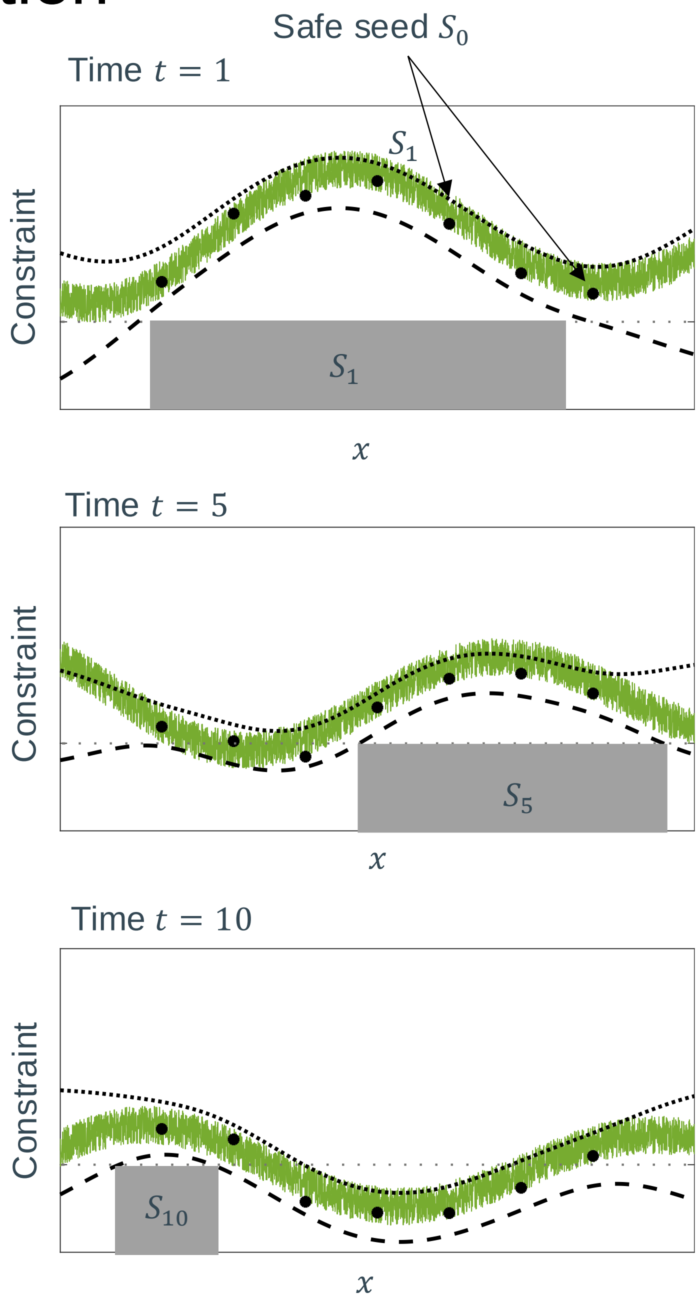
Safe Time-Varying Optimization based on Gaussian Processes with Spatio-Temporal Kernel

J. Li, M. Zagorowska, G. De Pasquale. A. Rupenyan, J. Lygeros

Research supported by NCCR Automation funded by the SNCF (grant no. 180545). M. Zagorowska acknowledges funding from the Marie Curie Horizon Postdoctoral Fellowship project RELIC (grant no 101063948) for writing, revisions, and the compressor case study. A. Rupenyan acknowledges support from the Johann Jakob Rieter foundation.

