A Stochastic Resource-Sharing Network for Electric Vehicle Charging

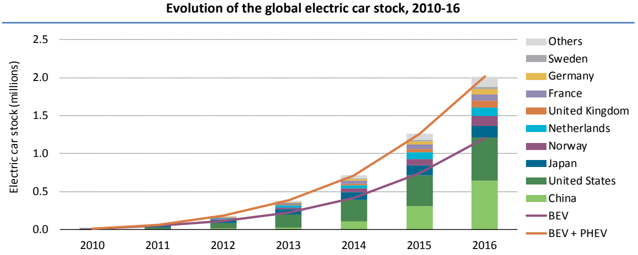
Bert Zwart

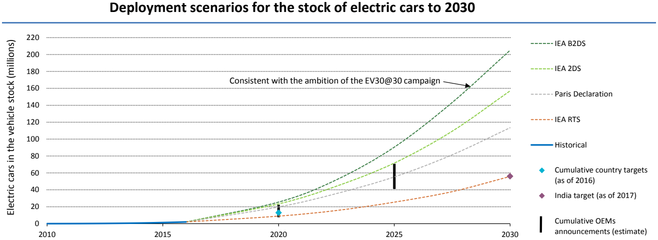
CWI Amsterdam and TU Eindhoven

Simons Institute
March 26, 2018

joint work with Angelos Aveklouris and Maria Vlasiou (TU Eindhoven)

The rise of Electric Vehicles





Types of charging

Bottlenecks in EV charging

Current:

Ability to charge a battery fast

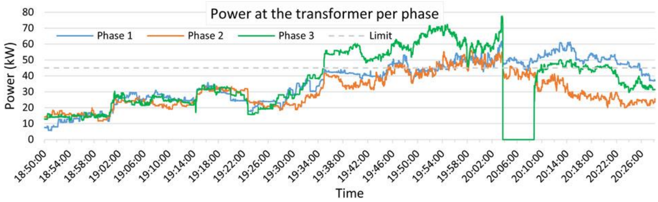
Number of charging stations

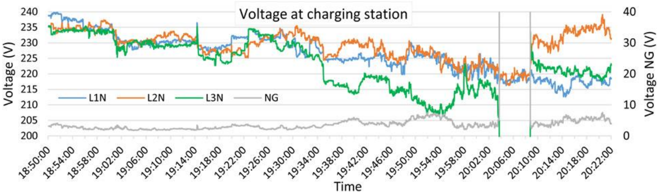
Future:

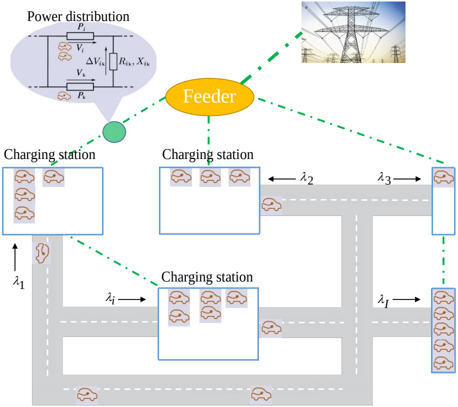
Capacity of the grid

A 2025 scenario: charging electric vehicles, baking pizzas, and melting a fuse in Lochem

From G. Hoogsteen, J. Hurink, G. Smit, et al. Proc. CIRED (2017):

