AWARE: Resilient Wide-Area Byzantine Consensus Using Adaptive Weighted Replication

Christian Berger, Hans P. Reiser
University of Passau, Germany

João Sousa, Alysson Bessani
LASIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal

Problem

1. **In geo-replicated systems**, the heterogeneous latencies of connections between replicas limit the system's ability to achieve fast consensus

➡ WHEAT uses Δ additional spare replicas and **weighted replication** to **faster make progress** by accessing a proportionally smaller quorum of replicas

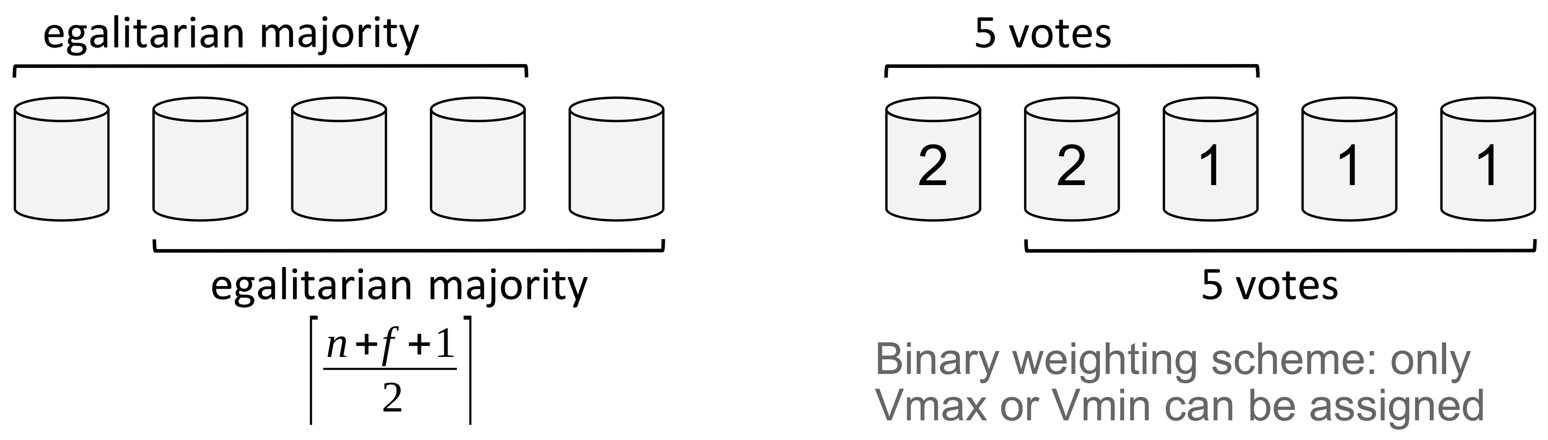
2. However, the benefit of weighted replication depends on choosing an optimal weight configuration (a non-trivial problem!)
3. The **environment of the SMR system** (i.e network characteristics) may change at runtime and thus the optimal configuration may also change

➡ AWARE enables geo-replicated systems to **adapt to their environment**

Weighted Replication

1) Egalitarian majority

2) Weighted (from *2f + 1* to *n - f* replicas)



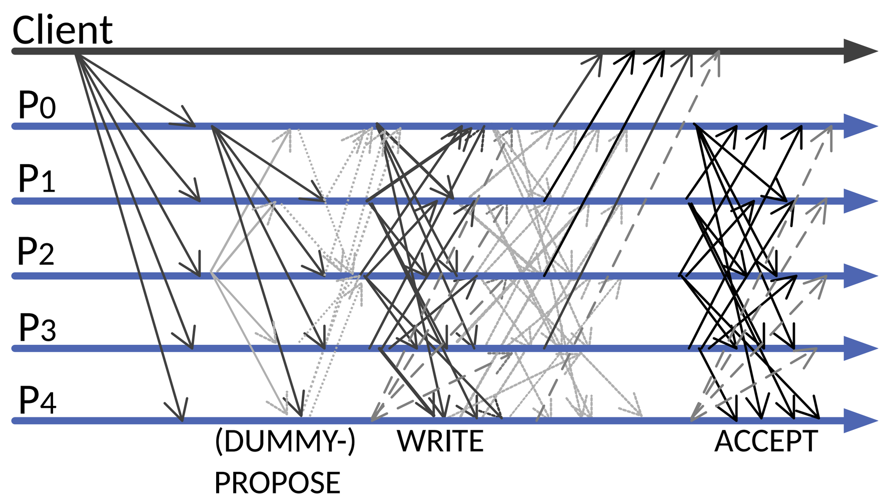
Binary weighting scheme: only Vmax or Vmin can be assigned