Motivation



- Solve

$$\max_{x} f(x,t)$$
$$\text{s.t. } c_i(x,t) \geq 0, i = 1, \dots, m$$

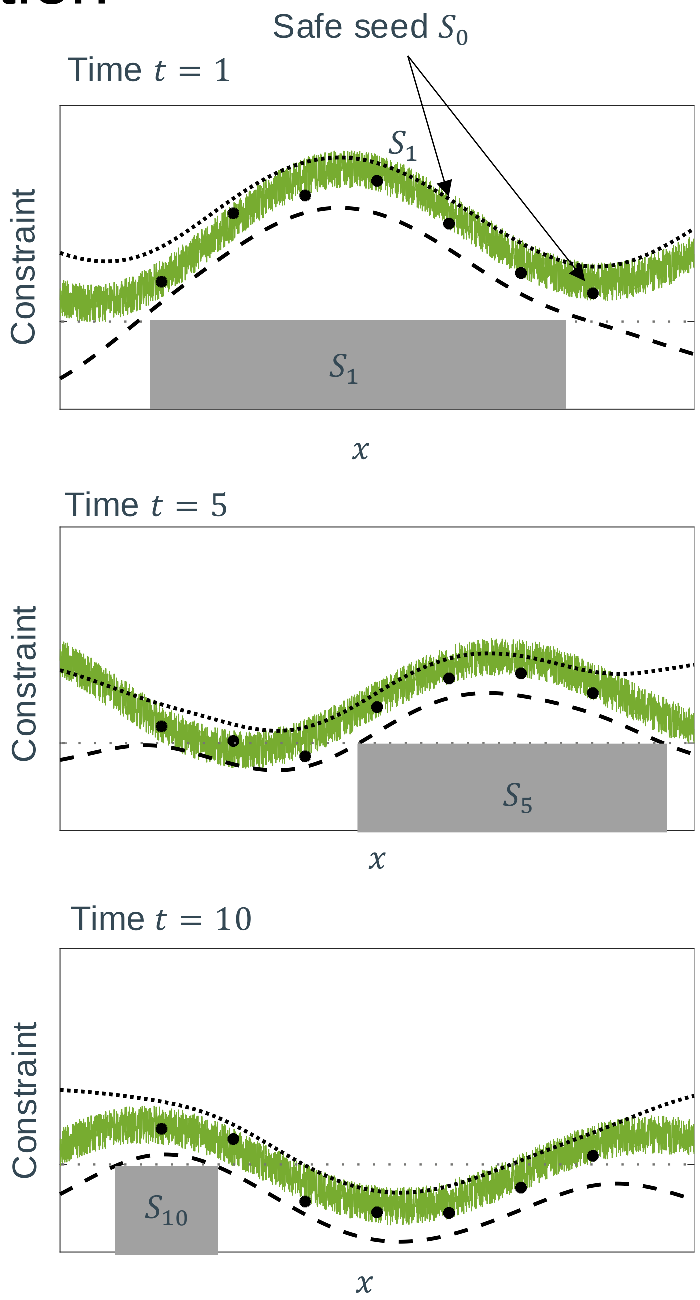
where the reward function and the constraints

- are **unknown** and can only be **sampled**,
- **change with time**

Challenges:

- Ensuring **safety** with time-varying constraints
- Finding **safe optimum** of time-varying objective

Motivation



- Solve

$$\max_{x} f(x,t)$$
$$\text{s.t. } c_i(x,t) \geq 0, i = 1, \dots, m$$

where the reward function and the constraints

- are **unknown** and can only **be sampled**
- **change with time**

Challenges:

- Ensuring **safety** with time-varying constraints
- Finding **safe optimum** of time-varying objective

Contribution	Handling changes in time	Safety	Optimality	Safe seed
TVSafeOPT	Spatio-temporal kernel	✓	✓	Only for $t = 0$

ETH zürich

TVSafeOPT – safe exploration

1. Provide **tight confidence intervals** $C_k(x, i)$ for the objective and constraints based on how **fast they change**, using the Lipschitz constant L

TVSafeOPT – safe exploration

1. Provide **tight confidence intervals** $C_k(x, i)$ for the objective and constraints based on how **fast they change**, using the Lipschitz constant L
2. Use the **lower** and the **upper bounds of** $C_k(x, i)$ to **update the safe set** S_k considering how **fast** the constraints i **change** with both **time** and **decisions**.

