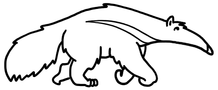
Cloud-Assisted On-Sensor Observation Classification in Latency-Impeded IoT Systems

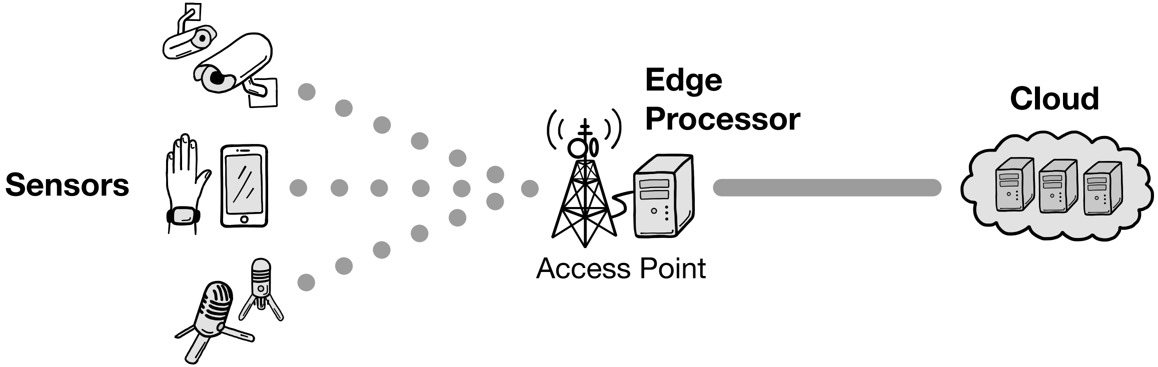
Igor Burago Marco Levorato

Department of Computer Science
University of California, Irvine

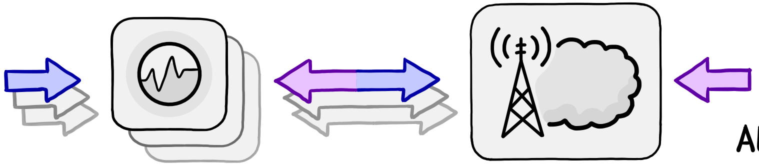


2019

Intelligent IoT in Principle



Information
Data to drive decisions



Intelligence
Algorithms to realize autonomy

Sensor

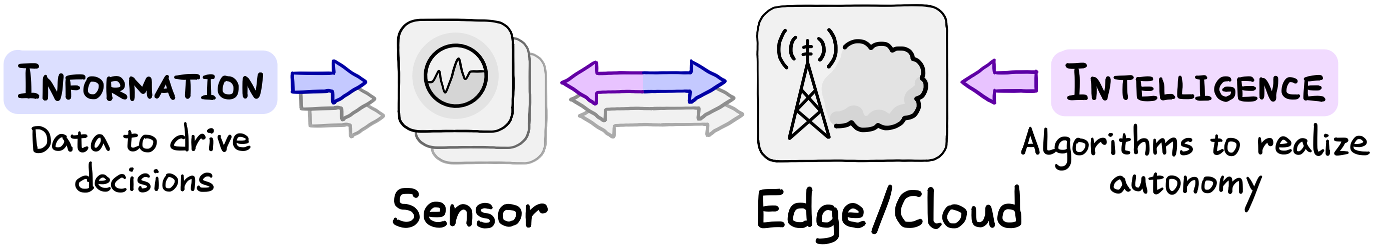
- Observes the environment
- Selectively informs the edge
- Adjusts its own operation

Edge/Cloud

- Accumulates relevant observations
- Exercises high-level control
- Assists on-sensor decision-making

Goal: Realize the alliance between information completeness and best intelligence

Intelligent IoT Challenge

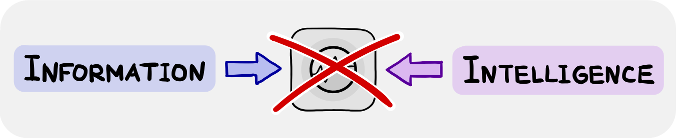


System is **constrained** by:

AND

- → **Sensor**

  High-quality decision cannot be made locally

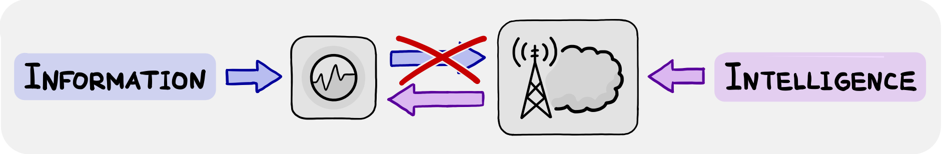


- → **Channel** with bottleneck in:

OR

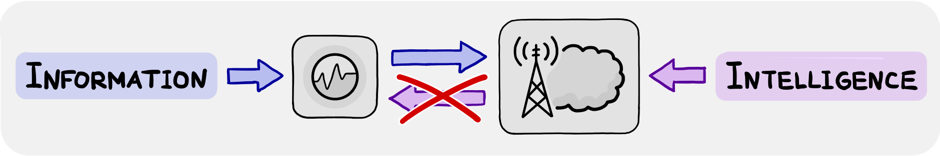
  - → **Bandwidth**

    Edge/Cloud cannot see all observations in full



  - → **Latency**

    Decision turnaround is too slow for reactive sensor adjustments



Approaches to Computation-Communication Tradeoff

Strategies vary in the amount of:

Decision
Quality

Approaches to Computation-Communication Tradeoff

Strategies vary in the amount of:

Decision Quality	On-Sensor Computation

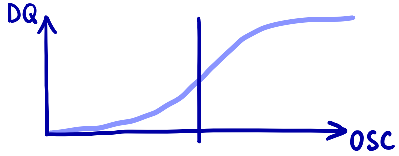
Approaches to Computation-Communication Tradeoff

Strategies vary in the amount of:	Decision Quality	On-Sensor Computation	Sensor-Cloud Communication

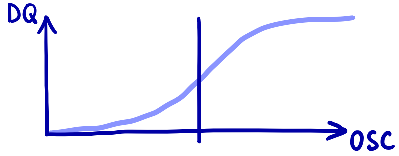
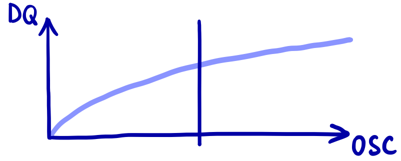
Approaches to Computation-Communication Tradeoff

	Decision Quality	On-Sensor Computation	Sensor-Cloud Communication
• Offload all decisions to cloud	Best	None	High

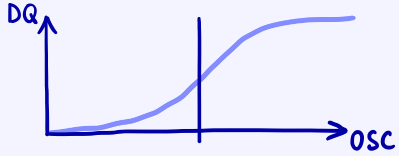
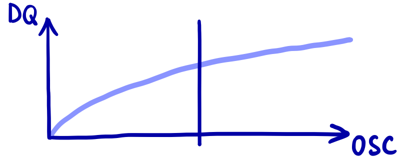
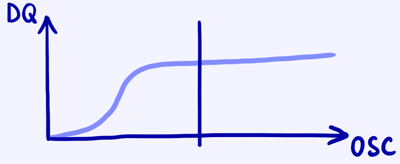
Approaches to Computation-Communication Tradeoff

	Decision Quality	On-Sensor Computation	Sensor-Cloud Communication
• Offload all decisions to cloud	Best	None	High
• Make all decisions on sensor			Low

Approaches to Computation-Communication Tradeoff

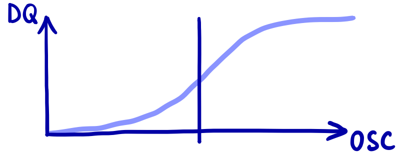
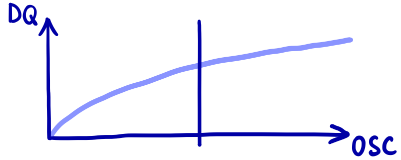
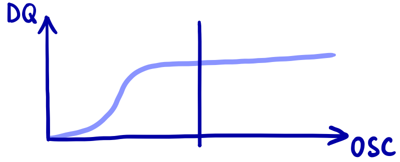
	Decision Quality	On-Sensor Computation	Sensor-Cloud Communication
• Offload all decisions to cloud	Best	None	High
• Make all decisions on sensor			Low
• Compress observations and offload all decisions to cloud			Medium

Approaches to Computation-Communication Tradeoff

	Decision Quality /	On-Sensor Computation	Sensor-Cloud Communication
• Offload all decisions to cloud	Best	None	High
• Make all decisions on sensor			Low
• Compress observations and offload all decisions to cloud			Medium
• Selectively transmit observations to cloud for decisions*			Low

