Trust-Aided Distributionally Robust Resource Allocation with Multi-Source Reference Information

Yanru Guo, Bo Zhou, Ruiwei Jiang, Siqian Shen, Xi (Jessie) Yang

Department of Industrial and Operations Engineering
University of Michigan at Ann Arbor

Outline

- Introduction
- Problem Description and Modeling
- Trust Update Process
- Computational Studies
- Conclusion

Outline

- **Introduction**
- Problem Description and Modeling
- Trust Update Process
- Computational Studies
- Conclusion

Motivation

Figure 1: Multi-source Information in wildfire events

Figure 1-1: https://news.wisc.edu/one-minute-data-from-uw-helps-nasa-detect-wildfires-faster/
Figure 1-2: https://www.nbcnews.com/mach/science/drones-are-fighting-wildfires-some-very-surprising-ways-ncna820966
Figure 1-3: https://www.doi.gov/wildlandfire/improving-wildfire-risk-reduction-through-ecosystem-mapping

Problem Context

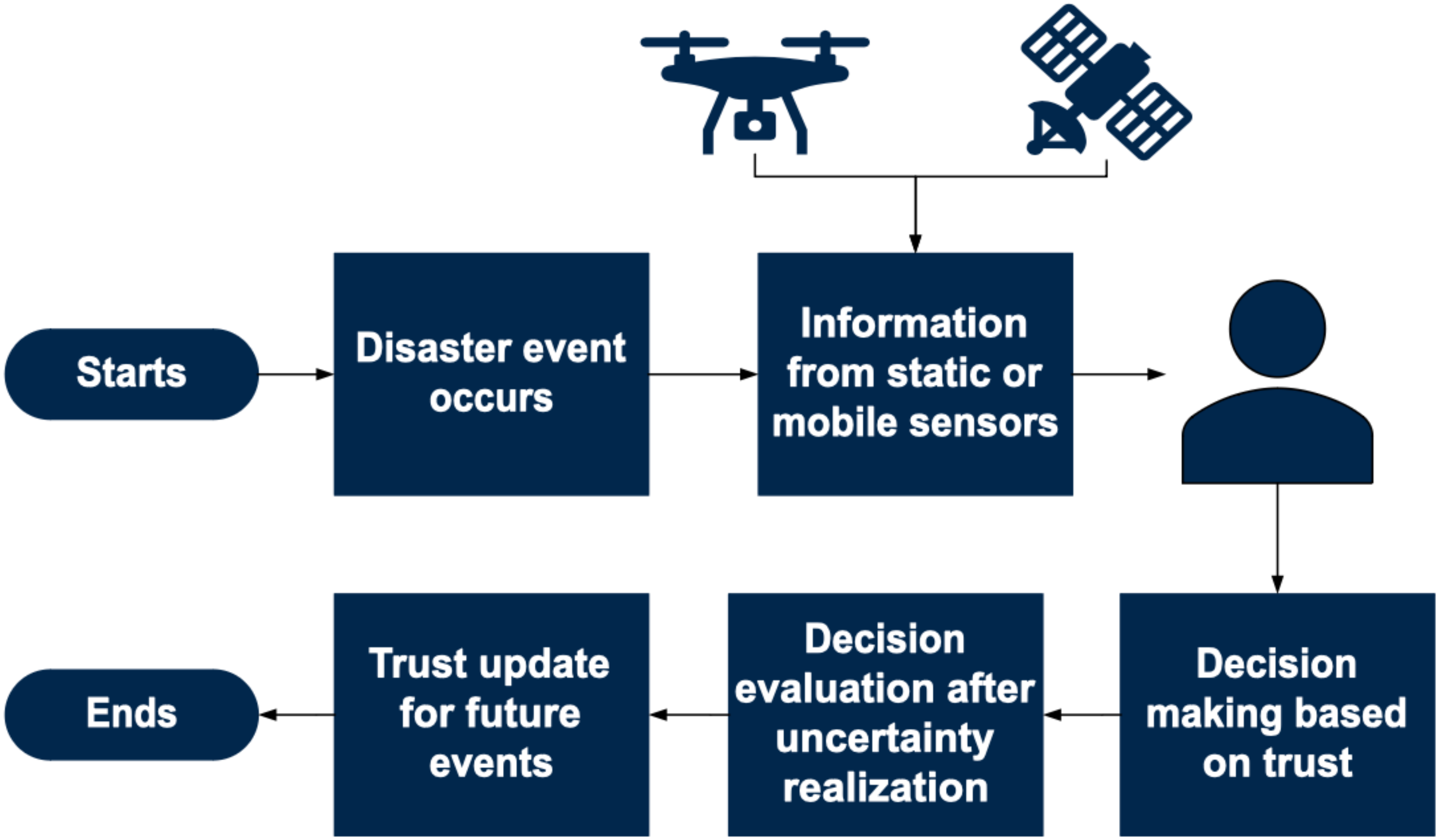


Figure 2: Overview of the problem

Outline

- Introduction
- **Problem Description and Modeling**
- Trust Update Process
- Computational Studies
- Conclusion

Resource Allocation under Uncertainty

- **Set**
  - Φ: set of subregions, $K = |\Phi|$.
- **Parameter**
  - c_k^u: unit penalty cost of unmet demand at subregion k, $\forall k \in [K]$.
  - c_k^o: unit penalty cost of overmet demand at subregion k, $\forall k \in [K]$.
  - B: resource budget.
  - ξ_k: random demand at subregion k for resources, $\forall k \in [K]$.
  - $\mathbb{P}_s$: $\boldsymbol{\xi}_s \sim \mathcal{N}(\boldsymbol{\mu}_s, \Sigma_s), \boldsymbol{\mu}_s = (\mu_{s1}, \ldots, \mu_{sK})^T, \Sigma_s = (\sigma_{s1}^2, \ldots, \sigma_{sK}^2)$
  - $\mathbb{P}_d$: $\boldsymbol{\xi}_d \sim \mathcal{N}(\boldsymbol{\mu}_d, \Sigma_d), \boldsymbol{\mu}_d = (\mu_{d1}, \ldots, \mu_{dK})^T, \Sigma_d = \left(\sigma_{d1}^2, \ldots, \sigma_{dK}^2\right)$
  - t_k: the trust in the drone at subregion k, $\forall k \in [K]$
- **Decision Variable**
  - x_k: the amount of resource assigned to subregion k, $\forall k \in [K]$.

Stochastic Programming Model (SP)

$$\inf_{\boldsymbol{x}\in\mathbb{X}} \mathbb{E}_{\boldsymbol{\xi}\sim\mathbb{P}}[\underbrace{(\boldsymbol{c}^u)^{\mathrm{T}}(\boldsymbol{\xi}-\boldsymbol{x})^+}_{\text{under-allocation penalty}} + \underbrace{(\boldsymbol{c}^o)^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{\xi})^+}_{\text{over-allocation cost}}]$$

$$s.t.\ \mathbb{X} = \left\{\boldsymbol{x}\in\mathbb{R}_+^K: \sum_{k=1}^{K} x_k \le B\right\}. \tag{1}$$

Apply the Monte Carlo Sampling Approach to replace $\mathbb{P}$ with an empirical distribution constructed by $|\Omega|$ scenarios with each scenario $\omega \in \Omega$ having an equal probability $p^\omega = \frac{1}{|\Omega|}$ and reformulate the problem as:

$$\min_{\boldsymbol{x}\in\mathbb{X}} \sum_{\omega\in\Omega} p^\omega\,[(\boldsymbol{c}^u)^{\mathrm{T}}(\boldsymbol{\xi}^\omega-\boldsymbol{x})^+ + (\boldsymbol{c}^o)^{\mathrm{T}}(x-\boldsymbol{\xi}^\omega)^+] \tag{2}$$