TVSafeOPT – safe exploration

1. Provide **tight confidence intervals** $C_k(x, i)$ for the objective and constraints based on how **fast they change**, using the Lipschitz constant L
2. Use the **lower** and the **upper bounds of** $C_k(x, i)$ to **update the safe set** S_k considering how **fast** the constraints i **change** with both **time** and **decisions**.
3. Extend safe exploration from SafeOPT to account for **time-varying settings** and pick the **most uncertain decision** using the **tight confidence interval** $C_k(x, i)$

TVSafeOPT – safe exploration

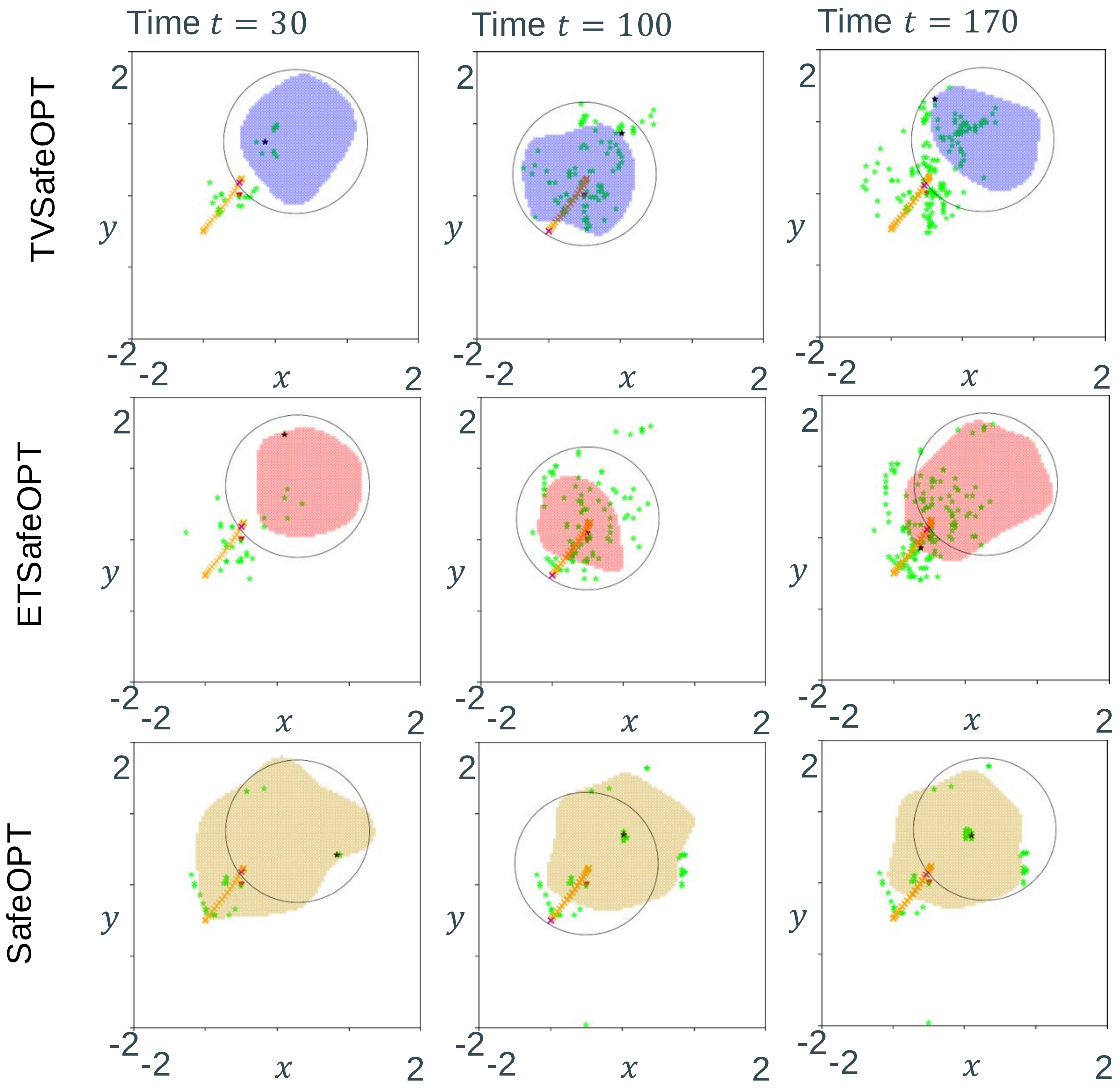
1. Provide **tight confidence intervals** $C_k(x, i)$ for the objective and constraints based on how **fast they change**, using the Lipschitz constant L
2. Use the **lower** and the **upper bounds of** $C_k(x, i)$ to **update the safe set** S_k considering how **fast** the constraints i **change** with both **time** and **decisions**.
3. Extend safe exploration from SafeOPT to account for **time-varying settings** and pick the **most uncertain decision** using the **tight confidence interval** $C_k(x, i)$