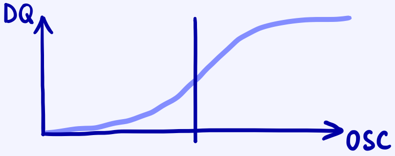
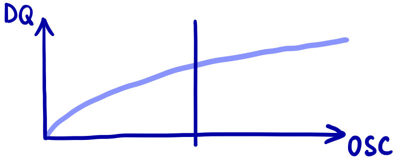
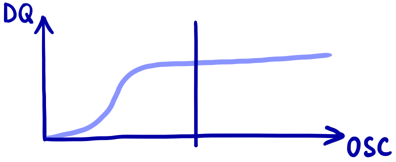
* I. Burago and M. Levorato, "Randomized Edge-Assisted On-Sensor Information Selection for Bandwidth-Constrained Systems", Asilomar Conference on Signals, Systems, and Computers, 2018.

Approaches to Computation-Communication Tradeoff

	Decision Quality	On-Sensor Computation	Sensor-Cloud Communication	Decision Turnaround
• Offload all decisions to cloud	Best	None	High	Delayed
• Make all decisions on sensor			Low	None
• Compress observations and offload all decisions to cloud			Medium	Delayed
• Selectively transmit observations to cloud for decisions*			Low	Delayed

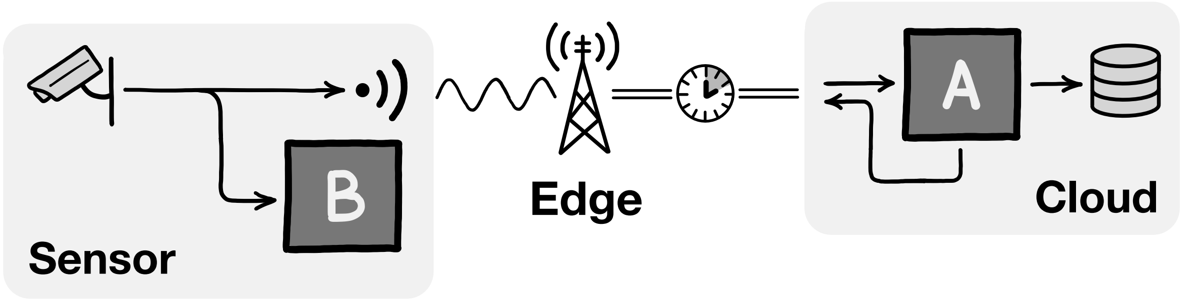
* I. Burago and M. Levorato, "Randomized Edge-Assisted On-Sensor Information Selection for Bandwidth-Constrained Systems", Asilomar Conference on Signals, Systems, and Computers, 2018.

Approaches to Computation-Communication Tradeoff

	Decision Quality / On-Sensor Computation		Sensor-Cloud Communication	Decision Turnaround
• Offload all decisions to cloud	Best	None	High	Delayed
• Make all decisions on sensor			Low	None
• Compress observations and offload all decisions to cloud			Medium	Delayed
• Selectively transmit observations to cloud for decisions*			Low	Delayed

* I. Burago and M. Levorato, "Randomized Edge-Assisted On-Sensor Information Selection for Bandwidth-Constrained Systems", Asilomar Conference on Signals, Systems, and Computers, 2018.

Sensor-Cloud Cooperation Model



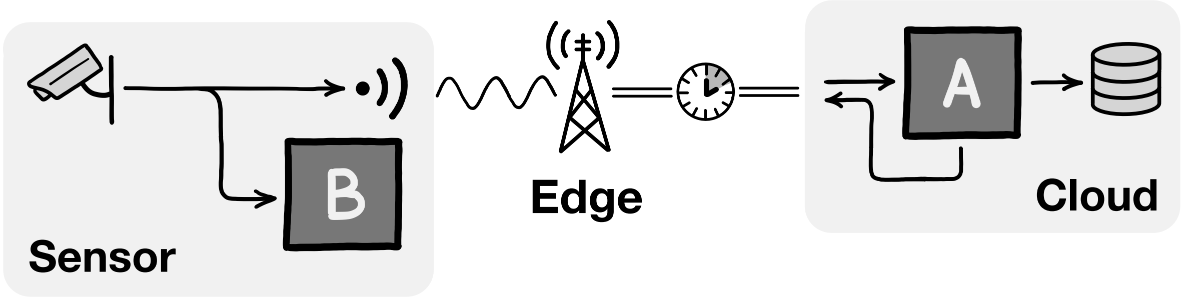
Model assumptions:

