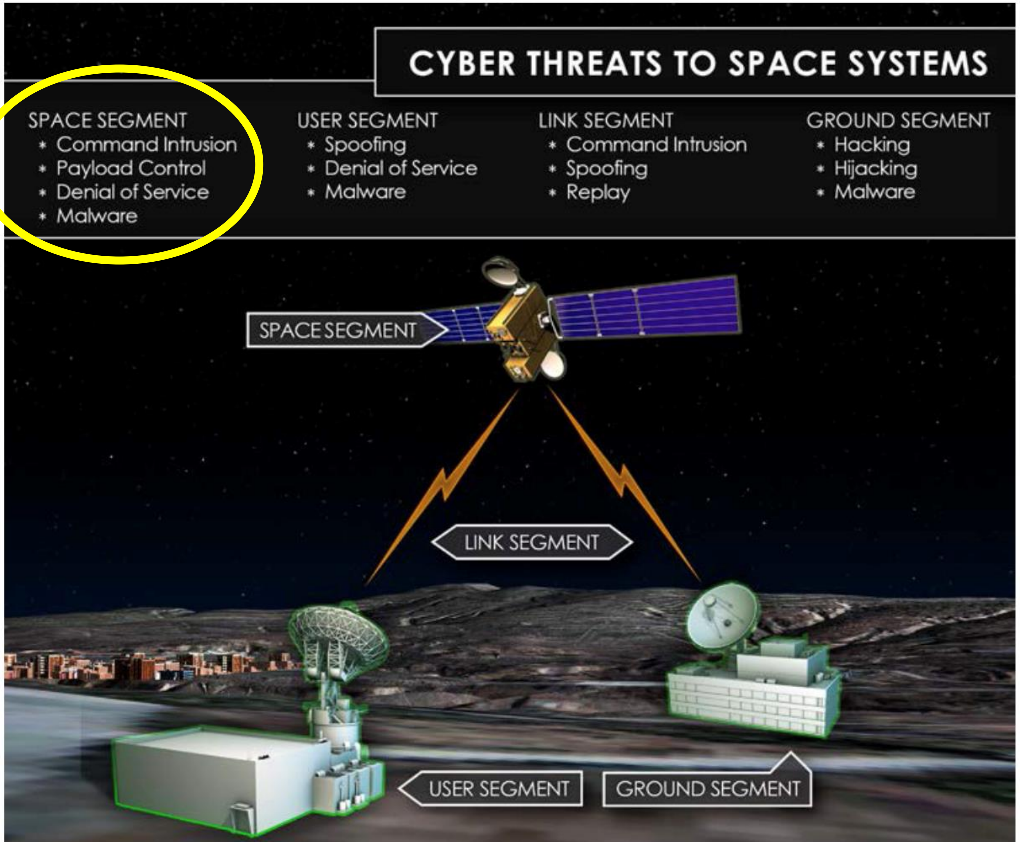
Optimizing Secure Multi-Party Computation in Satellite Proximity Operations

Caroline Fedele
University of Florida

7 December 2023

Security in Space



The Aerospace Corporation, 2019

- **Needs for both privacy and security in space**
- **Collisions have occurred**

IN-SPACE Cybersecurity

- **Growing number of satellites & expanding private sector**
- **Motivates autonomy needs**
- **Rendezvous & Proximity Operations (RPO)**
- **Near-field collision avoidance and characterization**

Project Overview

General goal: provide evaluation of secure satellite proximity operations using privacy-preserving computation

Demonstrate use of **secure multiparty computation (SMC)**, a method of operating on encrypted data, allowing private **satellite operations** to be conducted between mutually-distrustful agents

Previous work:

- Investigated where data privacy is needed in space
- Implemented SMC into matrix multiplication, attitude optimization, and other algorithms using the Sharemind SMC toolkit (3+ party, secret-sharing based protocol)
- Benchmarked time and memory overhead between each algorithm without and with SMC

Current work:

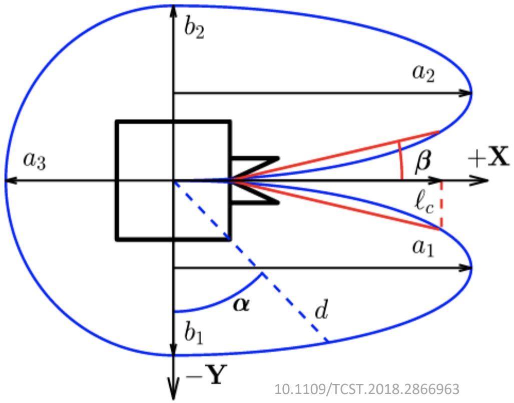
- Implementing SMC into more satellite proximity algorithms, demonstrating improvements
- Benchmarking various overhead measurements of 20+ secure protocol variants using MP-SPDZ

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

Rendezvous and Proximity Operations (RPO):

- On-board trajectory operation and replanning
  - E.g. docking, on-orbit servicing/refueling, formation flying
- RPO occurs on-board, autonomously
  - housed in guidance navigation and control (GNC) unit
- Needed at scales of < 500km between satellites

RPO example: docking



10.1109/TCST.2018.2866963

Ground station vs On-board Control

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

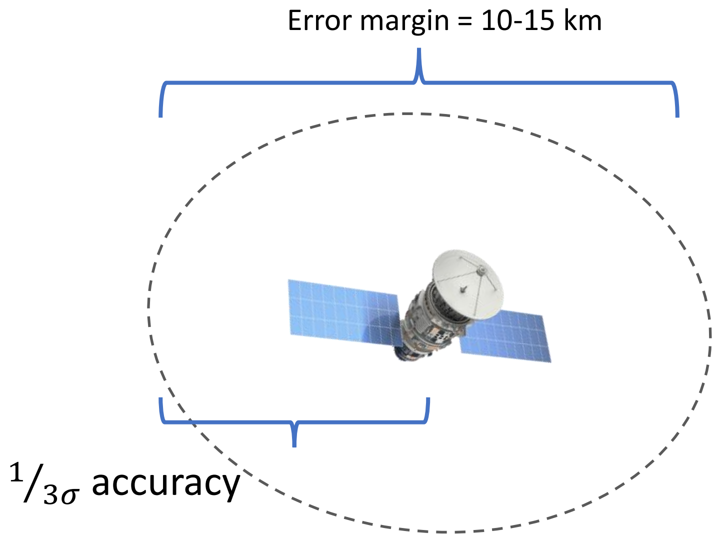
Problem: Capability Inference

Example: Collision Avoidance in RPO

- Minimum data to share with other satellites
  - position, velocity, ***covariance***

