Resilient Monotone Sequential Maximization

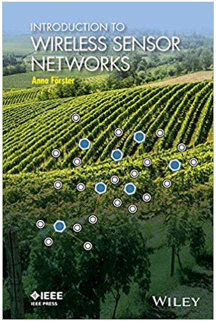
Vasileios Tzoumas
Post-doctoral associate
Massachusetts Institute of Technology
with
Ali Jadbabaie, George J. Pappas

Scheduling problems in control and sensing

Scheduling problems in control and sensing

Scheduling sensors for optimal Kalman filtering

Goal: Minimize minimum mean square error $\sum_{t=1}^{T} \text{trace}(\Sigma_t)$ by selecting different sensors to operate at each $t = 1, \ldots, T$.



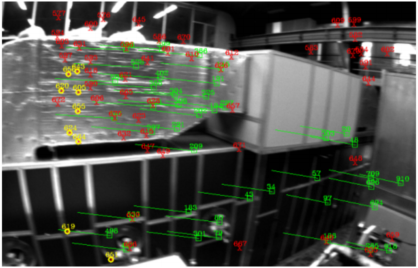
Complication: Bandwidth/battery considerations.
Problem: Schedule few sensors to activate at each step to achieve goal.

[1] [Gupta et al, Automatica'06]; [Vitus et al., Automatica'12].

Scheduling problems in control and sensing

Task-driven sensor scheduling for autonomous navigation

Goal: Minimize LQG cost by using different deployed sensors in the environment, as well as visual features, at each $t = 1, \ldots, T$.



Carlone and Karaman, IEEE TRO '18

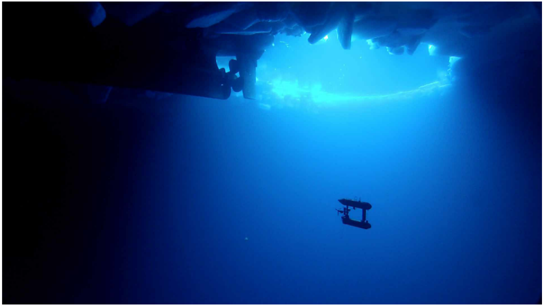
Complication: Power/computation limitations.
Situation: Not all sensing data are relevant to the task.
Problem: At each step activate only informative sensors towards goal.[1]

[1] [T, Carlone, Pappas, Jadbabaie, ACC'18]; [Pacelli, Majumdar, arxiv'18].

Scheduling problems in control and sensing

Scheduling motion plan for active information gathering

Goal: Maximize information about a process of interest by deploying team of mobile robots across a period of time.



Yang et al., Science Robotics '18

Situation: Each robot has discretized motion space.
Problem: Schedule robots' joint motion plan to achieve goal.[1]

[1] [Tokekar, IROS'14; Atanasov, ICRA'14; Corah, Michael, Aut. Robots '18].

Scheduling problems in control and sensing

All previous are monotone **sequential** maximization problems

Monotone sequential maximization:

Given:

- time horizon T;
- available sensors $\mathcal{V}_t$ at each time $t = 1, \ldots, T$;
- estimation objective f;
- sensing budgets α_t,

solve:

$$\max_{\mathcal{A}_1 \subseteq \mathcal{V}_1, |\mathcal{A}_1| \leq \alpha_1} \cdots \max_{\mathcal{A}_T \subseteq \mathcal{V}_T, |\mathcal{A}_T| \leq \alpha_T} f(\mathcal{A}_1, \ldots, \mathcal{A}_T).$$

[1] Additional contributions in control and sensing by: Bushnell; Clark; Cortes; Jovanovic; Krause; Le Ny; Mo; Motee; Olshevsky; Pasqualetti; Pequito; Poovendran; Roy; Sinopoli; Siami; Smith; Summers; Sundaram; Tomlin; Zampieri; ...

Sensors fail; features get occluded; robots get attacked[1]

Denial of Service in Sensor Networks

Unless their developers take security into account at design time, sensor networks and the protocols they depend on will remain vulnerable to denial-of-service attacks.