- Observations fall into two classes: – of interest to the sensor (positives)
  – and not (negatives)
- Sensor adjusts to its environment by reacting to positives
- Cloud can run the optimal classifier δ; sensor can not
- Sensor-cloud link has sufficient bandwidth
  but unacceptable latency for sensor adaptivity

B
- Sensor runs a low-complexity classifier,
  supplied by cloud, to make local decisions

A
- Cloud, via supervision of its classifier, trains the sensor's

Sensor-Cloud Cooperation Model

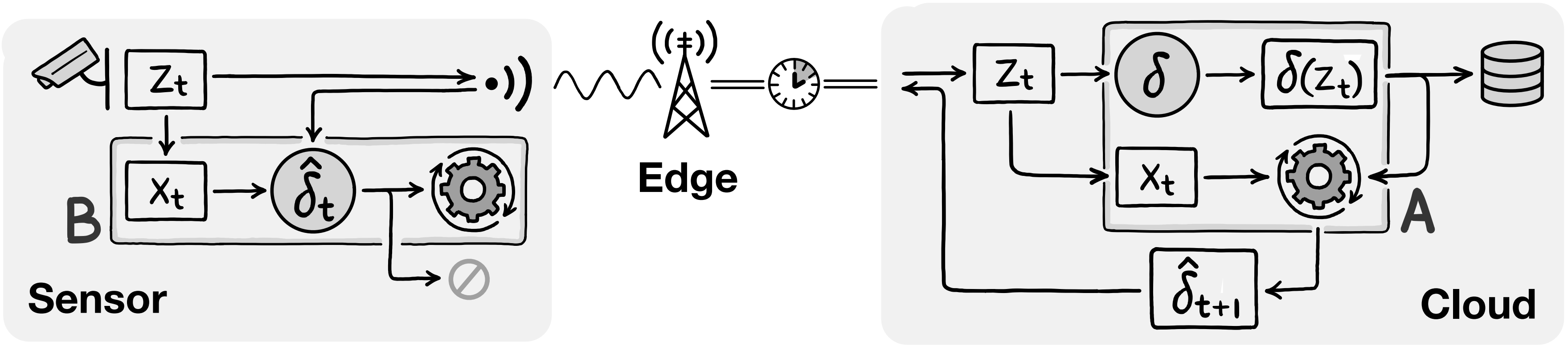


Factors at play

- Delayed channel
- No tolerance for missing decisions
- Auxiliary decision $\hat{\delta}$ to learn

- CPU, battery, etc.
- Computation constraints
- Complexity of $\hat{\delta}$

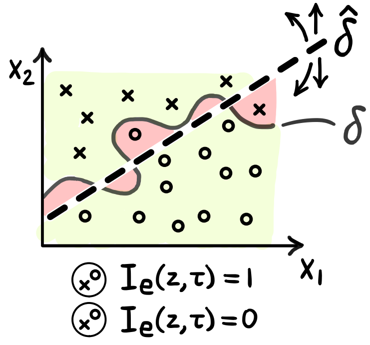
- Need to react to environment
- Cost of error in feature space
- Expected risk V to minimize



Sensor-Cloud Cooperation Model

Problem Formulation

For the on-sensor decision rule $\hat{\delta}: f(x,\theta) \gtrless \mu$



$$\underbrace{V(\tau) \triangleq \mathbb{E}\left[I_E(z,\tau) \, |r(u_\theta - \mu)| \right]}_{\text{Expected penalty for errors}} \longrightarrow \min_{\tau} \quad \text{s.t.} \quad \varphi(\tau) \leq \varphi_*$$

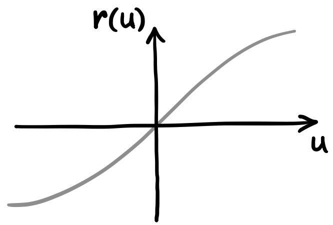
$\underbrace{\tau = \mathrm{vec}[\theta, \mu]}_{\text{Parameters}}$ $\underbrace{u_\theta \triangleq f(x,\theta)}_{\text{Decision function}}$ $\underbrace{\mu}_{\text{Threshold}}$ $\underbrace{\varphi(\tau) \triangleq \mathbb{E}[\hat{I}_\mu^{(1)}(u_\theta)]}_{\text{Probability of positives}}$ $\underbrace{\varphi_* > \pi_1 \triangleq \mathbb{E}[I_1(z)]}_{\text{Potential positive classifiability}}$

$\hat{I}_\mu^{(1)}$, I_1 $\in \{0,1\}$ indicate whether an observation z with features x is classified as a positive and is a true positive, respectively.