Stochastic systems

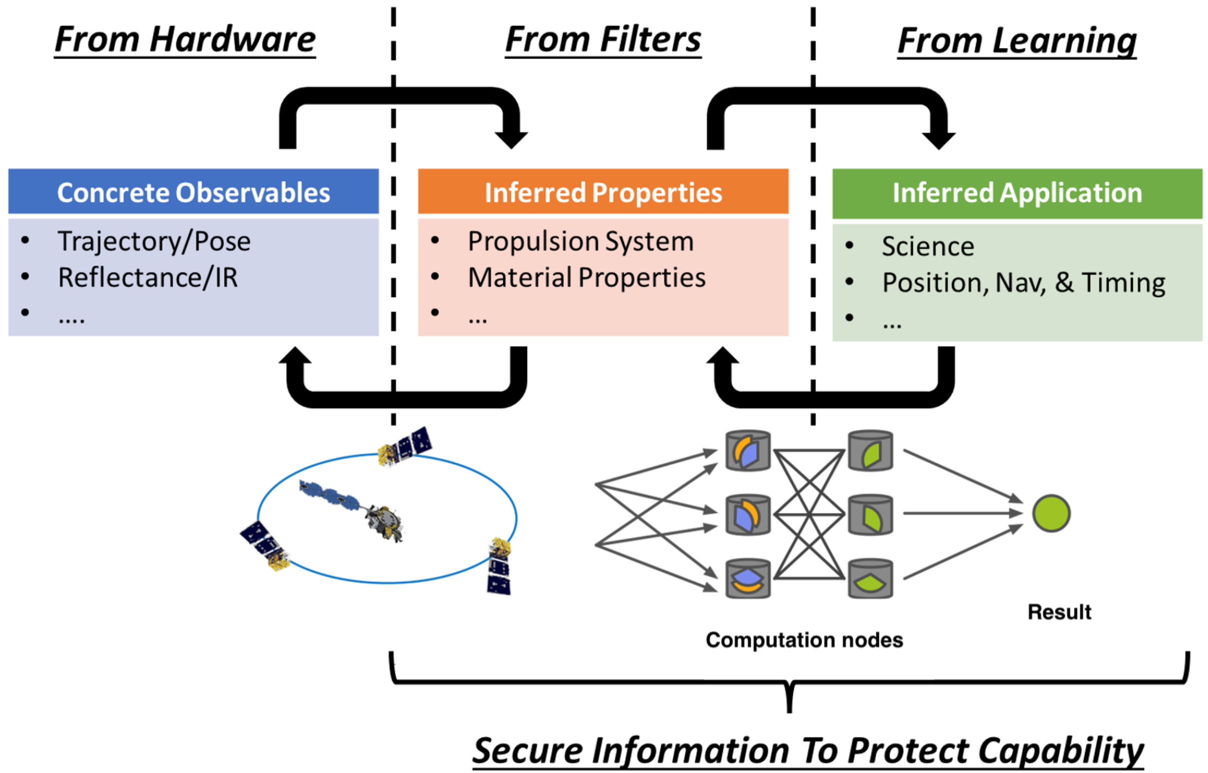
- Probabilistic, not deterministic
- Covariance matrices = quantify uncertainty
  - defined by ellipsoid
- Measure of TRUST, decisions based on accuracy



Problem: knowledge of error margins (covariance matrices) can lead to inferences on satellite capabilities, purpose, etc.

Solution: protect error margins using **privacy-preserving computation**

Characterization Problem



Privacy-Preserving Computation

Privacy-Preserving Computation (PPC)

- Allows for data to remain encrypted during computation
- Protects **physical integrity** of satellite during RPO and **data privacy** keeping data encrypted

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)
   1. *2-Party Computation (2PC):* e.g. Yao's garbled or BMR, binary circuit representation
   2. *Secret sharing*: 3+ parties, arithmetic circuit representation

Homomorphic Encryption (HE):

- Fully or Partially *homomorphic encryption* (FHE/PHE)
- "holy grail" of cryptography, providing strongest privacy guarantees at the cost of efficiency

Multi-Party Computation Tool

Security Models

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

Computation Domain

Mathematical structure of secret info

- Usually ring structure defined by integer operation with modulus or Galois (finite) field
- Binary circuits or arithmetic circuits
    - Mod prime, mod power 2

Underlying Primitives

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

Security model	Mod prime / GF(2^n)	Mod 2^k	Bin. SS	Garbling
Malicious, dishonest majority	MASCOT / LowGear / HighGear	SPDZ2k	Tiny / Tinier	BMR
Covert, dishonest majority	CowGear / ChaiGear	N/A	N/A	N/A
Semi-honest, dishonest majority	Semi / Hemi / Temi / Soho	Semi2k	SemiBin	Yao's GC / BMR
Malicious, honest majority	Shamir / Rep3 / PS / SY	Brain / Rep3 / PS / SY	Rep3 / CCD / PS	BMR
Semi-honest, honest majority	Shamir / ATLAS / Rep3	Rep3	Rep3 / CCD	BMR
Malicious, honest supermajority	Rep4	Rep4	Rep4	N/A
Semi-honest, dealer	Dealer	Dealer	Dealer	N/A

Table of supported protocols

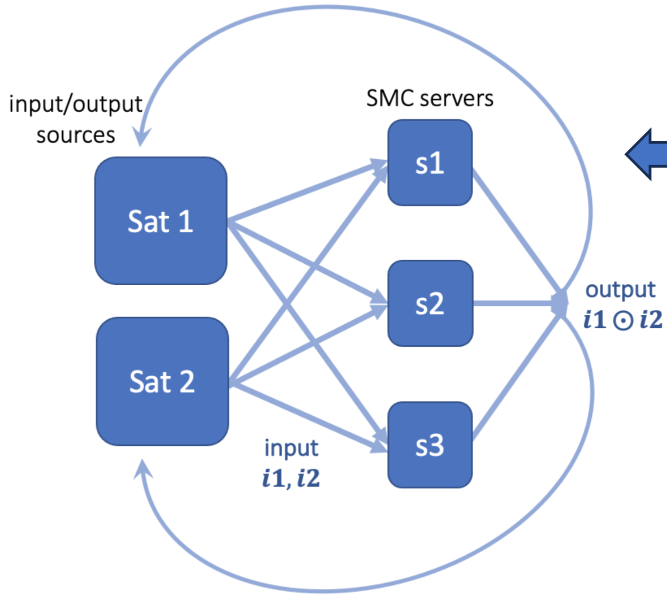
MP-SPDZ vs. Sharemind

Sharemind	MP-SPDZ
Ease of use for industry & non-security professionals	Prominent tool for academic research uses
C++ and proprietary SecreC code	Python
1 SMC approach – linear secret sharing (3+ parties)	Over 30 SMC variants (GC, OT, FHE, SS)
1 security model (semi-honest)	3 security models (semi-honest, malicious, covert)
1 trust option (honest majority)	2 trust options (honest or dishonest majority)
Black box – cannot see or modify source code	White box – can see and modify source code

SMC on Satellites

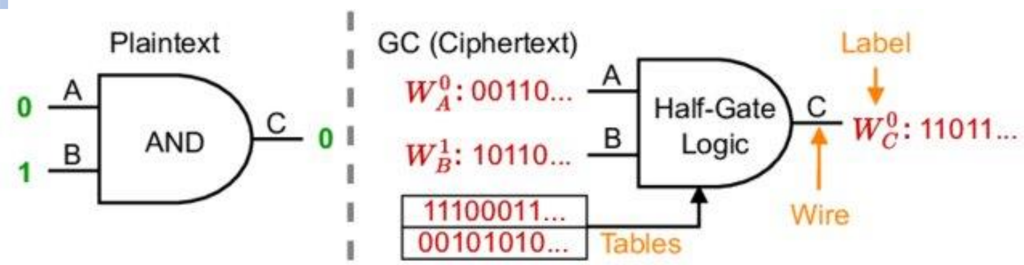
Our purpose

- Optimize protocol/variants for specific operations, informed by satellite algorithms that need privacy
- Demonstrate reasonable **efficiency** for each satellite operation
- Guarantees of **privacy** & **correctness** for each