Sensor networks hold the promise of facilitating large-scale, real-time data processing in complex environments. Their foreseeable applications will help protect and monitor military, environmental, safety-critical, or domestic infrastructures and resources. must form ad hoc relationships in a dense network with little or no preexisting infrastructure.

Protocols and algorithms operating in the network must support large-scale distribution, often with only localized interactions among nodes. The network must continue operating even after significant node

Wood and Stankovic, Computer '02

if we pick $\mathcal{A}_t$ to $\max_{|\mathcal{A}_t| \leq \alpha_t} f(\mathcal{A}_{1:T})$ → and then a $\mathcal{B}_t \subseteq \mathcal{A}_t$ gets deleted → we end up with $f(\mathcal{A}_{1:T} - \mathcal{B}_{1:T})$

[1] [Sless et al., AAMAS'14], [González-Banos et al., ICRA'02], [Roumeliotis et al., IROS'98].

Sensors fail; features get occluded; robots get attacked[1]

Denial of Service in Sensor Networks

Unless their developers take security into account at design time, sensor networks and the protocols they depend on will remain vulnerable to denial-of-service attacks.

Sensor networks hold the promise of facilitating large-scale, real-time data processing in complex environments. Their foreseeable applications will help protect and monitor military, environmental, safety-critical, or domestic infrastructures and resources. must form ad hoc relationships in a dense network with little or no preexisting infrastructure.

Protocols and algorithms operating in the network must support large-scale distribution, often with only localized interactions among nodes. The network must continue operating even after significant node

Wood and Stankovic, Computer '02

if we pick $\mathcal{A}_t$ to ~~$\max_{|\mathcal{A}_t| \leq \alpha_t} f(\mathcal{A}_{1:T})$~~ → and then a $\mathcal{B}_t \subseteq \mathcal{A}_t$ gets deleted → we end up with $f(\mathcal{A}_{1:T} - \mathcal{B}_{1:T})$

[1] [Sless et al., AAMAS'14], [González-Banos et al., ICRA'02], [Roumeliotis et al., IROS'98].

Resilient monotone sequential maximization

Problem

Given:

- *time horizon* T*;*
- *available sensors* $\mathcal{V}_t$ *at* $t = 1, \ldots, T$*;*
- *estimation objective* f *s.t. non-decreasing,* $f \geq 0$*, and* $f(\emptyset) = 0$*;* (e.g., Kalman filtering accuracy $\sum_{t=1}^{T} \text{trace}(\Sigma_t)$)
- *budgets* α_t *and* β_t *s.t.* $0 \leq \beta_t \leq \alpha_t \leq |\mathcal{V}_t|$*, for all* $t = 1, 2, \ldots, T$*,*

solve:

$$\max_{\mathcal{A}_1 \subseteq \mathcal{V}_1} \min_{\mathcal{B}_1 \subseteq \mathcal{A}_1} \cdots \max_{\mathcal{A}_T \subseteq \mathcal{V}_T} \min_{\mathcal{B}_T \subseteq \mathcal{A}_T} f(\mathcal{A}_1 - \mathcal{B}_1, \ldots, \mathcal{A}_T - \mathcal{B}_T),$$

such that:

$|\mathcal{A}_t| \leq \alpha_t$ *and* $|\mathcal{B}_t| \leq \beta_t$*, for all* $t = 1, \ldots, T$*.*

Symbol explanation:

- $\mathcal{A}_t$: selected sensors at time t;
- $\mathcal{B}_t$: failed sensors in $\mathcal{A}_t$.

Difficulty of problem

- Problem is at least NP-hard [Orlin et al., '16];
  e.g., inapproximable for Kalman filtering [Ye et al., ACC'18].

