Byzantine Fault Tolerance Models for Distributed Coordination in Dynamic Spectrum Sharing

Amy Babay*, Prashant Krishnamurthy*, Ilia Murtazashvili◊, Xiaoxuan Qin*

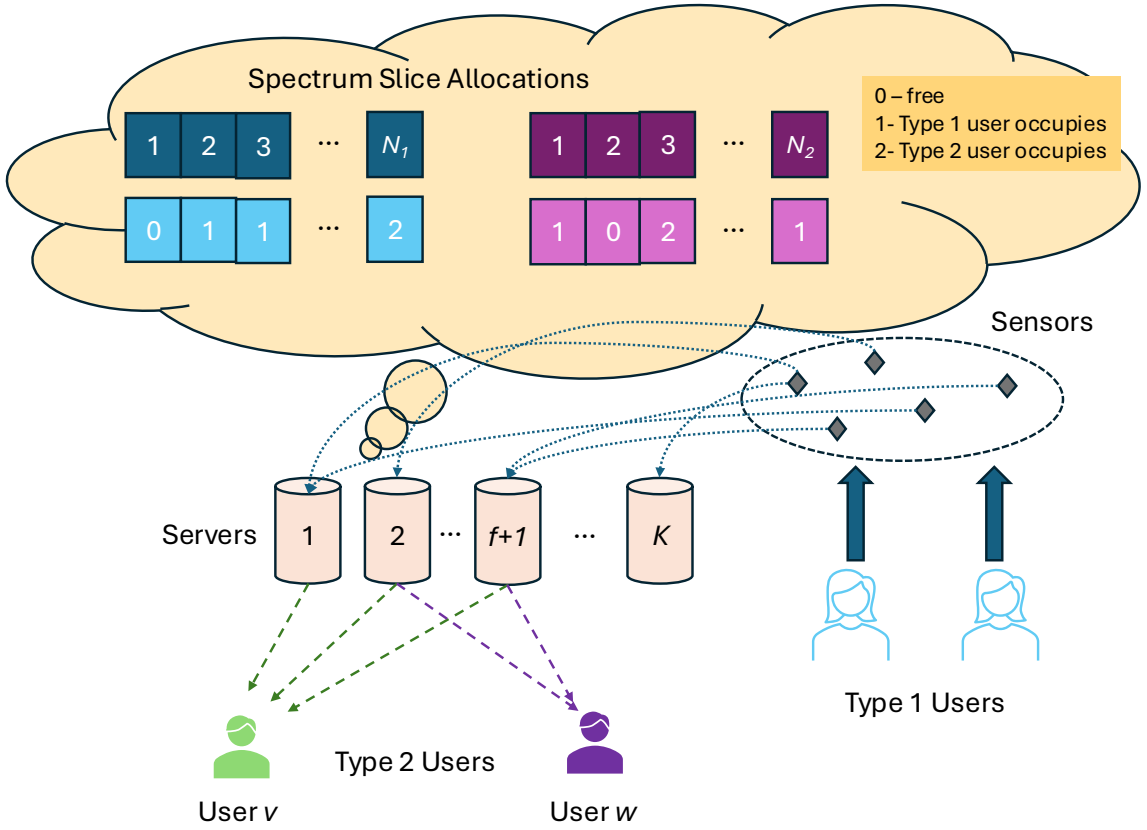
*Informatics and Networked Systems, ◊Public and International Affairs & Center for Governance and Markets
University of Pittsburgh

University of Pittsburgh

Dynamic Spectrum Sharing

We assume:

- 2 tier system
  - Type 1 users have priority; transmissions detected by sensors
  - Type 2 users issue requests for spectrum
- Servers maintain decentralized database of current transmissions and spectrum grants; make grant decisions