3 Party SMC: secret sharing

2 party SMC: garbled circuits



10.48550/arXiv.2211.13324

Methodology: Software

Security	Parties	Modulo	Protocol
Semi-honest	3	2^{64}	[AFL+16]
		128-bit prime	[AFL+16]
		128-bit prime	[CDM00]
	2	2^{64}	[DEF+19]-
		128-bit prime	[KOS16]-
		128-bit prime	[KPR18]- (semi HE)
		128-bit prime	[KPR18]- (somewhat HE)
Covert	2	128-bit prime	[KPR18]* (semi HE)
			[KPR18]* (somewhat HE)
Malicious	4	2^{64}	[DEK20]
Malicious	3	2^{64}	[EKO+20] (post-sacrifice)
		2^{64}	[ADEN19]
		128-bit prime	[LN17] (replicated)
		128-bit prime	[CGH+18] (replicated)
		128-bit prime	[LN17] (Shamir)
		128-bit prime	[CGH+18] (Shamir)
	2	2^{64}	[DEF+19]
		128-bit prime	[KOS16]

https://eprint.iacr.org/2020/521.pdf

Integrating SMC into satellite operations

- Testing different RPO algorithms
  - Quadratic Program
  - Conjunction Analysis

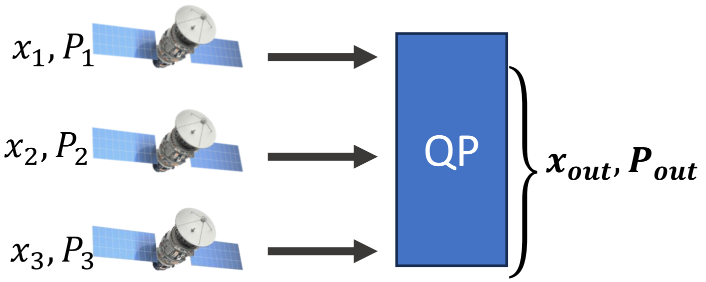
Software toolkit

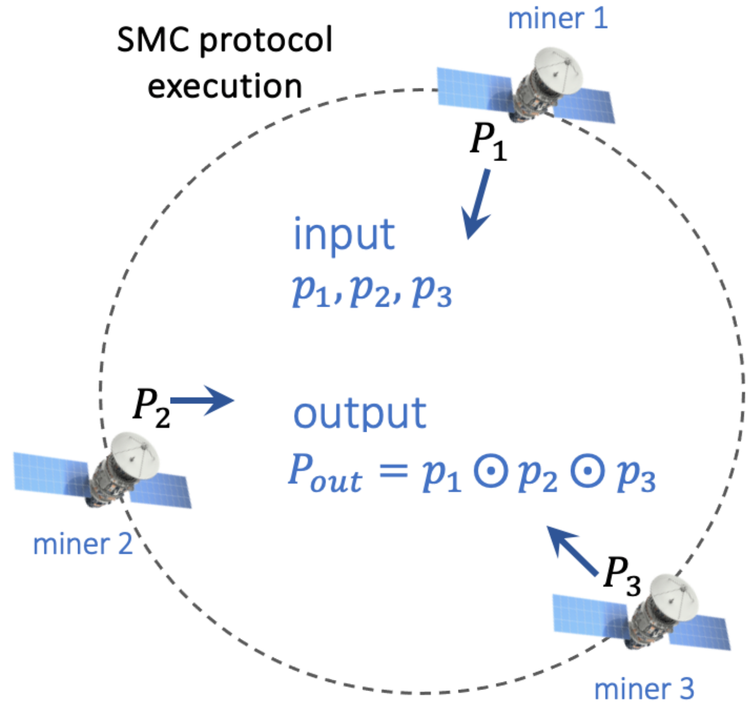
- MP-SPDZ
  - Platform for 30+ SMC operations
  - System of libraries based in python, designed for easy, even comparison between protocols variants

Quadratic Program: multi-point inspection

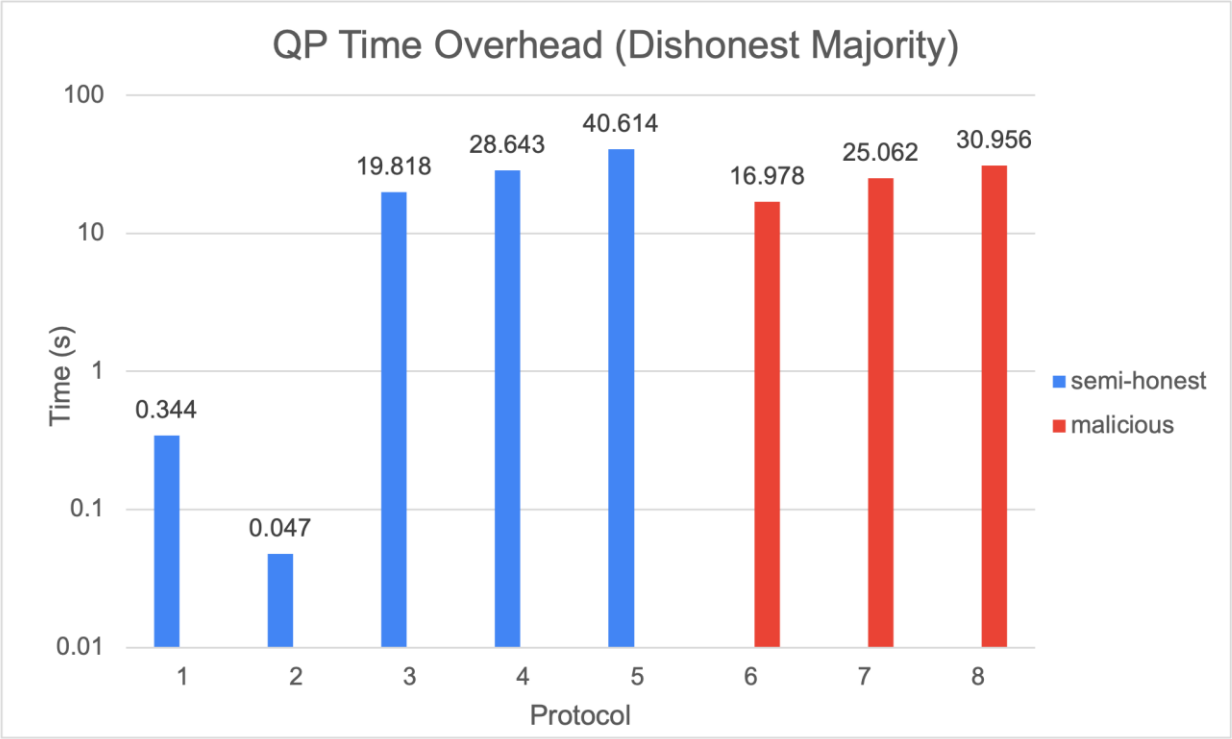
- Sensor Fusion optimization algorithm
- Need 3+ parties for 3 dimensional accuracy (secret sharing or homomorphic encryption)

- INPUT: position vector, *x*, and uncertainty matrix, *P*, for each satellite (only P is private)
- OUTPUT: optimized/most accurate {x, P} pair





