Protecting the Privacy of Computing Operations and Data in Autonomous Environments

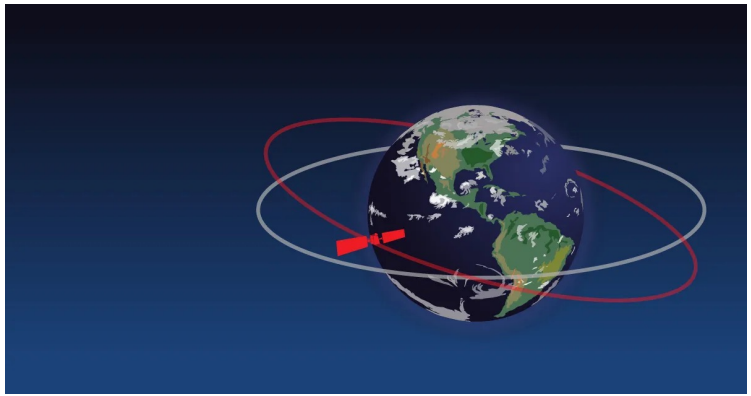
Kevin Butler & Caroline (Fedele) Brandon

University of Florida

April 2025

Protecting Information - Motivation

- Conjunction analysis: how to address potentially undisclosed satellites?
- More general question: How to prevent collisions without revealing what's in the air?
- Even more general: How to ensure data communicated by autonomous agents stays secure and private?



https://www.nasa.gov/cara/

Protecting Information - Motivation

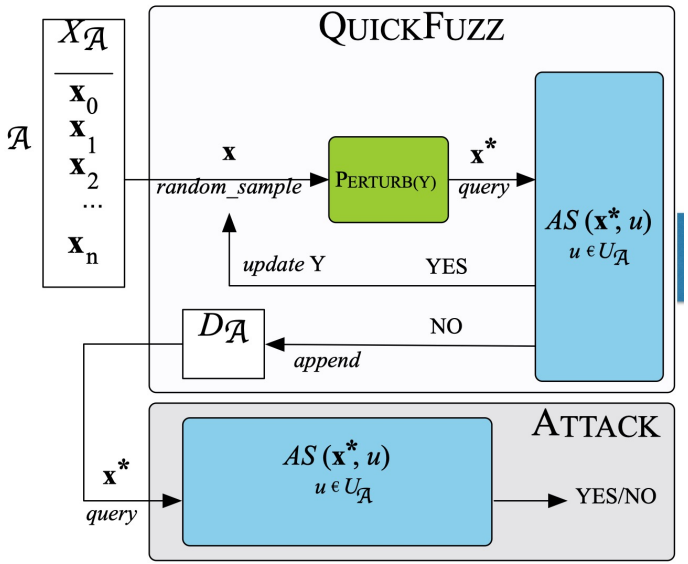
- Conjunction analysis: how to address potentially undisclosed satellites?
- More general question: How to prevent collisions without revealing what's in the air?
- Even more general: How to ensure data communicated by autonomous agents stays secure and private?
  - Data
  - System
    - Platform
    - Decision-making
  - User (real-time or training)

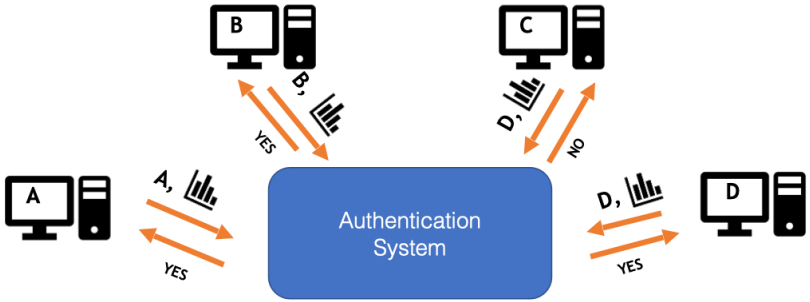
Students Involved

- Washington Garcia (ACT3/AFRL) – ***UDRI/ACT3***
- Caroline (Fedele) Brandon (2022/2023 AFRL/RV) – ***L3Harris***
- Grant Hernandez – ***Qualcomm – L3Harris***
- Brendan David-John ***– Virginia Tech***
- Joel Hirschmann

Authentication

- Examining limitations of authentication systems that employ machine learning and rely on “brittle” features



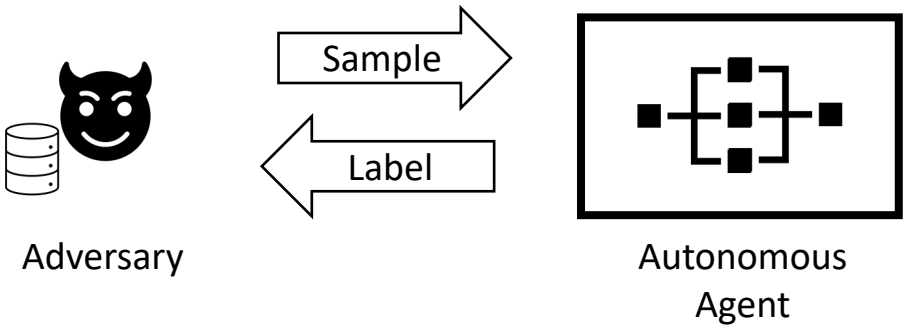


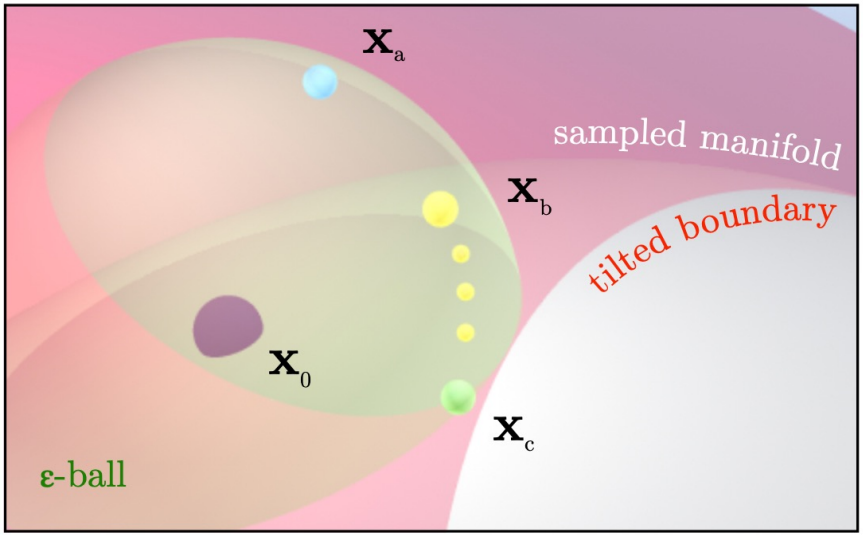
- Query-efficient fuzzing for adversarial sample crafting through feature extractor and heuristics for finding brittle features

Hard-Label Attacks

Hard-label adversarial machine learning attacks are a “grand-prize”:

- Adversary only needs ***query access*** to generate “label-flipped” samples (e.g., through compromised user)
- Hard-label attacks are gaining popularity, but not well characterized apart from convergence guarantees.





