7/27/2018

A Risk-Sensitive Reachability Problem for Safety of Stochastic Dynamic Systems in Discrete Time

Margaret Chapman, Jonathan Lacotte, Susmit Jha,
Donggun Lee, Claire Tomlin, Kevin Smith, Victoria Cheng,
David Freyberg

I.) Presents the formulation, why it's relevant, and how it relates to prior work

II.) Formalizes problem statement

III.) Presents an approximate computation method

IV.)* Presents disadvantages of the proposed computation method

V.) Application to stormwater infrastucture

*Please focus on this section.

Key References

- On Conditional-Value-at-Risk (CVaR)
  - J. Kisiala, Conditional Value-at-Risk: Theory and Applications, August 2015 (first few pages of Chapter 2, first few pages of Section 3.2)
  - A. Shapiro & D. Dentcheva, Lectures on Stochastic Programming Modeling and Theory, 2009 (first few pages of Section 6.3 Coherent Risk Measures; Average Value-at-Risk is another name for Conditional Value-at-Risk)

- On dynamic programming
  - D. P. Bertsekas, Dynamic Programming and Optimal Control, Vol 1, 2000. Athena Scientific.

- On a CVaR-algorithm
  - Chow, Tamar, Manor, and Pavone. (2015) Risk-Sensitive and Robust Decision-Making: a CVaR Optimization Approach, NIPS conference.

$\boxed{\text{I}}$ * Presents a risk-sensitive reachability problem as the computation of the sets,

$$S_\alpha^r = \left\{ x_0 \in S \;\middle|\; \text{there is a control policy } \pi \text{ such that } CVaR_\alpha^\pi \left(\begin{matrix}\text{extent of} \\ \text{constraint violation}\end{matrix} \right) < r \right\},$$

for $r \in \mathbb{R}$, $\alpha \in (0,1)$.

* Justifies why this formulation is relevant and how it relates to prior work in reachability.

7/18/2018

A Risk-Sensitive Reachability Problem for Safety of Stochastic Dynamic Systems in Discrete Time

- A classic reachability analysis problem for safety of dynamic systems is to compute the set of initial states from which the state trajectory is **guaranteed** to stay inside a given constraint set, K, under the assumption of unknown but **bounded** disturbances.

- Bertsekas (2000) (Dynamic Programming and Optimal Control, Vol. 1) provides the discrete-time version of this problem (p. 190, p. 46, Ex. 1.5), which follows.

- Given a system, $x_{k+1} = f_k(x_k, u_k, w_k)$, $x_k \in S$ state, $u_k \in C$ control, $w_k \in W_k(x_k, u_k)$ disturbance, one may compute a value function that specifies the safety of the state trajectory emanating from any initial condition, $x_0 \in S$,

$$J^*(x_0) := \inf_{\pi} \max_{\substack{w_k \in W_k \\ k=0,1,\dots,N-1}} \left\{ \sum_{k=0}^{N} g_k(x_k) \right\}, \quad \text{where} \quad g_k(x_k) = \begin{cases} 0 & \text{if } x_k \in K \\ 1 & \text{if } x_k \notin K \end{cases}, \quad \text{and} \qquad (1)$$

$\pi \in \widetilde{\Pi} := \{(\mu_0, \dots, \mu_{N-1}),\ \mu_k : S \to C\}$ is an admissible policy.

- $J^*(x_0)$ specifies safety because $J^*(x_0) = 0 \Leftrightarrow \exists\, \pi \in \widetilde{\Pi}$ such that $x_0 \in K, \dots, x_N \in K$ for all $w_0 \in W_0, \dots, w_{N-1} \in W_{N-1}$.

- In particular, the above min-max formulation assumes that (1) the disturbance is adversarial and (2) safety is well-described solely as a binary notion according to set membership. (inside K is good, outside K is bad, being far inside/outside K is equally good/bad)

- More generally, one can define the safe set as follows:

$$\{x_0 \in S \mid \exists\, \pi \in \widetilde{\Pi} \text{ s.t. } x_0 \in K, \dots, x_N \in K \text{ for all } w_0 \in W_0, \dots, w_{N-1} \in W_{N-1}\},$$

which can be computed via $\{x_0 \in S \mid J^*(x_0) = 0\}$. (2)

- But, in practice, disturbances do not usually behave adversarially.
  - Most storms do not cause major floods.
  - Most drivers are never involved in car chases.
  - Most drones (outside of war) are not intending to collide with each other.

Indeed, min-max formulations, such as (1), may produce overly conservative control policies with limited practical utility; the set of control policies that satisfies (1) or the safe set (2) may be artifically small.

- In applications where min-max formulations are not useful, one may consider a stochastic formulation, where the disturbance is assumed to be drawn from some known probability distribution, $w_k \sim \mathbb{P}_k$. (Techniques for estimating such distributions from data is active research - refs?)

- Abate et al. (2008) Automatica formulized the probabilistic reachability problem for safety of stochastic dynamic systems in discrete time* as follows.
  - The probabilistic safe set with level $\varepsilon \in [0,1]$ is $\{x_0 \in S \mid \exists \pi \in \Pi \text{ s.t. } x_0 \in K, \dots, x_N \in K \text{ with probability at least } 1-\varepsilon\}$,

    which equals $\left\{x_0 \in S \,\middle|\, \sup_{\pi \in \Pi} \mathbb{E}\left[\prod_{k=0}^{N} \mathbb{1}_{K}(x_k)\right] \geq 1-\varepsilon\right\}$, (3)

    which is equivalent to $\left\{x_0 \in S \,\middle|\, \inf_{\pi \in \Pi} \mathbb{E}\left[\max_{k \in [0,N]} \mathbb{1}_{K^c}(x_k)\right] \leq \varepsilon\right\}$, (4)

    where $\mathbb{1}_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$ is the indicator function.
  - Abate et al. 2008 proved that (3), (4) can be computed via dynamic programming algorithms.