$I_E(z,\tau) \triangleq (1 - I_1(z))\hat{I}_\mu^{(1)}(u_\theta) + I_1(z)(1 - \hat{I}_\mu^{(1)}(u_\theta))$ is the error indicator.

r is a loss function based on the "distance" $u_\theta - \mu$.

In the simplest case, $r(u_\theta - \mu) = u_\theta - \mu$.



Choice of Criterion

Error probability ($r(\cdot) \triangleq 1$):

$$\mathbb{E}[I_E(z,\tau)] \longrightarrow \min_\tau$$

Challenging to build a gradient estimator.

For instance, let $F(\tau) \triangleq \int_{\{x : f(x,\tau)>0\}} p(x)\,dx$.

Then $\nabla F(\tau) = -\int_{\{x:\, f(x,\tau)>0\}} \mathrm{div}_x [\Lambda(x,\tau) p(x)]\, dx,$ (A)

$$\Lambda(x,\tau) \triangleq \frac{\nabla_\tau f(x,\tau) (\nabla_x f(x,\tau))^T}{\|\nabla_x f(x,\tau)\|}$$

or $\nabla F(\tau) = \int_{\{x:\, f(x,\tau)=0\}} \frac{\nabla_\tau f(x,\tau)}{\|\nabla_x f(x,\tau)\|} p(x)\, d\sigma.$ (B)

(A) requires estimating $p(x)$ for $\nabla p(x)$. ← Practically impossible

(B) requires estimating a surface integral in X. ← Slow

Choice of Criterion

Error probability ($r(\cdot) \triangleq 1$):

$$\mathbb{E}\left[I_E(z,\tau)\right] \longrightarrow \min_\tau$$

Challenging to build a gradient estimator.

For instance, let $F(\tau) \triangleq \int_{\{x:f(x,\tau)>0\}} p(x)\,dx$.

Then $\nabla F(\tau) = -\int_{\{x:f(x,\tau)>0\}} \mathrm{div}_x\left[\Lambda(x,\tau)p(x)\right] dx,$ (A)

$$\Lambda(x,\tau) \triangleq \frac{\nabla_\tau f(x,\tau)\left(\nabla_x f(x,\tau)\right)^T}{\|\nabla_x f(x,\tau)\|}$$

or $\nabla F(\tau) = \int_{\{x:f(x,\tau)=0\}} \frac{\nabla_\tau f(x,\tau)}{\|\nabla_x f(x,\tau)\|} p(x)\,d\sigma.$ (B)

(A) requires estimating $p(x)$ for $\nabla p(x)$. ← Practically impossible

(B) requires estimating a surface integral in X. ← Slow

Expected risk (nontrivial $r(\cdot)$):

$$V(\tau) \triangleq \mathbb{E}\left[I_E(z,\tau)\,|r(u_\theta-\mu)|\right] \longrightarrow \min_\tau$$

Practical stochastic gradient is possible when

$I_E(z,\tau)$ and $r(\cdot)$ concord to allow $I_E(z,\tau)\cdot|r(\cdot)|$ to be differentiable

In $V(\tau)$: $r(u_\theta-\mu) > 0$ when $I_E(z,\tau)=1$

$r(u_\theta-\mu) \leq 0$ when $I_E(z,\tau)=0$

Decision Rule Update

Algorithm

$$\tau_{t+1} = \mathcal{H}(\tau_t - \gamma_t \widehat{G}(z_{t+1}, \tau_t)),$$

where $\widehat{G}(z,\tau) \triangleq \mathrm{vec}[J(z,\tau)\nabla_\theta f(x,\theta), -J(z,\tau)]$
is a stochastic quasi-gradient of $V(\tau)$;

$$J(z,\tau) \triangleq \begin{cases} +1, & \text{for false positive;} \\ 0, & \text{for correct;} \\ -1, & \text{for false negative;} \end{cases}$$

is the signed error indicator;

γ_t is learning rate; and

$\mathcal{H}(\tau)$ is a transformation encapsulating the a priori knowledge on parameter localization (or $\mathcal{H}(\tau) \equiv \tau$).