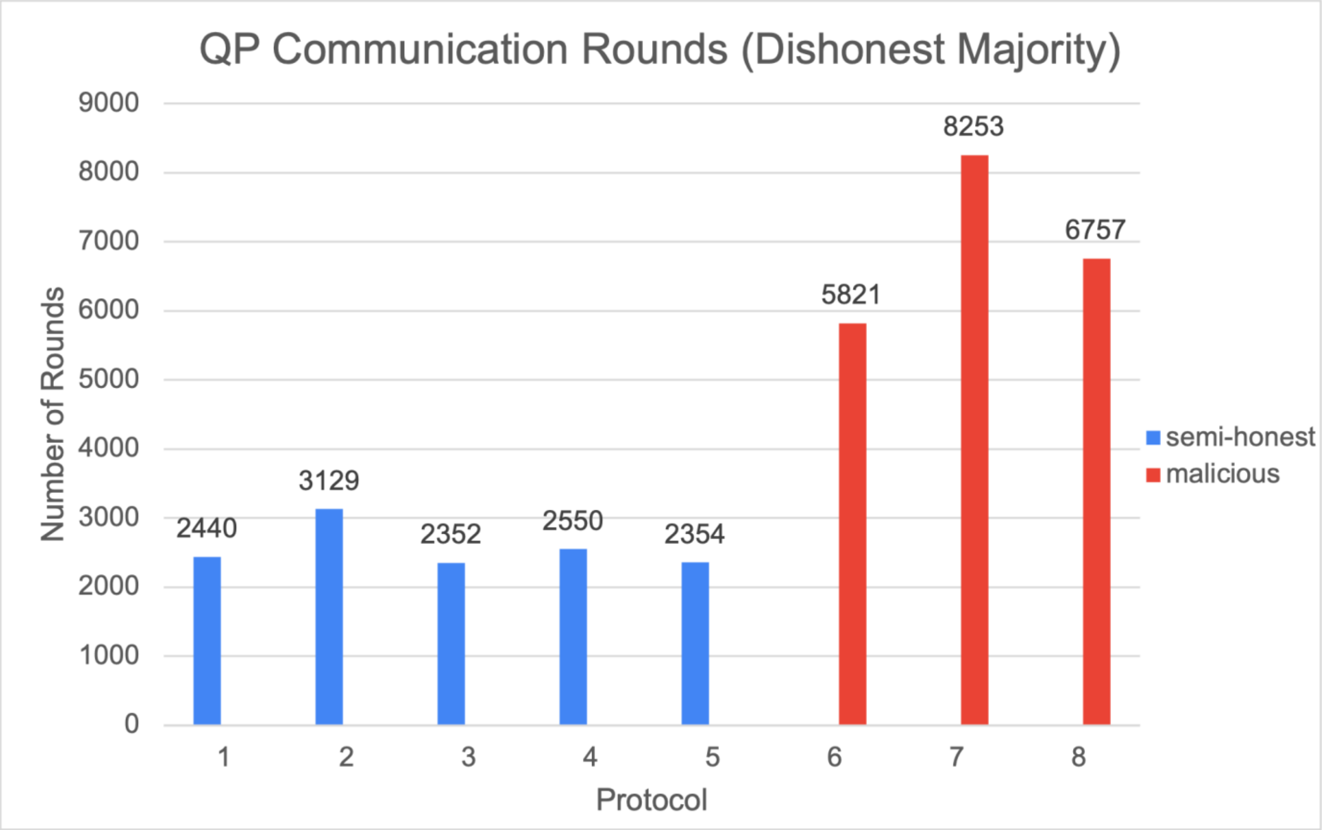
Evaluation: Quadratic Program



1-8 correspond to different arithmetic circuit protocols

Most efficient → #2 = semi2k, modulo 2^k oblivious transfer-based protocol

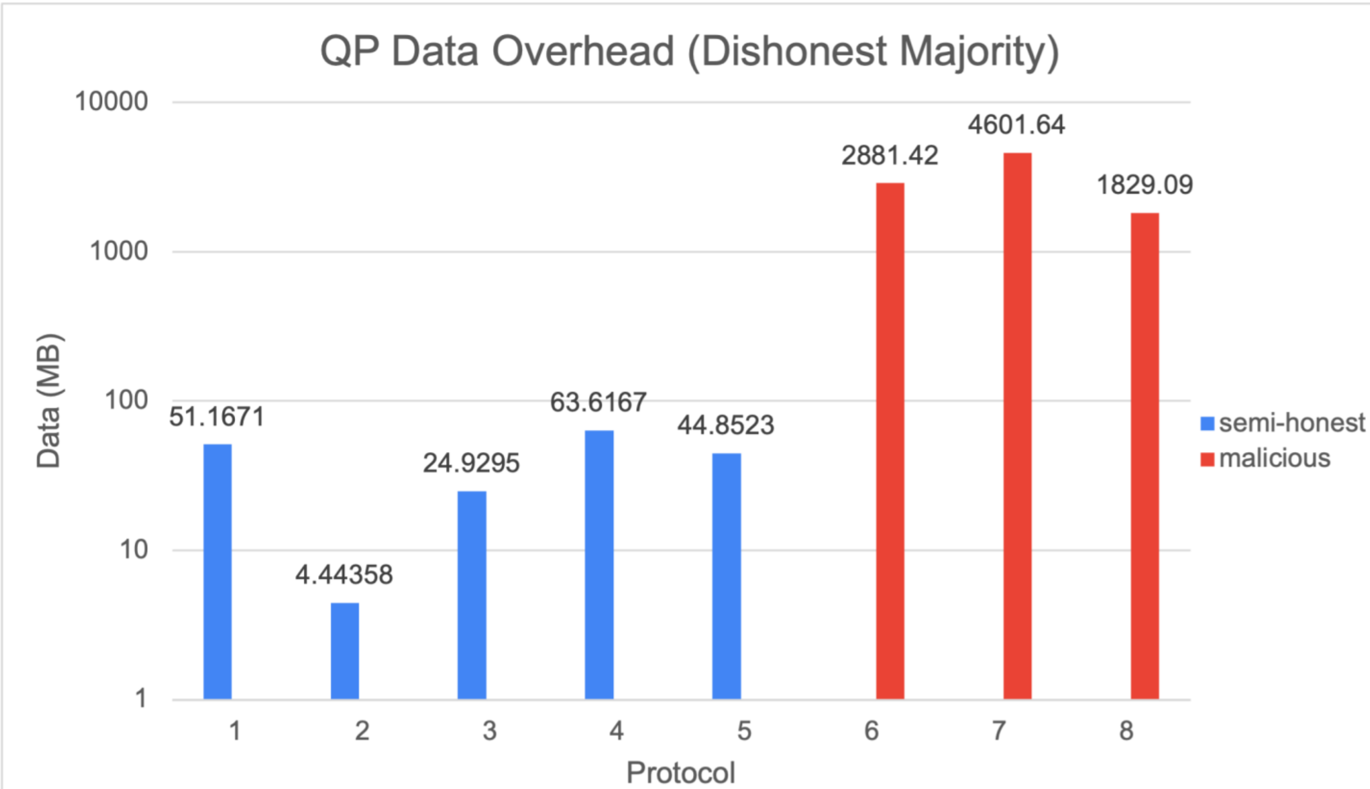
Evaluation: Quadratic Program



- Higher number of rounds for Malicious model

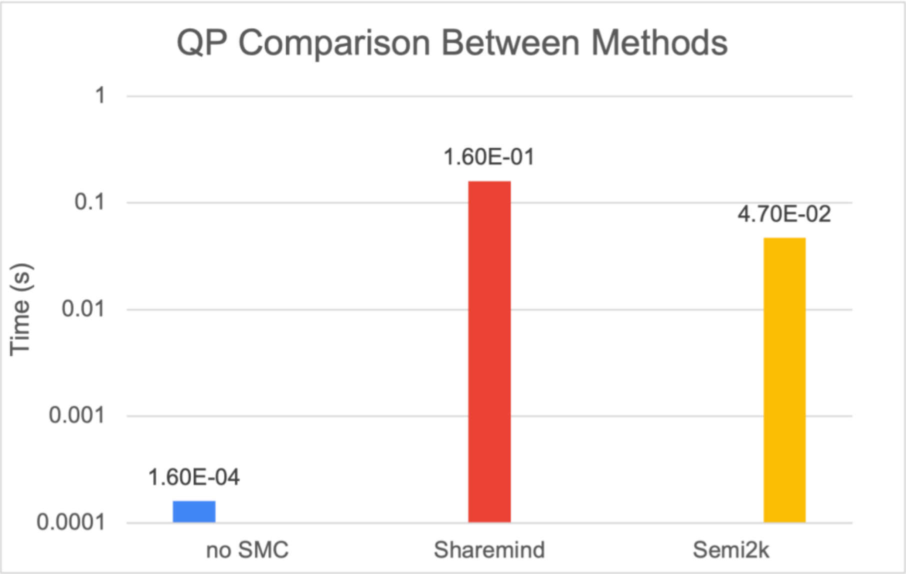
- significant factor in space applications

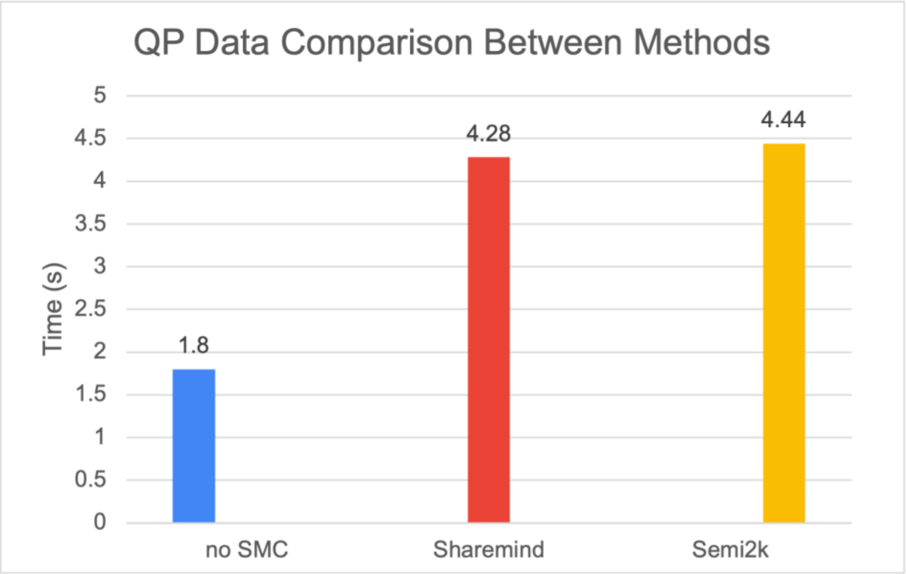
Evaluation: Quadratic Program



Most efficient time *and* memory → #2 = semi2k, modulo 2^k oblivious transfer-based protocol

Evaluation: Quadratic Program





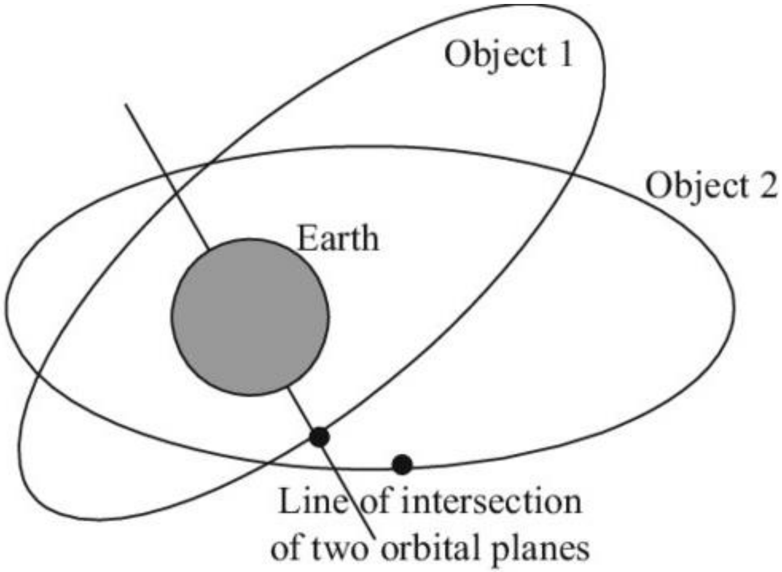
Further motivation for MP-SPDZ

- approx. 1 order of magnitude improvement over Sharemind
- about 2 orders of magnitude greater than without SMC (state-of-the-art)