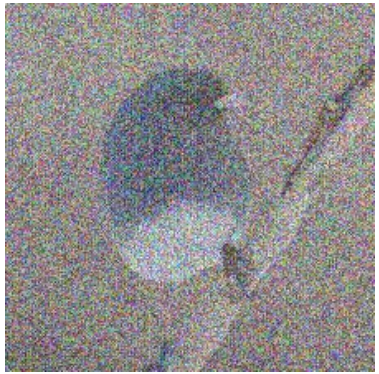
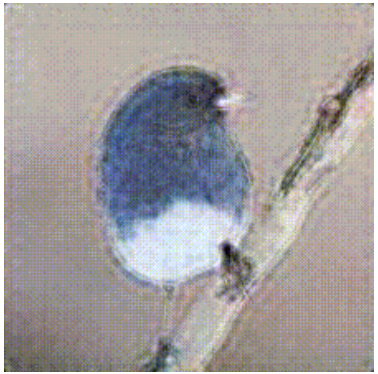
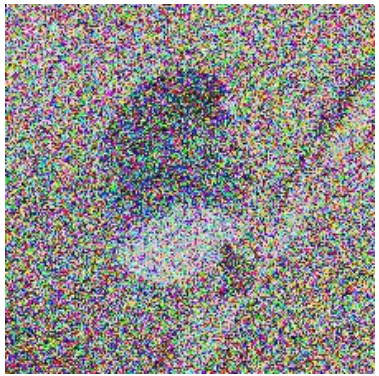
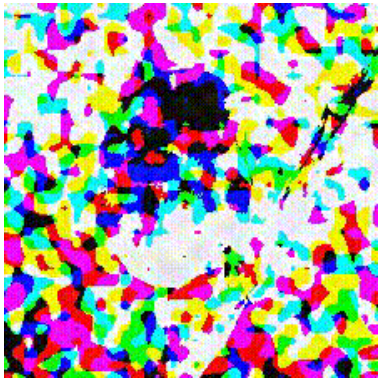
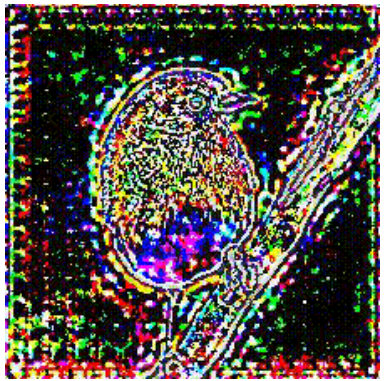
Subsampling and query efficiency

- Questions we sought to answer:
  - What advantages does search subsampling give the adversary?
  - How can we generalize the idea of search subsampling?
- We addressed this as an information-theoretic problem, leveraging the data processing inequality to derive a close—form solution of manifold-gradient mutual information

$$I(\mathcal{G},\mathcal{M})_\epsilon = 2\int_{\mathcal{M}^+} p(1,x^+)\log(\frac{p(1,x^+)}{p_{\mathcal{G}}(1)p_{\mathcal{M}}(x^+)})\,dx^+ + 2\int_{\mathcal{M}^+} p(-1,x^+)\log(\frac{p(-1,x^+)}{p_{\mathcal{G}}(-1)p_{\mathcal{M}}(x^+)})\,dx^+.$$

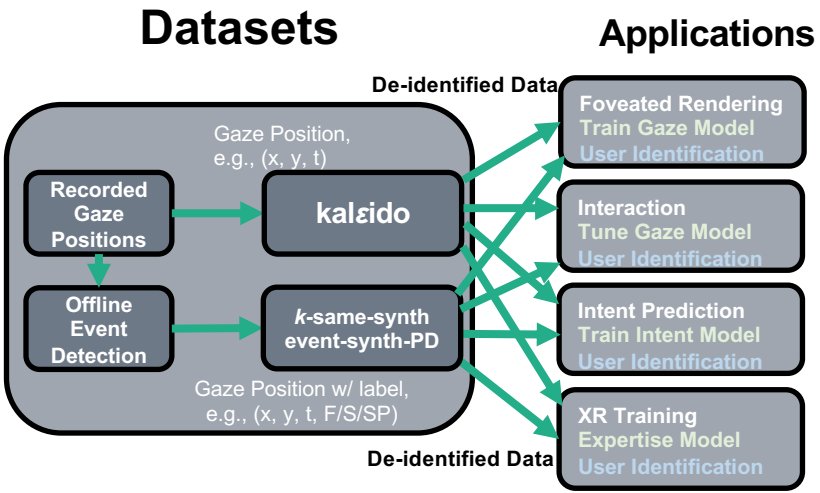
$$I(\mathcal{G},\mathcal{M})_\epsilon = \frac{2}{\sqrt{2\pi\sigma^2}}\sum_{i=1}^{||\mathcal{M}^+||}\exp(-\frac{(x_i^*-\theta)^2}{2\sigma^2})\cdot\beta_i^+\Delta_i + \frac{2}{\sqrt{2\pi\sigma^2}}\sum_{i=1}^{||\mathcal{M}^+||}\exp(-\frac{(x_i^*+\theta)^2}{2\sigma^2})\cdot\beta_i^-\Delta_i.$$

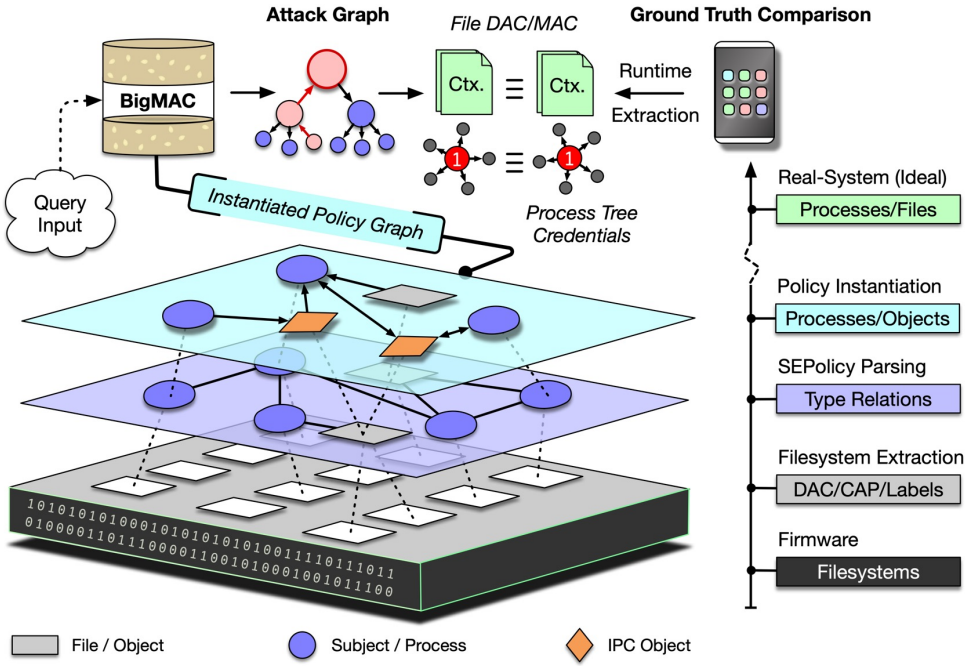
Results of Dimensionality Reduction

	HSJA	BiLN+HSJA	AE+Sign-OPT
Adversarial			
Diff			

Protecting Platforms and Users

- Reducing re-identification risk through differential privacy eye tracking mechanisms for both post-processed and real-time streaming data





