Discount-based Pricing and Capacity Planning for EV Charging under Stochastic Demand

Parthe Pandit, Samuel Coogan

ECE
UCLA
parthepandit@ucla.edu

ECE, CEE
Georgia Institute of Technology
sam.coogan@gatech.edu

29th June 2018

Motivation: Challenges due to widespread EV adoption

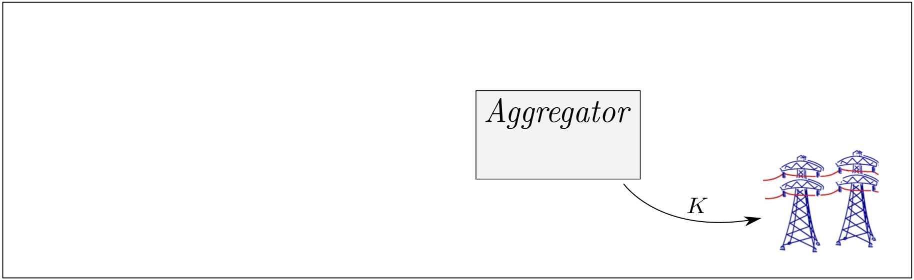
- Utilities need to plan capacity robust to load variability
- **Our focus**: *EV Charging Aggregators*
  - Buy from utility company, supply to EV users
  - For e.g. parking lots at airports, malls, and DC fast charging facilities etc.

- If an aggregator allows all users to charge at the fastest kW rate, Power capacity $(K) \propto$ Space $(M) \times$ Max charging rate $(r_{\max})$
- For e.g.: UCLA would require $20,000 \times 120$ kW $= 2.4$ GW

- **Problem**: Can we achieve a better scaling than the worst-case product above?
- **Solution**: Incentivise EV users to stay longer via *pricing*

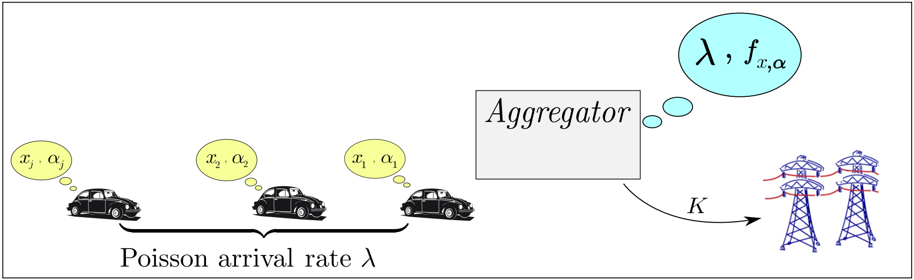
- ≈ 10 fold reduction for aggregator at the scale of UCLA

Problem formulation



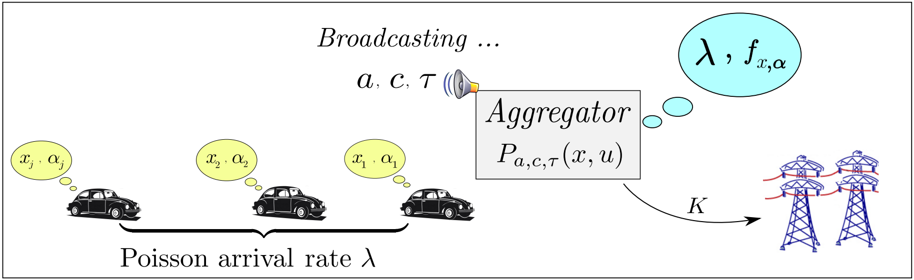
- Aggregator agrees with the utility not to exceed power usage by K (in kW)

Problem formulation



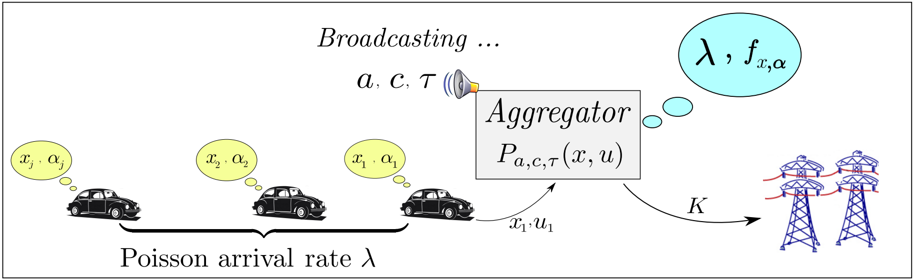
- Aggregator agrees with the utility not to exceed power usage by K (in kW)
- Users demand x_i (kW-hr) and impatience α_i (\$/hr), $\quad (x_i, \alpha_i) \sim f_{x,\alpha}$