Distributionally Robust Model (DRO)

For notation simplicity, we let $\ell(\boldsymbol{x}, \boldsymbol{\xi}) = (\boldsymbol{c}^u)^{\mathrm{T}}(\boldsymbol{\xi} - \boldsymbol{x})^+ + (\boldsymbol{c}^o)^{\mathrm{T}}(\boldsymbol{x} - \boldsymbol{\xi})^+$ and then we have the DRO model:

$$\inf_{\boldsymbol{x} \in \mathbb{X}} \left\{ \sup_{\mathbb{P} \in \mathcal{P}} \mathop{\mathbb{E}}_{\boldsymbol{\xi} \sim \mathbb{P}} [\ell(\boldsymbol{x}, \boldsymbol{\xi})] \right\} \tag{3}$$

where the definition of ambiguity set $\mathcal{P}$ is given by:

$$\mathcal{P} := \left\{ \mathbb{P} \in \mathcal{M}(\Xi) : d_W\left(\mathbb{P}, \hat{\mathcal{P}}_N\right) \leq \epsilon \right\} \tag{4}$$

$$d_W\left(\mathbb{P}, \hat{\mathcal{P}}_N\right) = \inf \left\{ \mathop{\mathbb{E}}_{(\boldsymbol{\xi}, \hat{\boldsymbol{\xi}}) \sim \Pi} \left[\| \boldsymbol{\xi} - \hat{\boldsymbol{\xi}} \| \right] \right\} \tag{5}$$

Trust-Aided Ambiguity Set

Recall:

$\mathbb{P}_s$: $\boldsymbol{\xi}_s \sim \mathcal{N}(\boldsymbol{\mu}_s, \Sigma_s), \boldsymbol{\mu}_s = (\mu_{s1}, \ldots, \mu_{sK})^T, \Sigma_s = (\sigma_{s1}^2, \ldots, \sigma_{sK}^2)$

$\mathbb{P}_d$: $\boldsymbol{\xi}_d \sim \mathcal{N}(\boldsymbol{\mu}_d, \Sigma_d), \boldsymbol{\mu}_d = (\mu_{d1}, \ldots, \mu_{dK})^T, \Sigma_d = \left(\sigma_{d1}^2, \ldots, \sigma_{dK}^2\right)$

t_k: the trust in the drone at subregion k, $\forall k \in [K]$

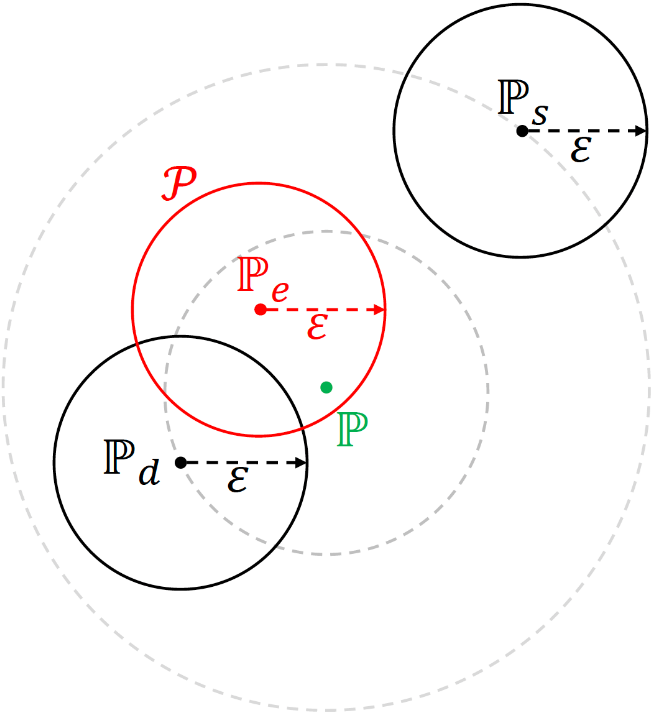


Figure 3 Trust-aided ambiguity set

We assume the empirical uniform distribution $\hat{\mathcal{P}}_N$ on N independent and identically distributed samples is sampled from a normal distribution $\mathbb{P}_e \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ satisfying the following conditions:

- For $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)^T$,

$$\mu_k = t_k \cdot \mu_{dk} + (1 - t_k) \cdot \mu_{sk}$$

- For $\Sigma = (\sigma_1^2, \ldots, \sigma_K^2)^T$,

$$\sigma_k^2 = t_k^2 \cdot \sigma_{dk}^2 + (1 - t_k)^2 \cdot \sigma_{sk}^2$$

DRO Reformulation [1]

Then, we can reformulate the **inner worst case problem** as an LP:

$$\inf_{\lambda, s_{ik}, \gamma_{ijk}} \lambda\epsilon + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} s_{ik} \tag{6}$$

$$s.t. \quad b_{jk} + \langle a_{jk}, \hat{\xi}_{ik} \rangle + \langle \gamma_{ijk}, d_k - C_k \hat{\xi}_{ik} \rangle \leq s_{ik}, \quad \forall i \in [N], j \in [J], k \in [K]$$

$$\| C_k^T \gamma_{ijk} - a_{jk} \|_* \leq \lambda \forall i \in [N], j \in [J], k \in [K]$$

$$\gamma_{ijk} \geq 0 \quad \forall i \in [N], j \in [J], k \in [K]$$

Finally, we reformulate our **original DRO** (3) as an LP:

$$\inf_{\mathbf{x}, \lambda, s_{ik}, \gamma_{ijk}} \lambda\epsilon + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} s_{ik} \tag{7}$$

$$\text{s.t.} \quad x \in \mathbb{X}$$

$$b_{jk} + \langle a_{jk}, \hat{\xi}_{ik} \rangle + \langle \gamma_{ijk}, d_k - C_k \hat{\xi}_{ik} \rangle \leq s_{ik}, \quad \forall i \in [N], j \in [J], k \in [K]$$

$$\| C_k^T \gamma_{ijk} - a_{jk} \|_* \leq \lambda \forall i \in [N], j \in [J], k \in [K]$$

$$\gamma_{ijk} \geq 0 \quad \forall i \in [N], j \in [J], k \in [K]$$

[1] Mohajerin Esfahani, P., & Kuhn, D. (2018). Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and tractable reformulations. *Mathematical Programming, 171*(1-2), 115-166.

Outline

- Introduction
- Problem Description and Modeling
- **Trust Update Process**
- Computational Studies
- Conclusion

