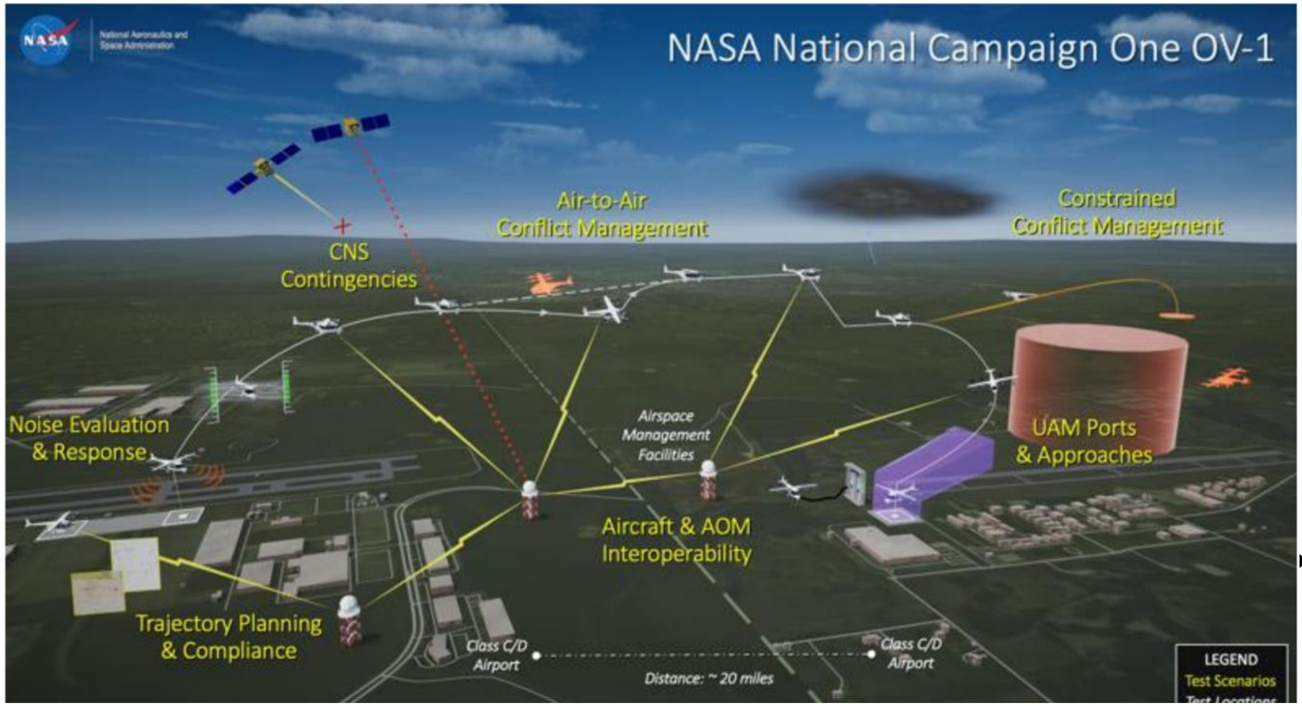
Airspace Management System Framework for Planning and Logistics

Srinivasa Salapaka
Department of Mechanical Science and Engineering
University of Illinois Urbana-Champaign

August 11, 2023

Credits: NASA / Lillian Gipson

Airspace Scenarios



- ⋆ **Physical Space - Airspace**
  - **sequence of intersections:**
  - multi-input multi-output, flexible locations and geometry; multi-lane and bidirectional
  - **air-corridors:**
  - bidirectional and multi-lane; flexible locations and geometry; different UAV speeds; hovering or holding patterns are possible

- ⋆ **Traffic Management System (cloud based)**
  - possibly **distributed** (sectorwise) or **centralized**
  - decide **network aspects** – allow how many vehicles, which new airways to open, etc.
  - assist **individual UAS** – relay perceived information, waypoint paths, etc.

- ⋆ **Ground Resources**
  - mobile or static landing stations
  - service stations – recharging etc.

- ⋆ **Vehicles (Unmanned Air Systems UAS)**
  - Heterogenous, multi-speed
  - Provide its own parameters (e.g. start and destination points, fuel, make, travel parameters (desired ETA etc) to aggregator
  - report path conditions to the aggregator
  - provide control parameters to track path/time dictated by aggregator

High-Level Objectives

- **abstract out planning/logistics problems**
  - ★ architectural details are evolving: focus **on broader, more adaptable** solutions
- **develop cloud/web-based analytical and design tools** for
  - ★ Traffic Management Systems (TMS)
  - ★ Individual Unmanned Aerial Systems (UASs)