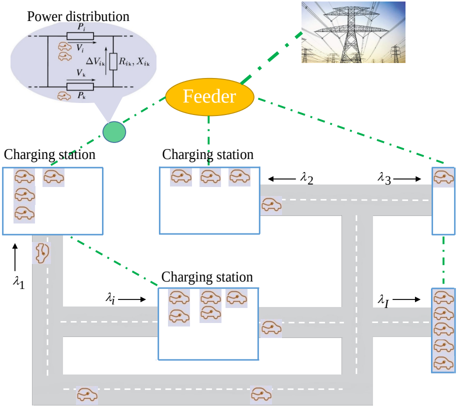






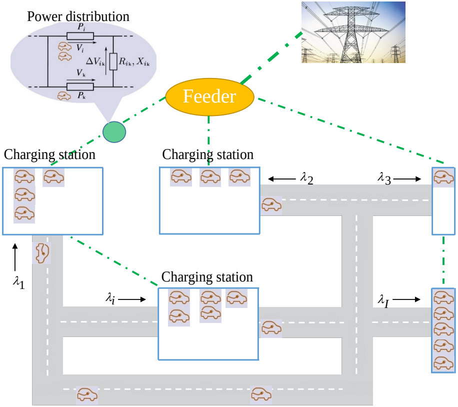
- I charging stations

EV charging: stochastic process on interacting networks



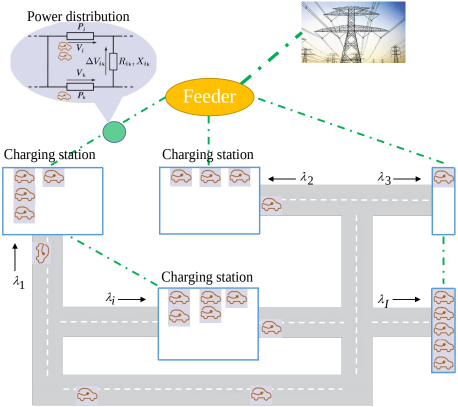
- I charging stations
- K_i parking spaces

EV charging: stochastic process on interacting networks



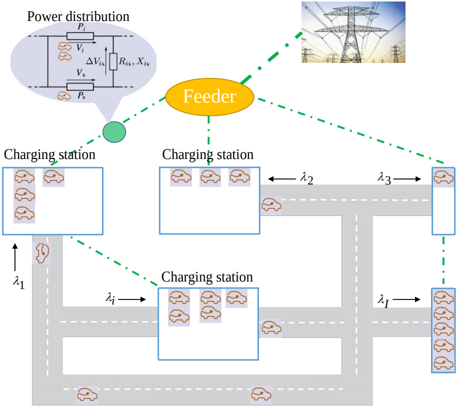
- I charging stations
- K_i parking spaces
- Arrivals: λ_i

EV charging: stochastic process on interacting networks



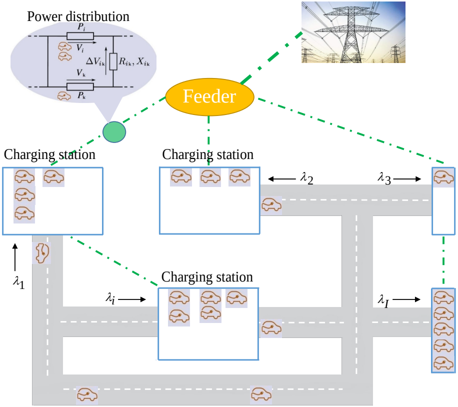
- I charging stations
- K_i parking spaces
- Arrivals: λ_i
- Charging requirement: B

EV charging: stochastic process on interacting networks



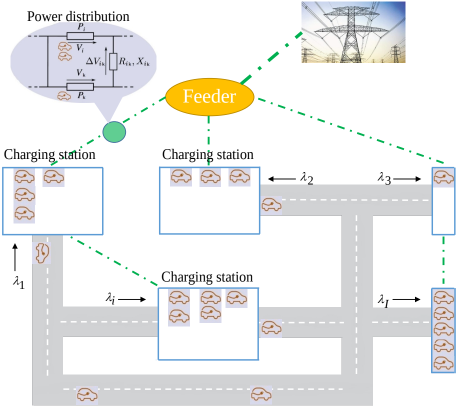
- I charging stations
- K_i parking spaces
- Arrivals: λ_i
- Charging requirement: B
- Parking time: D

EV charging: stochastic process on interacting networks



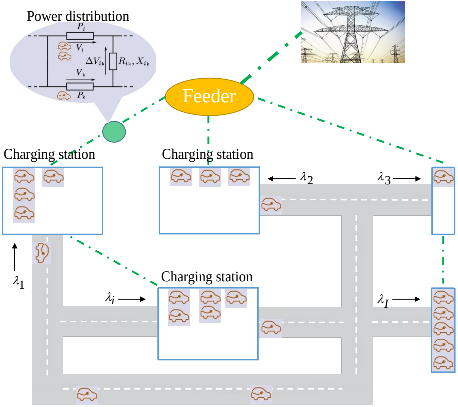
- I charging stations
- K_i parking spaces
- Arrivals: λ_i
- Charging requirement: B
- Parking time: D
- $Q_i(t)$: Number of EVs