Trust Update Process Overview

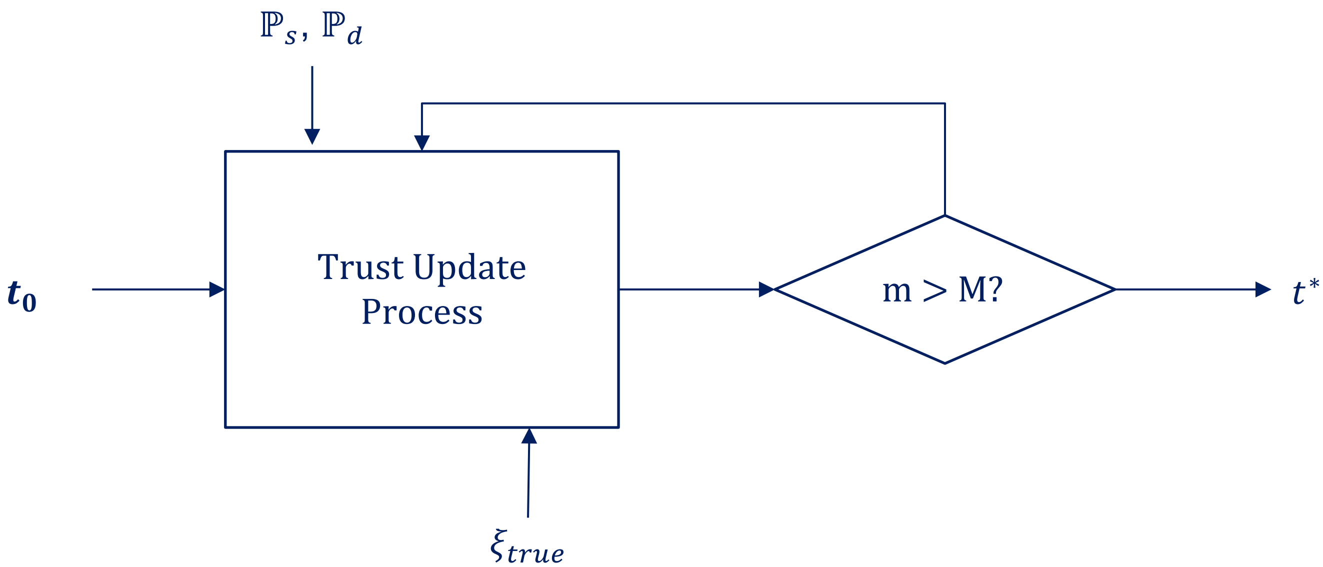


Figure 4 Trust update overview

Assumption

- The demand realization $\boldsymbol{\xi}_{\text{true}}$ follows normal distribution $\mathcal{N}(\boldsymbol{\mu}_{\text{true}}, \boldsymbol{\Sigma}_{\text{true}})$.
- The relative error $\boldsymbol{r}_s$ (or $\boldsymbol{r}_d$) is fixed: $\boldsymbol{\mu}_s = \langle \boldsymbol{\mu}_{\text{true}}, \boldsymbol{r}_s \rangle$ (or $\boldsymbol{\mu}_d = \langle \boldsymbol{\mu}_{\text{true}}, \boldsymbol{r}_d \rangle$), and $\mathbf{r}_s < \mathbf{1} < \mathbf{r}_d$.
- For each iteration m, $\boldsymbol{\xi}_{true}$ is different, and $\mathbb{P}_s$ and $\mathbb{P}_d$ will change accordingly.

Trust Update Process Overview

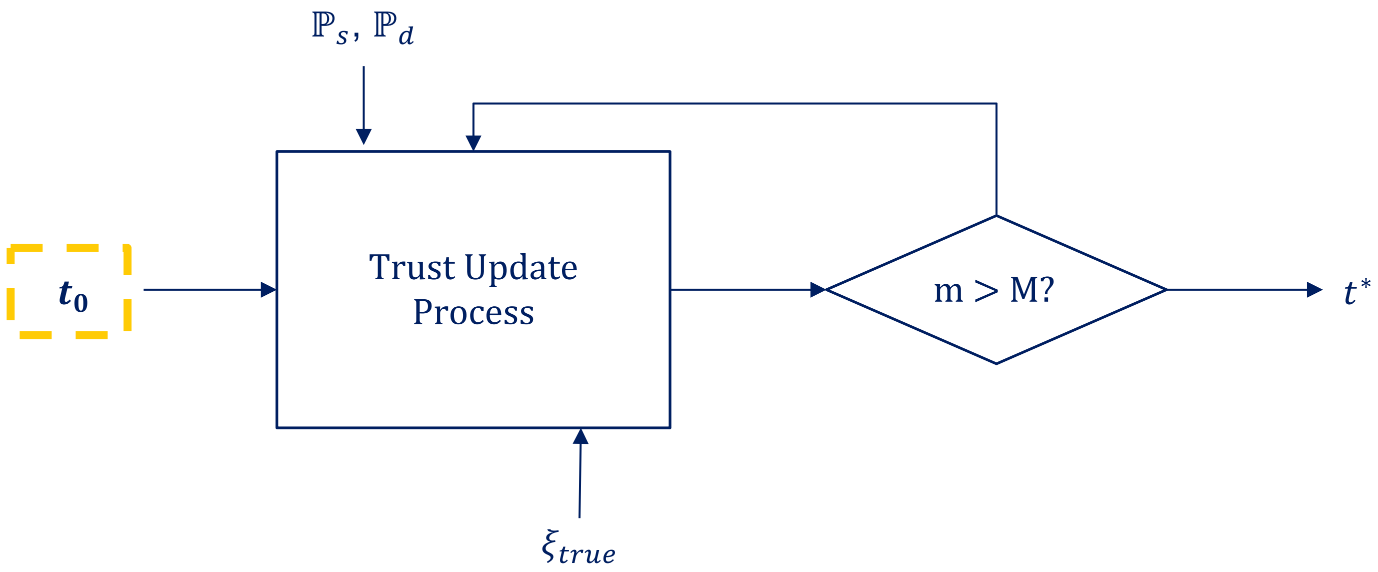


Figure 4 Trust update overview

We use M to denote the number of iterations in the trust update process. Before the trust update process begin, the decision maker holds an original trust $\mathbf{t}_0$ over the drone.