- Similar to the min-max formulation, the stochastic formulation above also assumes that safety is well-described solely as a binary notion according to set membership.

* Abate did the analysis for hybrid systems.

- However, this may not be an appropriate description for safety, especially when the system at hand is hard to control.
  - Water may occassionally overflow the banks of a stormwater detention pond. boundary of $\mathcal{K}$ hard to define Lidar map/perception has uncertainty
  - During a course of cancer treatment, some healthy cells may die in addition to cancer cells.

- In addition to quantifying set membership (does traj ever leave $\mathcal{K}$?), we may also want to quantify <u>the degree of constraint violation (or satisfaction)</u> exhibited by a given state trajectory.

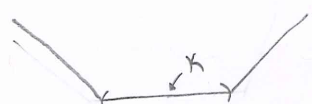
  We abbreviate " as <u>the degree of safety</u>.

- A natural way to quantify the degree of safety <u>at a given state</u> is to define a surface function that represents distance in some sense with respect to the boundary of $\mathcal{K}$ such that $x \in \mathcal{K} \Leftrightarrow g(x) < 0$.

  E.g. $\mathcal{K} = (0, 10)$. $g(x) = x(x-10)$, $g(x) = |x-5| - 5$.

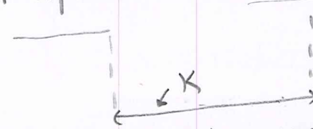
  (This technique is used in HJ Reachability Analysis for dynamics in <u>continuous-time</u> w/ unknown but bounded disturbances.)

  - Note that choosing $g(x)$



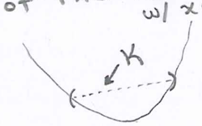
does not quantify the degree of constraint satisfaction

or



does not quantify the degree of constraint satisfaction or violation.

, so we will consider $g(x)$ of the form, w/ $x \in \mathcal{K} \Leftrightarrow g(x) < 0$.



- However, it is not immediately clear how to quantify the degree of safety of a given <u>state trajectory</u>.

  Candidates include:

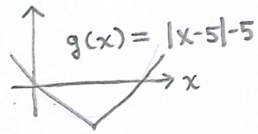
  $\sum_{k=0}^{N} g(x_k)$ (5) (adopted from (1) - <u>Bertsekas</u>)

  $\prod_{k=0}^{N} g(x_k)$, (6) (adopted from (3) - <u>Abate</u>)

  $\max_{k \in \{0,\dots,N\}} g(x_k)$ (7) (" (4) - <u>Abate</u>)

- We will argue that (5) and (6) are likely not suitable for safety using examples of realizations of the state trajectory.

$K = (0, 10)$



- Ex. 1a $(x_0, x_1, x_2) = (1, -1, 1)$. Realized traj exited K. $\sum_{k=0}^{2} g(x_k) = -1$

  ↑ realized traj ↓

- Ex. 1b $(x_0, x_1, x_2) = (1/3, 1/3, 1/3)$. Realized traj did not exit K. $\sum_{k=0}^{2} g(x_k) = -1$

} The realized value of $\sum g(x_k)$ does not indicate whether the traj left K.

- Ex. 2a $(x_0, x_1, x_2) = (1, -1, -2)$. Realized traj left K. $\prod_{k=0}^{2} g(x_k) = -1 \cdot 1 \cdot 2 = -2$

- Ex. 2b $(x_0, x_1, x_2) = (1, 1, 2)$. Realized traj did not leave K. $\prod_{k=0}^{2} g(x_k) = -1 \cdot -1 \cdot -2 = -2$

} The realized value of $\prod g(x_k)$ does not indicate whether traj left K.

- However, (7) is a suitable metric. Let g be such that $x \in K \Leftrightarrow g(x) < 0$.

  Let $(x_0, x_1, \dots, x_N)$ be any realized state trajectory.

  <u>Fact</u> $\max_{k \in \{0,\dots,N\}} g(x_k) < 0 \Leftrightarrow \forall k,\ g(x_k) < 0 \Leftrightarrow \forall k,\ x_k \in K$.

  - (7) is an appropriate metric for the degree of safety of a state trajectory b/c (7) indicates

    (i) whether the traj stays in K

    (ii) the degree of constraint satisfaction if traj $\in K$

    (iii) " " violation if traj $\notin K$

    } via minimum distance to boundary of K.

- Recall that our system is stochastic, $x_{k+1} = f_k(x_k, u_k, w_k)$ $w_k \sim P_k$, so

$$\max_{k\in\{0,\dots,N\}} g(x_k) \text{ is a random variable.}$$

- A natural way to evaluate a random variable is via expectation.

- We might consider the following value function,

$$V^*(x_0) := \min_{\pi\in\Pi} \mathbb{E}\left[\max_{k\in\{0,\dots,N\}} g(x_k)\right] \quad (8)$$

to quantify the degree of safety of the trajectory emanenating from $x_0 \in S$.

- Key limitations of (8) include: $V^*(x_0)$ being sufficiently small does not
  - ⓐ provide a probabilistic safety guarantee
  - ⓑ appreciate the reality that rare events may occur and be detrimental to safety

- There is another metric similar in spirit to expectation that does satisfy ⓐ and ⓑ, called <u>Conditional Value at Risk</u> (or Expected Shortfall) "CVaR"

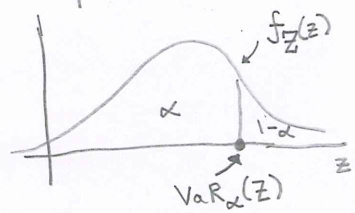
- Intuitively, CVaR is a function from the space of random variables to $\mathbb{R}$ that penalizes the outcomes that hurt the most (cite Shapiro encyclopedia).