EV charging: stochastic process on interacting networks



- I charging stations
- K_i parking spaces
- Arrivals: λ_i
- Charging requirement: B
- Parking time: D
- $Q_i(t)$: Number of EVs
- $Z_i(t)$: Number of uncharged EVs

EV charging: stochastic process on interacting networks



- I charging stations
- K_i parking spaces
- Arrivals: λ_i
- Charging requirement: B
- Parking time: D
- $Q_i(t)$: Number of EVs
- $Z_i(t)$: Number of uncharged EVs

Aim: efficient charging schedule while keeping voltage drop bounded

Related work

R. C. Larson and K. Sasanuma. Congestion pricing: A parking queue model. Journal of Industrial and Systems Engineering, 4(1):1–17, 2010.

E. Yudovina and G. Michailidis. Socially optimal charging strategies for electric vehicles. IEEE TAC, 60(3):837–842, 2015.

R. Carvalho, L. Buzna, R. Gibbens, F. Kelly. Critical behaviour in charging of electric vehicles. New J. Phys., 17(9): 095001, 2015.

J. Cruise, S. Shneer (2018). Almost finished work on stability properties.

Assessing voltage drop: a tractable load flow model

- **Linearized Distflow:** $W_{kk}^{lin} := |V_k^{lin}|^2$

$$W_{kk}^{lin} = W_{00} - 2 \sum_{\epsilon_{ls} \in \mathcal{P}(k)} R_{ls} \sum_{m \in \mathcal{I}(s)} z_m p_m,$$

  - $\boldsymbol{z} = (z_i, i \geq 1)$ denotes the number of uncharged EVs in the network at some particular time
  - **Each EV at node** i receives power p_i
  - $\sum_{m \in \mathcal{I}(s)} z_m p_m$ is the consumed power by subtree rooted in node s

Assessing voltage drop: a tractable load flow model

- **Linearized Distflow:** $W_{kk}^{lin} := |V_k^{lin}|^2$

$$W_{kk}^{lin} = W_{00} - 2 \sum_{\epsilon_{ls} \in \mathcal{P}(k)} R_{ls} \sum_{m \in \mathcal{I}(s)} z_m p_m,$$

  – $\boldsymbol{z} = (z_i, i \geq 1)$ denotes the number of uncharged EVs in the network at some particular time
  – **Each EV at node** i receives power p_i
  – $\sum_{m \in \mathcal{I}(s)} z_m p_m$ is the consumed power by subtree rooted in node s

- Some of our results also hold for more general AC (on trees)

Assessing voltage drop: a tractable load flow model

- **Linearized Distflow:** $W_{kk}^{lin} := |V_k^{lin}|^2$

$$W_{kk}^{lin} = W_{00} - 2 \sum_{\epsilon_{ls} \in \mathcal{P}(k)} R_{ls} \sum_{m \in \mathcal{I}(s)} z_m p_m,$$

  – $\boldsymbol{z} = (z_i, i \geq 1)$ denotes the number of uncharged EVs in the network at some particular time
  – **Each EV at node** i receives power p_i
  – $\sum_{m \in \mathcal{I}(s)} z_m p_m$ is the consumed power by subtree rooted in node s

- Some of our results also hold for more general AC (on trees)

- Next: how to schedule amount of power for each battery

Power allocation: use network utility maximization

- $\boldsymbol{z} = (z_i, i \geq 1)$: number of uncharged EVs in the network
- $\boldsymbol{p} = (p_i, i \geq 1)$: allocated power to vehicles at node i
- Each EV receives utility $u_i(p_i)$. Example: $u_i(p_i) = w_i \log(p_i)$

Power allocation: use network utility maximization

- $\boldsymbol{z} = (z_i, i \geq 1)$: number of uncharged EVs in the network
- $\boldsymbol{p} = (p_i, i \geq 1)$: allocated power to vehicles at node i
- Each EV receives utility $u_i(p_i)$. Example: $u_i(p_i) = w_i \log(p_i)$

$$\begin{aligned} \boldsymbol{p} = \arg\max \quad & \sum_{i=1}^{I} z_i u_i(p_i) \\ \text{subject to} \quad & z_i p_i \leq M_i, \quad 0 \leq p_i \leq c^{\max}, \\ & \underline{\upsilon}_i \leq W_{ii}^{lin}(\boldsymbol{p}, \boldsymbol{z}) \leq \overline{\upsilon}_i \end{aligned}$$