One Iteration in Trust Update Process

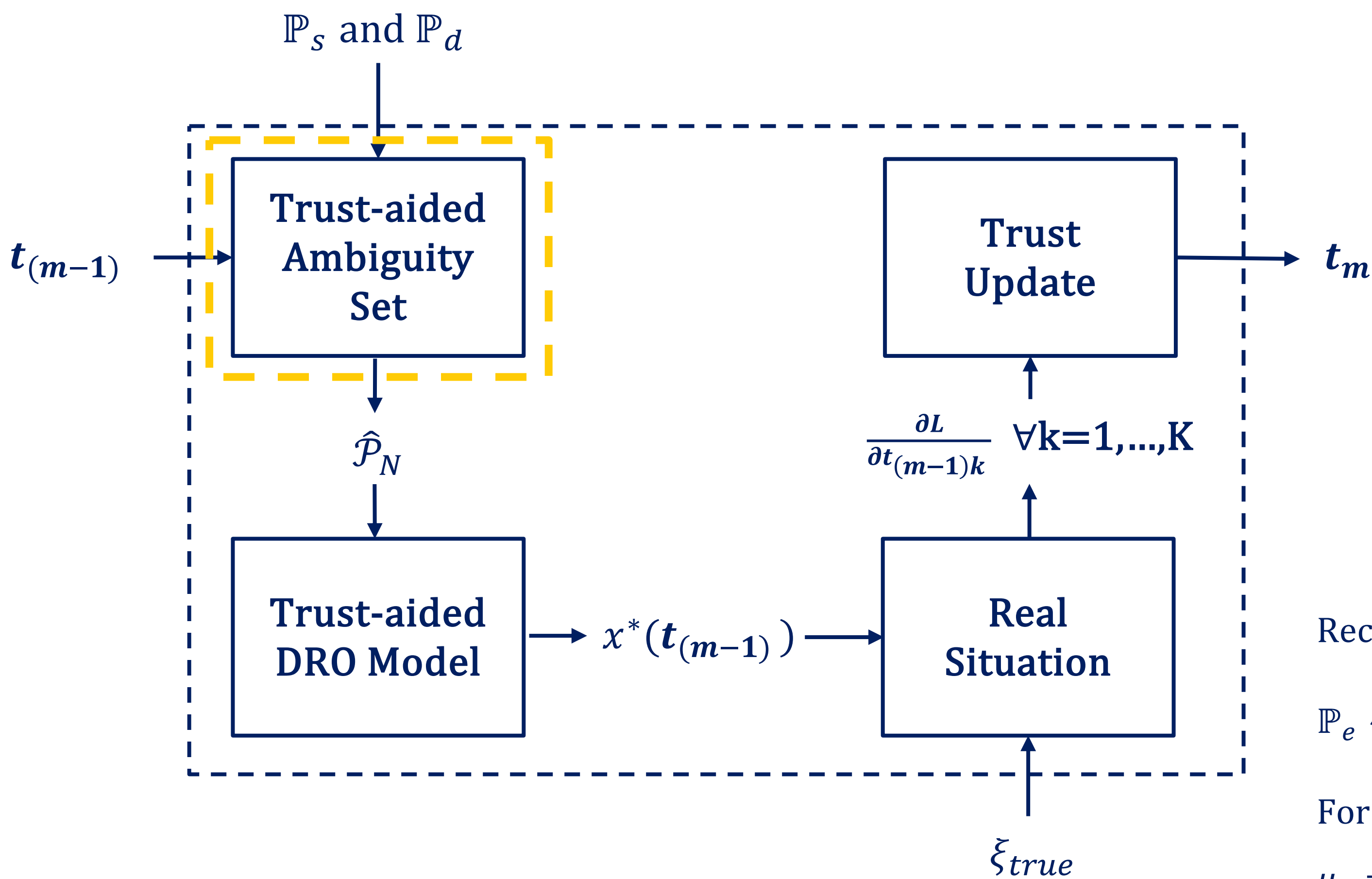


Figure 5 One iteration in trust update process

Recall:

$\mathbb{P}_e \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$

For $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)^T$,

$\mu_k = t_k \cdot \mu_{dk} + (1 - t_k) \cdot \mu_{sk}$

For $\Sigma = (\sigma_1^2, \ldots, \sigma_K^2)^T$,

$\sigma_k^2 = t_k^2 \cdot \sigma_{dk}^2 + (1 - t_k)^2 \cdot \sigma_{sk}^2$

One Iteration in Trust Update Process

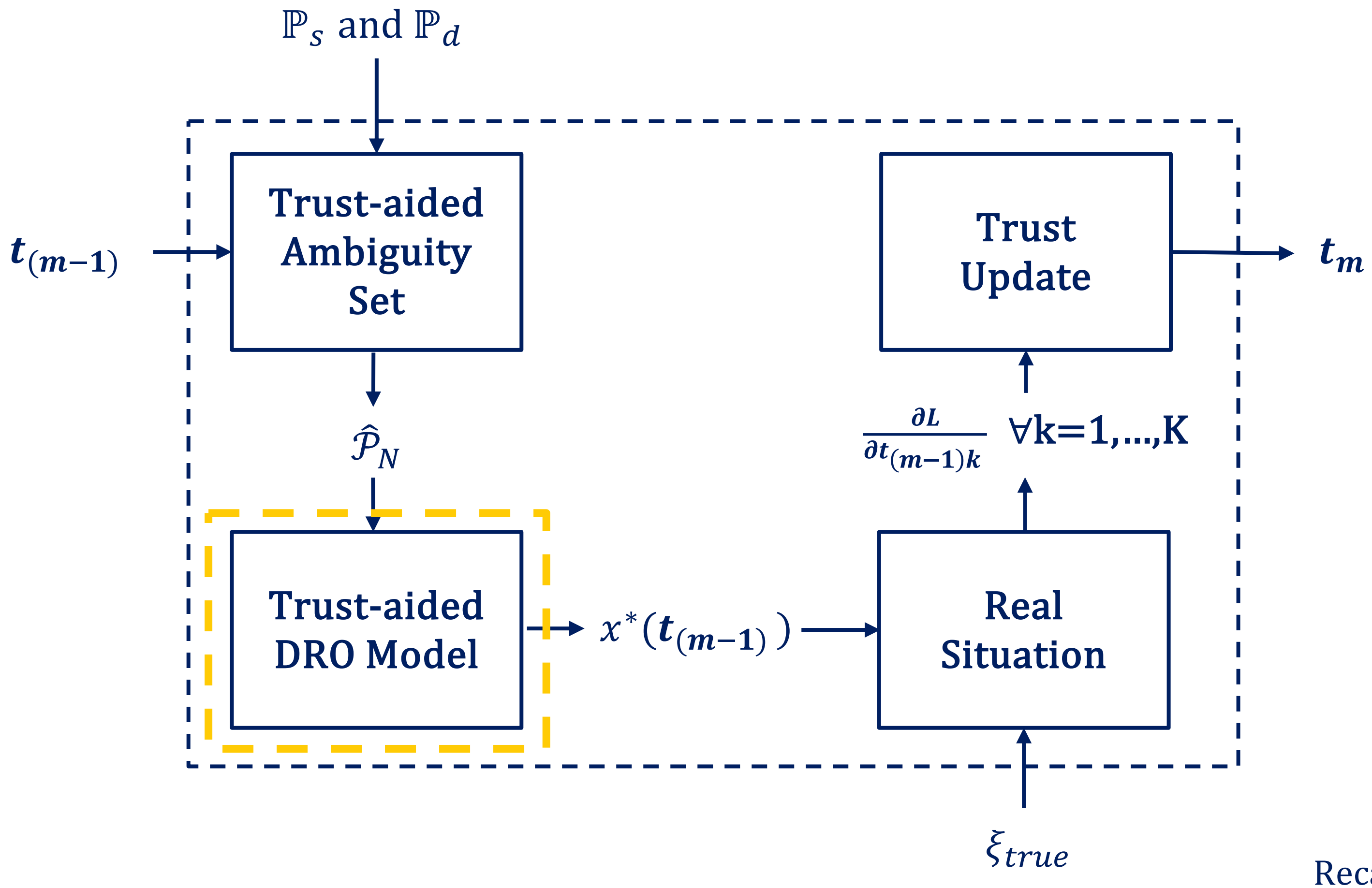


Figure 5 One iteration in trust update process

Recall:

$$\inf_{\boldsymbol{x}\in\mathbb{X}}\left\{\sup_{\mathbb{P}\in\mathcal{P}}\mathbb{E}_{\boldsymbol{\xi}\sim\mathbb{P}}[\ell(\boldsymbol{x},\boldsymbol{\xi})]\right\}$$