- **Theorem (informal):**
  For any $\delta \in (0,1)$, *every constraint* $c_i(x, t) \geq 0, i = 1, \ldots, m$, *holds at every time step* $t \geq 0$ *for all* $x \in S_k$ *with probability at least* $1 - \delta$, *if* $\sqrt{\beta_k} = B + \sigma\sqrt{2(\gamma_{km}^h + 1 + \ln 1/\delta)}$ *where* γ_{km}^h *depends on maximal mutual information obtained from GPs and the objective and constraints belong to RHKS with bound* B.

TVSafeOPT – safe exploration

- Comparison of safe sets computed by TVSafeOPT, ETSafeOPT, and SafeOPT at t=30, t=100, t=170

Initial safe seeds
Safe points
Maximizers
Current maximizers
True maximizers
Current true maximizers



TVSafeOPT – safe exploitation

- Focus on ensuring **optimality** when the system becomes **stationary**:

$$\max_{x} f(x,t) \quad \text{s.t. } c_i(x,t) \geq 0 \quad \Rightarrow \quad \max_{x} f(x) \quad \text{s.t. } c_i(x) \geq 0$$

TVSafeOPT – safe exploitation

- Focus on ensuring **optimality** when the system becomes **stationary**:

$$\max_x f(x,t) \quad \text{s.t. } c_i(x,t) \geq 0 \quad \Rightarrow \quad \max_x f(x) \quad \text{s.t. } c_i(x) \geq 0$$

- **Theorem (informal):**

For any $\delta \in (0,1)$, the value of the stationary reward $f(x_{k^*})$ will be within ϵ from the true optimum f^* in the reachable set in at most k^* steps:

$$|f(x_{k^*}) - f^*| \leq \epsilon$$

where k^*depends on the choice of β, maximal mutual information from the GPs, measurement noise, the initial safe seed S_0, and ϵ, and the reachable set is a subset of the largest possible set expanded from S_0 with the margin depending on $L(t)$ and ϵ.

TVSafeOPT – safe exploitation

- Focus on ensuring **optimality** when the system becomes **stationary**:

$$\max_x f(x,t) \quad \text{s.t. } c_i(x,t) \geq 0 \quad \Rightarrow \quad \max_x f(x) \quad \text{s.t. } c_i(x) \geq 0$$