- No known poly-time algorithm:
  - E.g., greedy can perform arbitrarily bad [Orlin et al., '16], since it is oblivious to failures:

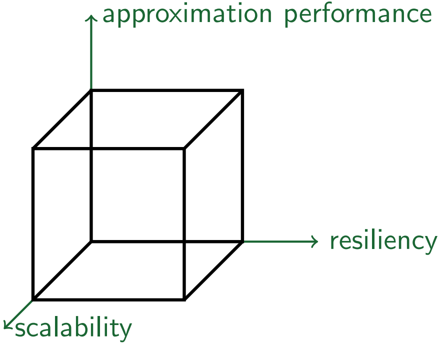
Greedy [Fisher et al., 1978]

1: $\mathcal{A}_t \leftarrow \emptyset$ for all $t = 1, \ldots, T$;
2: $\mathcal{A}_{1:T} \leftarrow \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_T$;
3: **while** $|\mathcal{A}_t| \leq \alpha_1$ or $\ldots$ or $|\mathcal{A}_T| \leq \alpha_T$ **do**
4: $\quad x \in \arg\max_{y \in \mathcal{V} \setminus \mathcal{A}_{1:T}} f(\mathcal{A}_{1:T} \cup \{y\})$;
5: $\quad \mathcal{A}_{1:T} \leftarrow \mathcal{A}_{1:T} \cup \{x\}$;
6: **end while**

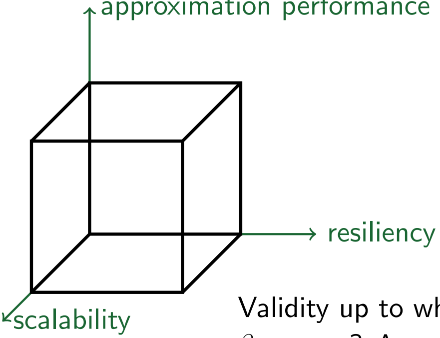
Output: Sets $\mathcal{A}_1, \ldots, \mathcal{A}_T$.

- f in control and sensing is usually non-submodular:
  - Min. mean square error of Kalman filter [Ye, Roy, Sundaram, ACC'18]
  - trace of inverse of controllability Gramian [Olshevsky, TCNS '17];
  - LQG cost [Summers, CDC '17; Tzoumas et al, arxiv '18];

Characteristics of good algorithm



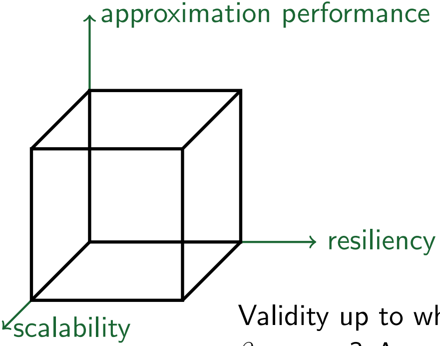
Characteristics of good algorithm



Validity up to what numbers β_t of failures?

$\underbrace{\beta_t \ll \alpha_t}_{\text{low resiliency}}$? $\underbrace{\text{Any } \beta_t \leq \alpha_t}_{\text{high resiliency}}$?

Characteristics of good algorithm

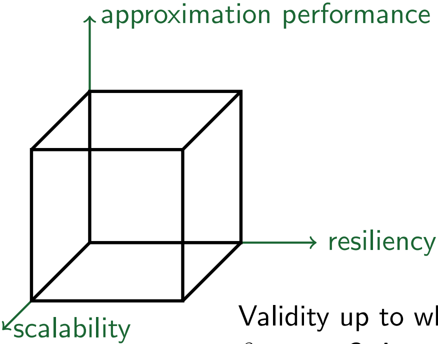


Validity up to what numbers β_t of failures?

$\underbrace{\beta_t \ll \alpha_t}_{\text{low resiliency}}$? $\underbrace{\text{Any } \beta_t \le \alpha_t}_{\text{high resiliency}}$?

Running time w.r.t. α_t, β_t, $|\mathcal{V}_t|$? Exponential? Linear time?