Objectives

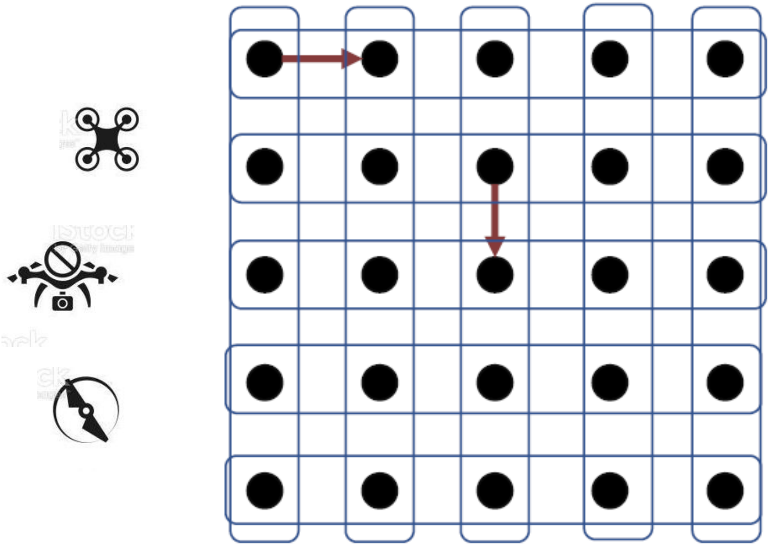
- **planning and logistic problems**
  - ⋆ resource allocation, scheduling, routing etc. with heterogeneous vehicles
    - e.g., **routing and scheduling of UAS** (determine sequence of waypoints and schedule them for each UAS)
    - e.g., **resource allocation and scheduling:** coverage by service stations, recharging schedules; service-vehicles schedules, etc.
    - e.g., **traffic-density design** to avoid congestion, determining relative ratios of different types, speeds etc
    - e.g., **efficiency and robustness design** of the UAS's speed profile for energy efficiency, time, and robustness to uncertainties, minimizing (ideally eliminating) holding patterns; trade-off optimality vs robustness
    - e.g., **respect constraints:** communication constraints, capacity constraints, dynamic constraints, topographical constraints

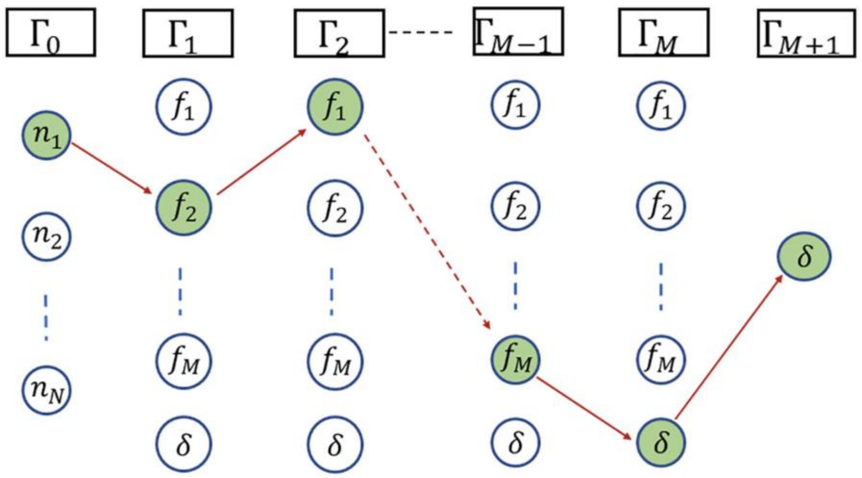
- **sensitivity analysis:** with respect to network parameters
    - e.g., with traffic volume, traffic density, UAS-type configuration, # of facilities, facility-types, communication error and delays etc.
  - ⋆ guide **prioritization of resource allocation**
  - ⋆ **quantifying risk assessment and resilience** - network's vulnerability to external disturbances or changes in operating conditions

- **offline ML training**:
  - ⋆ learning from offline digital twin airspace simulation systems – identify meta-parameters etc.
  - ⋆ useful for **contingency management**, **resilience and reliability studies, advisory/suggesting for network and UAS parameters, real-time optimization and scalable studies.**