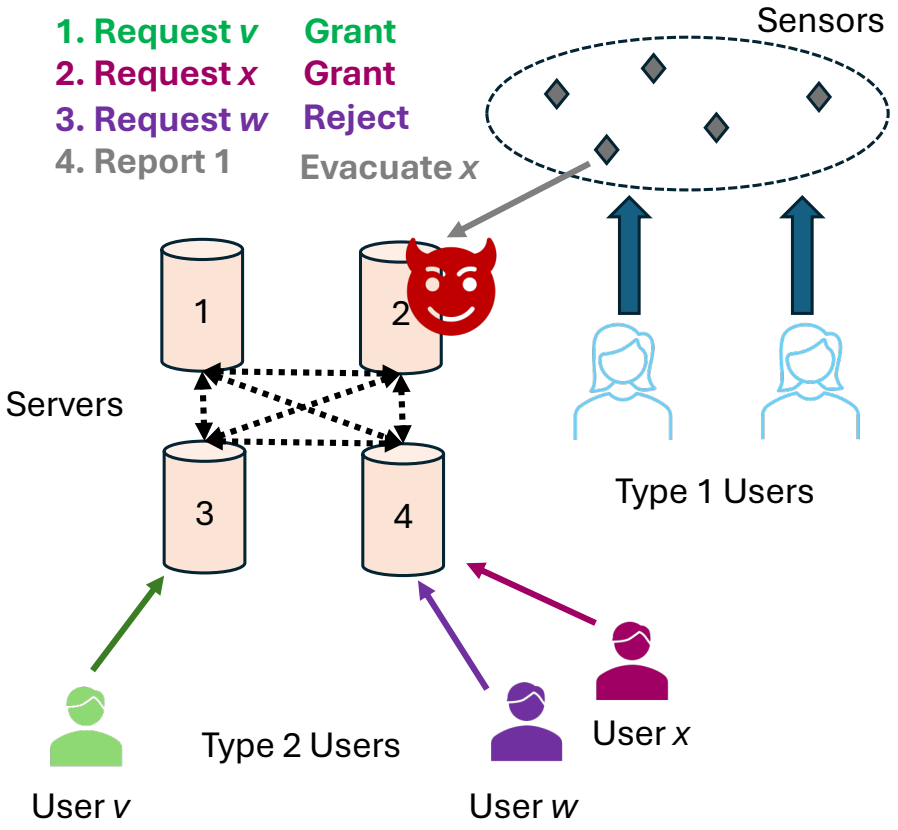


Challenges of Spectrum Sharing with Decentralized Coordination

- Technical Challenges:
  - How can we make spectrum allocation decisions that efficiently and fairly utilize the available spectrum in an environment where servers/sensors/users may:
    - Belong to different (potentially competing) entities
    - Experience failures (crashes, measurement errors, message loss, or even security breaches leading to compromises)
    - Have differing observations of physical properties (e.g. interference levels)
  - How can we ensure high performance of the coordination system?
    - Low latency for decision making, high throughput for serving requests
- Institutional Challenges:
  - Who is responsible for deploying servers and sensors?
  - Who determines the policies for making allocation decisions?
- Logistical Challenges:
  - What is the total size / cost / management overhead of such a system?

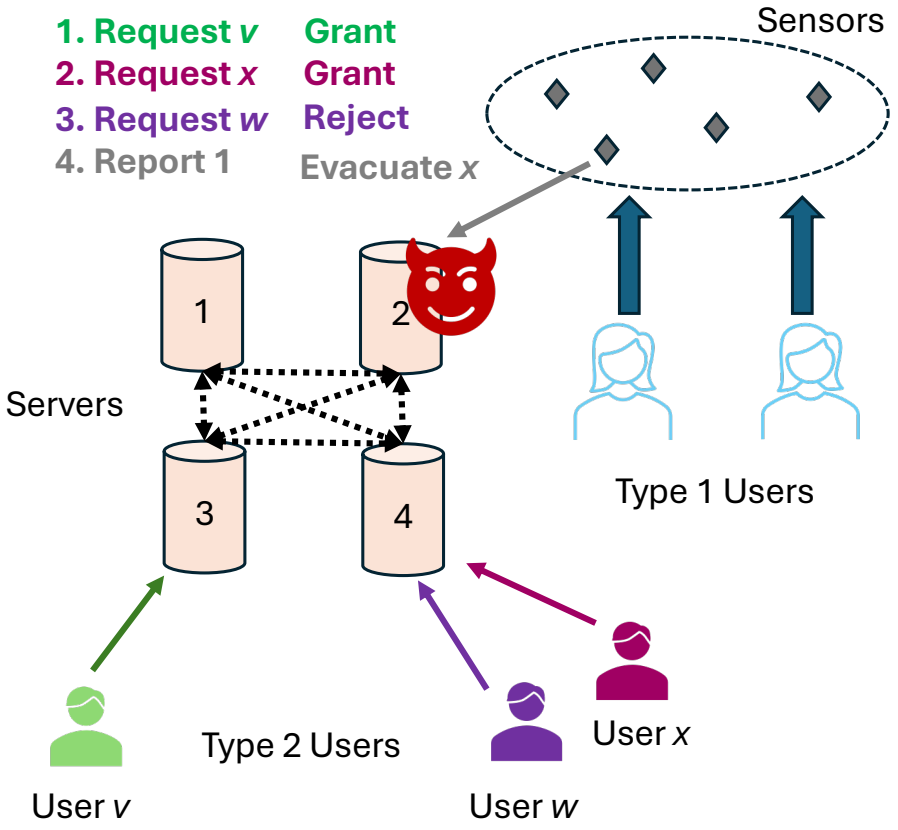
Tolerating Server Misbehavior with BFT: Sketch