- Information protection through access control on embedded/mobile platforms

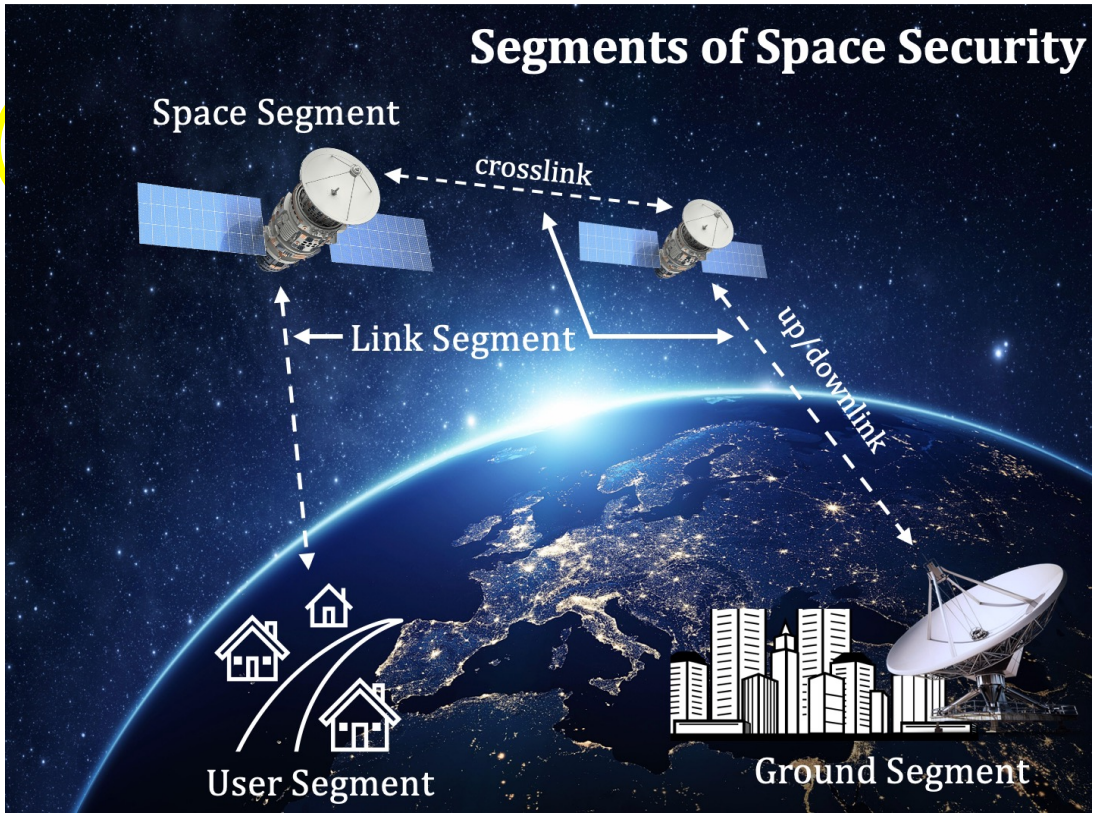
The Final Security Frontier: Securing Satellite Rendezvous and Proximity Operations with Secure Multi-party Computation

Caroline (Fedele) Brandon

L3Harris Technologies | University of Florida

April 2025

Security in Space



Congested & Contested

- Rapidly growing number of and dependence on satellites
- Protection of critical infrastructure
- Rendezvous & Proximity Operations (RPO) and In-Space Servicing, Assembly, Manufacturing (ISAM)
- Motivates autonomy needs
- Need to develop security standards and protocols

Recap

Using privacy preserving computation to secure data during satellite rendezvous & proximity operations (RPO)

- Explored methods and tools for secure multi-party computation (SMC)
- Worked with Kirtland AFRL to determine commonly-used satellite algorithms (hence RPO) and embedded hardware that is operable in space
- Determined the need for privacy during RPO, little existing research
- Demonstrated that SMC is a feasible approach and can be done in RPO algorithms given space constraints
- Optimized SMC protocol parameters for in-space constraints, and customized underlying cryptography for different algorithmic functions

12

Motivation: RPO

Rendezvous & proximity operations (RPO)

- On-orbit trajectory operation & replanning
  - Servicing (e.g., docking, refuel, repair), inspection, reconnaissance, formation flying, collision avoidance, debris removal
- Occurs autonomously on-board in guidance navigation and control (GNC)

	Ground station	On-Board
Distance between satellites	1-10 Mm	< 500 km
Time needed	Days-weeks	< 1 day
Speed	km /sec	m /sec
Method	conjunction analysis	RPO

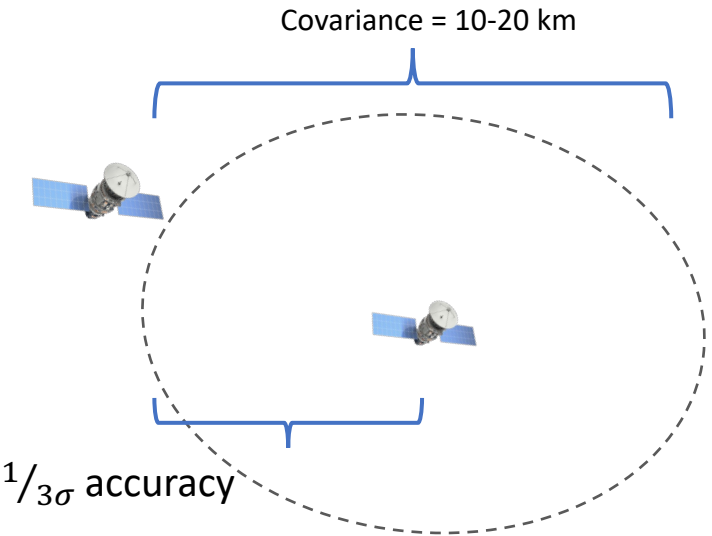
Problem: Proprietary Design Inference

Example: collision avoidance

- Data to share with other satellites

  position | velocity | **covariance**

- Covariance matrices quantify uncertainty
  - Calculated using intrinsic sensor variance
  - Measure of trust in probabilistic system



Problem: knowledge of covariance matrices can lead to knowledge of sensors on board

→ inference of satellite capabilities, purpose, etc.

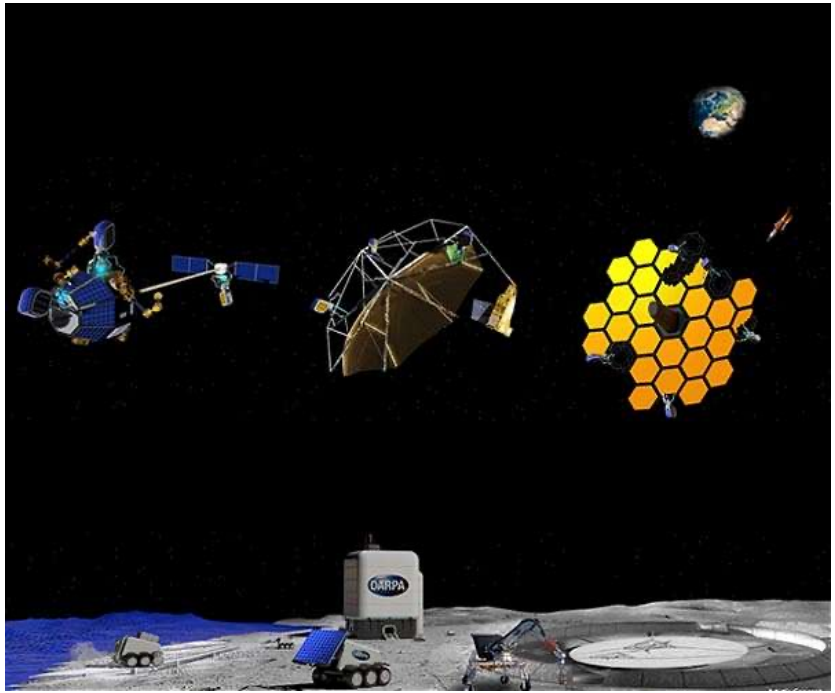
Problem: Proprietary Design Inference

Example: in-space manufacturing

- Integrated circuits, advanced materials, bioengineering, large assembly (Luvoir telescope)

Sensitive Values*	Threat Assumptions
Covariance matrices	Infer proprietary sensor info
Fuel levels	May infer satellite capacity/mission objectives
State-of-health telemetry (e.g. power, heat use)	Infer propulsion system,
Installation/servicing technology parameters	Infer IP (e.g. IC design, robotic arm capability)

*not exhaustive list, values are mission dependent



Solution: protect sensitive values using **privacy-preserving computation**

UF UNIVERSITY of FLORIDA · Duke UNIVERSITY · TEXAS The University of Texas at Austin · UC SANTA CRUZ

Privacy-Preserving Computation

Privacy-Preserving Computation

- Allows for data to remain encrypted during computation
- Protect **physical integrity** of satellite during RPO and **data privacy** keeping data encrypted
- Promising for other constrained systems (wireless sensor network, embedded devices)

Secure Multiparty Computation (SMC):

- Cryptographic protocol that allows set of mutually-distrusting parties to jointly compute a function on their inputs, without revealing information about inputs (millionaire's problem)

Computation Domain

Mathematical structure of secret info

- Binary circuits or arithmetic circuits
- Ring or finite field defined by integer operation with modulus prime or 2^k