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t.
the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}}\,\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau;$
5) a sequence of learning rates $\gamma_t > 0$, s.t.
$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty;$
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\left[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\, -\mathbb{E}[J(z,\tau)]\right].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x) \|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x) \|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t.
the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}} \|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau$;
5) a sequence of learning rates $\gamma_t > 0$, s.t.
$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E} V_* < \infty$;
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\Big[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\, -\mathbb{E}[J(z,\tau)]\Big].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa}, \quad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t.
the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}}\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau;$
5) a sequence of learning rates $\gamma_t > 0$, s.t.
$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty;$
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \operatorname{vec}\left[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\ -\mathbb{E}[J(z,\tau)]\right].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \le C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^\kappa\right] \le C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \le C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \le C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t.
   the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
   $$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \le C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \le V(\tau) + C_{\mathcal{H}}\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau$;
5) a sequence of learning rates $\gamma_t > 0$, s.t.
   $$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty$;
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
  $$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\Big[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\ -\mathbb{E}[J(z,\tau)]\Big].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t.
the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}}\,\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau$;
5) a sequence of learning rates $\gamma_t > 0$, s.t.
$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*$, $\quad \mathbb{E}V_* < \infty$;
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\left[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\, -\mathbb{E}[J(z,\tau)]\right].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t. the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies

$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$

4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}}\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau$;
5) a sequence of learning rates $\gamma_t > 0$, s.t.

$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty$;
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:

$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\Big[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\ -\mathbb{E}[J(z,\tau)]\Big].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t. the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}}\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau;$
5) a sequence of learning rates $\gamma_t > 0$, s.t.
$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

E.g., if $\gamma_t = Ct^{-a}$, $C, a > 0$, then $\frac{\kappa+1}{2\kappa} < a \leq 1$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty;$
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\Big[\mathbb{E}[J(z,\tau)\nabla_\theta f(x,\theta)],\ -\mathbb{E}[J(z,\tau)]\Big].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[(C_f(x))^\kappa\right] \leq C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \leq C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\left[C_{\nabla f}(x)\right] \leq C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t.
the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies
$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \leq C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$
4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \leq V(\tau) + C_{\mathcal{H}}\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau;$
5) a sequence of learning rates $\gamma_t > 0$, s.t.
$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty;$
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:
$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \operatorname{vec}\left[\mathbb{E}\left[J(z,\tau)\nabla_\theta f(x,\theta)\right], -\mathbb{E}\left[J(z,\tau)\right]\right].$$

Decision Rule Update Guarantees

Theorem 1 (Algorithm convergence) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) a continuously differentiable decision function $f(x,\theta)$, s.t. $\forall x, \theta', \theta''$

$$|f(x,\theta'') - f(x,\theta')| \le C_f(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\big[(C_f(x))^\kappa\big] \le C_{f,\kappa}, \qquad \kappa > 1,$$
$$|\nabla_\theta f(x,\theta'') - \nabla_\theta f(x,\theta')| \le C_{\nabla f}(x)\,\|\theta''-\theta'\|, \qquad \mathbb{E}\big[C_{\nabla f}(x)\big] \le C_{\nabla f};$$

3) a feature-extraction function $\chi(z)$, s.t. the c.d.f. F_{u_θ} of the random variable $u_\theta \triangleq f(\chi(z),\theta)$ satisfies

$$|F_{u_\theta}(u'') - F_{u_\theta}(u')| \le C_F\,|u''-u'|, \qquad \forall \theta, u', u'';$$

4) a mapping $\mathcal{H}(\tau)$, s.t. $V(\mathcal{H}(\tau)) \le V(\tau) + C_{\mathcal{H}}\|\mathcal{H}(\tau)-\tau\|^2 \quad \forall \tau;$
5) a sequence of learning rates $\gamma_t > 0$, s.t.

$$\sum_{t=1}^{\infty} \gamma_t = \infty, \qquad \sum_{t=1}^{\infty} \gamma_t^{2\lambda} < \infty, \qquad \lambda \triangleq \frac{\kappa}{\kappa+1};$$

the algorithm:

- converges in criterion: $\lim\limits_{t\to\infty} V(\tau_t) \overset{a.s.}{=} V_*, \quad \mathbb{E}V_* < \infty;$
- achieves the necessary condition of extremum for $V(\tau)$ on a subsequence:

$$\liminf_{t\to\infty} \|G(\tau_t)\| \overset{a.s.}{=} 0, \qquad G(\tau) \triangleq \mathrm{vec}\Big[\mathbb{E}\big[J(z,\tau)\nabla_\theta f(x,\theta)\big], -\mathbb{E}\big[J(z,\tau)\big]\Big].$$

Decision Rule Update Guarantees

Theorem 2 (Constraint satisfaction)

Under the assumptions of Theorem 1, if

1) condition 2) of Theorem 1 holds for $\kappa = \infty$, i.e.,
   $|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta'' - \theta'\|$, where $\mathbb{E}\left[(C_f(x))^\kappa\right] \leq C_{f,\kappa} \quad \forall \kappa$;
2) p.d.f. $p(x)$ of observations in the feature space that is continuous and has compact support;
3) $\mathcal{H}(\tau + \gamma\Delta) - \tau = \gamma\Psi(\tau)\Delta + O(\gamma^2)$ for some bounded matrix, $\|\Psi(\tau)\| \leq C_\Psi \ \forall \tau$;

then, on the trajectory of the algorithm,
$\varphi(\tau_t)$ converges to $\pi_1 < \varphi_*$ in mean square and w.p. 1.

Decision Rule Update Guarantees

Theorem 2 (Constraint satisfaction)

Under the assumptions of Theorem 1, if

1) condition 2) of Theorem 1 holds for $\kappa = \infty$, i.e.,
   $|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta'' - \theta'\|$, where $\mathbb{E}\left[(C_f(x))^\kappa\right] \leq C_{f,\kappa} \quad \forall \kappa$;
2) p.d.f. $p(x)$ of observations in the feature space that is continuous and has compact support;
3) $\mathcal{H}(\tau + \gamma\Delta) - \tau = \gamma\Psi(\tau)\Delta + O(\gamma^2)$ for some bounded matrix, $\|\Psi(\tau)\| \leq C_\Psi \ \forall \tau$;

then, on the trajectory of the algorithm,
$\varphi(\tau_t)$ converges to $\pi_1 < \varphi_*$ in mean square and w.p. 1.

Decision Rule Update Guarantees

Theorem 2 (Constraint satisfaction)

Under the assumptions of Theorem 1, if

1) condition 2) of Theorem 1 holds for $\kappa = \infty$, i.e.,
   $|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta'' - \theta'\|$, where $\mathbb{E}\left[(C_f(x))^\kappa\right] \leq C_{f,\kappa} \quad \forall \kappa$;
2) p.d.f. $p(x)$ of observations in the feature space that is continuous and has compact support;
3) $\mathcal{H}(\tau + \gamma\Delta) - \tau = \gamma\Psi(\tau)\Delta + O(\gamma^2)$ for some bounded matrix, $\|\Psi(\tau)\| \leq C_\Psi \quad \forall \tau$;

then, on the trajectory of the algorithm,
$\varphi(\tau_t)$ converges to $\pi_1 < \varphi_*$ in mean square and w.p. 1.

Decision Rule Update Guarantees

Theorem 2 (Constraint satisfaction)

Under the assumptions of Theorem 1, if

1) condition 2) of Theorem 1 holds for $\kappa=\infty$, i.e.,
   $|f(x,\theta'')-f(x,\theta')| \leq C_f(x)\,\|\theta''-\theta'\|$, where $\mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa} \quad \forall \kappa$;
2) p.d.f. $p(x)$ of observations in the feature space that is continuous and has compact support;
3) $\mathcal{H}(\tau+\gamma\Delta)-\tau=\gamma\Psi(\tau)\Delta+O(\gamma^2)$ for some bounded matrix, $\|\Psi(\tau)\| \leq C_\Psi \; \forall\tau$;

then, on the trajectory of the algorithm,
$\varphi(\tau_t)$ converges to $\pi_1 < \varphi_*$ in mean square and w.p. 1.