- BFT State Machine Replication (SMR)
  - All servers receive all **user requests** and **sensor reports**
  - Servers use BFT SMR to agree on the **order** in which to process requests/reports
  - Upon processing a request:
    - Servers generate ***grant*** or ***reject*** response; user may begin transmitting upon receiving ***f+1* grant responses** from different servers
  - Upon processing a sensor report:
    - Servers generate ***evacuate*** command for any interfering Type 2 users; users must stop transmitting upon receiving ***f+1* evacuate commands** from different servers



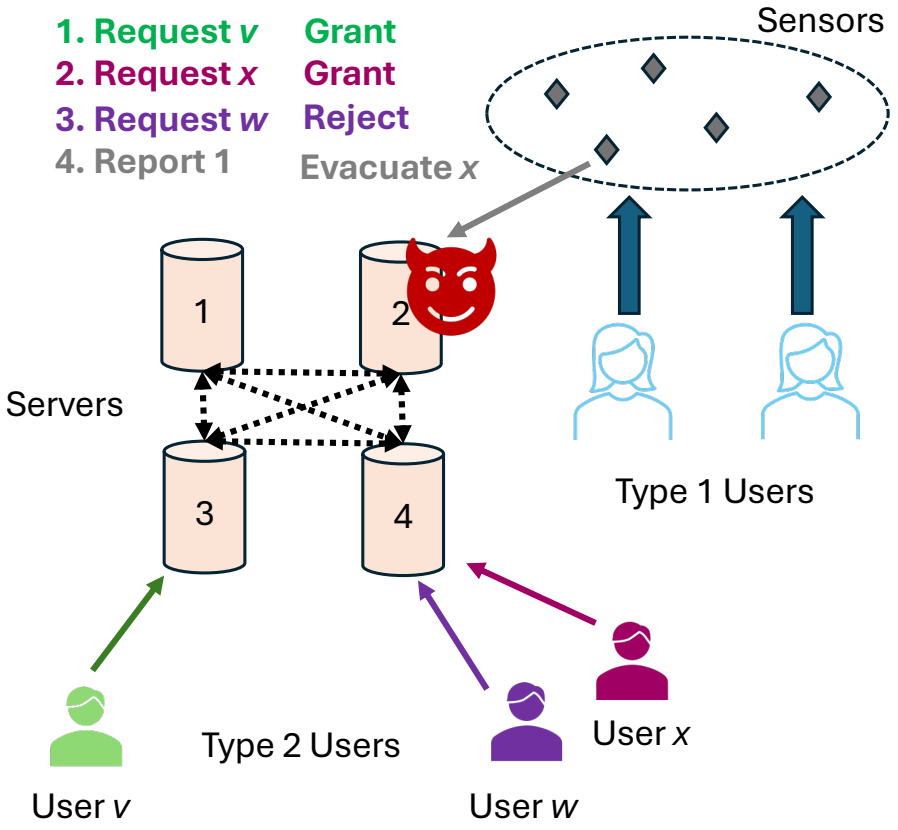
Tolerating Server Misbehavior with BFT: Model Mismatch

- Does NOT meet application needs as described
  - **What if a sensor is temporarily disconnected?**
    - Upon reconnection, want to agree on *latest state*, not reliable stream of updates
  - **What if a user is temporarily disconnected?**
    - Must stop transmitting to protect Type 1 users
  - **What if a sensor is compromised?**
  - **What if a user ignores decisions?**