Problem formulation



- Aggregator agrees with the utility not to exceed power usage by K (in kW)
- Users demand x_i (kW-hr) and impatience α_i (\$/hr), $(x_i, \alpha_i) \sim f_{x,\alpha}$
- Aggregator broadcasts parameters of pricing function
- Pricing function incentives users to provide longer service time deadlines

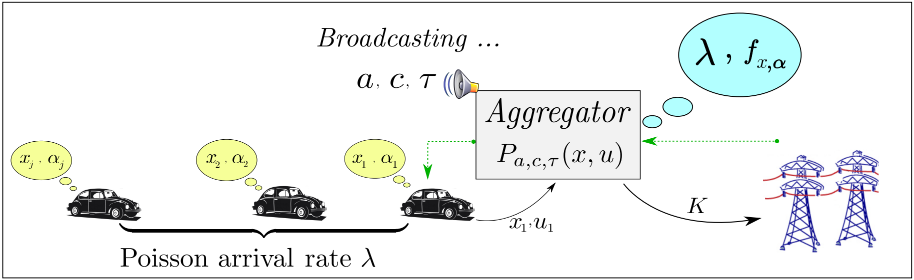
Problem formulation



- Aggregator agrees with the utility not to exceed power usage by K (in kW)
- Users demand x_i (kW-hr) and impatience α_i (\$/hr), $(x_i, \alpha_i) \sim f_{x,\alpha}$
- Aggregator broadcasts parameters of pricing function
- Pricing function incentives users to provide longer service time deadlines
- User i chooses their service time/deadline - u_i (in hr) based on the total cost

$$u_i = \underset{u \geq 0}{\operatorname{argmin}} \underbrace{P_{a,c,\tau}(x_i, u)}_{\text{Monetary cost}} + \underbrace{\alpha_i \, u}_{\text{Opportunity cost}}$$

Problem formulation



- Aggregator agrees with the utility not to exceed power usage by K (in kW)
- Users demand x_i (kW-hr) and impatience α_i (\$/hr), $(x_i, \alpha_i) \sim f_{x,\alpha}$
- Aggregator broadcasts parameters of pricing function
- Pricing function incentives users to provide longer service time deadlines
- User i chooses their service time/deadline - u_i (in hr) based on the total cost

$$u_i = \operatorname*{argmin}_{u \geq 0} \underbrace{P_{a,c,\tau}(x_i, u)}_{\text{Monetary cost}} + \underbrace{\alpha_i\, u}_{\text{Opportunity cost}}$$

- **Note**: Aggregator could be distributed since sum of Poisson processes is a Poisson process

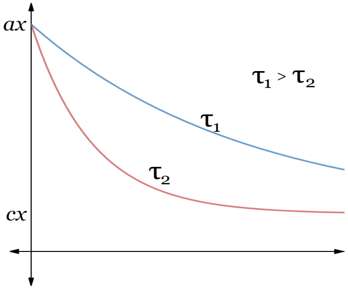
Discount-based pricing function

- Desired properties from $P(x, \cdot)$ to incentivise longer deadlines u_i
  - **Decreasing**: Users pay lower price for longer service times
  - E.g. Patient users (i.e. lower α_i, longer u_i) get more discount
  - **Convex**: Discounts are diminishing as service time increases
- We consider the following pricing function
- For charging an EV by x kW-hr in time u hr is,

$$P_{a,c,\tau}(x, u) = x(ae^{-u/\tau} + c)$$

Discount-based pricing function

$P(x, u)$ v/s u



Interpreting the parameters:

$$P(x, u) = x(ae^{-u/\tau} + c)$$

- c - Base price (\$/kW-hr),
- a - Surge price (\$/kW-hr)
- τ - Suggested service time (hr)

- Service time decision by user i to minimize total cost,

$$u_i = \tau \cdot \log\left(\frac{ax_i}{\tau\alpha_i}\right)$$

- \$ amount paid by user i with demand x_i and impatience α_i

$$P_{a,c,\tau}(x_i, u_i) = c\, x_i + \tau\, \alpha_i$$

Capacity planning under stochastic demand

- Let $f_{x,\alpha}$ be such that $\mathbb{E}u = b\tau$, $\mathbb{E}\left(\frac{x}{u}\right) = \frac{\mu}{\tau}$ and $\mathrm{Var}\left(\frac{x}{u}\right) = \frac{1}{2}\left(\frac{\nu}{\tau}\right)^2$
- Let the maximum rate of charging a *single* EV be $r_{\max}$ (in kW)