Example: Routing+Scheduling of Vehicles

- which vehicle should cross which intersection when – minimize commute time





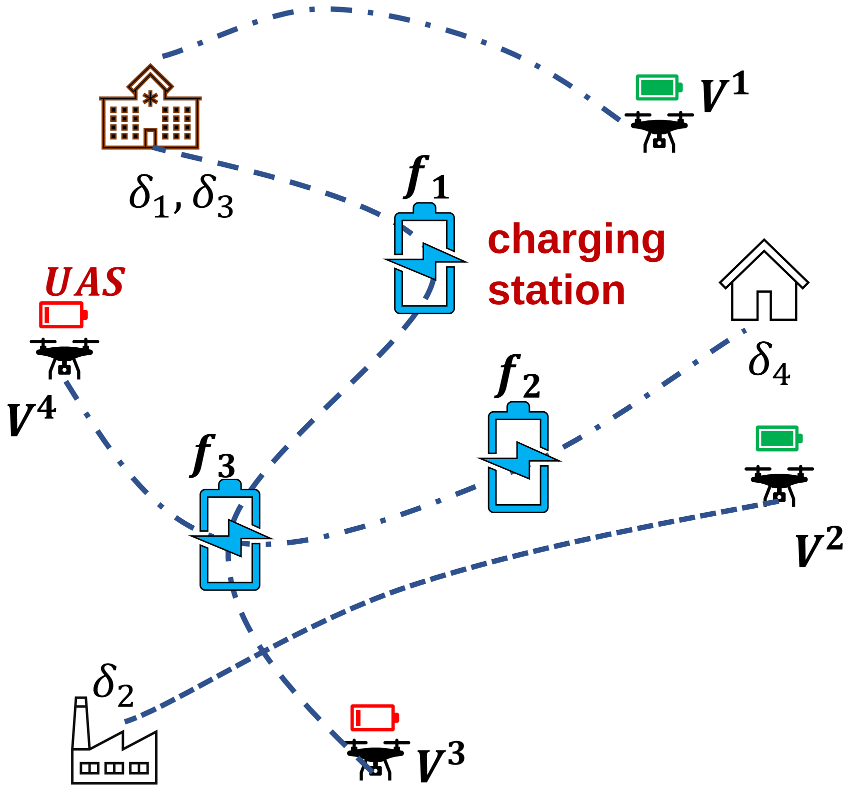
- determine optimal routes – sequence of intersections
- determine time spent between two successive intersections
- Respect constraints – congestion (node capacity), speed, and obstacle avoidance constraints.

- decision variables

$\boldsymbol{\eta^n_{j|i}(k)} \in \{\mathbf{0}, \mathbf{1}\}$: is $= 1$ if n^{th} vehicle decides to go from intersection i at the k^{th} stage to intersection j at the $(k+1)^{th}$ stage

$\boldsymbol{t^n_j(k)}$: time required to reach j^{th} intersection at the k^{th} stage

Other Planning/Logistic Examples:



- UAS, charging UGVs, and destination targets.
- each drone: pre-defined charging capacity and an initial charge
- UAS can choose to go to the destination directly or via a subset of charging facilities to ensure battery capacity

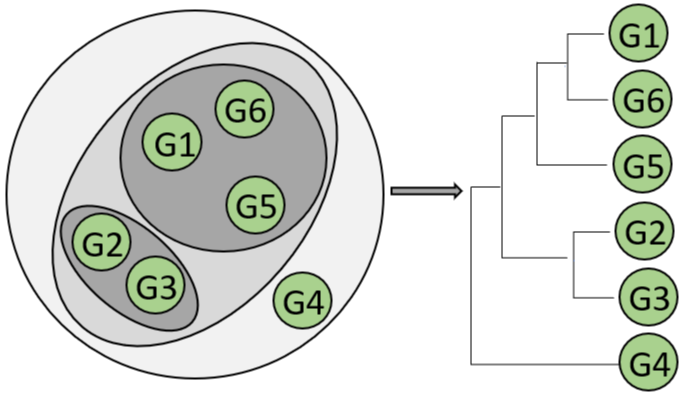
- **Routing+Scheduling:**
  - objective: determine routes and inter-node times to minimize the total weighted travel time over all UAS
  - constraints: No UAS runs out of charge, congestion constraint
- **Scheduling on predefined routes:**
  - objective: each drone has a mission (predefined sequence of nodes to go to). schedule drones to minimize the total weighted travel time
  - constraints: processing time constraint, precedence order constraint, and congestion constraint
- **Routing+Resource Allocation:**
  - Objective: determine locations of charging stations to minimize total weighted time (distance) over all UAS
  - Constraints: No UAS runs out of charge, obstacles are avoided