- A related function from the space of random variables to $\mathbb{R}$ is Value-at-Risk. (VaR)

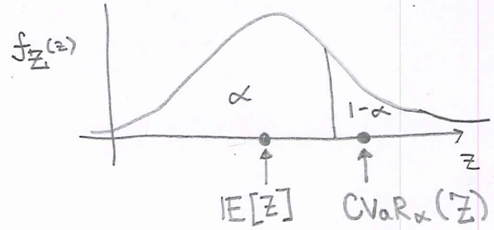
VaR of r.v Z at confidence level $\alpha \in (0,1)$ is the α-quantile of the distribution of Z.

if Z continuous r.v.

probability density function of Z



- For continuous r.v. Z, CVaR of Z at confidence level $\alpha \in (0,1)$, is $\mathbb{E}[Z \mid Z \geq VaR_\alpha(Z)]$.



- Why is CVaR a good option?

a "risk-sensitive" optimal control problem

$$W^*(x_0) := \min_{\pi \in \Pi} CVaR_\alpha\left[\max_{k \in \{0,\dots,N\}} g(x_k)\right]$$

$$W^*(x_0) < 0 \Rightarrow \exists \pi \text{ s.t. } CVaR_\alpha\left[\max_k g(x_k)\right] < 0$$

$$\Downarrow$$

$$\exists \pi \text{ s.t. } \underbrace{\mathbb{P}\left[\max_k g(x_k) \geq 0\right]}_{} < 1-\alpha$$

$$\| $$

$$\mathbb{P}[\exists k \text{ s.t. } g(x_k) \geq 0]$$

$$\|$$

$$\mathbb{P}[\exists k \text{ s.t. } x_k \notin \mathcal{X}]$$

- there is a probabilistic safety guarantee for $\underline{\underline{\text{any}}}$ g such that $g(x) < 0 \Leftrightarrow x \in \mathcal{X}$

- does penalize more harmful outcomes

vs.

a "risk-neutral" optimal control problem

$$V^*(x_0) := \min_{\pi \in \Pi} \mathbb{E}\left[\max_{k \in \{0,\dots,N\}} g(x_k)\right]$$

$$V^*(x_0) < 0 \Rightarrow \exists \pi \text{ s.t. } \mathbb{E}\left[\max_k g(x_k)\right] < 0.$$

$$\mathbb{E}[Z] = \sum z_i \mathbb{P}[Z = z_i]$$

- no known probabilistic safety guarantee for $g(x) \neq$ indicator function

- does not penalize the more harmful outcomes

- Recall the classic reachability analysis problem for safety:

  to compute the set of initial conditions from which the state trajectory is "guaranteed" to stay inside $\mathcal{K}$.

- We can apply this concept to our risk-sensitive optimal control problem and define a risk-sensitive r-sublevel set,

$$S_{(r)}^{\alpha} := \left\{ x_0 \mid \min_{\pi} \mathrm{CVaR}_{\alpha}\left(\max_{k} g(x_k) \right) < r \right\}.$$

- In particular, if $r = 0$, then the states in $S(0)$ also satisfy a probabilistic safety guarantee.

  Further, as r decreases, the states in $S(r)$ can be considered more safe.

- Our problem is to compute S_r^{α} for various values of (α, r).

  ↑ set of initial conditions from which there is a control policy that can make the risk of "large" constraint violations "small."

Wh

- What are the ~~benefits~~ features of S_r^{α}?
  - stochastic dynamics in discrete-time w/ non-adversarial disturbances
  - quantifies the degree of safety (constraint violation or satisfaction); appreciates situations where safety cannot be well-described solely as a binary notion in terms of set membership.
  - provides a probabilistic safety guarantee for $r=0$
  - appreciates the reality that rare harmful outcomes (outcomes that are detrimental to safety.) can occur by explicitly penalizing these outcomes as opposed to penalizing average outcomes.

$\boxed{\text{II}}$ Presents formal problem statement, and its relation to probabilistic safety

System Model

$$x_{k+1} = f_k(x_k, u_k, w_k) \qquad k = 0, 1, \ldots, N-1$$

$x_k \in S$ state

$u_k \in C$ control input (non-random)

$w_k \sim \mathbb{P}_k$ random disturbance given by some probability distribution
- independent of u_k
- may depend on x_k

$\Pi := \{(\mu_0, \mu_1, \ldots, \mu_{N-1}),\ \mu_k : S \to C\}$ set of admissible control policies

Safety criterion : $x_k \in K$, the constraint set, for all $k = 0, 1, \ldots, N$.

$g : S \to \mathbb{R}$ is a function that characterizes the extent of constraint violation/satisfaction, such that $x \in K \Leftrightarrow g(x) < 0$.

Note 1) $\xi_y^\pi(k)$ denotes the random state at time k under a given policy $\pi \in \Pi$, starting from a given (non-random) initial condition, $y \in S$, at time 0.

2) $\max_{k \in \{0,1,\ldots,N\}} g(\xi_y^\pi(k))$ is a random variable of the worst extent of constraint violation over the finite time horizon under policy π, starting from initial condition $y \in S$.