Power allocation: use network utility maximization

- $\boldsymbol{z} = (z_i, i \geq 1)$: number of uncharged EVs in the network
- $\boldsymbol{p} = (p_i, i \geq 1)$: allocated power to vehicles at node i
- Each EV receives utility $u_i(p_i)$. Example: $u_i(p_i) = w_i \log(p_i)$

$$\begin{aligned} \boldsymbol{p} = \arg\max \quad & \sum_{i=1}^{I} z_i u_i(p_i) \\ \text{subject to} \quad & z_i p_i \leq M_i, \quad 0 \leq p_i \leq c^{\max}, \\ & \underline{v}_i \leq W_{ii}^{lin}(\boldsymbol{p}, \boldsymbol{z}) \leq \overline{v}_i \end{aligned}$$

- Key challenge: implementation by market mechanism. [Kelly (1997): communication networks.]
- This talk: assess performance

Solvable special case: product-form property

Proportional fairness in line network

Consider line network with $K_i = M_i = \infty$ for all i, $c^{\max} = \infty$.
Set $\bar{R}_i = \sum_{j=1}^{i} R_j$.
If $u_i(p_i) = \log p_i$, then for every $\boldsymbol{n} \in \mathbb{N}_+^I$,

$$\lim_{t\to\infty} \mathbb{P}(\boldsymbol{Z}(t) = \boldsymbol{n}) = (1-\rho)(\sum_{i=1}^{I} n_i)! \prod_{i=1}^{I} \frac{\rho_i^{n_i}}{n_i!},$$

provided $\rho = \sum_{i=1}^{I} \rho_i = \sum_{i=1}^{I} \lambda_i \mathbb{E}[B]\overline{R}_i/\delta < 1$.

Solvable special case: product-form property

Proportional fairness in line network

Consider line network with $K_i = M_i = \infty$ for all i, $c^{\max} = \infty$.
Set $\bar{R}_i = \sum_{j=1}^{i} R_j$.
If $u_i(p_i) = \log p_i$, then for every $\boldsymbol{n} \in \mathbb{N}_+^I$,

$$\lim_{t\to\infty} \mathbb{P}(\boldsymbol{Z}(t) = \boldsymbol{n}) = (1-\rho)(\sum_{i=1}^{I} n_i)! \prod_{i=1}^{I} \frac{\rho_i^{n_i}}{n_i!},$$

provided $\rho = \sum_{i=1}^{I} \rho_i = \sum_{i=1}^{I} \lambda_i \mathbb{E}[B]\overline{R}_i/\delta < 1$.

- Proof idea: reduction to multi-class Processor queue

Solvable special case: product-form property

Proportional fairness in line network

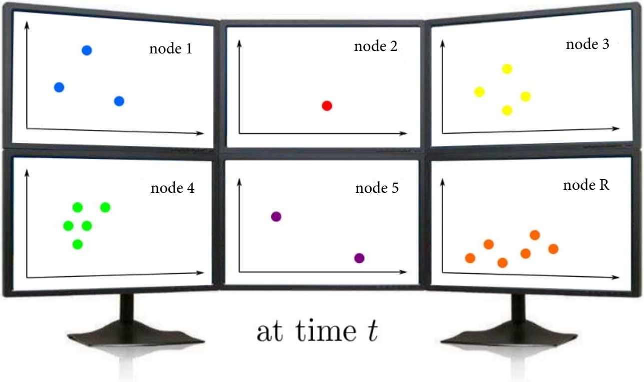
Consider line network with $K_i = M_i = \infty$ for all i, $c^{\max} = \infty$.
Set $\bar{R}_i = \sum_{j=1}^{i} R_j$.
If $u_i(p_i) = \log p_i$, then for every $\boldsymbol{n} \in \mathbb{N}_+^I$,

$$\lim_{t\to\infty} \mathbb{P}(\boldsymbol{Z}(t) = \boldsymbol{n}) = (1-\rho)(\sum_{i=1}^{I} n_i)! \prod_{i=1}^{I} \frac{\rho_i^{n_i}}{n_i!},$$

provided $\rho = \sum_{i=1}^{I} \rho_i = \sum_{i=1}^{I} \lambda_i \mathbb{E}[B]\overline{R}_i/\delta < 1$.

- Proof idea: reduction to multi-class Processor queue
- New application of Processor Sharing after mainframe computer systems (70s) and communication networks (90s).