Characteristics of good algorithm



f_{approx} finitely close to f_{opt}?

$$\underbrace{\frac{f_{\text{approx}}}{f_{\text{opt}}} \geq 0?}_{\text{no guarantee}} \quad \underbrace{\frac{f_{\text{approx}}}{f_{\text{opt}}} = 1?}_{\text{exact}}$$

Validity up to what numbers β_t of failures?

$\underbrace{\beta_t \ll \alpha_t}_{\text{low resiliency}}$? $\underbrace{\text{Any } \beta_t \leq \alpha_t}_{\text{high resiliency}}$?

Running time w.r.t. α_t, β_t, $|\mathcal{V}_t|$? Exponential? Linear time?

Literature review

Earlier work considers **non-sequential** variant (for **placement** instead of scheduling):

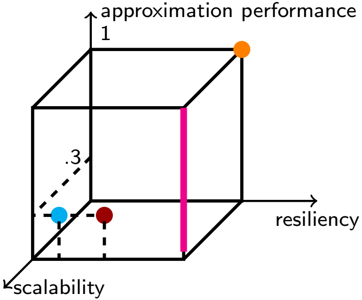
$$\max_{\mathcal{A}\subseteq\mathcal{V}, |\mathcal{A}|\leq\alpha} \; \min_{\mathcal{B}\subseteq\mathcal{A}, |\mathcal{B}|\leq\beta} \; f(\mathcal{A}-\mathcal{B}).$$

Literature review

Earlier work considers **non-sequential** variant (for **placement** instead of scheduling):

$$\max_{\mathcal{A}\subseteq\mathcal{V},|\mathcal{A}|\leq\alpha}\ \min_{\mathcal{B}\subseteq\mathcal{A},|\mathcal{B}|\leq\beta}\ f(\mathcal{A}-\mathcal{B}).$$

- For f **monotone** and **submodular**, algorithms by:
  - Krause et al., JMLR '08: Exponential time $O(\alpha^\beta)$.
  - Orlin et al., '16, Cevher et al., ICML '18: Low resiliency $\beta \leq \sqrt{\alpha}$, $\alpha/\log\alpha$.
  - Tzoumas et al., IEEE CDC '17: Guarantee for f with bounded curvature.

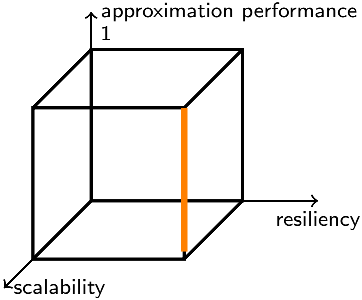


Literature review

Earlier work considers **non-sequential** variant (for **placement** instead of scheduling):

$$\max_{\mathcal{A}\subseteq\mathcal{V},|\mathcal{A}|\leq\alpha} \; \min_{\mathcal{B}\subseteq\mathcal{A},|\mathcal{B}|\leq\beta} \; f(\mathcal{A}-\mathcal{B}).$$

- For **monotone** functions with **bounded curvature**, using algorithm by Tzoumas et al. CDC '17:
  - Cevher et al., NeurIPS '18: Cardinality Constraint.
  - Tzoumas et al., arxiv '18: Any matroid constraint.



Our claim of innovation

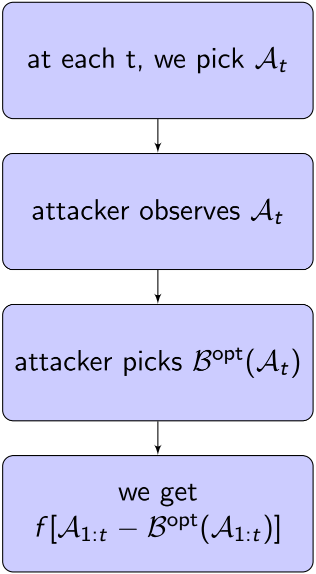
First algorithm for Problem s.t.:

- *Adaptive*: $\mathcal{A}_t$ is adapted to history of attacks $\mathcal{B}_1, \ldots, \mathcal{B}_{t-1}$;
- *Scalable:* $O[(\alpha_t - \beta_t)|\mathcal{V}|]$ running time per step $t = 1, \ldots, T$;
- *Resilient:* valid for any number of failures $\beta_{1:T}$;
- *Non-zero approximation performance:* guaranteed performance for any monotone function with bounded curvature.