⋮

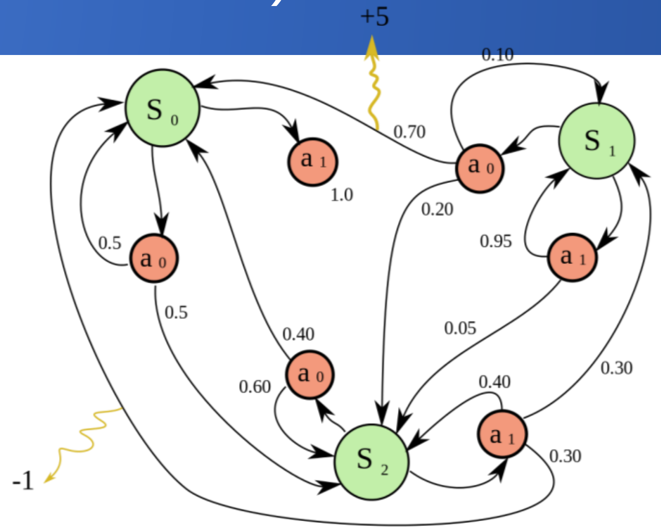
- **Various other combinations**

Abstraction: Parameterized Sequential Decision Making (Para-SDM) Problems

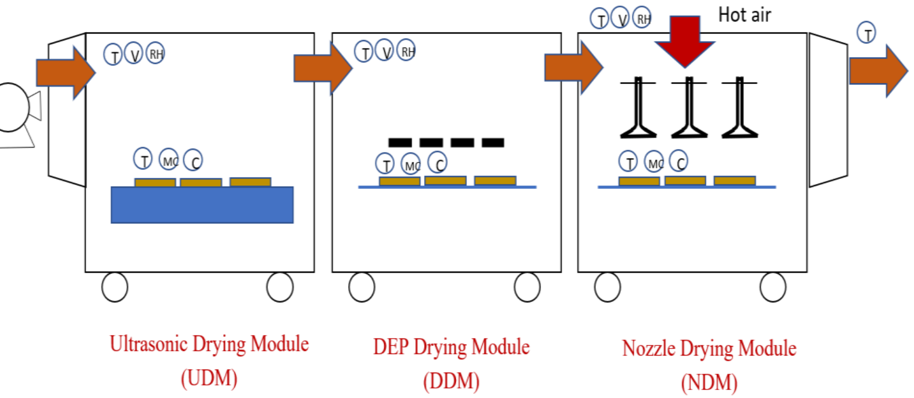
- Common Aspect:
  - ★ **Simultaneous Sequential Decision Making and Parameter Optimization**



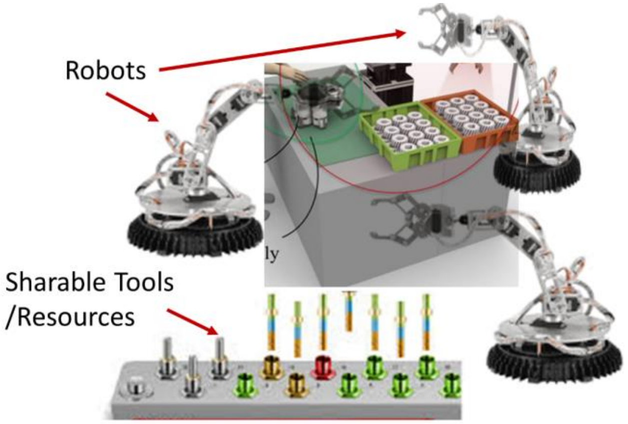
Hierarchical Clustering



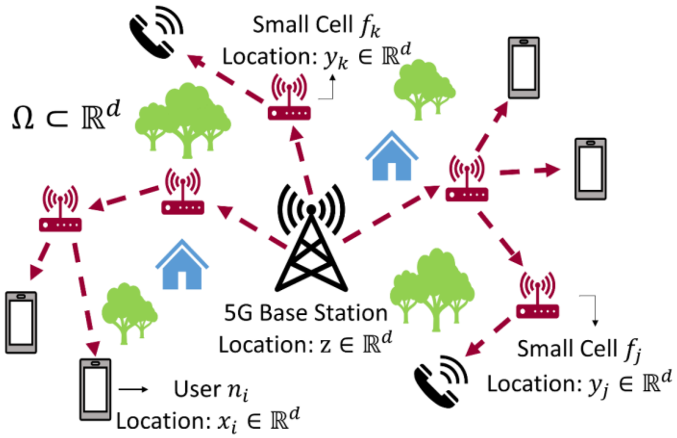
Markov Decision Processes



Industrial Process Optimization



Job-Shop Scheduling



5G Small Cell Network Design

A General Framework

Statistical Physics

Maximum Entropy Principle

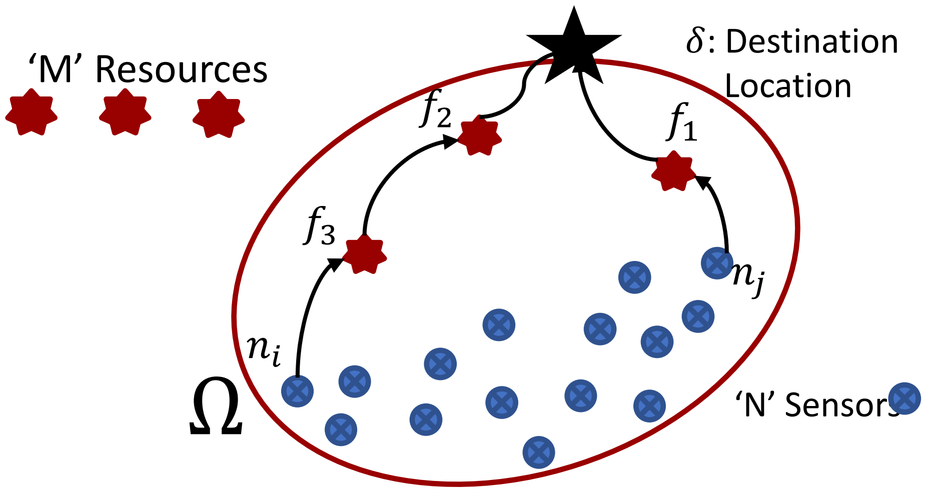
Combinatorial Optimization Problems

Clustering, Data Classification, Network Aggregation, Routing, Scheduling, Traveling Salesman Problems, Resource Allocation, …