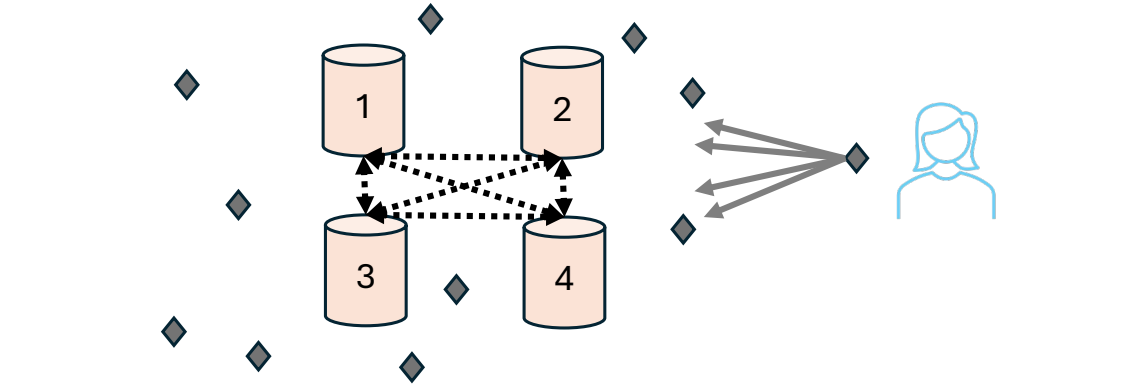
Tolerating Server Misbehavior with BFT: Policy implications

- Property rights / policy implications:
  - Implementation is decentralized, but **grant authority is centralized**
  - For each request, severs apply **deterministic rules** to determine whether it should be granted; rules are identical for all servers (including those deployed by different organizations)

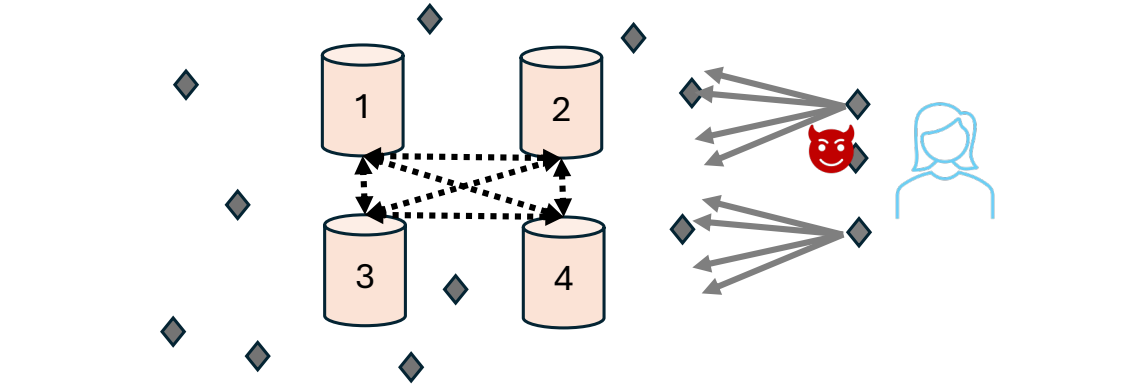


What if a sensor is compromised?

- Fully Trusted
  - All sensors are correct -> single detection report is enough



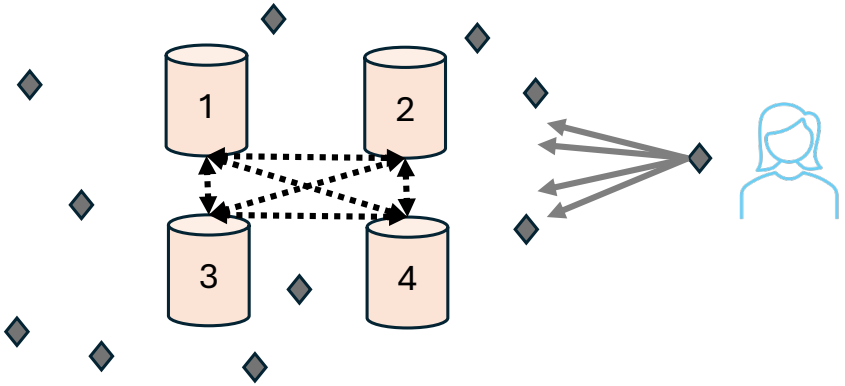
- Untrusted
  - Up to *g* sensors may be Byzantine
  - Require *g+1* reports to consider a transmission detected