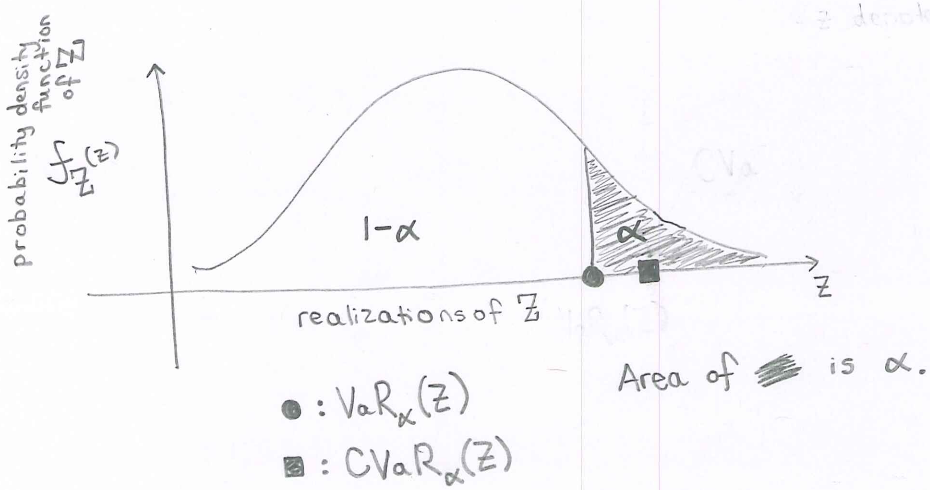
The problem is to compute the family of <u>risk-sensitive sets</u>,

$$S_\alpha^r := \left\{ x_0 \in S \,\middle|\, \exists\, \pi \in \Pi \text{ such that } \mathrm{CVaR}_\alpha^\pi\left(\max_{k \in \{0,1,\ldots,N\}} g(\xi_{x_0}^\pi(k)) \right) < r \right\},$$

for $r \in \mathbb{R}$, $\alpha \in (0,1)$.

<u>Note</u> For a continuous random variable Z, $\mathrm{CVaR}_\alpha(Z) = \mathbb{E}[Z \mid Z \geq \mathrm{VaR}_\alpha(Z)]$,

where $\mathrm{VaR}_\alpha(Z) := \min\{c \mid \mathbb{P}[Z \geq c] \leq \alpha\}$.



Remark The states in S^o_α enjoy a probabilistic safety guarantee. What does this mean?

$$S^o_\alpha := \left\{x_0 \in S \;\middle|\; \exists\, \pi \in \bar{\Pi} \text{ such that } \mathrm{CVaR}^\pi_\alpha\left(\max_{k\in\{0,1,\dots,N\}} g(\xi^\pi_{x_0}(k))\right) < 0\right\} \subseteq \left\{x_0 \in S \;\middle|\; \exists\, \pi \in \bar{\Pi} \text{ s.t. } \mathbb{P}\{\exists\, k\in\{0,\dots,N\} \text{ s.t. } \xi^\pi_{x_0}(k) \notin K\} < \alpha\right\}.$$

The probability of exiting the constraint set, K, is less than α.

Proof

$$\mathrm{CVaR}^\pi_\alpha\left(\max_{k\in\{0,\dots,N\}} g(\xi^\pi_{x_0}(k))\right) < 0 \quad \Rightarrow \quad \mathbb{P}\left\{\max_{k\in\{0,\dots,N\}} g(\xi^\pi_{x_0}(k)) \geq 0\right\} < \alpha, \text{ by Lemma 1 (see next page)} \qquad (1)$$

with $Z := \max_{k\in\{0,\dots,N\}} g(\xi^\pi_{x_0}(k))$ and $\delta := \alpha$.

$$\mathbb{P}\left\{\max_{k\in\{0,\dots,N\}} g(\xi^\pi_{x_0}(k)) \geq 0\right\} = \mathbb{P}\left\{\exists\, k\in\{0,\dots,N\} \text{ such that } g(\xi^\pi_{x_0}(k)) \geq 0\right\} \qquad \text{(by definion of max)}$$

$$= \mathbb{P}\left\{\exists\, k\in\{0,\dots,N\} \text{ such that } \xi^\pi_{x_0}(k) \notin K\right\}. \qquad (g(x) < 0 \Leftrightarrow x \in K) \qquad (2)$$

Take any $x_0 \in S^o_\alpha$. Then, $\exists\, \pi \in \bar{\Pi}$ such that $\mathrm{CVaR}^\pi_\alpha\left(\max_{k\in\{0,\dots,N\}} g(\xi^\pi_{x_0}(k))\right) < 0$.

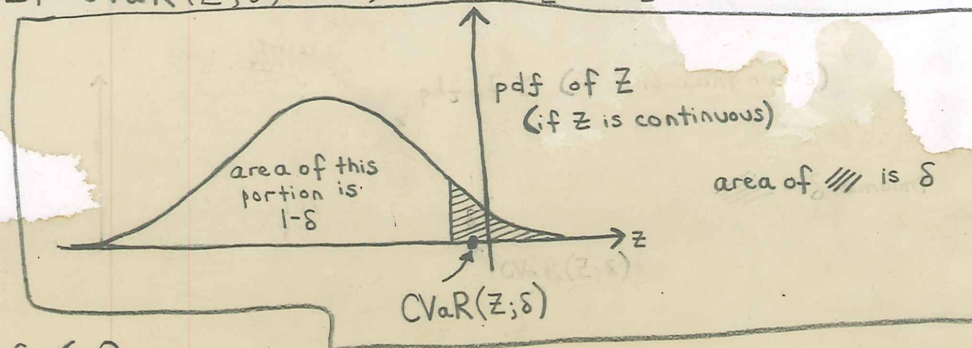
So, $\exists\, \pi \in \bar{\Pi}$ such that $\mathbb{P}\left\{\max_{k\in\{0,\dots,N\}} g(\xi^\pi_{x_0}(k)) \geq 0\right\} < \alpha$ by (1).