One Iteration in Trust Update Process

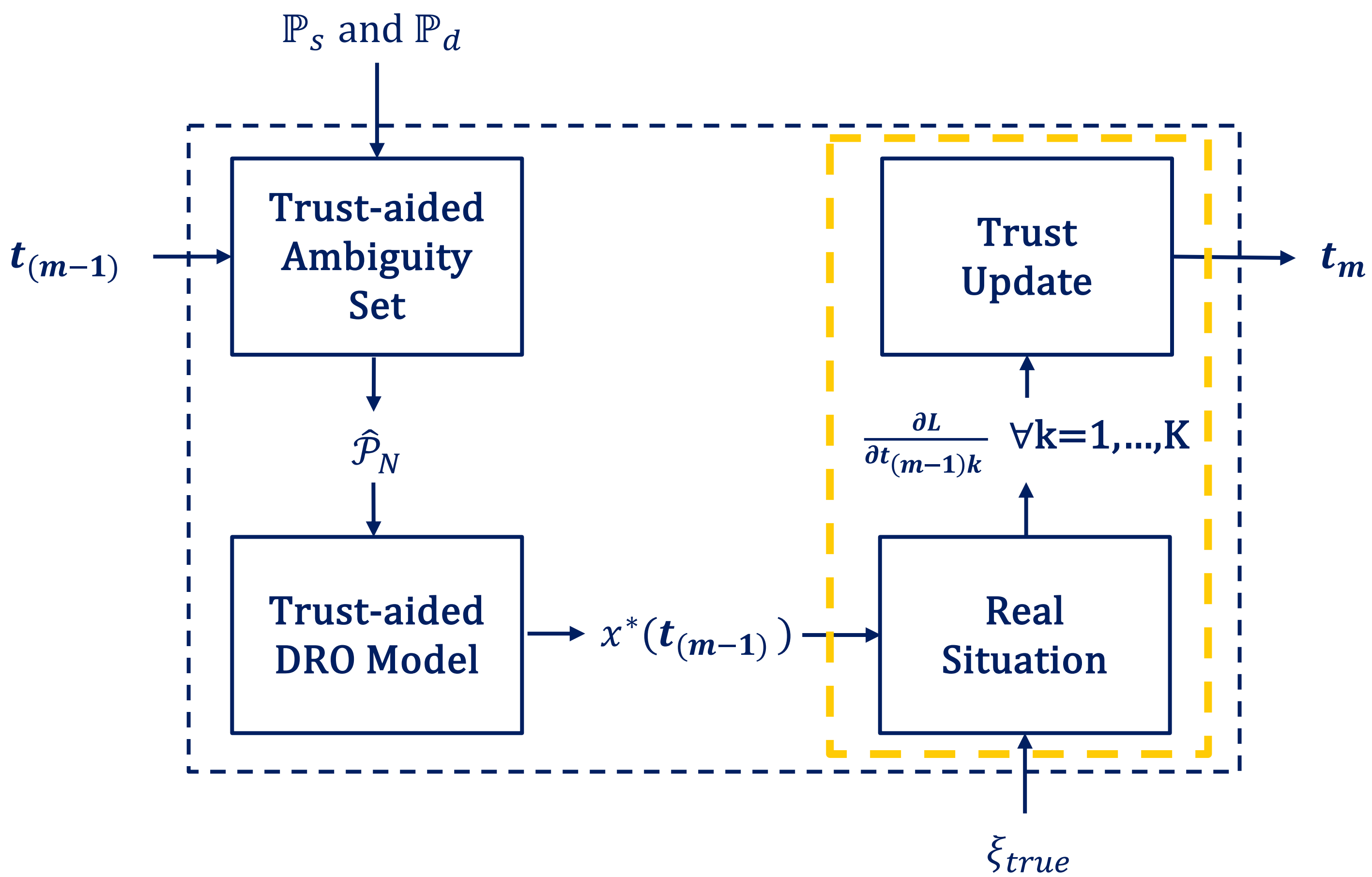


Figure 5 One iteration in trust update process

One Iteration in Trust Update Process

$$
\begin{aligned}
L &= \ell(\mathbf{x}^*, \boldsymbol{\xi}_{\text{true}}) \\
&= (\mathbf{c}^u)^{\mathrm{T}}[\boldsymbol{\xi}_{\text{true}} - \mathbf{x}^*(\mathbf{t})]^+ + (\mathbf{c}^o)^{\mathrm{T}}[\mathbf{x}^*(\mathbf{t}) - \boldsymbol{\xi}_{\text{true}}]^+ \\
&= \sum_{k=1}^{K} \left[c_k^u \left[\xi_{\text{true},k} - x_k^*(\mathbf{t})\right]^+ + c_k^o \left[x_k^*(\mathbf{t}) - \xi_{\text{true},k}\right]^+ \right]
\end{aligned} \tag{8}
$$

Ideally, we would like to update the trust t_{mk} based on the partial derivative.

$$
t_{mk} = \begin{cases} t_{(m-1)k} + w^s, & \dfrac{\partial L}{\partial t_{(m-1)k}} < 0 \\ t_{(m-1)k}, & \dfrac{\partial L}{\partial t_{(m-1)k}} = 0 \\ t_{(m-1)k} - w^f, & \dfrac{\partial L}{\partial t_{(m-1)k}} > 0 \end{cases}
$$

Due to the complexity of (8), we use the incremental trial-and-error method to update trust with $w_s = w_f = 0.01$.

Trust Update Process Overview

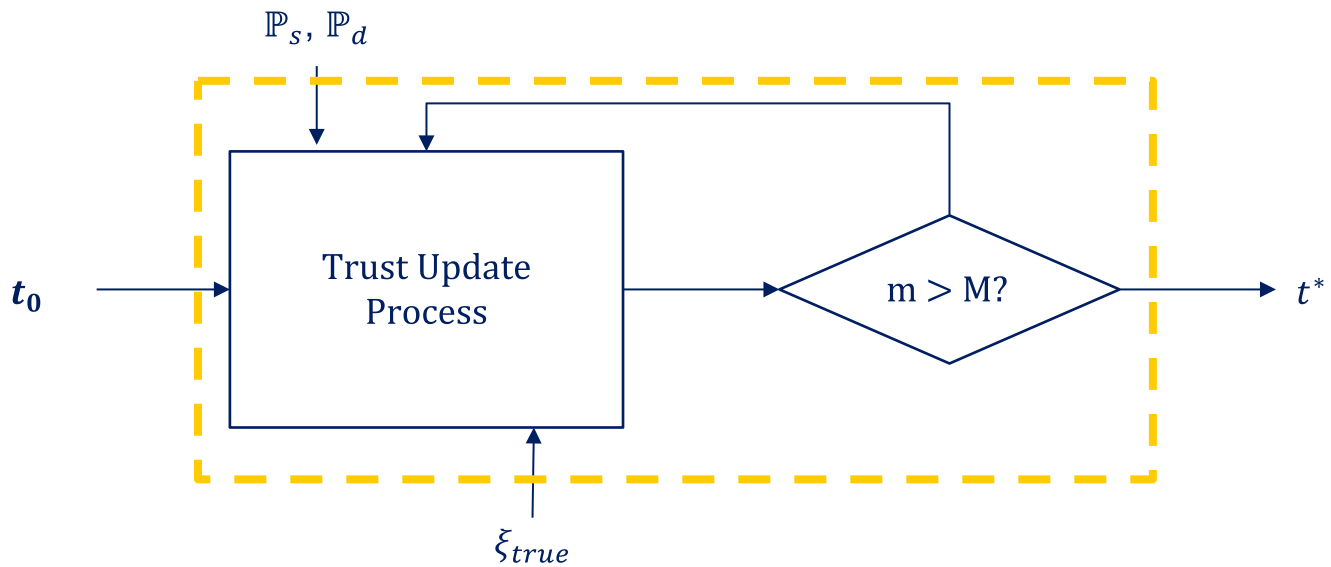


Figure 4 Trust update overview

Based on these assumptions, as m grows, we should see t_{mk} starts fluctuating near t_k^{ideal}, which satisfies

$$\mu_{\text{true},k} = t_k^{\text{ideal}} \mu_{dk} + \left(1 - t_k^{\text{ideal}}\right) \mu_{sk}$$