- QP: need < 10 s to compute. This is < 0.1 s so well within efficiency

Algorithm 2: Alfano's Method

Alfano's Method: conjunction analysis

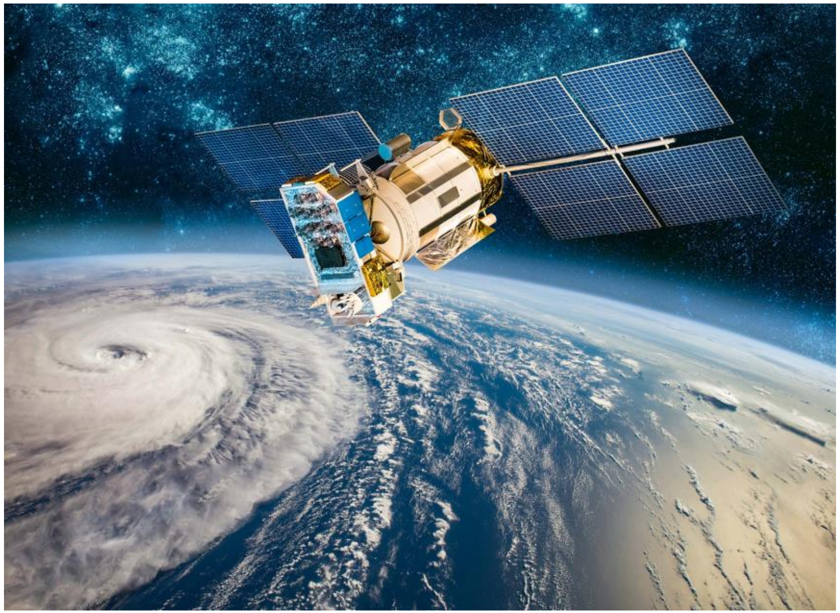
- Calculate collision probability between two spherical objects
- Assume linear orbital dynamics: one satellite stationary relative to other
- 2 party SMC problem, no trusted 3rd party (Garbled Circuits or Oblivious Transfer)
- INPUT = $\{x_{1,2}, v_{1,2}, R_{1,2}, C_{1,2}\}$ for satellite *1* and *2*, only covariance matrices, $C_{1,2}$, are private
- OUTPUT = p, probability of collision



$$p \leftarrow \frac{1}{2\pi\sigma_x\sigma_y}\int_{-R}^{R}\int_{-\sqrt{R^2-x^2}}^{\sqrt{R^2-x^2}} f(x,y)dydx \qquad \text{where} \qquad f(x,y) = \exp\left[\frac{-1}{2}\left[\left(\frac{x-x_m}{\sigma_x}\right)^2 + \left(\frac{y-y_m}{\sigma_y}\right)^2\right]\right]$$