★ Weighted replication is **safe** and does **not violate the resilience bound *f***

Monitoring

★ AWARE uses **reliable self-monitoring as decision-making basis** for adapting replicas' voting weights or leader position at runtime

1. **We measure the Propose latency of non-leaders:** periodically, an alternately selected dummy leader broadcasts a dummy proposal
2. **We measure the Write latency:** replicas immediately respond by sending acknowledgments



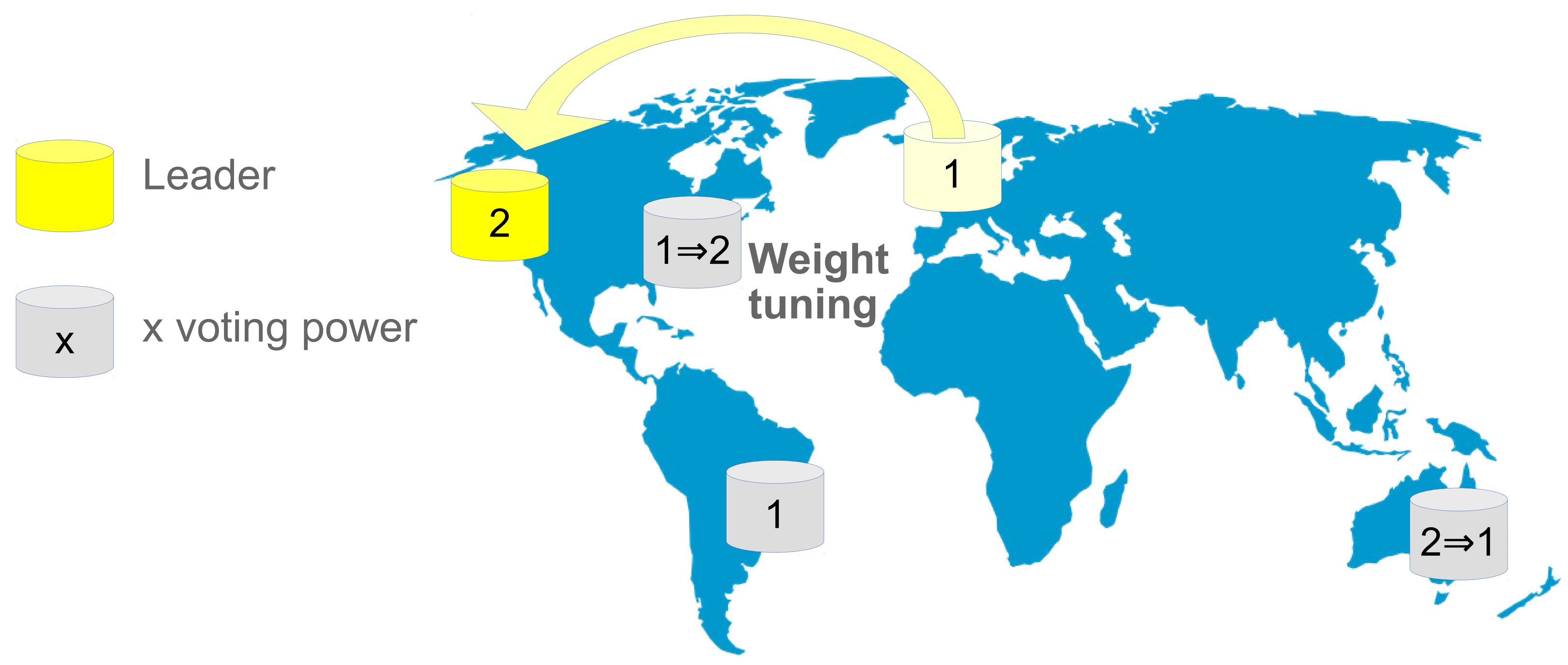
★ Replicas periodically **disseminate their measurements with total order,** thus maintain the same latency matrix after some specific consensus instance