Trust Update Process Overview

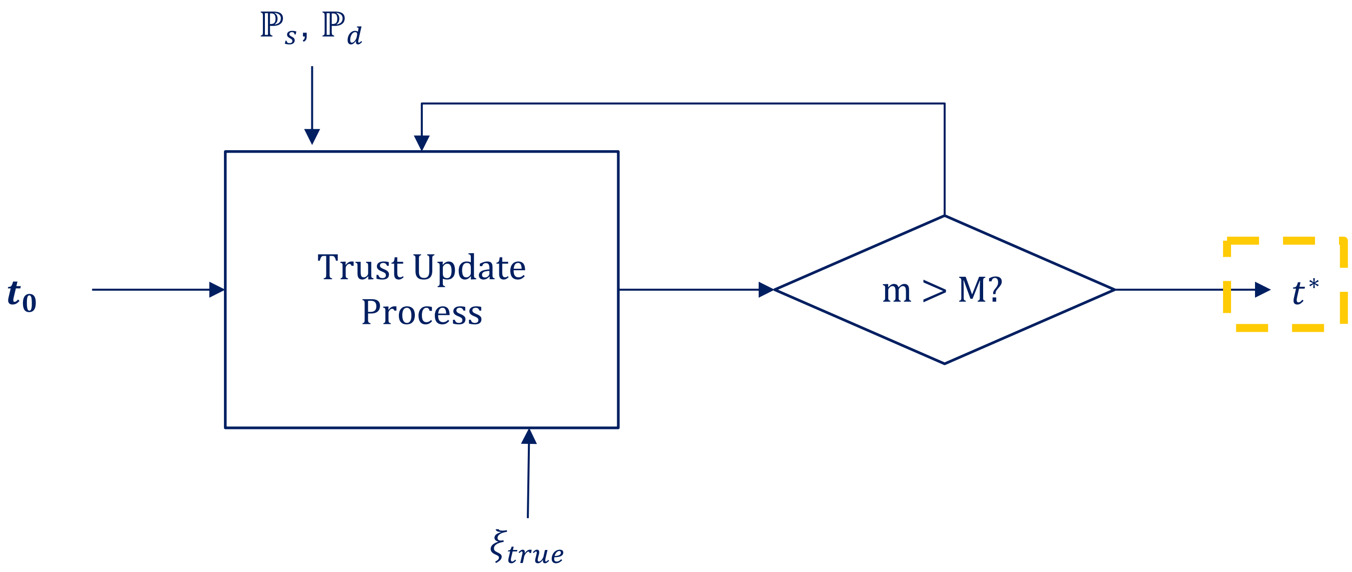


Figure 4 Trust update overview

Trust Value Selection

After M iterations in the trust update process, we can get an applicable trust value t^* through two methods:

1) The applicable trust value t^* equals the max trust value of the fluctuating interval, $t^* = t^*_{Max}$;
2) The applicable trust value t^* equals the average of the trust value in the fluctuating interval, $t^* = t^*_{Avg}$.

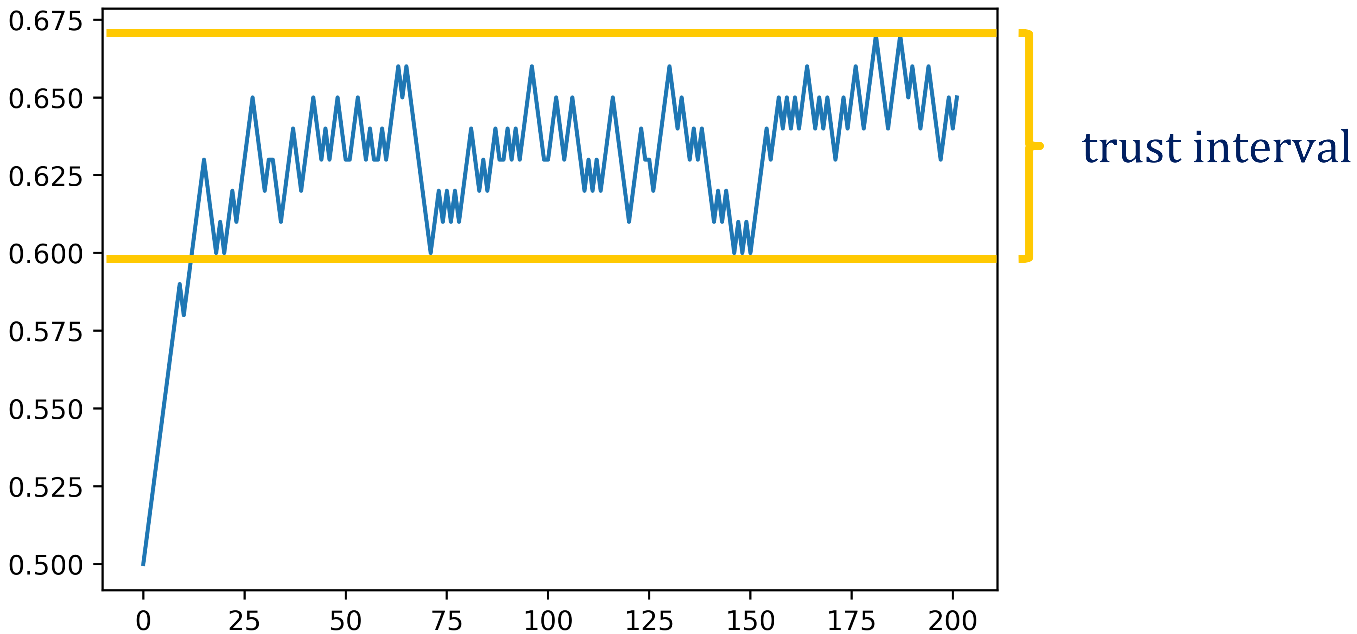


Figure 6 Trust interval

Outline

- Introduction
- Problem Description and Modeling
- Trust Update Process
- **Computational Studies**
- Conclusion

Experimental Setup

- $K = 3$ regions, $M = 200$ iterations, $N = 1000$ samples
- $\boldsymbol{c^u} = [5000, 10000, 15000]^{\mathrm{T}}$, $\boldsymbol{c^o} = [1, 1, 1]^{\mathrm{T}}$, $B = 500$
- $\boldsymbol{\xi_{true}} \sim \mathcal{N}(\boldsymbol{\mu_{true}}, \Sigma_{true})$, each element in $\boldsymbol{\mu_{true}}$ is an integer value sampled from $[100,200]$, and $\boldsymbol{\sigma_{true}} = 0.01 * \boldsymbol{\mu_{true}}$
- $\boldsymbol{r_s} = (0.7, 0.6, 0.3)^{\mathrm{T}}$, $\boldsymbol{r_d} = (1.1, 1.2, 1.1)^{\mathrm{T}}$, $\boldsymbol{\sigma_s} = 0.02 * \boldsymbol{\mu_s}$, $\boldsymbol{\sigma_d} = 0.02 * \boldsymbol{\mu_d}$
- $\boldsymbol{t_0} = (0.5, 0.5, 0.5)^{\mathrm{T}}$
- For out-of-sample tests, $\mathrm{H} = 100$, each element in $\boldsymbol{\mu_{true}}$ in event h is a uniform integer random value sampled from $[100,200]$ and $\boldsymbol{\sigma_{true}} = 0.01 * \boldsymbol{\mu_{true}}$

Trust Update Process with baseline setting

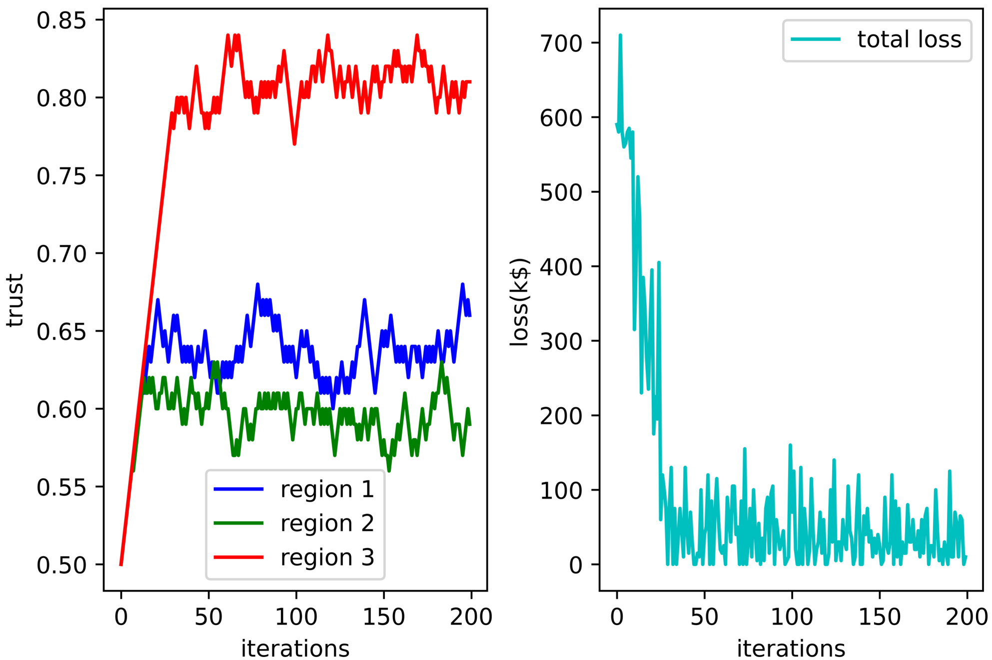


Figure 7 Trust update process with the baseline setting

Table 1 Trust update process with the baseline setting

K	M	Trust Interval	Avg. Loss (k$)	Time (sec.)
3	200	[0.60,0.68] [0.56,0.63] [0.77,0.84]	41.68	1360.03

(Trust Interval denotes the range of fluctuating trust, Avg. Loss denotes the average losses with trust in the fluctuating interval in thousand dollars, and Time denotes the total computation time in seconds.)