Underlying Primitives

- Secret Sharing
- Oblivious Transfer
- Homomorphic Encryption
- Garbled Circuits

Security Model

Adversarial Settings

- Honest vs. dishonest majority – assumption of behavior of parties
- Semi-honest vs. malicious corruption – passive vs. active adversary

Case 1: Honest majority, semi-honest model

- MPC operation within organization, trusted computing parties
- Preventing accidental leakage

Case 2: Dishonest majority, semi-honest model

- Compromised coalition of multiple stakeholders with common mission goal
- Prevent passive adversaries from learning proprietary information

Case 3: Dishonest majority, malicious model

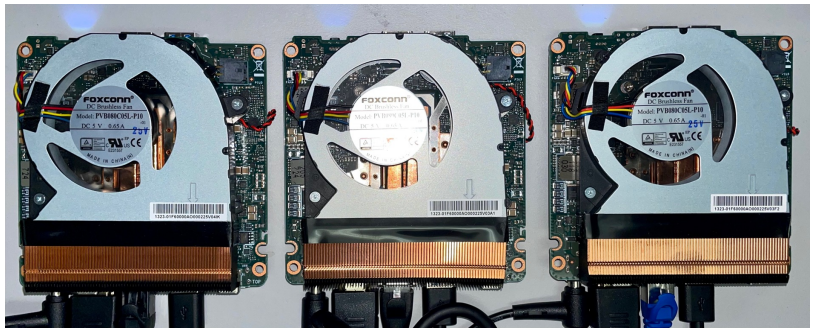
- MPC done between untrustworthy organizations
- Expectation of attack on protocol itself

Implementation

Benchmarks

- Small satellites assumed for this research
- “passively safe” model – optimizes calculations for time and fuel levels
- Network emulates latency and bandwidth in space communications (500km distance, 10 Mbsp bandwidth)

On Board		Network	
Execution time	30 sec - 5 min	Frequency	S, X, Ku, Ka, optical
Memory	10 MB	Rate	10 Mbps – 1 Gbps
Examples	Dove constellation, OneWeb and SpaceX’s Ku-Ka-band satellites, Kepler system, Spire’s small satellites		



Hardware setup: 3 Nvidia Nanos – each 4 ARM CPUs, 1.47GHz clock speed, over 1 Gbps port

Finding hardware for use in space

- Commercial off-the-shelf (COTS)
- Sufficient radiation tolerance
  - e.g. 1-1.5 years for small satellites
- Sufficient power & efficiency with limited resources

SMC Configurations

MP-SPDZ SMC Compiler

- Dishonest majority, semi-honest:
  hemi, semi, semi2k*, soho, temi
- Dishonest majority, malicious:
  mascot, mama, spdz2k
- Honest majority**, semi-honest:
  rep-field, rep-ring, shamir
- Honest majority, malicious:
  maliciously-secure variations of rep-field, rep-ring, and shamir

Table of protocols

Protocol	Specifics	Other protocols in family
mascot	GF(p) field-based, OT-based preprocessing	mama, SPDZ2k (mod 2^k)
semi	GF(p), stripped version of mascot, OT for preprocessing (hemi/soho/temi use SHE)	semi2k (mod 2^k), hemi, temi, soho
replicated-field	Replicated secret shares over field GF(p), supports all arithmetic (+/-/x/÷), minimal communication	mal-rep-field, ps-rep-field, sy-rep-field
replicated-ring	Replicated secret shares over ring (mod 2^k), faster mult & bitwise ops, does not support division	mal-rep-ring, ps-rep-ring, sy-rep-ring
shamir	Threshold secret sharing over field, higher comms, stronger security capabilities	mal-shamir, sy-shamir

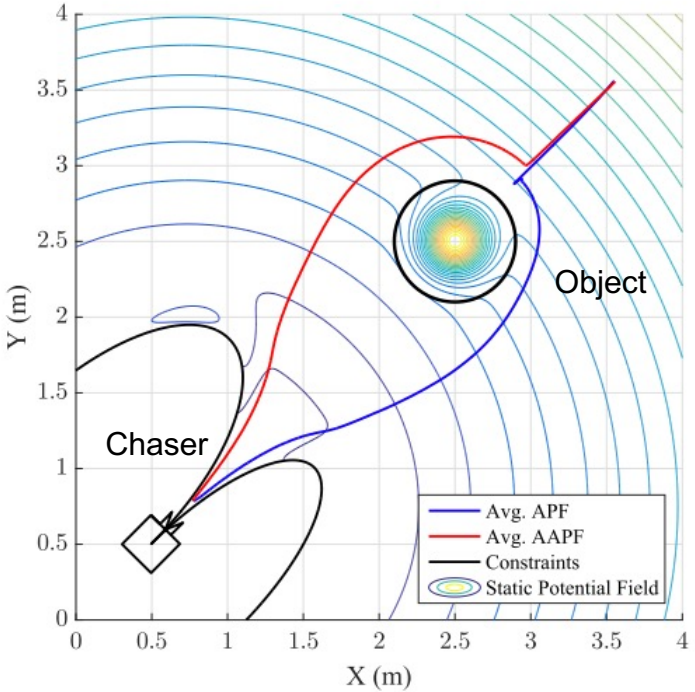
Collision Avoidance

Artificial Potential Function (APF):

- Docking, servicing, collision avoidance
  - On-board trajectory control
- Assume linear orbital dynamics: one satellite stationary relative to other

Parameter	Value	Privacy
$\mathbf{c}_{pos}, \mathbf{o}_{pos}, \mathbf{t}_{pos}$	vectors of chaser, object, target positions respectively	public
$k_a, \mathbf{K}_{acc}$	Gain values	public
$\mathbf{B}_{inv}$	Input guidance parameter	private
$\mathbf{Q}_a$	Target covariance matrix	private
$\mathbf{c}_{vel}$	Chaser's velocity	private
ψ, σ	Chaser's height, width	private
$\mathbf{N}$	Object's covariance matrix	private

Satellite trajectory to avoid obstacle

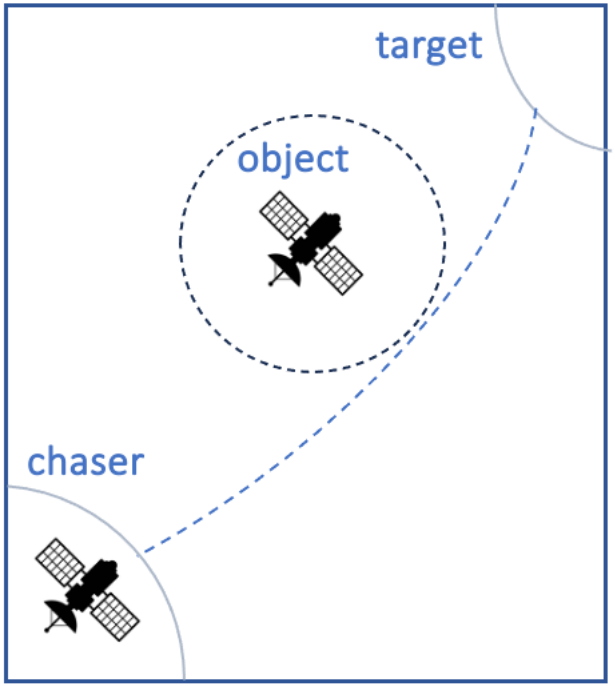


10.1007/978-981-10-2963-9_5

Collision Avoidance

How it works:

- 2 satellites: Chaser and Obstacle
- 3 positions (3x3 in space): chaser, obstacle, target



Attractive Potential

$$\phi_a = \frac{k_a}{2} \boldsymbol{r}_{ct} \boldsymbol{Q}_a \boldsymbol{r}_{ct}$$

Repulsive Potential

$$\phi_r = \exp[\frac{-\boldsymbol{r}_{co} \boldsymbol{N} \boldsymbol{r}_{co}}{\sigma}]$$