Thus, $\exists\, \pi \in \bar{\Pi}$ such that $\mathbb{P}\left\{\exists\, k\in\{0,\dots,N\} \text{ s.t. } \xi^\pi_{x_0}(k) \notin K\right\} < \alpha$ by (2).

$$\therefore \quad x_0 \in \left\{x_0 \in S \;\middle|\; \exists\, \pi \in \bar{\Pi} \text{ s.t. } \mathbb{P}\{\exists\, k\in\{0,\dots,N\} \text{ s.t. } \xi^\pi_{x_0}(k) \notin K\} < \alpha\right\}. \quad \blacksquare$$

Lemma 1

Let $\delta \in (0,1)$, and Z is a random variable. If $\mathrm{CVaR}(Z;\delta) < 0$, then $\mathbb{P}[Z \geq 0] < \delta$.



Proof:

see J. Kisiala, 2015 eqs (3.8) and (3.9)

$$\mathrm{CVaR}(Z;\delta) = \min_{c \in \mathbb{R}} \left\{ \frac{1}{\delta}\mathbb{E}\left[(Z-c)_+\right] + c \right\}$$

$\mathrm{CVaR}(Z;\delta) < 0 \Rightarrow \exists\, c \in \mathbb{R}$ such that $\frac{1}{\delta}\mathbb{E}\left[(Z-c)_+\right] + c < 0$.

Because $\frac{1}{\delta}\mathbb{E}\left[(Z-c)_+\right] \geq 0$, we must have $c < 0$.

$$\frac{1}{\delta}\mathbb{E}\left[(Z-c)_+\right] + c < 0$$
$$c < -\frac{1}{\delta}\mathbb{E}\left[(Z-c)_+\right] \leq 0$$
$$\therefore\ c < 0$$

$\therefore\ \exists\, c < 0$ such that $\frac{1}{\delta}\mathbb{E}\left[(Z-c)_+\right] + c < 0$.

Set $\alpha := -c$.

$\exists\, \alpha > 0$ such that $\frac{1}{\delta}\mathbb{E}\left[(Z+\alpha)_+\right] - \alpha < 0$

$$\mathbb{E}\left[(Z+\alpha)_+\right] < \delta\alpha$$

$$\mathbb{P}[Z \geq 0] = \mathbb{P}\left[(Z+\alpha)_+ \geq \alpha\right] \underset{\text{Markov's Inequality}}{\leq} \frac{\mathbb{E}\left[(Z+\alpha)_+\right]}{\alpha} < \delta \qquad \blacksquare$$

$Z+\alpha \geq \alpha \ \cap\ Z+\alpha \geq 0$

$$\mathbb{P}\left[\max(Z+\alpha,0) \geq \alpha\right] = \mathbb{P}\left[Z+\alpha \geq \alpha \cap Z+\alpha \geq 0\right] + \mathbb{P}\left[\cancel{0 \geq \alpha \cap Z+\alpha < 0}\right]$$

0 because $\alpha > 0$

$$= \mathbb{P}\left[Z+\alpha \geq \alpha\right]$$
$$= \mathbb{P}[Z \geq 0]$$

Key properties of $S_\alpha^r := \left\{ x_0 \in S \;\middle|\; \exists \pi \in \Pi \text{ such that } CVaR_\alpha^\pi\left(\max_{k \in \{0,1,\dots,N\}} g(\xi_{x_0}^\pi(k)) \right) < r \right\}$.

$\alpha \in (0,1)$, $r \in \mathbb{R}$; S: state space

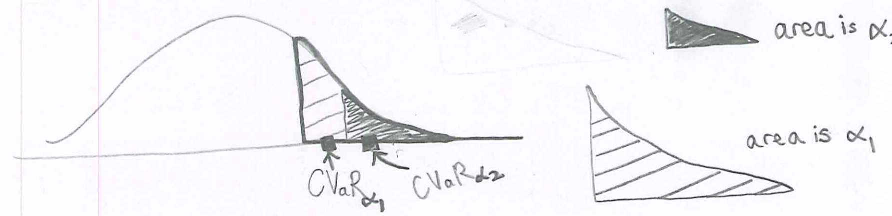
- S_α^r is the set of initial conditions, from which the risk of large constraint violations by the state trajectory can be made small.

- $x_0 \in S_\alpha^0 \implies \exists \pi \in \Pi$ such that $\mathbb{P}\{\exists k \in \{0,\dots,N\} \text{ s.t. } \xi_{x_0}^\pi(k) \notin K\} < \alpha$.

  The probability of violating the constraints can be made small.

- Fix $\alpha \in (0,1)$. $r_1 \geq r_2 \implies S_\alpha^{r_2} \subseteq S_\alpha^{r_1}$.

- Fix $r \in \mathbb{R}$. $\alpha_1 \geq \alpha_2 \implies S_{\alpha_2}^r \subseteq S_{\alpha_1}^r$.



$\therefore \; S_{\alpha_2}^{r_2} \subseteq S_{\alpha_1}^{r_2} \subseteq S_{\alpha_1}^{r_1}$ for any $r_1 \geq r_2$, $1 > \alpha_1 \geq \alpha_2 > 0$.

- In words, smaller r and/or smaller α corresponds to being more safe.

- $$S_\alpha^r = \left\{ x_0 \in S \;\middle|\; \min_{\pi \in \Pi} CVaR_\alpha^\pi\left(\max_{k \in \{0,\dots,N\}} g(\xi_{x_0}^\pi(k)) \right) < r \right\}.$$

  S_α^r: risk-sensitive set ; S: state space

$\boxed{\text{III}}$ Presents a method for computing approximate risk-sensitive safe sets.