According to the two methods for calculating the applicable trust value, we get $t^*_{Avg} = (0.64, 0.60, 0.81)^{\mathrm{T}}$ and $t^*_{Max} = (0.68, 0.63, 0.84)^{\mathrm{T}}$. In later comparison, we use t^*_{Max} in TA-DRO(1) and t^*_{Avg} in TA-DRO(2).

Out-of-sample performances with different methods

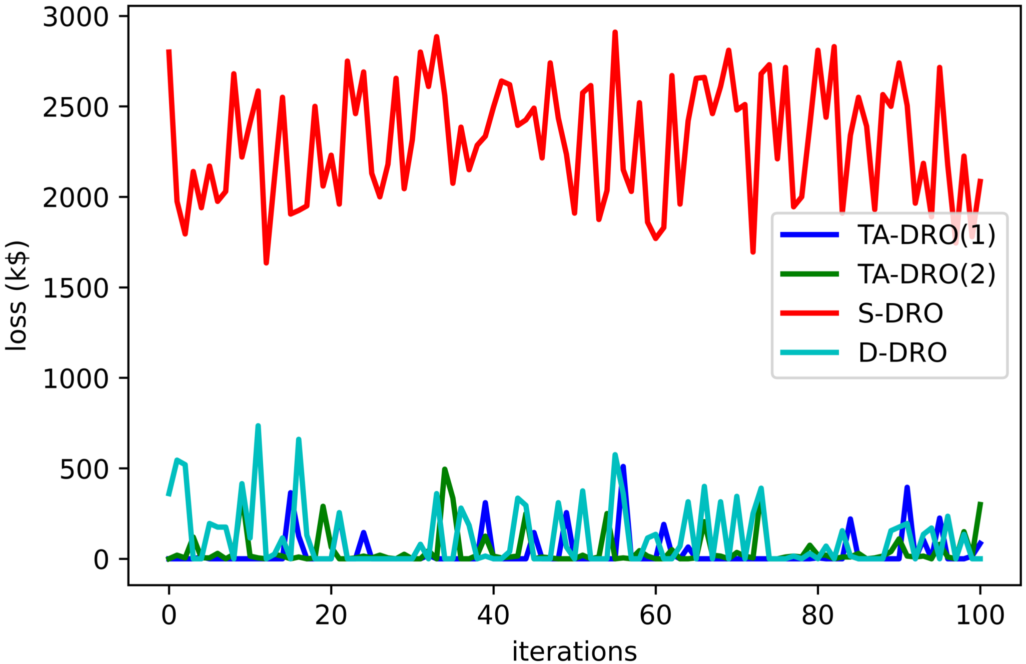


Figure 8 Out-of-sample performances (with trust v.s. without trust)

Table 2 Out-of-sample performances (with trust v.s. without trust)

K	M	H	Method	Avg. Loss (k$)	Time (sec.)
3	200	100	TA-DRO(1)	34.27	102.55
			TA-DRO(2)	43.91	99.73
			S-DRO	2311.83	98.60
			D-DRO	114.07	112.10

(Avg. Loss denotes the average loss of 100 out-of-sample events, and Time denotes the total time used for 100 out-of-sample events)

We compare TA-DRO with S-DRO and D-DRO and see TA-DRO has better out-of-sample performances with similar computation time and less average loss.

Out-of-sample performances with different methods

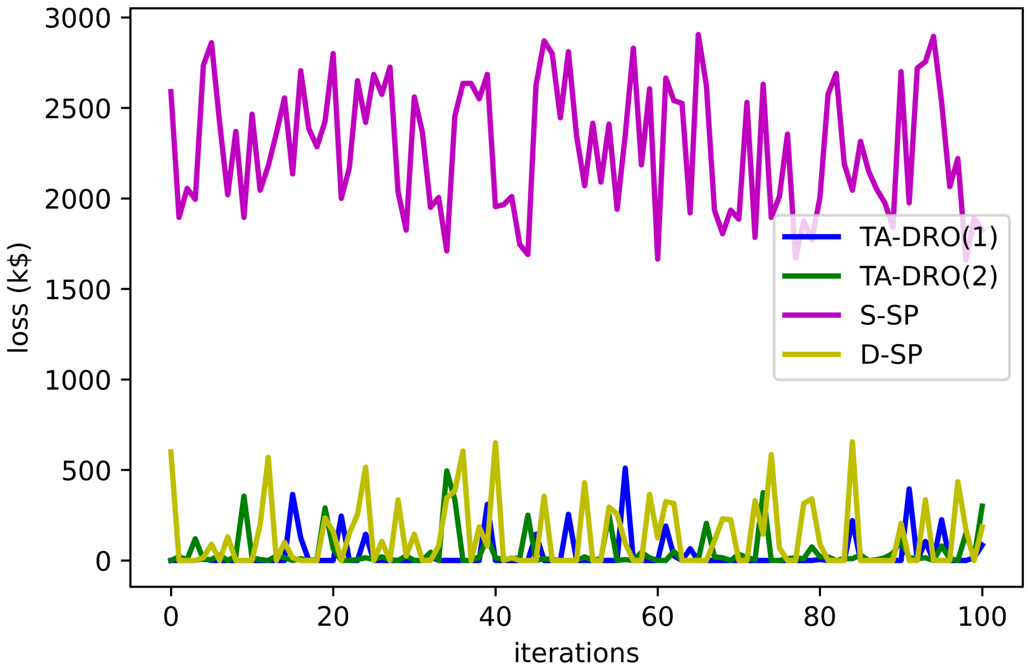


Figure 9 Out-of-sample performances (DRO v.s. SP)

Table 3 Out-of-sample performances (DRO v.s. SP)

K	M	H	Method	Avg. Loss (k$)	Time (sec.)
3	200	100	TA-DRO(1)	34.27	102.55
			TA-DRO(2)	43.91	99.73
			S-SP	2276.83	84.35
			D-SP	128.53	88.46

(Avg. Loss denotes the average loss of 100 out-of-sample events, and Time denotes the total time used for 100 out-of-sample events)

We compare TA-DRO with S-SP and D-DP and see TA-DRO has better out-of-sample performances with slightly higher computation time but less average loss. We also see that under current setting TA-DRO(1) with t^*_{Max} performs better.