PARA-SDM
Network Design and Planning problems, Supply Chain problems, MDPs, RL, …

Machine Learning MDPs, RL

Baseline Para-SDM: Facility Location with Path Optimization



- **"find simultaneously optimal paths and facility locations"**

$$\min_{\substack{\mathcal{Y},\eta(\gamma|i)\in\{0,1\}: \\ \sum_\gamma \eta(\gamma|i)=1}} D = \sum_{i=1}^{N} \rho_i \sum_{\gamma\in G} \boldsymbol{\eta(\gamma|i)}\, d(n_i,\gamma)$$

- application example areas: supply-chain networks, sensor networks, last-mile delivery

- decision variables
  - ★ **Path γ**: sequence of facilities from nodes to destination $n_i \to \boldsymbol{f_{r_1}} \to \boldsymbol{f_{r_2}} \to \cdots \boldsymbol{f_{r_q}} \to \boldsymbol{\delta}$
  - ★ **Parameters $\mathcal{Y}$:** facility locations $\mathcal{Y} := \{y_j\}$

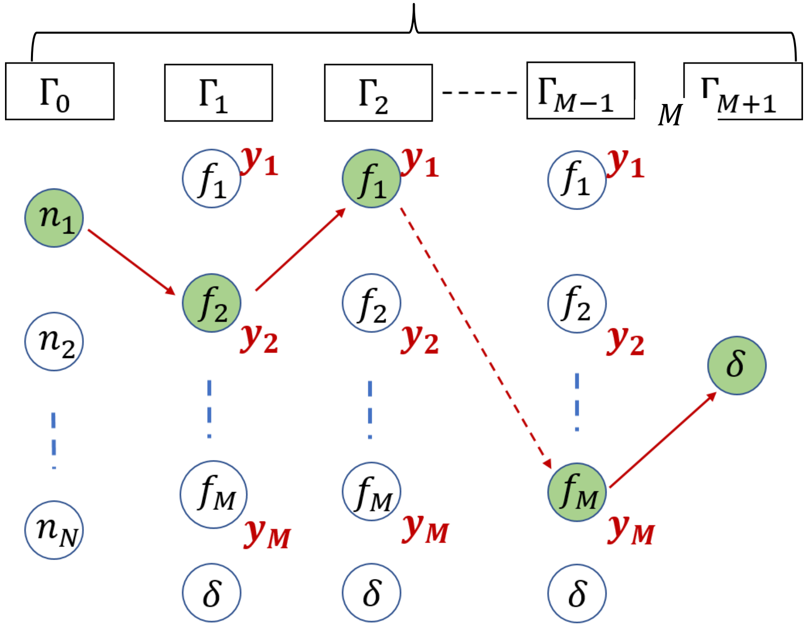
★ **combinatorial configurations**
- exponential # of paths γ;
- $\{\eta(\gamma|i)\}$ is combinatorial
- continuum of facility locations $\boldsymbol{y}$

★ **computationally complex**
- non-convex, NP-hard
- sequential algorithms : sub-optimal

★ **combinatorial viewpoint** for para-SDMs

Facility Location with Path Optimization: Baseline Para-SDM

Stages Illustration



$$J^{\mu}(s) = \sum_{t=0}^{\infty} d_t(x_t, x_{t+1})$$

subject to

$$x_{t+1} = \mu(a_t|x_t), \quad x_0 = s$$

+

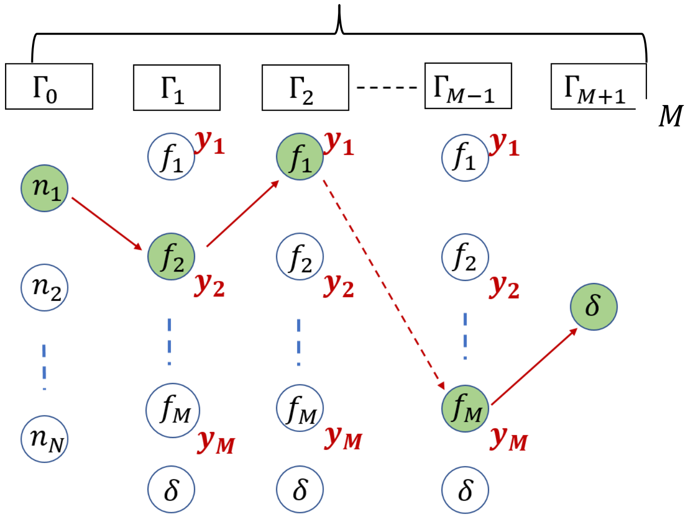
other constraints on states and actions

★ **MDPs: Shortest Path Problems**

★ **Para-SDMs: Shortest Path + Facility Location Problems**

Solve Shortest Path + Facility Location Problem: Baseline Para-SDM

Stages Illustration



$$\min_{\substack{\mathcal{Y},\eta(\gamma|i)\in\{0,1\}:\\ \sum_\gamma q(\gamma|i)=1}} D = \sum_{i=1}^{N} \rho_i \sum_{\gamma\in G} \eta(\gamma|i) d(x_i,\gamma)$$