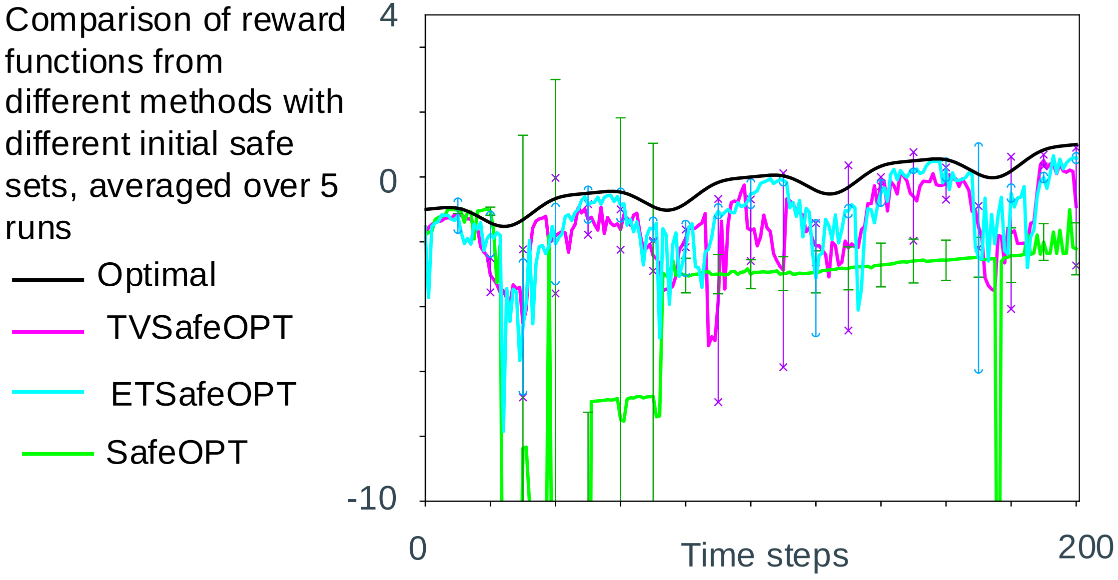
- **Theorem (informal):**

For any $\delta \in (0,1)$, the value of the stationary reward $f(x_{k^*})$ will be within ϵ from the true optimum f^* in the reachable set in at most k^* steps:

$$|f(x_{k^*}) - f^*| \leq \epsilon$$

where k^* depends on the choice of β, maximal mutual information from the GPs, measurement noise, the initial safe seed S_0, and ϵ, and the reachable set is a subset of the largest possible set expanded from S_0 with the margin depending on $L(t)$ and ϵ.

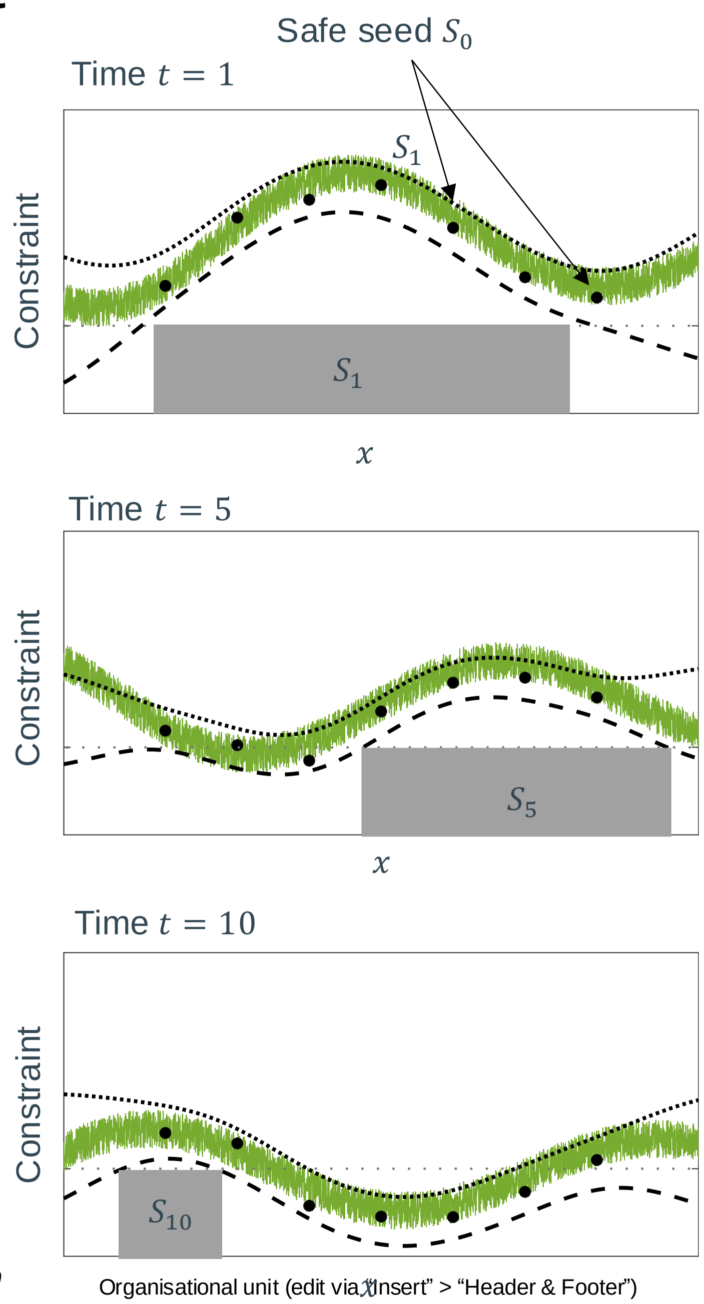
Comparison of reward functions from different methods with different initial safe sets, averaged over 5 runs