Theorem (Space and Power Capacity, for constant power charging)

For an arrival rate of λ, at any time, with 99% confidence, we have

1. *Demand for space will not exceed*

$$M = \lambda\, b\tau + \sqrt{5\lambda\, b\tau} + 3$$

2. *Power delivered will not exceed*

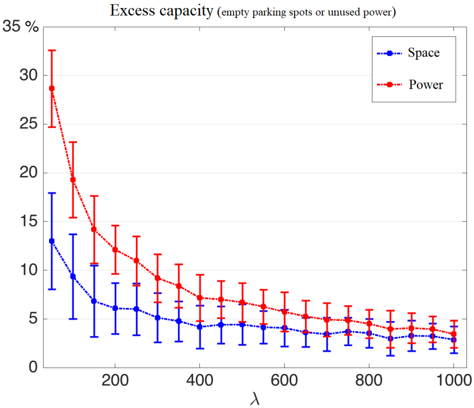
$$K = \lambda\, b\mu + (\mu + \nu)\sqrt{6\lambda\, b/\tau} + 6r_{\max}$$

- Similar to $Y \sim \mathcal{N}(\mu, \sigma^2)$, then with 99% confidence, $Y \le \mu + 3\sigma$.
- $K \propto \mathcal{O}(M + r_{\max})$ instead of $K \propto \mathcal{O}(M \times r_{\max})$
- Optimal rate with respect to λ. A lower bound of $K = \Omega(\lambda)$ exists.
- **Tradeoff:** Space capacity $M = \widetilde{\mathcal{O}}(\tau)$. Power capacity $K = \widetilde{\mathcal{O}}(1/\sqrt{\tau})$.

Numerical simulations - Excess capacity v/s λ

Excess capacity $\propto \frac{1}{\sqrt{\lambda}}$

Excess $:= \frac{\text{Theorem}-\text{Actual}}{\text{Actual}} \times 100$. For 100 simulations of 8 hr each.



Numerical simulations - Capacity v/s Confidence

Excess capacity $\propto \frac{\log(\frac{1}{\delta})}{\sqrt{\lambda}}$ for $1-\delta$ confidence

Excess$= \frac{\text{Theorem}-\text{Actual}}{\text{Actual}} \times 100$. For 100 simulations of 8 hr each.

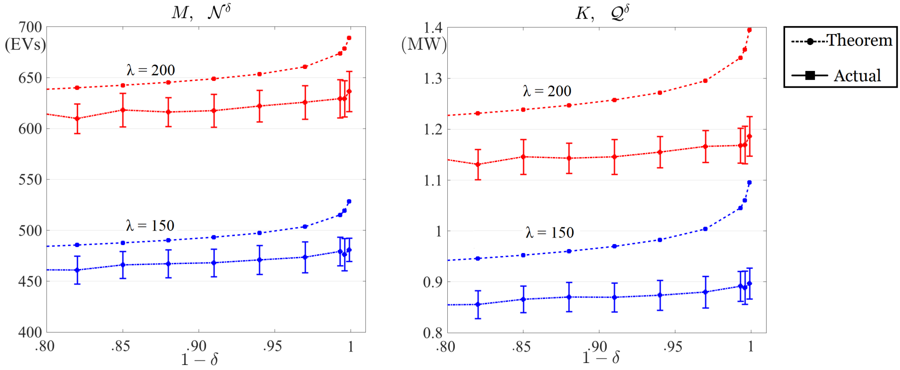


Figure: $\mathcal{N}^\delta$ is the $1-\delta$ percentile of occupied space, $\mathcal{Q}^\delta$ is the $1-\delta$ percentile of power delivery.

Summary

- Pricing can help incite some desired behaviour in *impatient* users
- Probabilistic constraints allow reducing installation capacity drastically from $K \propto \mathcal{O}(M \times r_{\max})$ to $K \propto \mathcal{O}(M + r_{\max})$
  - Example: $\approx$ 10 fold reduction for UCLA.
- We characterized probability of failure that helps plan for back-up capacity (battery banks/generators)
- Other aggregators with impatient users
  - *Cloud computing*: Client machines upload FLOPS to a server with a deadline for the computation. Users get discounts for waiting longer.
  - *Cab aggregators*: Users get a discount for waiting longer (Uber Pool or Lyft Line), allowing for more efficient resource allocation by aggregator.

Thank You!