$\boldsymbol{r}_{ct}, \boldsymbol{r}_{co}$= distance between chaser and target, obstacle respectively; k_a, σ, ψ are constants; $\boldsymbol{Q}_a$= covariance matrix of Target, $\boldsymbol{N}$ = covariance of obstacle

Control Forces

$$\boldsymbol{u}(\boldsymbol{x_c}, \boldsymbol{x_t}, \boldsymbol{x_{vel}}) = -\boldsymbol{B}^{-1} \boldsymbol{K}_a (\boldsymbol{x_{vel}} + \nabla \phi_{tot}), \quad \text{where}$$

$$\nabla \phi_{tot} = k_a \boldsymbol{Q}_a \boldsymbol{r}_{ct} + \frac{2\psi}{\sigma} \exp[\frac{-\boldsymbol{r}_{co} \boldsymbol{N} \boldsymbol{r}_{co}}{\sigma}] \boldsymbol{N} \boldsymbol{r}_{co}$$

APF Pseudocode

Public parameters: $k_a \in \mathbb{R}, \mathbf{K_{acc}} \in \mathbb{R}^3$ (gain and shaping factors)
All matrices ($\mathbf{K_{acc}}, \mathbf{Q_a}, \mathbf{N}, \mathbf{B_{inv}}$) are assumed to be diagonal and replaced with vectors of their diagonal elements

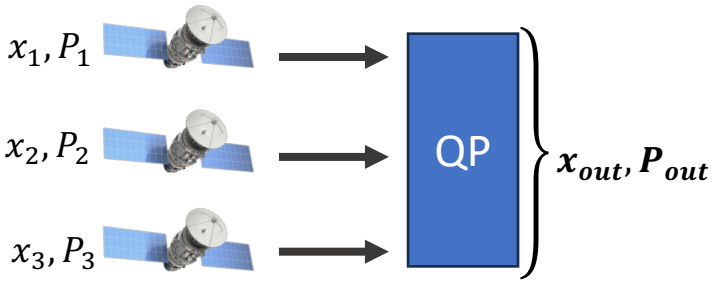
1: **procedure** APF($\mathbf{c_{pos}}, \mathbf{o_{pos}}, \mathbf{t_{pos}} \in \mathbb{R}^3, [\![\mathbf{c_{vel}}]\!], [\![\mathbf{Q_a}]\!], [\![\mathbf{N}]\!], [\![\mathbf{B_{inv}}]\!] \in \mathbb{R}^3, [\![\sigma]\!], [\![\psi]\!] \in \mathbb{R}$) ▷ coordinates $\mathbf{c_{pos}}$, $\mathbf{o_{pos}}$, and $\mathbf{t_{pos}}$ are public; private $\mathbf{N}$ is contributed by the object; private $\mathbf{c_{vel}}, \mathbf{Q_a}, \mathbf{B_{inv}}, \sigma$, and ψ are contributed by the chaser
2: $\mathbf{rCt} \leftarrow \mathbf{t_{pos}} - \mathbf{c_{pos}}$ ▷ chaser-target distance
3: **if** $(\mathbf{rCt}[2] > \pi)$ **then** ▷ wrap angle between $-\pi$ and π
4: $\mathbf{rCt}[2] \leftarrow \mathbf{rCt}[2] - 2\pi$
5: **else if** $(\mathbf{rCt}[2] < -\pi)$ **then**
6: $\mathbf{rCt}[2] \leftarrow \mathbf{rCt}[2] + 2\pi$
7: **end if**
8: $\mathbf{rCo} \leftarrow \mathbf{o_{pos}} - \mathbf{c_{pos}}$ ▷ chaser-object distance
9: **if** $(\mathbf{rCo}[2] > \pi)$ **then** ▷ wrap angle between $-\pi$ and π
10: $\mathbf{rCo}[2] \leftarrow \mathbf{rCo}[2] - 2\pi$
11: **else if** $(\mathbf{rCo}[2] < -\pi)$ **then**
12: $\mathbf{rCo}[2] \leftarrow \mathbf{rCo}[2] + 2\pi$
13: **end if**
14: $[\![\nabla\phi_A]\!] \leftarrow (k_a \cdot \mathbf{rCt}) \cdot [\![\mathbf{Q_a}]\!]$ ▷ attractive potential gradient
15: $[\![x]\!] \leftarrow (\mathbf{rCo} \cdot \mathbf{rCo}) \odot [\![\mathbf{N}]\!]$
16: $[\![\text{constVio}]\!] = 0$
17: **if** $([\![x]\!] < 1)$ **then**
18: $[\![\text{constVio}]\!] = 1$
19: **end if**
20: $[\![x]\!] \leftarrow -[\![x]\!]/[\![\sigma]\!]$
21: $[\![\nabla\phi_R]\!] \leftarrow 2[\![\psi]\!]/[\![\sigma]\!] \cdot e^{[\![x]\!]} \cdot (\mathbf{rCo} \cdot [\![\mathbf{N}]\!])$ ▷ repulsive potential gradient
22: $[\![\mathbf{u}]\!] \leftarrow (-\mathbf{K_{acc}}) \cdot [\![\mathbf{B_{inv}}]\!] \cdot ([\![\mathbf{c_{vel}}]\!] + [\![\nabla\phi_A]\!] + [\![\nabla\phi_R]\!])$ ▷ compute control force
23: **return** $[\![\mathbf{u}]\!], [\![\text{constVio}]\!]$
24: **end procedure**

→ **expensive operation**

Algorithm 2: Quadratic Program

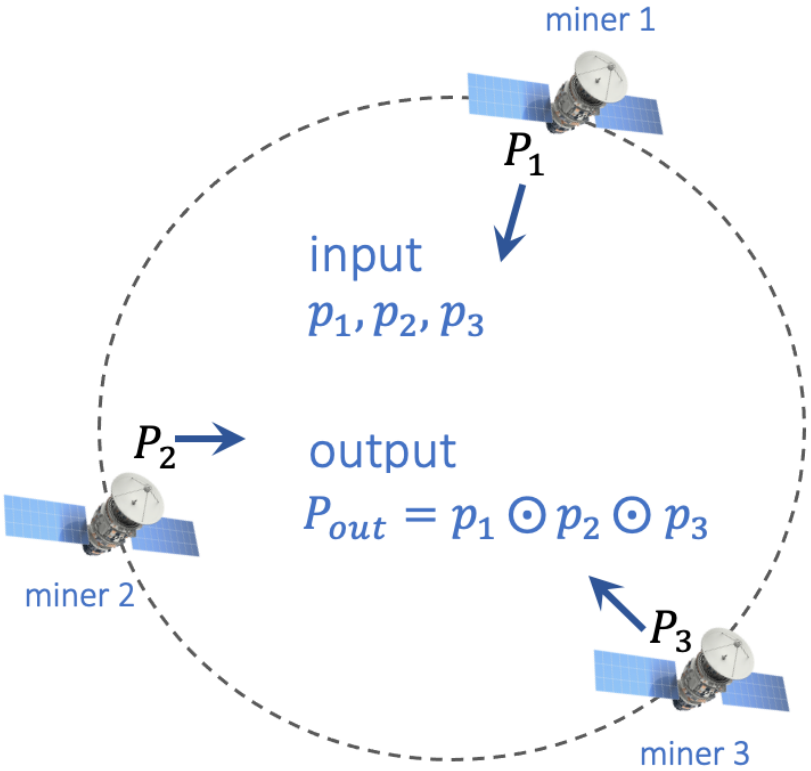
Quadratic Program: multi-point inspection

- Sensor Fusion optimization algorithm
- Need 3+ parties for 3 dimensional accuracy
- Allows for preprocessing expensive computations



Shared parameters

- Measured positions: x_1, x_2, x_3 → **Public**
- Position covariance: P_1, P_2, P_3 → **Private**