An algorithm that computes the Conditional value-at-risk of a sum of costs has been proposed by Chow et al. 2015.

Chow, Tamar, Manor, Pavone

Risk-Sensitive and Robust Decision-Making a CVaR Optimization Approach

We propose to approximate CVaR of the maximum cost in terms of CVaR of a sum of costs, which then would be computed via the algorithm in Chow et al. 2015.

Recall, $S_\alpha^r = \{x_0 \in S \mid \min_{\pi \in \Pi} \mathrm{CVaR}_\alpha^\pi \left(\max_{k \in \{0,\dots,N\}} g(\xi_{x_0}^\pi(k)) \right) < r \}$.

<u>Lemma 2</u>

Take $\alpha \in (0,1)$ and $r \in \mathbb{R}$.

Define $U_\alpha^r := \{x_0 \in S \mid \min_{\pi \in \Pi} \mathrm{CVaR}_\alpha^\pi \left(\sum_{k=0}^{N} e^{g(\xi_{x_0}^\pi(k))} \right) < e^r \}$.

Define $\mathcal{O}^r := \{x_0 \in S \mid g(x_0) < r\}$.

Then, $U_\alpha^r \subseteq S_\alpha^r \subseteq \mathcal{O}^r$.

<u>Remark</u>

U_α^r is an underapproximation of the risk-sensitive set, S_α^r, and $\mathcal{O}^r$ is an overapproximation of S_α^r. While we don't know how to compute S_α^r exactly, we can compute U_α^r (using Chow et al. 2015) and $\mathcal{O}^r$ (I believe).

To prove Lemma 2, we need the following result:

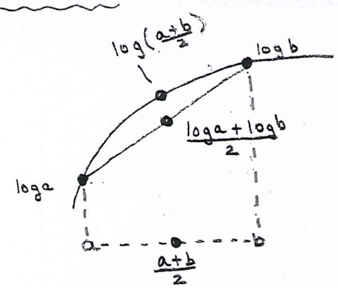
<u>Lemma 3</u> $\quad \mathrm{CVaR}_\alpha(\log(Z)) \le \log(\mathrm{CVaR}_\alpha(Z)) \quad \forall\, Z \in \left\{\begin{matrix}\text{random variable with}\\ \text{finite 1st-order moment}\end{matrix}\right\}$

<u>Proof</u>:

Because $\mathrm{CVaR}_\alpha(\cdot)$ is a coherent riskmeasure, there is a set of probability density functions, A_α, such that $\mathrm{CVaR}_\alpha(Z) = \sup_{f \in A_\alpha} \mathbb{E}_f[Z] \;\; \forall\, Z \in \mathcal{L}_1(\Omega, \mathcal{F}, P)$.

see Shapiro, <u>Lectures on Stochastic Programming Modeling and Theory</u>, Chapter 6.3, Coherent Risk Measures, equation (6.40)

For any $f \in A_\alpha$, $\mathbb{E}_f[\log(Z)] \le \log(\mathbb{E}_f[Z])$, by concavity of the logarithm.



For any $f \in A_\alpha$, $\quad \log(\mathbb{E}_f[Z]) \le \log\Big(\underbrace{\sup_{f \in A_\alpha} \mathbb{E}_f[Z]}_{= \,\mathrm{CVaR}_\alpha(Z)}\Big)$, since log is increasing.

$\therefore \quad \mathbb{E}_f[\log(Z)] \le \log(\mathrm{CVaR}_\alpha(Z)) \quad \forall\, f \in A_\alpha.$

So, $\log(\mathrm{CVaR}_\alpha(Z))$ is an upper bound for $\{\mathbb{E}_f[\log(Z)] : f \in A_\alpha\}$.

$$\therefore \quad \underbrace{\overset{\text{l.u.b.}}{\sup_{f \in A_\alpha} \mathbb{E}_f[\log(Z)]}}_{=\,\mathrm{CVaR}_\alpha(\log(Z))} \le \overset{\text{u.b.}}{\log(\mathrm{CVaR}_\alpha(Z))}$$

l.u.b. stands for the least upper bound, which is the supremum.

u.b. stands for an upper bound.

$$\therefore \quad \mathrm{CVaR}_\alpha(\log(Z)) \le \log(\mathrm{CVaR}_\alpha(Z)). \quad \blacksquare$$

Fact: 1) $\max\{y_1, \ldots, y_n\} \le \log(e^{y_1} + \cdots + e^{y_n}) \le \max\{y_1, \ldots, y_n\} + \log n$, $y \in \mathbb{R}^n$

"log-sum-exp" is a differentiable, analytic approximation of the max function.

2) log-sum-exp is convex on $\mathbb{R}^n$

Ref: Boyd & Vandenberghe, Convex Optimization, Cambridge University Press, 2004. 7th printing with corrections 2009. See Chapter 3, Convex functions, p. 72, Log-sum-exp bullet point

$$U_\alpha^r := \left\{x_0 \in S \,\middle|\, \min_{\pi \in \Pi} \mathrm{CVaR}_\alpha^\pi\left(\sum_{k=0}^{N} e^{g(\xi_{x_0}^\pi(k))}\right) < e^r\right\}$$

$$S_\alpha^r = \left\{x_0 \in S \,\middle|\, \min_{\pi \in \Pi} \mathrm{CVaR}_\alpha^\pi\left(\max_{k \in \{0,\ldots,N\}} g(\xi_{x_0}^\pi(k))\right) < r\right\}$$

$$\mathcal{O}^r := \left\{x_0 \in S \,\middle|\, g(x_0) < r\right\}$$

Proof of Lemma 2

We will show $U_\alpha^r \subseteq S_\alpha^r \subseteq \mathcal{O}^r$.