➡ AWARE maintains these **synchronized** matrices for both Propose and Write latencies $\hat{M}^P$ and $\hat{M}^W$ used for decisions later

Self-Optimization

★ AWARE continuously strives for latency gains at runtime. We **optimize voting weights** and **leader position** to **minimize consensus latency**

Leader relocation



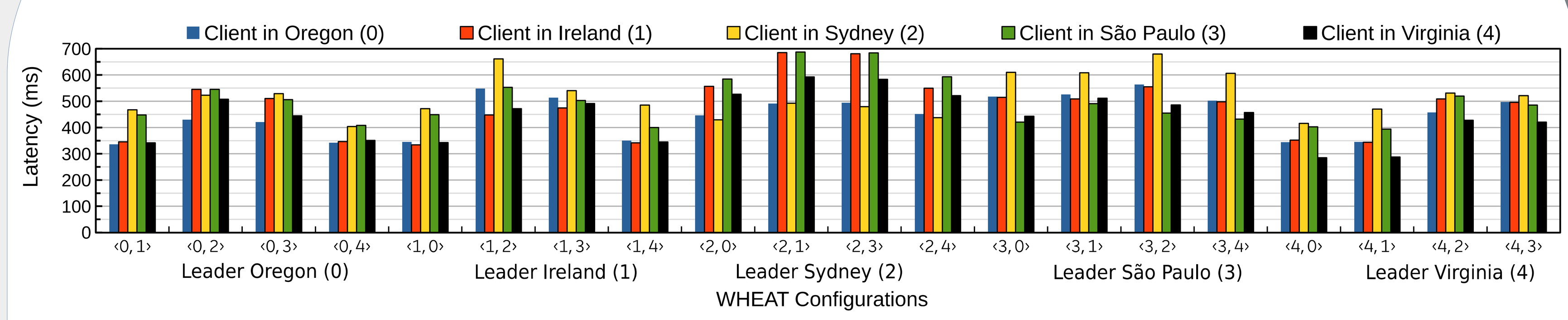
★ After specific consensus instances, replicas **deterministically solve** an optimization problem: *PredictLatency* is a function to predict the latency of the consensus protocol using the measured latencies in $\hat{M}^P$, $\hat{M}^W$ and all possible weight distributions W ∈ **W** and permitted leaders l ∈ **L**:

$$\langle \hat{l}, \hat{W} \rangle = \underset{W \in \mathfrak{W}, l \in \mathfrak{L}}{\arg\min} \; PredictLatency(l, W, \hat{M}^P, \hat{M}^W)$$

➡ Replicas **safely reconfigure** to a new weight or leader configuration if it minimizes the system's consensus latency