QP Pseudocode

Algorithm 2 Quadratic Program

1: **procedure** QP($\mathbf{pos}_1, \mathbf{pos}_2, \mathbf{pos}_3 \in \mathbb{R}^3, [\![\mathbf{P}_1^{-1}]\!], [\![\mathbf{P}_2^{-1}]\!], [\![\mathbf{P}_3^{-1}]\!] \in \mathbb{R}^{3\times3}$) ▷ measured coordinates $\mathbf{pos}_1, \mathbf{pos}_2, \mathbf{pos}_3$ are public; private inverse covariance matrix $[\![\mathbf{P}_i^{-1}]\!]$ is contributed by participant i

2: $[\![\mathbf{M}]\!] \leftarrow 2([\![\mathbf{P}_1^{-1}]\!] + [\![\mathbf{P}_2^{-1}]\!] + [\![\mathbf{P}_3^{-1}]\!])$

3: $[\![\mathbf{v}]\!] \leftarrow 2([\![\mathbf{P}_1^{-1}]\!][\![\mathbf{pos}_1]\!] + [\![\mathbf{P}_2^{-1}]\!][\![\mathbf{pos}_2]\!] + [\![\mathbf{P}_3^{-1}]\!][\![\mathbf{pos}_3]\!])$

4: $[\![\mathbf{e}]\!] \leftarrow [\![\mathbf{M}]\!]^{-1}[\![\mathbf{v}]\!]$

5: **return** $[\![\mathbf{e}]\!]$ ▷ position uncertainty

6: **end procedure**

→ **expensive operation**

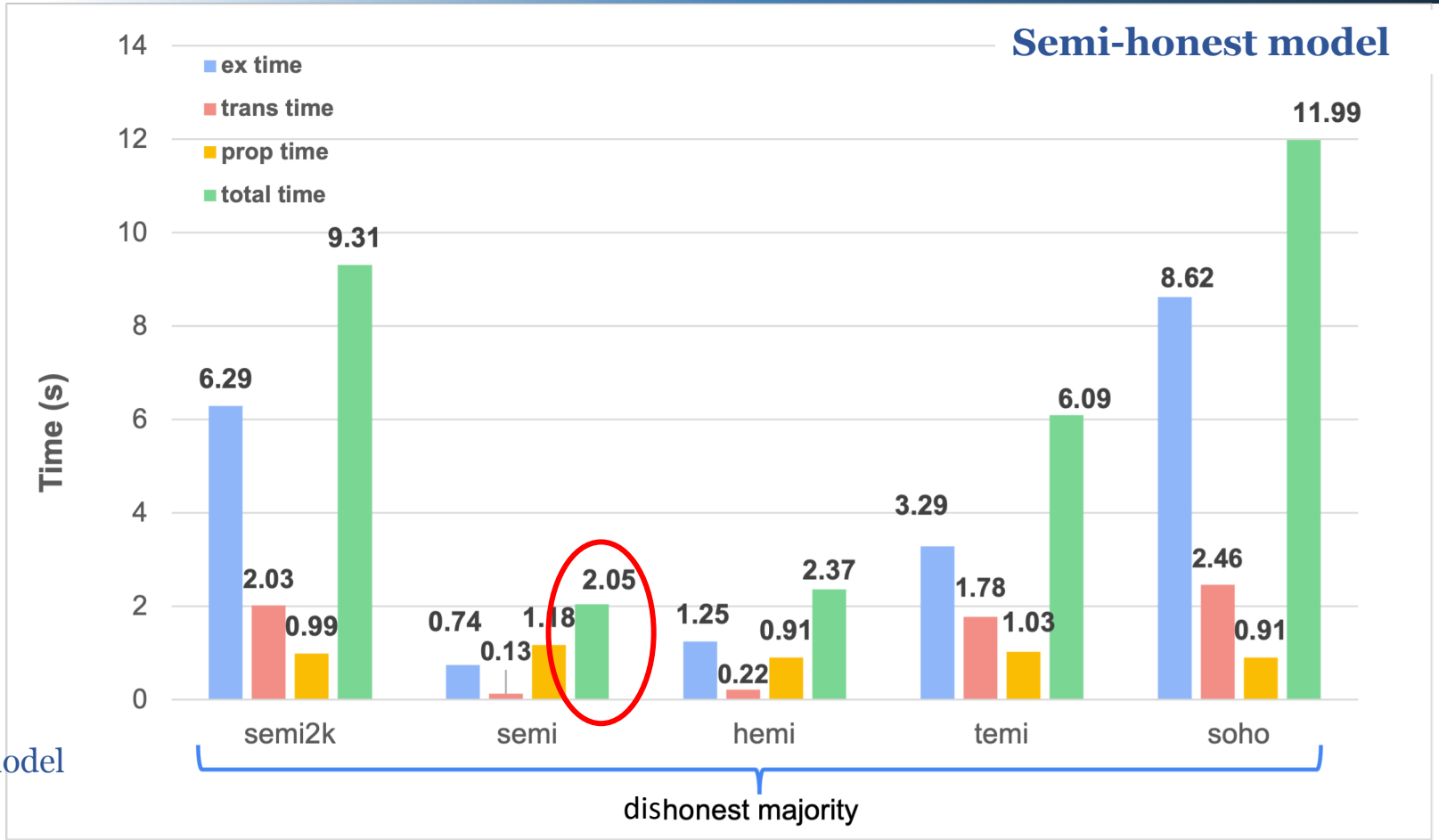
MPC Optimizations

- **CODE:** reduced computational complexity and redundancy in both APF and QP code
  - Represent 3x3 diagonal matrices as 3x1 vectors
  - Vectorized multiplications, comparisons, other operations
  - Imported pre-computed values from Chaser and Object, rather than hardcode
  - Eliminated code “fluff”, unused variables, and redundancy (i.e. combined loops)
  - Used fixed point (sfix, cfix) over floating point (expensive in MP-SPDZ)
  - Removed unnecessarily privatized variables, making them public instead
    - i.e. positions in space are not particularly sensitive

MPC Optimizations

- **NETWORK:** used MP-SPDZ optimization parameters
  - --direct: instructs protocol to bypass usual preprocessing steps for certain gates, enabling direct communication between parties, typically for input or output gates
  - --batch size: controls how many independent inputs or operations are processed together in a single batch; optimizes preprocessing
    - Ex. For HE, 1 multiplication costs same as 10,000. Can reduce batch size to reduce comms cost, though potentially increasing number of rounds
  - --budget: sets limit for number of preprocessing elements (e.g., triples, bits, etc.) to be used or generated, preventing overuse of resources in constrained environments.
    - Controls tradeoff between compile speed/memory usage and communication rounds during execution

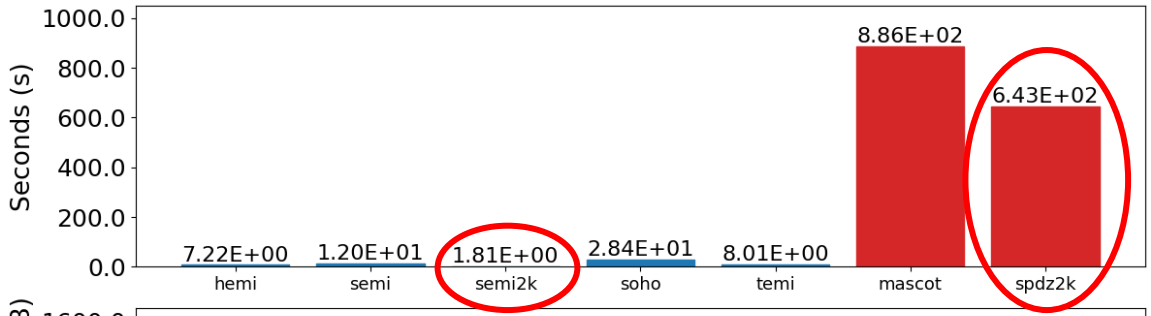
Previous APF evaluation



(Malicious model omitted)