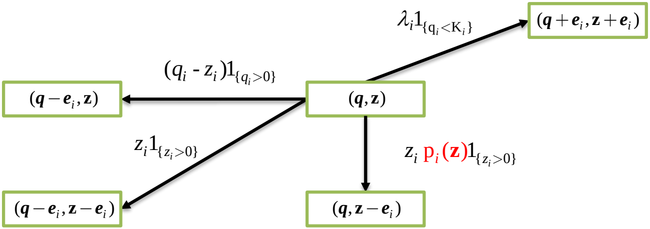
General case: Measure-valued process



Particles move to the left at rate $\boldsymbol{p}_r(z)$ and to the south at rate z_r

Computational analysis of stochastic model: hard

- Markov Chain (MC) model (assuming exponential rv's)

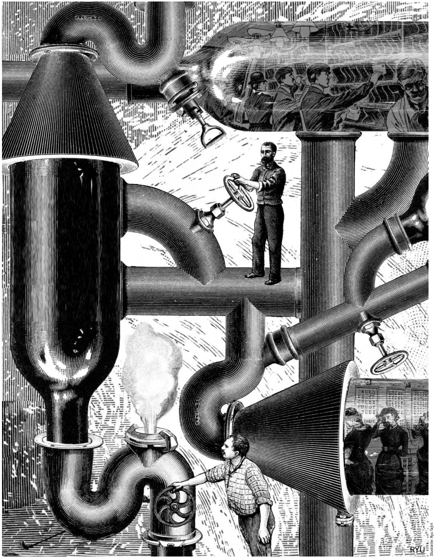


$\boldsymbol{p}(\boldsymbol{z}) = \arg\max_{\boldsymbol{p}} \sum_{i=1}^{I} z_i u_i(p_i)$

- For a line network with identical parking lots of size K and R nodes, computing the equilibrium distribution of the MC requires solving $O(K^2R)$ convex optimization problems.

- Things are even worse for non-exponential distributions, so we need another approach.

Fluid approximation



Fluid approximation

Scaling:

- Consider a family of models indexed by n
- Capacity of power in the n^{th} system: nM_i
- Arrival rate in the n^{th} system: $n\lambda_i$
- Number of parking spaces in the n^{th} system: nK_i

- Fluid approximation at time $t \geq 0$: $\boldsymbol{z}(t) = (z_i(t), i \geq 1)$
- $z_i(t) \to z_i^*$ as $t \to \infty$ and z_i^* is an invariant point

Main result for fluid approximation

Characterization of performance

$\boldsymbol{z}^*$ is given by $z_i^* = \frac{\Lambda_i^*}{g_i^{-1}(\Lambda_i^*)}$, with

$$\Lambda^* = \arg\max \sum_{i=1}^{I} G_i(\Lambda_i)$$

$$\text{subject to} \quad \Lambda_i \leq M_i, \quad 0 \leq \Lambda_i \leq g_i(c^{\max}),$$

$$\underline{\upsilon}_i \leq W_{ii}^{lin}(\boldsymbol{\Lambda}) \leq \overline{\upsilon}_i,$$

$G_i'(\cdot) = u_i'(g_i^{-1}(\cdot))$ with
$g_i(x) := \gamma_i \mathbb{E}[\min\{Dx, B\}]$ and $\gamma_i = \min\{\lambda_i, K_i \mathbb{E}[D]\}$.

Main result for fluid approximation

Characterization of performance

$\boldsymbol{z}^*$ is given by $z_i^* = \frac{\Lambda_i^*}{g_i^{-1}(\Lambda_i^*)}$, with

$$\Lambda^* = \arg\max \sum_{i=1}^{I} G_i(\Lambda_i)$$

$$\text{subject to} \quad \Lambda_i \leq M_i, \quad 0 \leq \Lambda_i \leq g_i(c^{\max}),$$

$$\underline{\upsilon}_i \leq W_{ii}^{lin}(\boldsymbol{\Lambda}) \leq \overline{\upsilon}_i,$$

$G_i'(\cdot) = u_i'(g_i^{-1}(\cdot))$ with
$g_i(x) := \gamma_i \mathbb{E}[\min\{Dx, B\}]$ and $\gamma_i = \min\{\lambda_i, K_i \mathbb{E}[D]\}$.

Fraction of cars that get succesfully charged: $P(D > Bz_i^*/\Lambda_i^*)$.