- $\eta(\gamma|i) = \prod_{k=1}^{M-1} \eta_k(\gamma_{k+1}|\gamma_k)$
- $\eta_k(\gamma_{k+1}|\gamma_k) = 1$ if γ_k to γ_{k+1} $|\{\eta_k\}| = \mathcal{O}(M^3)$
- Polynomial-order decision variables

$$\max_{\mathcal{Y},\{p_k\}} \quad H := -\sum_{i=1}^{N} \rho_i \sum_{\gamma\in G} \overbrace{\left(\prod_k p_k(\gamma_{k+1}|\gamma_k)\right)}^{p(\gamma|i)} \log\left(\prod_k p_k(\gamma_{k+1}|\gamma_k)\right)$$

subject to

$$D := \sum_{i=1}^{N} \rho_i \sum_{\gamma\in G} \prod_k p_k(\gamma_{k+1}|\gamma_k)\, d(x_i,\gamma) = d_0$$

Free Energy $F = D - \frac{1}{\beta} H$

★ **Path Probabilities $p(\gamma|i)$**

– $p_k(\gamma_{k+1}|\gamma_k) = e^{-\beta d_k} \dfrac{\sum_{\gamma_{k+2}^M} e^{-\beta \sum_{t=k+1}^{M} d_t(\gamma_t,\gamma_{t+1})}}{\sum_{\gamma_{k+1}^M} e^{-\beta \sum_{t=k}^{M} d_t(\gamma_t,\gamma_{t+1})}}$

★ **Free Energy**

– $F = -\frac{1}{\beta} \sum_{\gamma_0} p_{\gamma_0} \log \sum_{\gamma_1,\ldots,\gamma_M} e^{-\beta \sum_{t=0}^{M} d_t(\gamma_t,\gamma_{t+1})}$

★ **Facility Locations:**

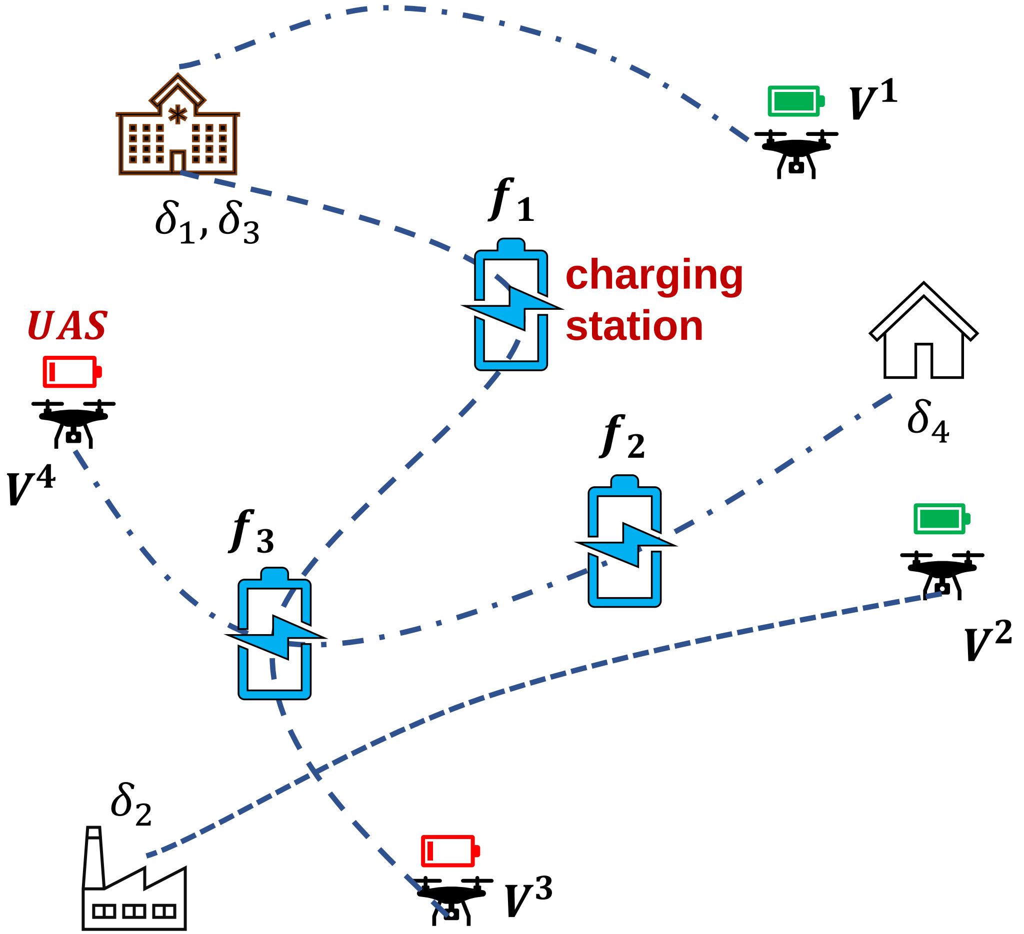
– $\mathcal{Y}$ obtained from $\frac{\partial F}{\partial y} = 0$