Evaluation

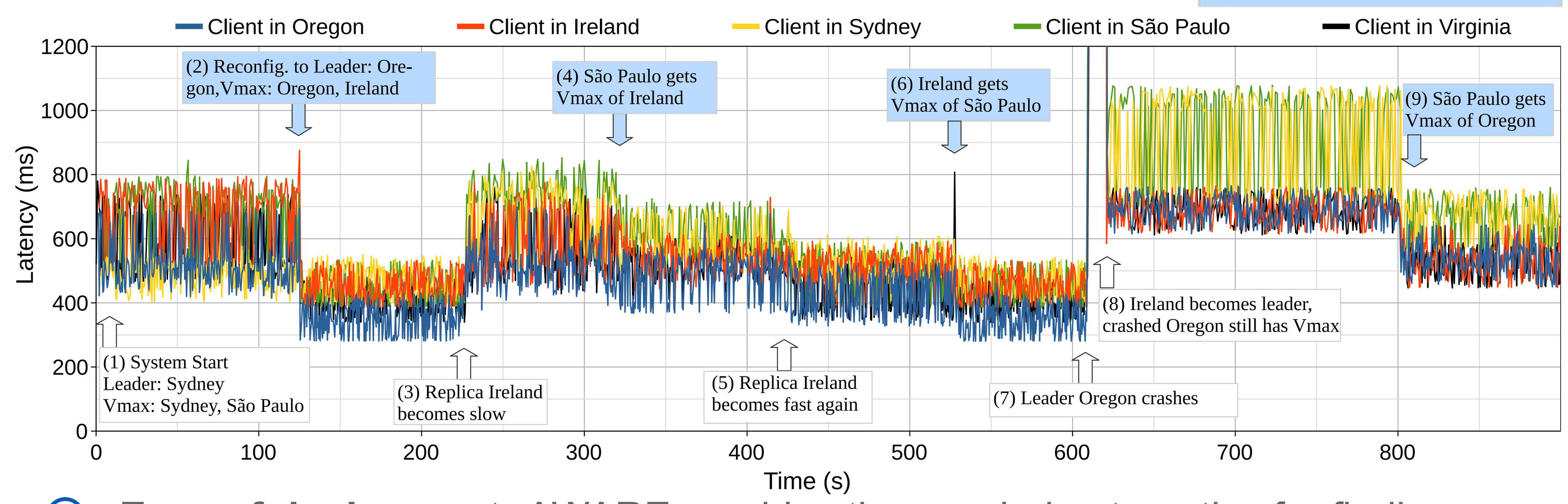
★ **Measured average request latency of 11th to 90th percentile across clients in different regions**



Configuration <L, R> means L is the leader and R is the other replica (besides the leader) with a voting weight of Vmax = 2

1. **Tuning voting weights** can reduce latency (compare configs with same leader)
2. **Leader relocation** may be necessary for achieving optimal consensus latency
3. **A global optimum does not exist** but a few pareto-optimal configurations dominate poorer performing configurations

★ **Runtime behavior of AWARE**



1. **Ease of deployment:** AWARE provides the *needed automation* for finding an optimal configuration by tuning voting weights and/or relocating the leader
2. **Adjusting to varying network conditions:** if the quality of communication links varies, AWARE *dynamically adjusts to new conditions* by shifting high voting power to replicas that are the fastest in a recent time frame
3. **Compensating for faults:** even if f replicas with high voting power become unavailable and restrict quorum variability, for non-malicious behavior, AWARE detects this and restores the availability of up to f (Vmax − Vmin) voting power by redistributing high voting weights

Conclusions

★ World-spanning Byzantine consensus systems can benefit from dynamic self-optimizing approaches in combination with weighted replication

★ AWARE enriches the idea of weighted replication by providing the needed automation to adapt to changing environmental conditions

★ Evaluation results show that the potential for latency and throughput gains is substantial. Specifically, the best configuration performs about 38.7% faster on average in terms of observed latency across clients' sites than the median

UNIVERSITÄT PASSAU

OptSCORE funded by Deutsche Forschungsgemeinschaft DFG

This work was partially supported through projects
- OptSCORE2 (RE3490/2-2) by DFG
- IRCoC (PTDC/EEISCR/6970/2014) by FCT
- Abyss (PTDC/EEI-SCR/1741/2041) by FCT

and LASIGE Research Unit (UID/CEC/00408/2019)

U LISBOA UNIVERSIDADE DE LISBOA

Ciências ULisboa

FCT Fundação para a Ciência e a Tecnologia