Idea of the proof: Little is known

- Little's law: (Expected number of customers)= (arrival rate) $\times$ (sojourn time)
- Snapshot principle: constant service rate in equilibrium
- $z_i^* = \gamma_i \mathbb{E}[\min\{D, \frac{B}{p_i(\mathbf{z}^*)}\}]$
- Add this approximate version of Little's law to Karush-Kuhn-Tucker (KKT) equations for optimization problem that defines $p(z)$.
- Still works for AC in this case, as convex relaxation of associated semi-definite program is exact.

Validating Distflow (two lots, Markovian model)

Table: Stochastic model

K	$\mathbb{E}[Z_1], \mathbb{E}[Z_2]$ (Distflow)	$\mathbb{E}[Z_1], \mathbb{E}[Z_2]$ (AC)	Rel. error
10	$4.5336, 4.6179$	4.6801, 4.6924	3.13 %, 1.58 %
20	$14.0174, 14.0385$	14.1725, 14.1948	1.09 %, 1.10 %

Table: Fluid approximation

K	z_1^*, z_2^* (Distflow)	z_1^*, z_2^* (AC)	Rel. error
10	$4.5769, 4.5769$	$4.7356, 4.7513$	3.35%, 3.67%
20	$14.0300, 14.0300$	$14.1849, 14.2069$	1.09%, 1.25%
30	$23.6820, 23.6820$	$23.8357, 23.8597$	0.64%, 0.74%
40	$33.4293, 33.4293$	$33.5823, 33.6073$	0.45%, 0.53%
50	$43.2330, 43.2330$	$43.3857, 43.4112$	0.35%, 0.41%

Validating the fluid approximation

Relative error		
K	Distflow	AC
10	$0.95\%, 0.86\%$	$1.18\%, 1.25\%$
20	$0.09\%, 0.06\%$	$0.08\%, 0.08\%$

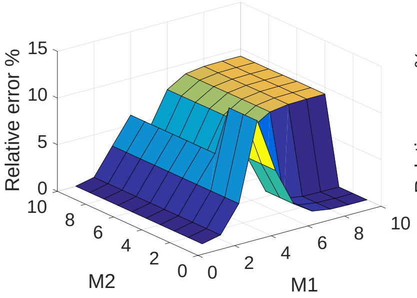


Figure: $\boldsymbol{K} = (10, 10)$ and $\boldsymbol{\lambda} = (12, 12)$

Weight-setting: fairness

Consider a line network, recall $\bar{R}_i = \sum_{j=1}^{i} R_i$.

Fairness property

Assume that $M_i = c^{\max} = \infty$, and $u_i(p_i) = w_i \log(p_i)$. If $w_i = \overline{R}_i$, then we have that $p_i(\boldsymbol{z}) = p(\boldsymbol{z}) > 0$. Moreover, $p(\boldsymbol{z}) = \delta\Big(\sum_{i=1}^{I} \overline{R}_i z_i\Big)^{-1}$.

Not very efficient

Weight-setting: maximizing throughput

- **Goal**: Choose weights that maximize the fraction of EVs that get successfully charged under weighted proportional fairness, i.e., $u_i(p_i) = w_i \log(p_i)$

Weight-setting: maximizing throughput

- **Goal**: Choose weights that maximize the fraction of EVs that get successfully charged under weighted proportional fairness, i.e., $u_i(p_i) = w_i \log(p_i)$
- For a line network topology, $M_i = c^{\max} = \infty$, it can be shown that this fraction can be optimized by choosing the weights as follows:

$$\max_{\boldsymbol{w}} \quad \sum_{i=1}^{I} \gamma_i \mathbb{P}(w_i D > \overline{R}_i B)$$

$$\text{subject to} \quad \sum_{i=1}^{I} \gamma_i \mathbb{E}[\min\{w_i D, \overline{R}_i B\}] \leq \delta$$

Non-convex, depends on the joint distribution of (B, D)

$B = HD$, H deterministic

Weight-finding problem can be transformed into

$$\max_{\boldsymbol{x}} \quad \sum_{i=1}^{I} \gamma_i x_i$$

$$\text{subject to} \quad \sum_{i=1}^{I} \mathbb{E}[D]\gamma_i \overline{R}_i H x_i \leq \delta, \quad x_i \in \{0, 1\}$$

Knapsack problem

Yields distributionally robust solution (i.e. homogeneous user preferences are the worst if the system is overloaded)

$B = HD$, H decreasing hazard rate.