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

Takeaway



Source: verdict.co.uk

Current Work:

- Optimizing MP-SPDZ protocols for QP algorithm
- Testing Alfano's method with MP-SPDZ
- AIAA (SciTech) Conference paper accepted

Future Work:

- Cybersecurity Conference paper in February (USENIX '24)
- Further examinations of space characterization issue and areas where privacy can be beneficial

Acknowledgements

Kevin Butler

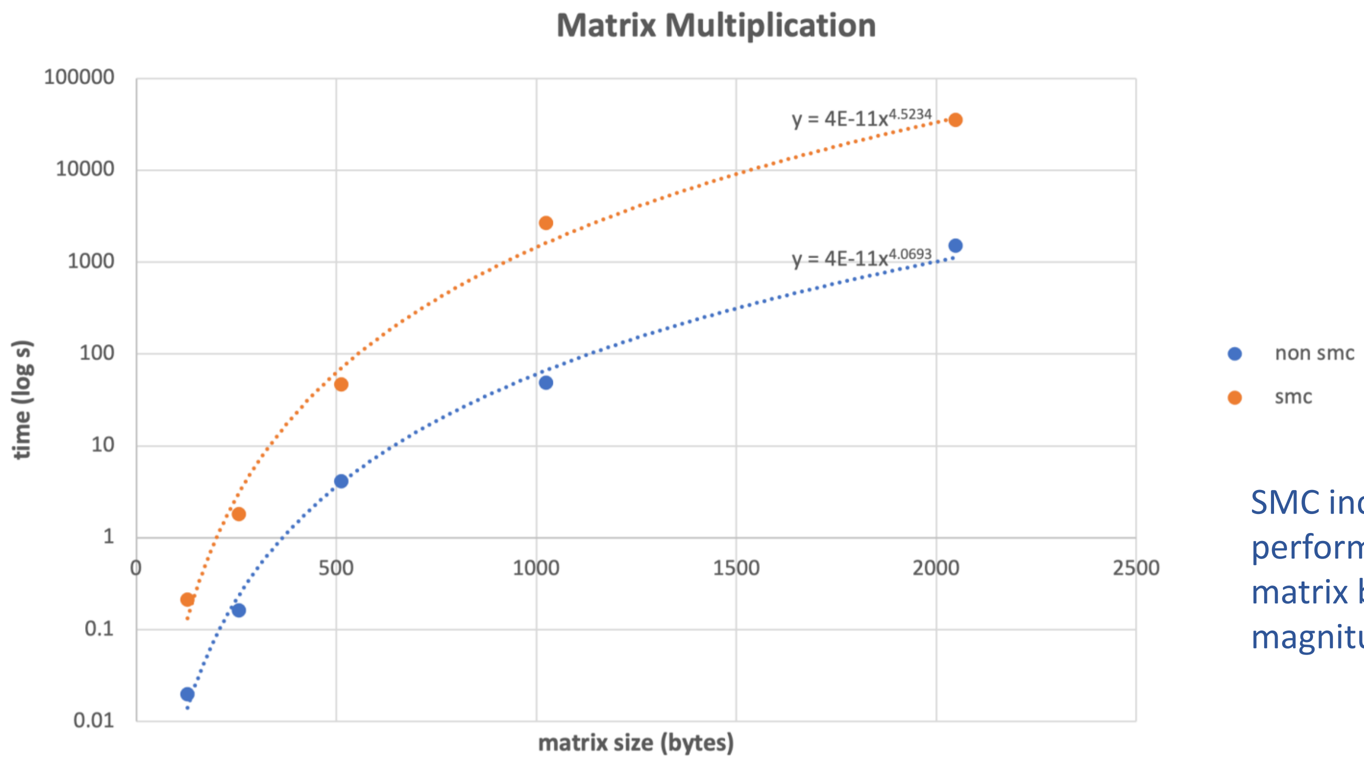
Tyler Lovelly

Chris Petersen

Carson Stillman

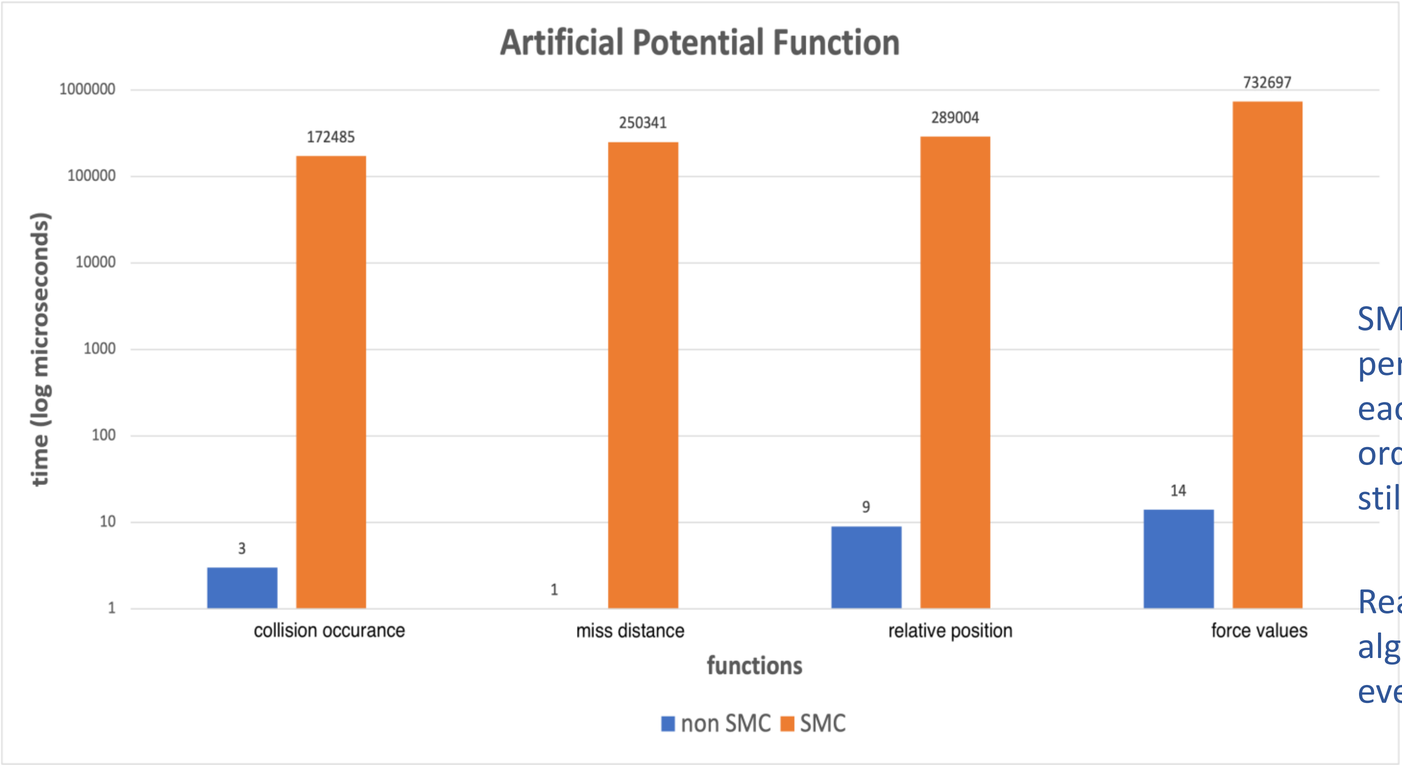
Questions?