Note that

$$\mathrm{CVaR}_\alpha^\pi\left(\max_{k \in \{0,\ldots,N\}} g(\xi_{x_0}^\pi(k))\right) \le \mathrm{CVaR}_\alpha^\pi\left(\log\left(\sum_{k=0}^{N} e^{g(\xi_{x_0}^\pi(k))}\right)\right)$$

by monotonicity of CVaR and by Fact *1 (see top of page).

$$\le \log\left(\mathrm{CVaR}_\alpha^\pi\left(\sum_{k=0}^{N} e^{g(\xi_{x_0}^\pi(k))}\right)\right)$$

by Lemma 3 (see p. 16).

Take $x_0 \in U_\alpha^r$. Then, $\exists\, \pi \in \Pi$ such that $\log\left(\mathrm{CVaR}_\alpha^\pi\left(\sum_{k=0}^{N} e^{g(\xi_{x_0}^\pi(k))}\right)\right) < r$.

So, $\exists\, \pi \in \Pi$ such that $\mathrm{CVaR}_\alpha^\pi\left(\max_{k \in \{0,\ldots,N\}} g(\xi_{x_0}^\pi(k))\right) < r$, by the above chain of inequalities.

$\therefore\ x_0 \in S_\alpha^r$.

$\therefore\ U_\alpha^r \subseteq S_\alpha^r$.

Now, take $x_0 \in S_\alpha^r$.

Then, $\exists\, \pi \in \Pi$ such that $\text{CVaR}_\alpha^\pi\left(\max_{k \in \{0,1,\dots,N\}} g(\xi_{x_0}^\pi(k)) \right) < r$.

Since $g(x_0) = g(\xi_{x_0}^\pi(0)) \leq \max_{k \in \{0,\dots,N\}} g(\xi_{x_0}^\pi(k))$ with probability 1, and CVaR is monotonic,

$$g(x_0) = \text{CVaR}_\alpha^\pi\left(g(x_0) \right) \leq \text{CVaR}_\alpha^\pi\left(\max_{k \in \{0,\dots,N\}} g(\xi_{x_0}^\pi(k)) \right) < r.$$

$g(x_0)$ is deterministic.

see line just above

see 2nd line on this page

$\therefore\ x_0 \in \{ y \in S \mid g(y) < r \} = \mathcal{O}^r$.

$\therefore\ S_\alpha^r \subseteq \mathcal{O}^r$. $\blacksquare$

* $\boxed{\text{IV}}$ Presents disadvantages of the method for computing approximate risk-sensitive safe sets

* Please look at this closely

- The method that we proposed for computing approximate risk-sensitive safe sets relies on an algorithm by Chow et al. 2015 (Risk-sensitive and Robust-decision-Making: a CVaR Optimization approach) that is able to compute,

infinite time!

$$\min_{\pi \in \Pi_\infty} \text{CVaR}_\alpha \left(\sum_{t=0}^{\infty} \text{cost}(x_t) \,\middle|\, x_0, \pi = \{\mu_0, \mu_1, \ldots\} \right)$$

- The algorithm that Chow et al. 2015 proposes does <u>not</u> satisfy Bellman's principle of optimality in the traditional sense.

- In particular, for Chow's algorithm,

$$\{\mu_0^*, \mu_1^*, \ldots\} \in \arg\min_{\pi} \text{CVaR}_\alpha \left(\sum_{t=0}^{\infty} \text{cost}(x_t) \,\middle|\, x_0, \pi = \{\mu_0, \mu_1, \ldots\} \right) \not\Rightarrow \{\mu_i^*, \mu_{i+1}^*, \ldots\} \in \arg\min_{\pi} \text{CVaR}_\alpha \left(\sum_{t=i}^{\infty} \text{cost}(x_t) \,\middle|\, x_i, \pi = \{\mu_i, \mu_{i+1}, \ldots\} \right).$$

In words:

Suppose that at x_0 the policy that is optimal for $\text{CVaR}_\alpha(\cdot)$ is $\{\mu_0^*, \mu_1^*, \ldots\}$, and under this policy state x_i occurs at time i with non-zero probability.

Then, the truncated policy $\{\mu_i^*, \mu_{i+1}^*, \ldots\}$ is <u>not</u> necessarily optimal for the subproblem, where we start at state x_i at time i, under the metric $\text{CVaR}_\alpha(\cdot)$.

- This might be a problem because most people in our community expect algorithms to satisfy Bellman's principle of optimality, and may not trust algorithms that don't.

- I believe that Chow's algorithm satisfies a "Bellman-like principle of optimality" on the augmented state space $S \times (0,1)$, where α becomes a state as well.

  $x \in S,\ \alpha \in (0,1)$.

- However, the Chow paper does not provide a clear relationship between their algorithm and the form of Bellman's principle of optimality that the community is familiar with (see Bertsekas, Dynamic Programming and Optimal Control, Vol. I).

- Before we use Chow's algorithm, I think that we need to fill this gap (i.e. relate Chow's algorithm to Bellman's principle of optimality clearly).

- The concept of <u>time-consistency</u> has come up several times in the multistage stochastic programming literature.

- From conversations with Aviv Tamar, I think that time consistency is another name for Bellman's principle of optimality.

- The definitions of time consistency appear to be different depending on the author.