Algorithm

Notation:

- $\mathcal{A}_{1:t} = \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_t$.



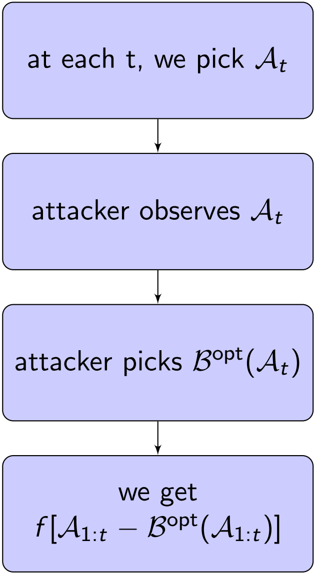
Algorithm

Idea: Given $\mathcal{B}_{1:t-1}$ do:

Pick $\mathcal{A}_t = \mathcal{A}_1 \cup \mathcal{A}_2$

"bait"
$|\mathcal{A}_1| = \beta_t$.

$\mathcal{A}_2 \subseteq \mathcal{V}_t - \mathcal{A}_1$;
$|\mathcal{A}_2| = \alpha_t - \beta_t$.



Algorithm[1]

Idea: Given $\mathcal{B}_{1:t-1}$ do:

Pick $\mathcal{A}_t = \mathcal{A}_1 \cup \mathcal{A}_2$

"bait"
$|\mathcal{A}_1| = \beta_t$.

$\mathcal{A}_2 \subseteq \mathcal{V}_t - \mathcal{A}_1$;
$|\mathcal{A}_2| = \alpha_t - \beta_t$.

Order $\mathcal{V}_t = \{z_1, \ldots, z_{|\mathcal{V}|}\}$ s.t.: $f(z_1) \geq \ldots \geq f(z_{|\mathcal{V}_t|})$

↓

Pick bait $\mathcal{A}_1 = \{z_1, \ldots, z_{\beta_t}\}$

↓

Pick $\mathcal{A}_2$ greedily from $\mathcal{V}_t - \mathcal{A}_1$ to maximize $f(\mathcal{A}_{1:t-1} - \mathcal{B}_{1:t-1}, \mathcal{A}_2)$

↓

Return $\mathcal{A}_t = \mathcal{A}_1 \cup \mathcal{A}_2$

[1] Making adaptive the algorithm in [T, Gatsis, Jadbabaie, Pappas, CDC '17].

Total curvature of monotone f

Definition [Sviridenko et al., arxiv'13]:[1]
For **monotone** f, then f's total curvature is defined as:

$$c_f \triangleq 1 - \min_{v \in \mathcal{V}} \min_{\mathcal{A},\mathcal{B} \subseteq \mathcal{V} \setminus \{v\}} \frac{f(\{v\} \cup \mathcal{A}) - f(\mathcal{A})}{f(\{v\} \cup \mathcal{B}) - f(\mathcal{B})}.$$

Interpretation:

- c_f measures how $\mathcal{V}$'s elements *substitute* each other;
- if f submodular $\Rightarrow c_f = \kappa_f$ [Conforti, Cornuéjols, Disc. Math.'84]:

$$\kappa_f \triangleq 1 - \min_{v \in \mathcal{V}} \frac{f(\mathcal{V}) - f(\mathcal{V} \setminus \{v\})}{f(\{v\})}.$$

[1] Other definitions: [Das, Kempe, ICML'11]; [Iyer et al., NeurIPS'13]; [Wang et al., Comb. Opt.'14].