Example: $H \sim \mathrm{Pareto}(\alpha, \kappa)$, i.e., $\mathbb{P}(H > x) = (\frac{\kappa}{x+\kappa})^\alpha$, $x \geq 0$, $\alpha > 1$, $\kappa > 0$.

Weight-finding problem can be transformed into a convex programming problem:

$$\max_{\boldsymbol{y}} \quad \sum_{i=1}^{I} \gamma_i (1 - y_i^{\alpha/(\alpha-1)})$$

$$\text{subject to} \quad \sum_{i=1}^{I} \frac{\mathbb{E}[D]\kappa\gamma_i \overline{R}_i}{1-\alpha}(y_i - 1) \leq \delta, \quad 0 \leq y_i \leq 1$$

Assume 10 nodes with $\sum_{i=1}^{10} \frac{\mathbb{E}[B]\gamma_i \overline{R}_i}{\delta} = 1.2$

The optimal weights are given by:

Table

Det.	0.06	0.09	0.11	0.12	0.13	0.14	0.15	0.16	0	0	8
Pareto (3)	0.12	0.11	0.10	0.10	0.09	0.09	0.09	0.08	0.08	0.08	9.5
Pareto (1.1)	0.11	0.10	0.10	0.09	0.09	0.09	0.09	0.09	0.09	0.09	10

The more variability, the better the system performance

Dynamic fluid equations ($K_r = \infty$)

$$z_r(t) = \int_0^t \lambda_i \mathbb{P}(D_r > t - s; B_r > \int_s^t \boldsymbol{p}_r(\boldsymbol{z}(u))du)ds \qquad r = 1, ..., R, t \geq 0$$

Recall

$$\begin{aligned} \boldsymbol{p}(\boldsymbol{z}) = \arg\max \quad & \sum_{i=1}^{I} z_i u_i(p_i) \\ \text{subject to} \quad & z_i p_i \leq M_i, \quad 0 \leq p_i \leq c^{\max}, \\ & \underline{v}_i \leq W_{ii}^{lin}(\boldsymbol{p}, \boldsymbol{z}) \leq \overline{v}_i \end{aligned}$$

Unique solution thanks to the properties: $\boldsymbol{p}(\boldsymbol{z})$ is Lipschitz continuous, $\boldsymbol{p}(\boldsymbol{z})$ is nonincreasing in $\boldsymbol{z}$.

Convergence to equilibrium

Proposition: $\boldsymbol{z}(t) \to \boldsymbol{z}^*$.

Proof: Define

$$l_r = \liminf_{t\to\infty} z_r(t) \qquad u_r = \limsup_{t\to\infty} z_r(t).$$

Using the dynamic fluid equations, it can be shown that

$$l_r = \lambda_r \mathbb{E}[\min\{D, \frac{B}{M_r(\boldsymbol{l},\boldsymbol{h})}\}] \quad u_r = \lambda_r \mathbb{E}[\min\{D, \frac{B}{m_r(\boldsymbol{l},\boldsymbol{h})}\}].$$

with

$$M_r(\boldsymbol{l},\boldsymbol{h}) = \sup_{\boldsymbol{z}\in[\boldsymbol{l},\boldsymbol{h}]} p_r(\boldsymbol{z}) \quad m_r(\boldsymbol{l},\boldsymbol{h}) = \inf_{\boldsymbol{z}\in[\boldsymbol{l},\boldsymbol{h}]} p_r(\boldsymbol{z})$$

using monotonicity of $\boldsymbol{p}(\boldsymbol{z})$ and the characterization of $\boldsymbol{z}^*$ using Little's law, uniqueness follows

Concluding comments

- A. Aveklouris, M. Vlasiou, and B. Zwart, A Stochastic Resource-Sharing Network for Electric Vehicle Charging, https://arxiv.org/pdf/1711.05561. (Submitted to the special energy issue of IEEE Transactions on Control of Network Systems)

- Multiclass extension: see preprint

- Currently working out a rigorous justification of the fluid scaling, allowing time-varying arrival rates

- Challenges: time-dependent behavior, controlling arrival rates of cars, incorporating markets explicity, reducing communication overhead by discretizing time, adding reactive power support, allowing for additional (noisy) behavior of other types of users, ...