- Shapiro says that time consistency is the principle that "at every state of the system, optimality of our decisions should not depend on scenarios which we know cannot happen in the future." (p.321) But then he explains time consistency using Bellman's principle of optimality. (p.322) Sec. 6.7 Multistage Risk Averse Optimization <u>Lectures on Stochastic Programming Modeling and Theory</u>

- Ruszczyński 2010 defines time consistency in terms of Bellman's principle of optimality. (see <u>Risk-averse dynamic programming for Markov decision processes</u>) Math. Program., 2010

  ($CVaR_\alpha(\sum_{t=0}^{\infty} Z_t)$ is also not time-consistent)

- Shapiro states that the formulation $CVaR_\alpha(\max_{t=1,\ldots,T} Z_t)$ is <u>not</u> a time-consistent formulation (see Shapiro, 2009, On a time consistency concept in risk averse multistage stochastic programming, <u>Operations Research Letters</u>.)

- I'm concerned that our work will be criticized harshly for not using a "time consistent" formulation, however I'm not sure how to make a formulation that is both "time consistent" and meaningful in the context of reachability analysis for safety.

- I think that the middle ground is to relate Chow's algorithm to a "Bellman-like principle of optimality."

- For each state $x_0 \in S$ (where S is finite), and $y \in U \subset (0,1)$ (where U is finite), Chow's algorithm provides a good estimate for $V(x_0, y) := \min_{\pi \in \Pi_\infty} \mathrm{CVaR}_y\left(\sum_{t=0}^{\infty} \mathrm{cost}(x_t) \,\middle|\, x_0, \pi\right)$.

$\Pi_\infty := \{(\mu_0, \mu_1, \dots),\ \mu_k : \text{history} \to \text{action probability space}\}$

- For finitely many $\alpha \in (0,1)$, we want to compute an underapproximation of the risk-sensitive set,

$$S_\alpha^r := \left\{ x_0 \in S \;\middle|\; \exists\, \pi \in \Pi \text{ s.t. } \mathrm{CVaR}_\alpha^\pi\left(\max_{k=0,1,\dots,N} g(\xi_{x_0}^\pi(k)) \right) < r \right\},$$

by computing,

$$U_\alpha^r := \left\{ x_0 \in S \;\middle|\; \min_{\pi \in \Pi} \mathrm{CVaR}_\alpha^\pi\left(\sum_{k=0}^{N} e^{g(\xi_{x_0}^\pi(k))} \right) < e^r \right\},$$

for any $r \in \mathbb{R}$, using Chow's algorithm, where $\Pi := \{(\mu_0, \mu_1, \dots, \mu_{N-1}),\ \mu_k : S \to C\}$.

- How do we adapt Chow's algorithm to a finite time horizon?

- Perhaps, we want to evaluate the real system, which operates in continuous time, over a 24h time horizon.

$k = 0, 1, 2, \ldots, N$

$t = 0, \Delta t, 2\Delta t, \ldots, N\Delta t$

Want 1) $N \Delta t = 24h$

2) N to be large enough so that the error bound (see Thm 7, Chow 2015) is small enough

depends on discount factor.

* 1) & 2) determine the discretization time step, Δt, for our model.

*Does Chow's algorithm work for time-varying probability distributions?

$w_k \sim P_k$

- But, we also need to somehow incorporate a discount factor into the sum, $\sum_{k=0}^{N} e^{g(\xi^{\pi}_{x_0}(k))}$, in order to use Chow's algorithm. *How do we do this?*

- Or, maybe, it would make more sense to derive a dynamic programming update (like eq. 6 in Chow 2015) for a fixed $N \in \mathbb{N}$?

(also like Proposition 1.3.1 in Bertsekas' Dynamic Programming and Optimal Control)

Perhaps use the technique in Kene's paper in eq. 18 "A minimum discounted reward formulation for computing reachable sets"

- Another challenge with using Chow's algorithm is that our set of admissible policies is $\Pi^{x} := \{(\mu_0, \mu_1, \ldots, \mu_{N-1}); \mu_k : \underset{\text{state space}}{S} \to \underset{\text{action space}}{C}\}$, while for Chow et al. μ_k: history → action probability space.

$\boxed{\text{V}}$ Presents an application domain for risk-sensitive reachability that does not require real-time computation.

Consider the problem of choosing designs for stormwater infrastructure, where future weather is unknown and funding for infrastructure is limited. What is a reasonable way to evaluate candidate designs? Our recent work (Sustech 2018 submission) presents standard methods (run a fixed precipitation pattern called a design storm through a computer model and assess the outcome in simulation; choose the final design to meet given outcome metrics) in industry and suggests that using reachability analysis might improve upon/augment standard methods. Conventional reachability analysis, where the disturbance is assumed to live in a bounded set and act adversarially, produces too conservative (usually empty) safe sets that are marginally usefl for informing design choices. So, our Sustech submission treated the disturbance (storm) as a deterministic time-varying signal in order to show safe sets and interpret them in the context of stormwater design.

Thus, our Sustech submission did*not* appreciate the uncertainty that arises from how land responds to incoming precipitation.
(The precipitation pattern is fixed in the design stage; the pattern is called a design storm and is given by city regulations.)

① We propose to run the design storm through EPA's computer model (well-established software called SWMM) many times under different parameters to obtain a series of surface runoff plots over time.

② From these plots, we can estimate the probability distribution of our disturbance (the surface runoff) over time.
(The probability distribution will be time varying, I believe. $w_k \sim P_k$
I'm not sure if Chow's algorithm can handle this.)

③ The length of the design storm is 24h. We can choose the discretization step for our dynamics model (presented in Sustech submission; odes based on physics of how water flows between 2 ponds) to accomodate this storm length. We discretize our dynamics model (which is continuous time in the Sustech submission.)

④ We compute risk-sensitive safe sets on the discretized dynamics model, and interpret these sets in the context of stormwater design
(similar to Sustech submission but now with the risk-sensitive safe sets).