Comparison of TVSafeOPT and ETSafeOPT with respect to SafeOPT, showing the average and the standard deviation results from five runs with random initial safe sets

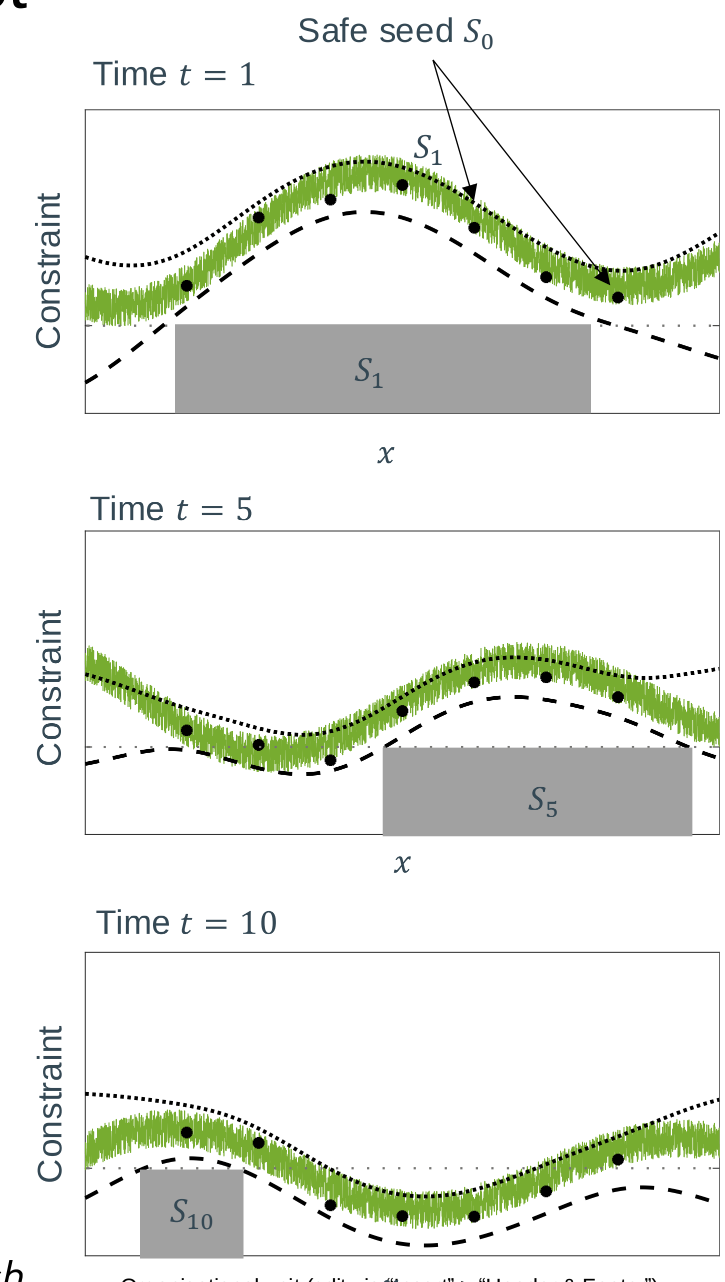
	ETSafeOPT	TVSafeOPT
Violations	-84.4%±1.7%	-99.99%±0.01%
Coverage ratio	-30.9%±2.9%	-21.0%±1.3%
Cumulative regret	-73.6%±14.7%	-66.9%±14.4%