Total curvature of monotone f

Definition [Sviridenko et al., arxiv'13]:
For **monotone** f, then f's total curvature is defined as:

$$c_f \triangleq 1 - \min_{v \in \mathcal{V}} \min_{\mathcal{A}, \mathcal{B} \subseteq \mathcal{V} \setminus \{v\}} \frac{f(\{v\} \cup \mathcal{A}) - f(\mathcal{A})}{f(\{v\} \cup \mathcal{B}) - f(\mathcal{B})}.$$

Properties:

- $c_f \leq 1$;
- NP-hard to compute $\Rightarrow$ approximated with upper bounds:
  - Restricted concave-smooth functions [Dimakis et al., Annals of Stat. '18];
  - Concave over modular functions [Iyer et al, NeurIPS '13];
  - Min. mean square error [Chamon, Ribeiro, GSIP '16, CDC '17];
  - LQG cost for task-driven sensing [T, Carlone, Pappas, Jadbabaie, '18];
  - LQG cost for actuation [Summers, Kamgarpour, CDC '18].

Algorithm's performance

Theorem

Algorithm:

- *(Resiliency) is valid for any* $0 \leq \beta_t \leq \alpha_t \leq |\mathcal{V}_t|$*;*
- *(Scalability) runs in* $(\alpha_t - \beta_t)|\mathcal{V}|$ *time;*
- *(Approximation performance) guarantees for f* **monotone***:*

$$\frac{f_{approx}}{f_{opt}} \geq (1 - c_f)^4.$$

Algorithm's performance

Theorem

Algorithm:

- *(Resiliency) is valid for any* $0 \leq \beta_t \leq \alpha_t \leq |\mathcal{V}_t|$*;*
- *(Scalability) runs in* $(\alpha_t - \beta_t)|\mathcal{V}|$ *time;*
- *(Approximation performance) guarantees for* f **monotone***:*

$$\frac{f_{approx}}{f_{opt}} \geq (1 - c_f)^4.$$

Tightness of bound:

- For $\max\limits_{|\mathcal{A}| \leq \alpha} f(\mathcal{A})$ best possible bound is $(1 - c_f)$ [Sviridenko et al,'13];
- For $\max\limits_{|\mathcal{A}| \leq \alpha} \min\limits_{\mathcal{B} \subseteq \mathcal{A}, |\mathcal{B}| \leq \beta} f(\mathcal{A} - \mathcal{B})$ best known bound is $(1 - c_f)^2$;[1]

[1] Achieved modifying proof in [Tzoumas et al., arxiv'18].

Functions with $c_f < 1$, and applications

Functions:

- Restricted concave-smooth functions [Dimakis et al., Ann. Stat. '18];
- Concave over modular functions [Iyer et al, NeurIPS '13];
- Min. mean square error [Chamon, Ribeiro, GSIP '16, CDC '17];
- LQG cost for task-driven sensing [T, Carlone, Pappas, Jadbabaie, '18];
- LQG cost for actuation [Summers, Kamgarpour, CDC '18].

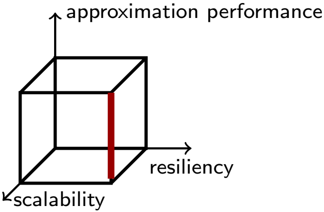
Applications (see also ref. above):

- Statistical learning:
  - Experiment design;
  - Data selection;
  - Dictionary selection.
- Control and sensing:
  - Task-driven sensor and actuator scheduling;
  - Active information gathering.
- Operations research:
  - Optimal budget allocation [Bian et al., ICML'17; Ahmed, Atamturk, Math. Prog. '11].

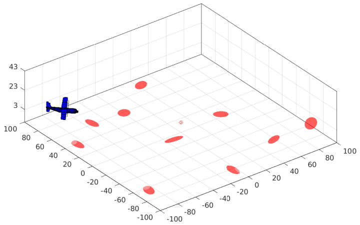
Summary of results

First algorithm for Problem s.t.:

- *Adaptive*: $\mathcal{A}_t$ is adapted to history of attacks $\mathcal{B}_1, \ldots, \mathcal{B}_{t-1}$;
- *Scalable:* $O[(\alpha_t - \beta_t)|\mathcal{V}_t|]$ running time per step $t = 1, \ldots, T$;
- *Resilient:* valid for any number of failures $\beta_{1:T}$;
- *Non-zero approximation performance:* guaranteed performance for any monotone function with bounded curvature.