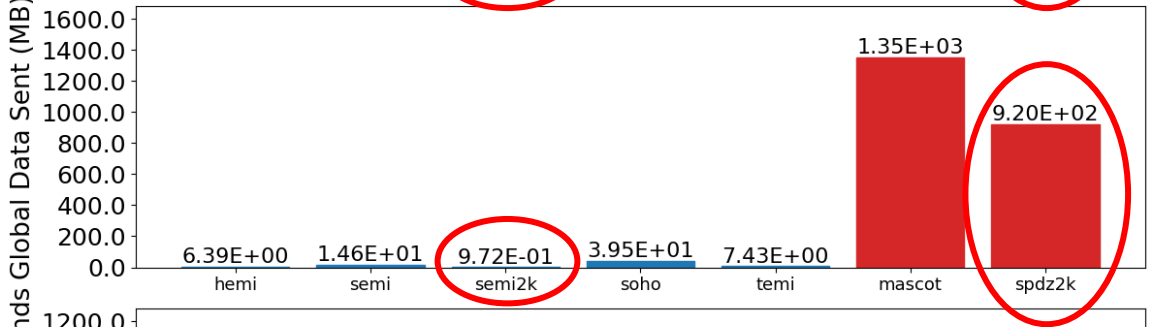
Evaluation – APF Benchmarks

Total execution time



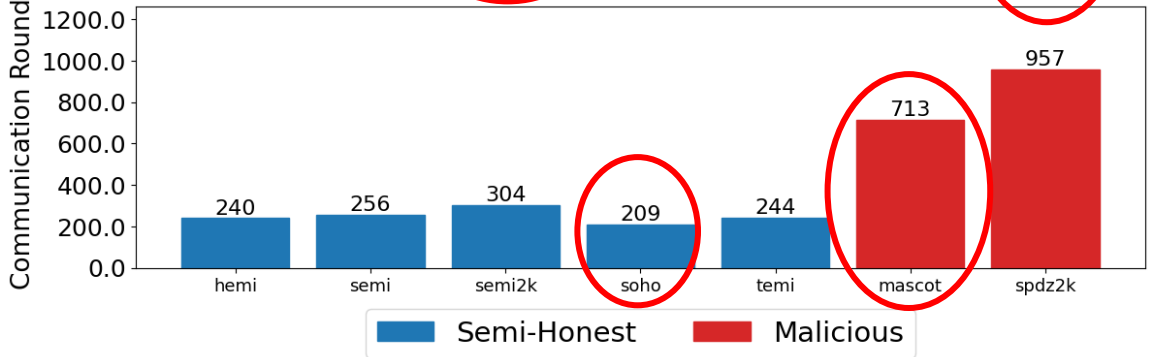
→ semi2k, spdz2k

Data shared



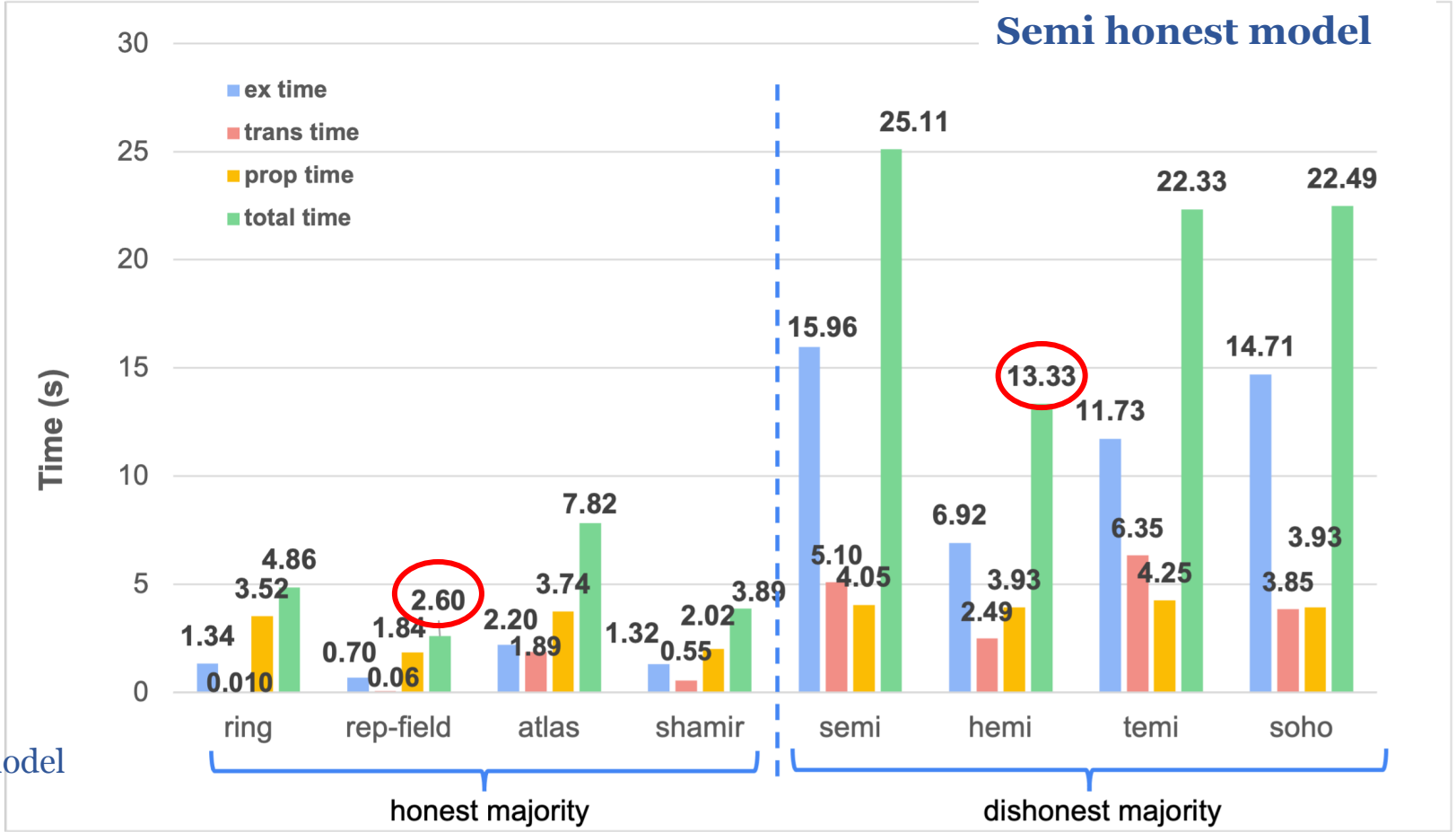
→ semi2k, spdz2k

Communication rounds



→ soho, mascot

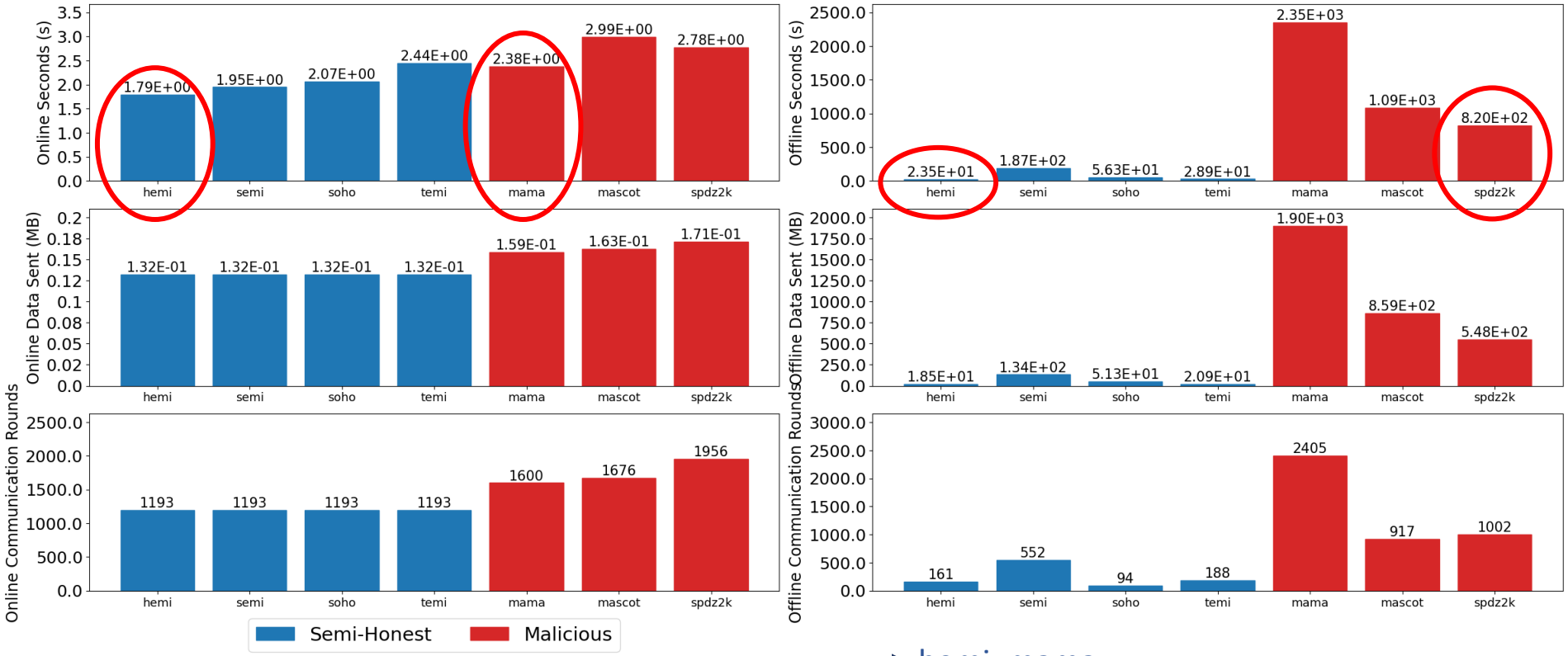
Previous QP Evaluation



(Malicious model omitted)

Evaluation: QP Benchmarks

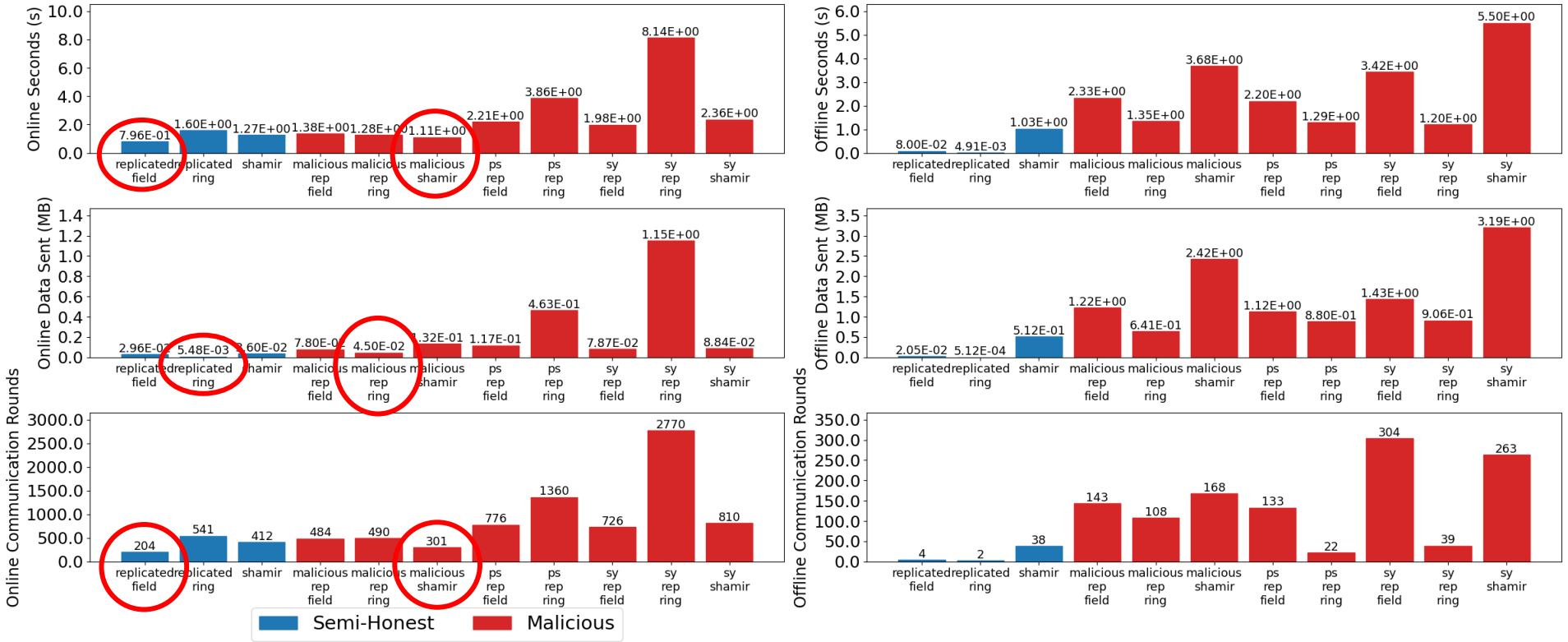
Dishonest Majority



→ hemi, mama

Evaluation: QP Benchmarks

Honest Majority



→ hemi, mama

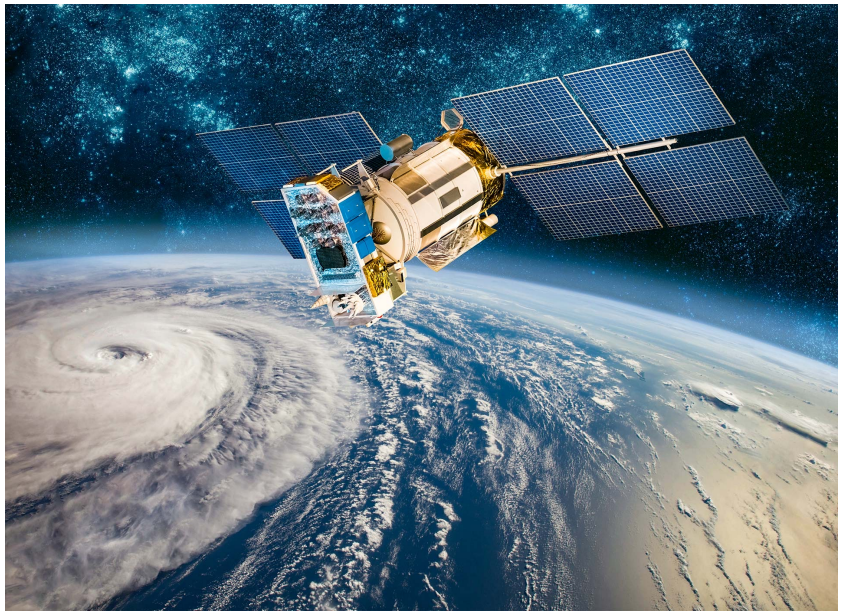
Observations

- **Number of communication rounds**
  - Varies based on computation
  - Lowering round complexity of implementation → very important
  - Optimized using the --budget parameter, i.e. set to 100,000 instead of default 1,000

- **3 party dishonest & honest majority protocols**
  - Semi-honest and malicious settings have comparable performance for online time
  - Reasonable to use malicious-secure setting if preprocessing is possible

- **Honest majority protocols**
  - Shamir is fastest online for malicious-secure protocols
    - Optimized shamir using --direct parameter to implement direct communications between parties rather than interpolation and “star-shaped” communications

Takeaway



Source: verdict.co.uk

Conclusion:

- Space is *highly* constrained → security & privacy solutions must be *highly* customized
  - Parameters can be tweaked to improve given protocol for given algorithm
  - Lowering round complexity of implementation → very important
- Values are very reasonable for RPO constraints of < 5 minutes and < 10 MB:
  - 2 party: 1.81 sec, 0.97 MB (1.2x faster)
  - 3 party (DM, online): 1.79 sec, 0.13 MB (7.4x faster)
  - 3 party (HM, online): 0.79 sec, 0.03 MB (3.3x faster)

Acknowledgements

Kevin Butler

Marina Blanton

Chrispy Petersen

Carson Stillman

Joel Hirschmann

Sara Rampazzi

Questions?