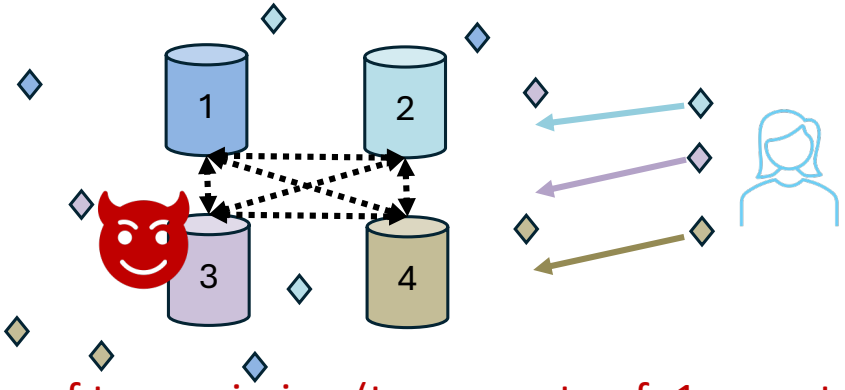
2g+1 sensors must be in-range of transmission (to guarantee g+1 report it)

What if sensors are deployed by different organizations?

- Global
  - All servers receive all sensor reports



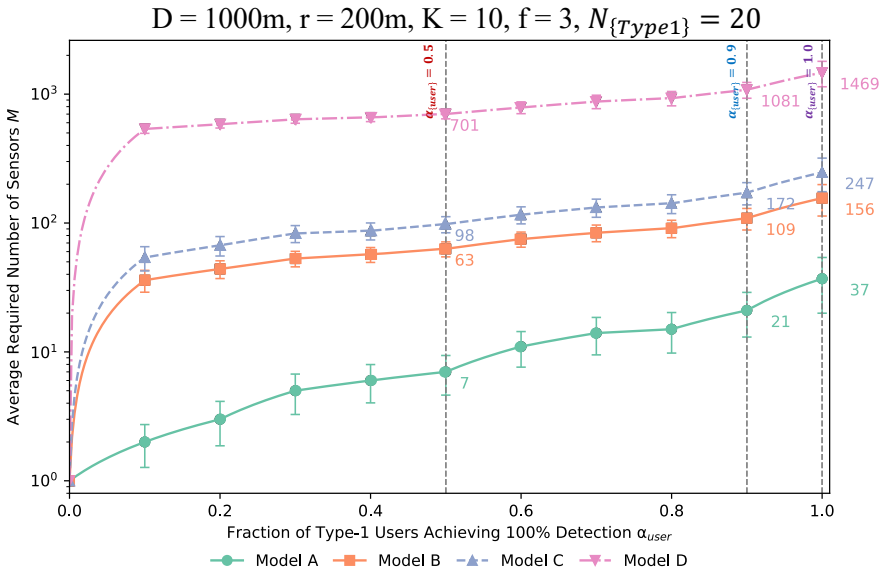
- Private
  - Each server receives reports from its "own" sensors (e.g. those deployed by same organization)



2f+1 servers must have sensors in-range of transmission (to guarantee f+1 report it)

Sensor Deployment Model: Impact on Number of Required Sensors

- **Results:**
  - **Baseline (global-fully trusted)**: **~37 sensors** sufficient to detect all Type 1 users
  - For **Model B (global-untrusted)** and **Model C (private-fully trusted)**, a Type 1 user is only detected once 7 sensors are in-range
    - Model C additionally requires that each of the 7 belongs to a different server
    - Require **4-10x more sensors** vs baseline
  - **Model D (private-untrusted)** tolerates the most compromises, but requires many sensors (**over 1400 for 100% detection**)



The number of sensors needed for detecting a given percentage of Type-1 users in *all* Monte Carlo trials

Outcomes so far…

- **BFT SMR** provides a potential technical framework for **decentralized coordination** of spectrum sharing
- Building a **correct** BFT Dynamic Spectrum Sharing System is not trivial
- Not obvious what **sensor and user threat models** are feasible to tolerate
- BFT models do not necessarily support **decentralized policy setting**
  - We're exploring regional and dynamic models that do

Thank you!

- Supported by seed grant from SpectrumX: An NSF Spectrum Innovation Center (NSF Award No. 2132700)