Annealing

- Minimize for every β starting from 0 to ∞ ($\Leftrightarrow$ decreasing d_0)
- $\beta = 0 \Rightarrow$ convex free energy $\Rightarrow$ (Global minima = uniform distribution)
- $\beta \to \infty \Rightarrow [p(\gamma|\gamma_0) \to \eta(\gamma|\gamma_0)] \Rightarrow$ (hard associations = solution)

Routing + Scheduling Problem

- **Scenario:**
  - ⋆ charging stations
    - – processing time for UAS
  - ⋆ n^{th}UAS $\boldsymbol{V^n} = (\boldsymbol{\ell_0^n}, \boldsymbol{\ell_d^n}, \boldsymbol{c_0^n}, \boldsymbol{T^n(0)}, \boldsymbol{S^n}, \boldsymbol{F.C.R^n})$
    - – $\boldsymbol{\ell_0^n}, \boldsymbol{\ell_d^n}$: entry and exit location
    - – $\boldsymbol{c_0^n}$: initial charge
    - – $\boldsymbol{T^n(0)}$: entry time
    - – $\boldsymbol{F.C.R^n}$: full-charge range



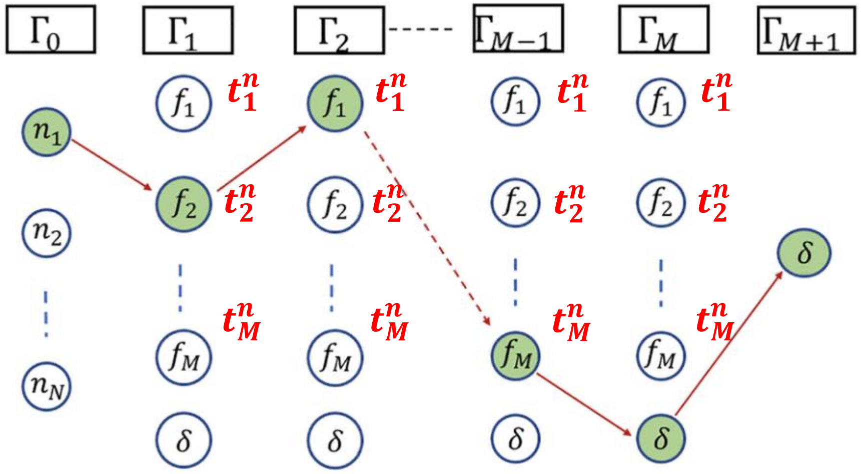
- **Objective: find simultaneously**

1. **Shortest time paths (routes) for all UAS**
2. **time each UAS spends in each corridor (in between successive intersections or facilities)**

- **Constraints:**
  - ⋆ **UAS are never without charge**
  - ⋆ **avoid congestion at charging stations**
  - ⋆ **UAS within maximum speed**

Problem Statement: Routing + Scheduling in FLPO form

Stage Illustration for drone n



Decision Variables

- t_j^n – Time taken to reach node j, $1 \le j \le M$
- $\eta_{j|i}^n(k) \in \{0,1\}$ – i to j transition in stage k

Total cost for a drone : $\boldsymbol{D^n = T^n(K) + C^n(K)}$

- Time taken : $T^n(K) = \Pi_{k=0}^{K} \eta_{j|i}^n(k) \sum_{k=0}^{K} (t_{j(k+1)}^n - t_{i(k)}^n)$
- Penalty incurred : $C^n(K) = \Pi_{k=0}^{K} \eta_{j|i}^n(k) \sum_{k=0}^{K} c_k^n(i,j)$
- $c_k^n(i,j)$ – penalty due to constraints

Solution Approach Using the MEP Framework

$$D = \min_{t_j^n, \eta_{j|i}^n(k)} \sum_n p_n \Pi_{k=0}^{K} \eta_{j|i}^n(k) \sum_{k=0}^{K} (t_{j(k+1)}^n - t_{i(k)}^n) + c_k^n$$