Evaluation: Matrix Multiplication



SMC increases time to perform algorithm on each matrix by 1-1.5 orders of magnitude

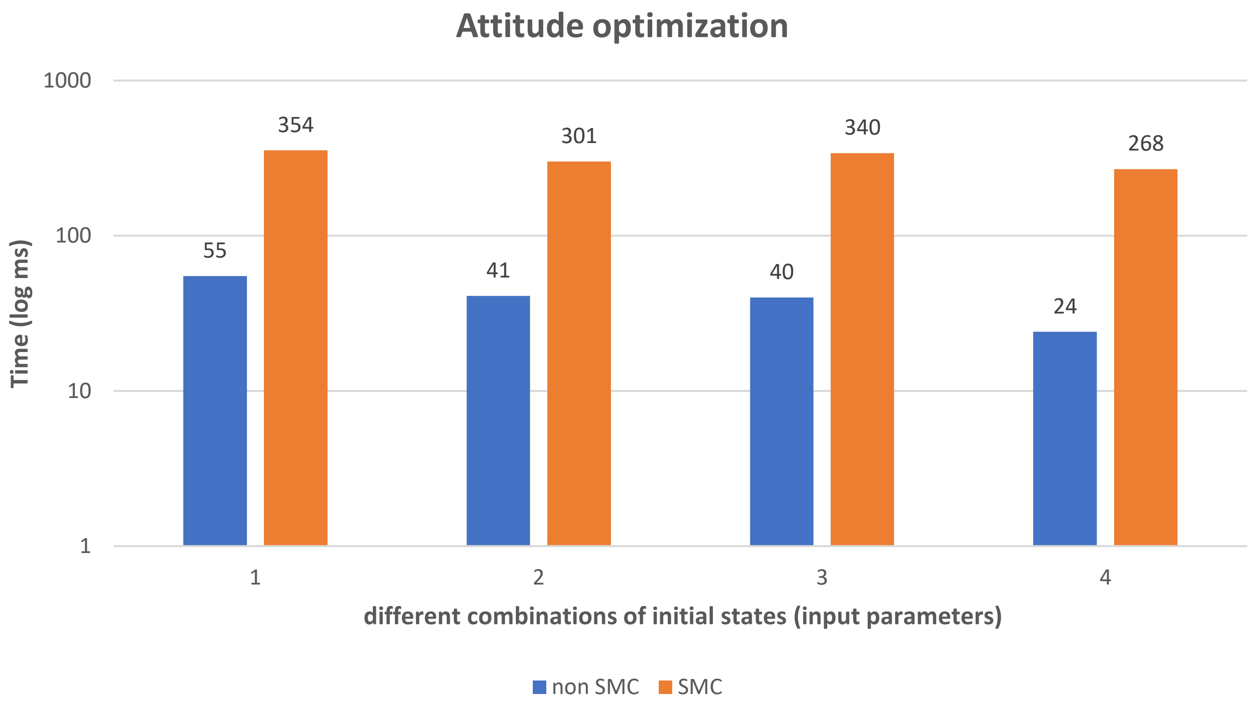
Evaluation: APF



SMC increases time to perform algorithm on each function by 4-5 orders of magnitude, still < 1s to execute

Reasonable since algorithm refreshes every 30 sec– 15 min

Evaluation: Optimization

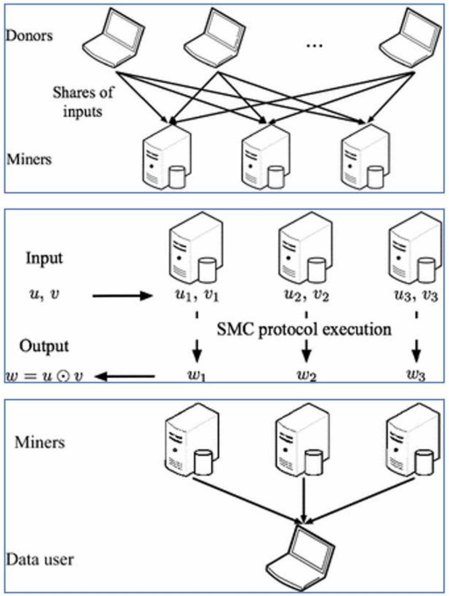


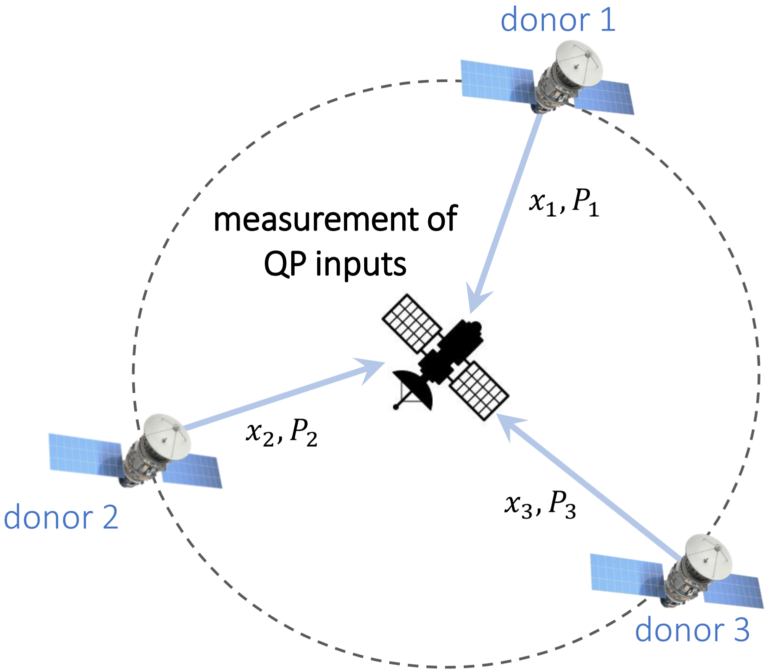
SMC increases time to perform algorithm on each function by ~1 order of magnitude