Decision Rule Update Guarantees

Theorem 2 (Constraint satisfaction)

Under the assumptions of Theorem 1, if

1) condition 2) of Theorem 1 holds for $\kappa = \infty$, i.e.,
$|f(x,\theta'') - f(x,\theta')| \leq C_f(x)\,\|\theta'' - \theta'\|$, where $\mathbb{E}\left[(C_f(x))^{\kappa}\right] \leq C_{f,\kappa} \quad \forall \kappa;$
2) p.d.f. $p(x)$ of observations in the feature space that is continuous and has compact support;
3) $\mathcal{H}(\tau + \gamma\Delta) - \tau = \gamma \Psi(\tau)\Delta + O(\gamma^2)$ for some bounded matrix, $\|\Psi(\tau)\| \leq C_\Psi \quad \forall \tau;$

then, on the trajectory of the algorithm,
$\varphi(\tau_t)$ converges to $\pi_1 < \varphi_*$ in mean square and w.p. 1.

Decision Rule Update Guarantees

Algorithm (Hyperplane separator):

$$\hat{\theta}_{t+1} = \theta_t - \gamma_t J(z_{t+1}, \tau_t)\left(\mathbf{I} - \theta_{t+1}\theta_{t+1}^T\right) x_{t+1},$$
$$\mu_{t+1} = \mu_t + \gamma_t J(z_{t+1}, \tau_t)$$
$$\tau_{t+1} = \mathrm{vec}\left[\hat{\theta}_{t+1} / \|\hat{\theta}_{t+1}\|, \mu_{t+1}\right]$$

In other words,

$$f(x, \theta) \triangleq \frac{\langle x, \theta \rangle}{\|\theta\|},$$
$$\mathcal{H}(\tau) \triangleq \mathrm{vec}[\theta / \|\theta\|, \mu].$$

Decision Rule Update Guarantees

Algorithm (Hyperplane separator): Guar

$$\hat{\theta}_{t+1} = \theta_t - \gamma_t J(z_{t+1}, \tau_t)(\mathbf{I} - \theta_{t+1}\theta_{t+1}^T) x_{t+1},$$
$$\mu_{t+1} = \mu_t + \gamma_t J(z_{t+1}, \tau_t)$$
$$\tau_{t+1} = \mathrm{vec}\left[\hat{\theta}_{t+1}/\|\hat{\theta}_{t+1}\|,\ \mu_{t+1}\right]$$

In other words,

$$f(x,\theta) \triangleq \frac{\langle x, \theta \rangle}{\|\theta\|},$$
$$\mathcal{H}(\tau) \triangleq \mathrm{vec}[\theta/\|\theta\|, \mu].$$

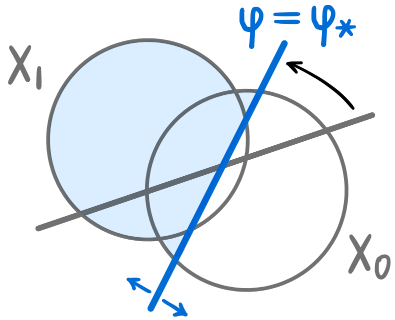
Theorem 3 (Hyperplane separator) Given

1) a sample of i.i.d. observations $\{z_t\}$;
2) p.d.f. $p(x)$ of observations in the feature space that is continuous and have compact support;
3) $\sum_{t=1}^{\infty} \gamma_t = \infty, \quad \sum_{t=1}^{\infty} \gamma_t^2 < \infty;$

the algorithm converges in criterion and achieves

$$\lim_{t\to\infty} \left\| \mathbb{E}_z[J(z_{t+1}, \tau_t)(\mathbf{I} - \theta_{t+1}\theta_{t+1}^T) x_{t+1}] \right\| \overset{a.s.}{=} 0,$$
$$\lim_{t\to\infty} \mathbb{E}_z[J(z_{t+1}, \tau_t)] \overset{a.s.}{=} 0,$$

i.e., $\varphi(\tau_t)$ converges to $\pi_1 < \varphi_*$ w.p. 1.



Cloud-Assisted On-Sensor Observation Classification in Latency-Impeded IoT Systems

Igor Burago
iburago@uci.edu

Marco Levorato
levorato@uci.edu

Intelligent and Autonomous Systems Lab
https://iasl.ics.uci.edu

Papers

- http://ib.fyi/isit19
- http://ib.fyi/acssc18