Impact



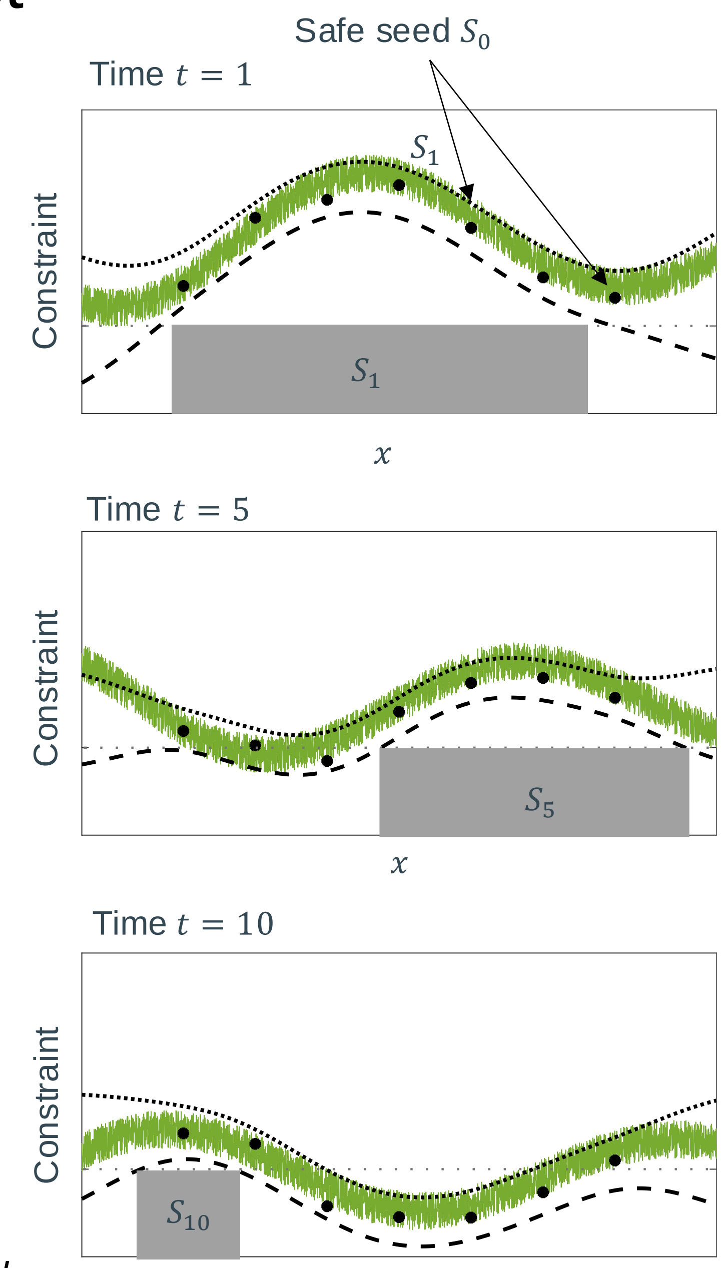
- TVSafeOPT:
  - extends SAFEOPT to **handle time-varying** optimization **problems**

Impact



- TVSafeOPT:
  - extends SAFEOPT to **handle time-varying** optimization **problems**
  - **adapts** to **changes in time** and **maintains fewer unsafe decisions** in its safe sets for time-varying problems than existing algorithms

Impact



- TVSafeOPT:
  - extends SAFEOPT to **handle time-varying** optimization **problems**
  - **adapts** to **changes in time** and **maintains fewer unsafe decisions** in its safe sets for time-varying problems than existing algorithms
  - is capable of **safely transferring safety** of the decisions into the future and will **find** the **near-optimal decision** when the reward function **stops changing**