Algorithm refreshes every 10 seconds, still reasonable to use SMC

Optimization Algorithm

satellite setup

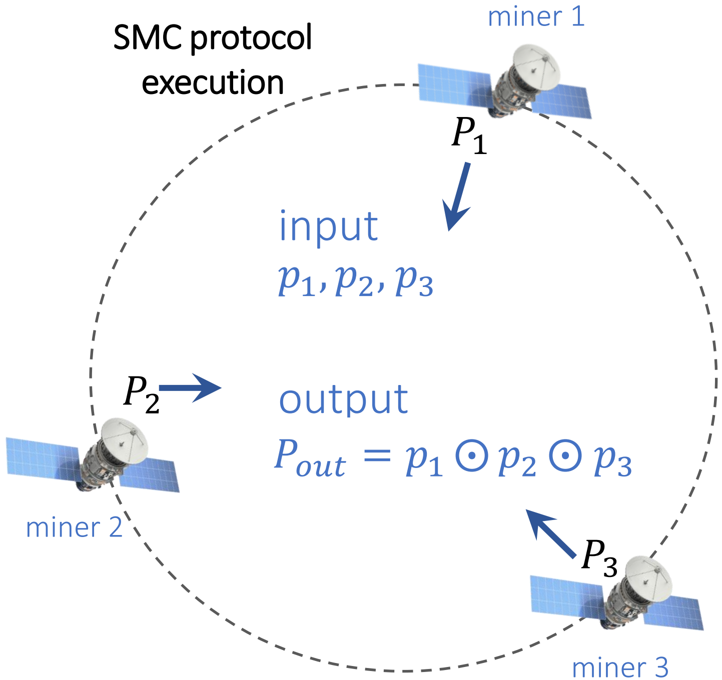


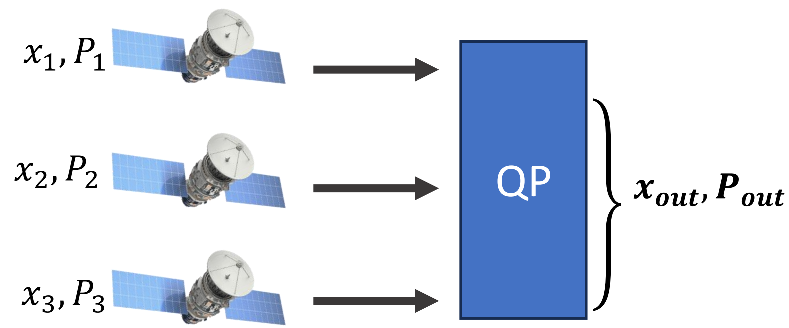


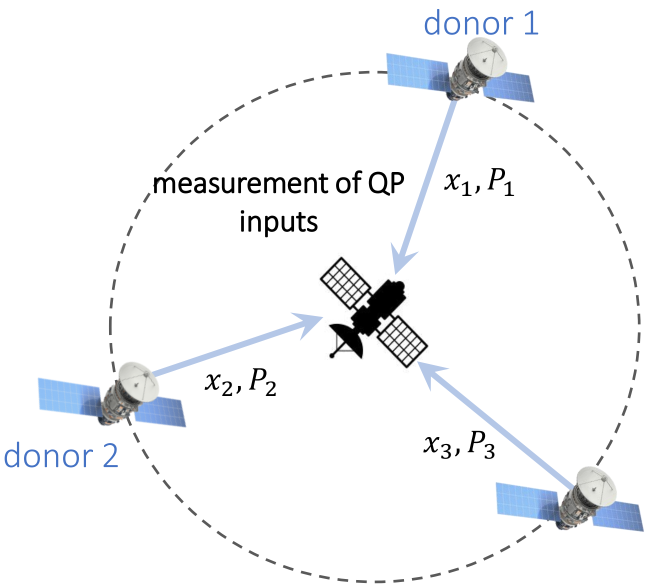
(p_1, p_2)

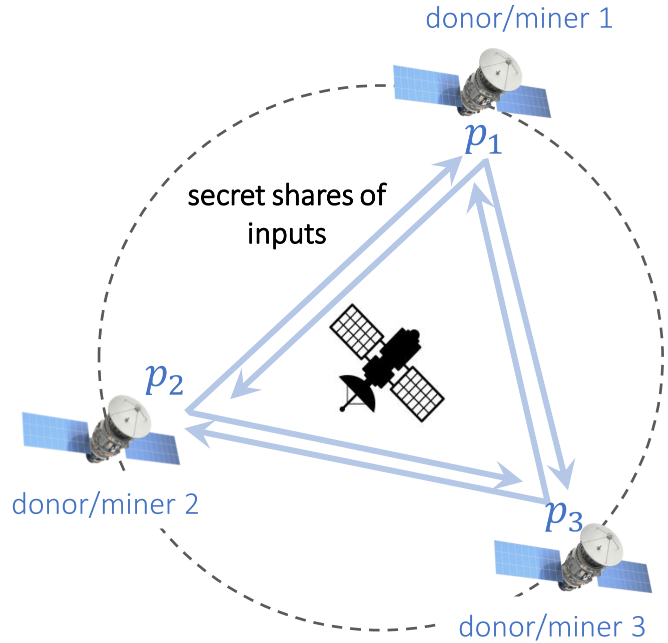
(p_1, p_3)

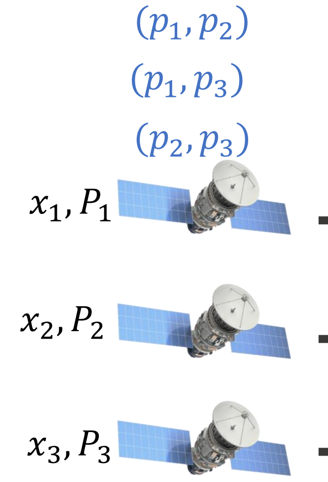
(p_2, p_3)






donor 1
measurement of QP inputs
x_1, P_1
x_2, P_2
x_3, P_3
donor 2

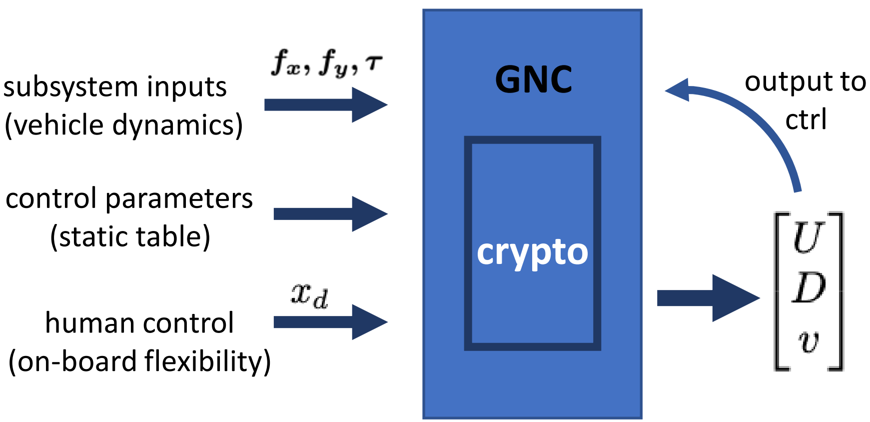

donor/miner 1
p_1
secret shares of inputs
p_2
p_3
donor/miner 2
donor/miner 3


(p_1, p_2)
(p_1, p_3)
(p_2, p_3)
x_1, P_1
x_2, P_2
x_3, P_3

Docking Algorithm

Another example: Artificial Potential Function (APF)

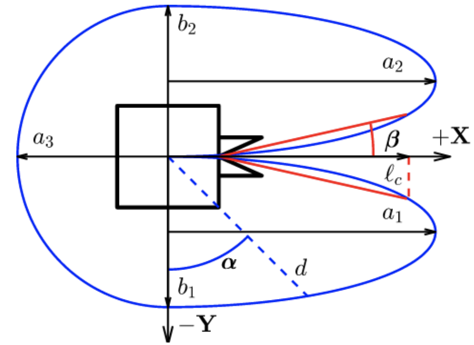
- Scenario: docking & collision avoidance at close range
  - On-board trajectory control
- Linear (relative) equations of motion



$$\begin{bmatrix} U \\ D \\ v \end{bmatrix} = apf(f_x, f_y, \tau, x_d, \ldots, \text{static params})$$

encrypted values

Keep-out zone potential



$$\phi_r = \sum_{i=1}^{N} \psi_i \exp\left[-\frac{\mathbf{r}_{co_i}^T \mathbf{N}_i \mathbf{r}_{co_i}}{\sigma_i}\right]$$

10.1109/TCST.2018.2866963

<u>Ground</u> Strong security Well-understood threat model Easier to manage/update systems	**<u>Link</u>** Strong security Well-understood threat model Established comms security practice
<u>User</u> Strong ground security Weak space security Human control/interaction Need in-space cooperation