Simulations: LQG control and sensing[1]



Scenario: UAV moves in a 3D space.

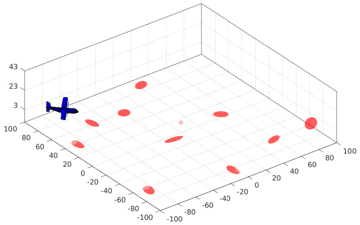
UAV's model: double-integrator, with state $x_k = [p_k, v_k]^\top$ where:

- p_k = position;
- v_k = velocity.

Objective: Land UAV at position [0, 0, 0] with 0 velocity.

[1][T, Carlone, Pappas, Jadbabaie, ACC'18; arxiv'18].

Simulations: LQG control and sensing[1]



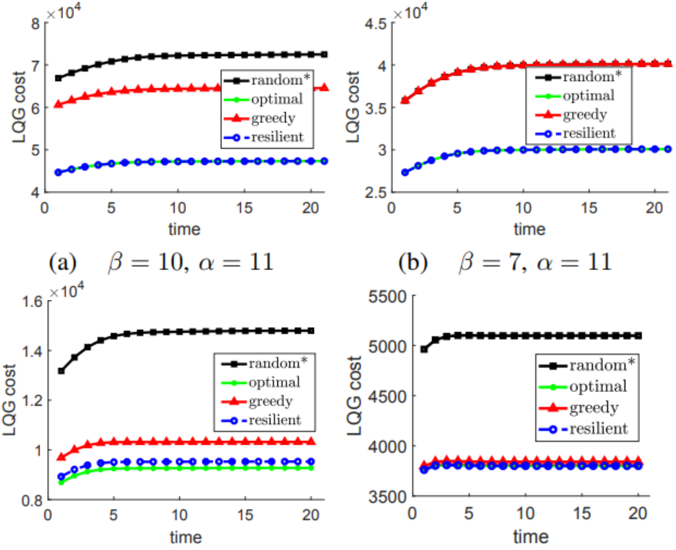
Available sensors and landmarks for localization:

- 1 GPS (measuring position);
- 1 altimeter;
- 1 stereo camera;
- 10 sensors on the ground;

Sensor selection metric: LQG cost.

[1] [T, Carlone, Pappas, Jadbabaie, ACC'18; arxiv'18].

Simulation results



(a) $\beta = 10$, $\alpha = 11$ (b) $\beta = 7$, $\alpha = 11$

(c) $\beta = 4$, $\alpha = 11$ (d) $\beta = 1$, $\alpha = 11$

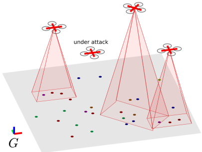
Summary, and extensions

First algorithm for Problem s.t.:

- *Adaptive*: $\mathcal{A}_t$ is adapted to history of attacks $\mathcal{B}_1, \ldots, \mathcal{B}_{t-1}$;
- *Scalable:* $O[(\alpha_t - \beta_t)|\mathcal{V}_t|]$ running time per step $t = 1, \ldots, T$;
- *Resilient:* valid for any number of failures $\beta_{1:T}$;
- *Non-zero approximation performance:* guaranteed performance for any monotone function with bounded curvature.

Where do we go from here:

- Beyond curvature: c_f can be provably < 1; is this enough?
- Resilient active target coverage: Active target coverage asks to max $\sum f_t$ where f_t unknown a priori;[1] f_t is the number of targets covered at time t.
  - Contrast that to $\sum \text{tr}(\Sigma_t)$ for Kalman filt., where $\sum \text{tr}(\Sigma_t)$ is computable given a sensor selection.



[1] [Zhou, Tzoumas, Pappas, Tokekar, IEEE RAL'19].