status updating protocol," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 7, pp. 2071–2089, 2023.
- [16] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Age-optimal information updates in multihop networks," in *IEEE Int. Symp. Inf. Theory (ISIT)*. IEEE, 2017, pp. 576–580.