Sensitivity analysis with varying M

Table 4 Trust update process with varying M

K	M	Trust Interval	Avg. Loss (k$)	Time (sec.)
3	10	N/A N/A N/A	N/A	75.79
	50	[0.62,0.66] [0.59,0.62] [0.78,0.82]	43.25	341.97
	100	[0.61,0.68] [0.57,0.63] [0.77,0.84]	47.07	679.38
	200	[0.60,0.68] [0.57,0.63] [0.77,0.84]	41.68	1360.03
	300	[0.59,0.68] [0.56,0.63] [0.77,0.84]	43.72	2049.65

Scalability Analysis

Table 5 Trust update process with varying K

K	M	Time (sec.)
3		1360.03
5	200	3944.50
10		17012.75

(Time denotes the total computation time in seconds.)

The increase of number of regions not only leads to an increase in computation time for solving TA-DRO once, but also causes the incremental trial-and-error update process to take more time.

Outline

- Introduction
- Problem Description and Modeling
- Trust Update Process
- Computational Studies
- **Conclusion**

Conclusion

- We construct a **trust-aided ambiguity set** and build a **TA-DRO** for solving a disaster relief resource allocation problem.
- We propose a **trust update process** to reflect human trust varying on different sources of information.
- We conduct various experiments based on hypothetical data. Results indicate that under current assumptions the **TA-DRO performs better** than DRO solely based on information from the satellite or the drone.

Future research to address the following limitation:

- the current way of information fusion
- the assumption of fixed optimistic (or conservative) prediction given by the satellite (or the drone)
- a static one-stage problem

Continuing Study – Probability Fusion

$$\min_{x\in\mathcal{X}}\left\{\max_{\mathbb{P}\in\mathcal{P}}\mathbb{E}_{\xi\sim\mathbb{P}}[Q(x,\xi)]\right\}$$

Satellite $\tilde{\xi}_s$

$$\xi=\tilde{\xi}_s+\Delta\xi_s$$

$$\Delta\xi_s\sim\mathbb{P}^e_{\Delta,s}\qquad \mathbb{P}^e_{\Delta,s}(\Delta\xi_s)=\frac{1}{N}\sum_j\delta\left(\Delta\xi_s-\Delta\hat{\xi}_{s,j}\right)$$

$$\mathbb{P}^e_s(\xi)=\frac{1}{N}\sum_j\delta\left(\xi-\tilde{\xi}_s-\Delta\hat{\xi}_{s,j}\right)$$

$$\mathcal{P}_s=\left\{\mathbb{P}:\Xi\to R\,\middle|\,\begin{matrix}\mathbb{E}_{\mathbb{P}}[1]=1\\ d_W(\mathbb{P},\mathbb{P}^e_s)\le\varepsilon_s\end{matrix}\right\},\ \varepsilon_s=\varepsilon_s\left(\Delta\hat{\xi}_{s,1:N},\beta\right)$$

Drone $\tilde{\xi}_d$

$$\xi=\tilde{\xi}_d+\Delta\xi_d$$

$$\Delta\xi_d\sim\mathbb{P}^e_{\Delta,d}\qquad \mathbb{P}^e_{\Delta,d}(\Delta\xi_d)=\frac{1}{N}\sum_j\delta\left(\Delta\xi_d-\Delta\hat{\xi}_{d,j}\right)$$

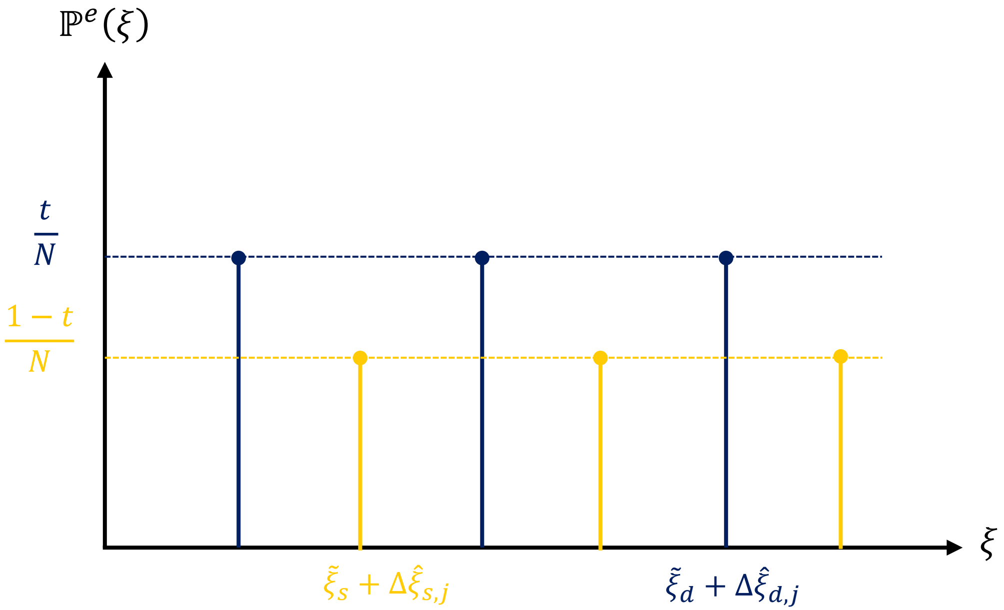
$$\mathbb{P}^e_d(\xi)=\frac{1}{N}\sum_j\delta\left(\xi-\tilde{\xi}_d-\Delta\hat{\xi}_{d,j}\right)$$

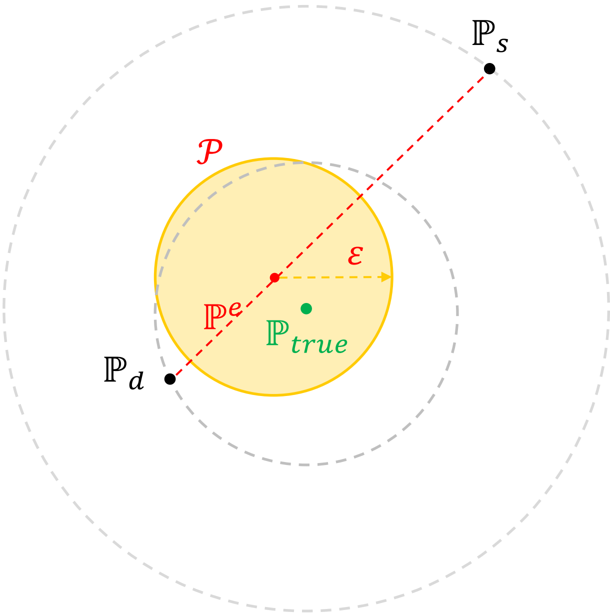
$$\mathcal{P}_d=\left\{\mathbb{P}:\Xi\to R\,\middle|\,\begin{matrix}\mathbb{E}_{\mathbb{P}}[1]=1\\ d_W(\mathbb{P},\mathbb{P}^e_d)\le\varepsilon_d\end{matrix}\right\},\ \varepsilon_d=\varepsilon_d\left(\Delta\hat{\xi}_{d,1:N},\beta\right)$$

Continuing Study - Probability Fusion

$$\mathbb{P}^e(\xi) = (1-t)\mathbb{P}_s^e(\xi) + t\mathbb{P}_d^e(\xi) = \frac{1-t}{N}\sum_j \delta_{\tilde{\xi}_s + \Delta\hat{\tilde{\xi}}_{s,j}} + \frac{t}{N}\sum_j \delta_{\tilde{\xi}_d + \Delta\hat{\tilde{\xi}}_{d,j}}$$

$$\mathcal{P} = \left\{ \mathbb{P}: \Xi \to R \,\middle|\, \begin{array}{l} \mathbb{E}_{\mathbb{P}}[1] = 1 \\ d_W(\mathbb{P}, \mathbb{P}^e) \le \varepsilon \end{array} \right\}, \varepsilon = \varepsilon\left(\Delta\hat{\tilde{\xi}}_{s,1:N}, \Delta\hat{\tilde{\xi}}_{d,1:N}, \beta, t\right)$$





Thanks for listening!