- Replace binary associations with a probability associations

$$\{0,1\} \ni \eta_{j|i}^n \rightarrow p_{j|i}^n(k) \in [0,1], \forall k$$

- Minimize free energy iteratively for $\beta > 0,\ \beta = \epsilon \rightarrow \infty,\ 0 < \epsilon \ll 1$

$$F = \min_{t_j^n, p_{j|i}^n(k)} D - {}^{1}/_{\beta} H,$$

$$\text{where } H = \Pi_{k=0}^{K} p_{j|i}^n(k) \sum_{k=0}^{K} \log p_{j|i}^n(k)$$

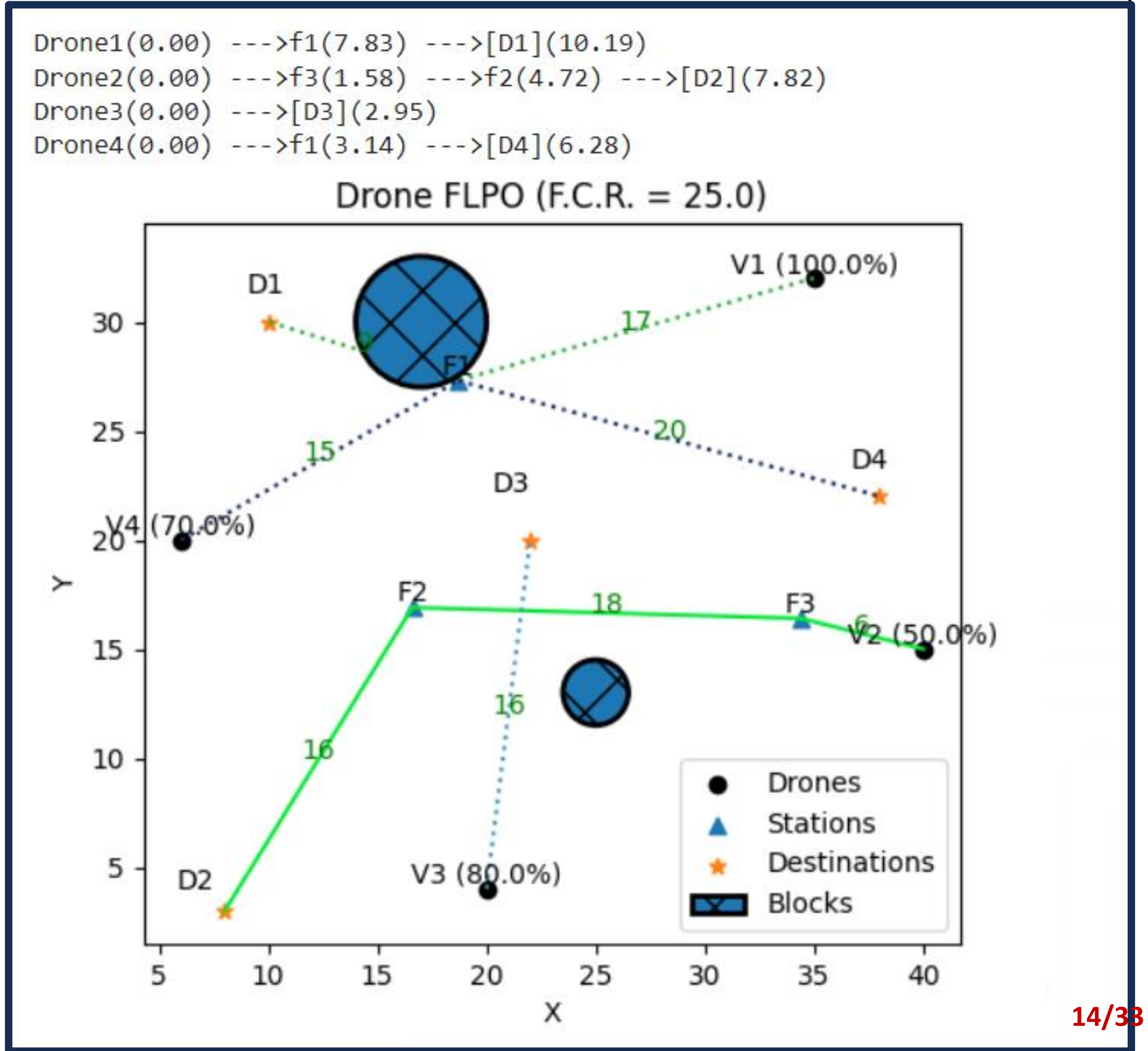
Annealing

- Minimize for every β starting from 0 to ∞
- $\beta = 0 \Rightarrow$ convex free energy $\Rightarrow$ (Global minima = uniform distribution)
- $\beta \rightarrow \infty \Rightarrow [p(\gamma|\gamma_0) \rightarrow \eta(\gamma|\gamma_0)] \Rightarrow$ (hard associations = solution)

Simulations: scheduling + routing + congestion avoidance

PTF : processing time frame (charging time at each station) PTF = 2.0

PLAY THE VIDEO!

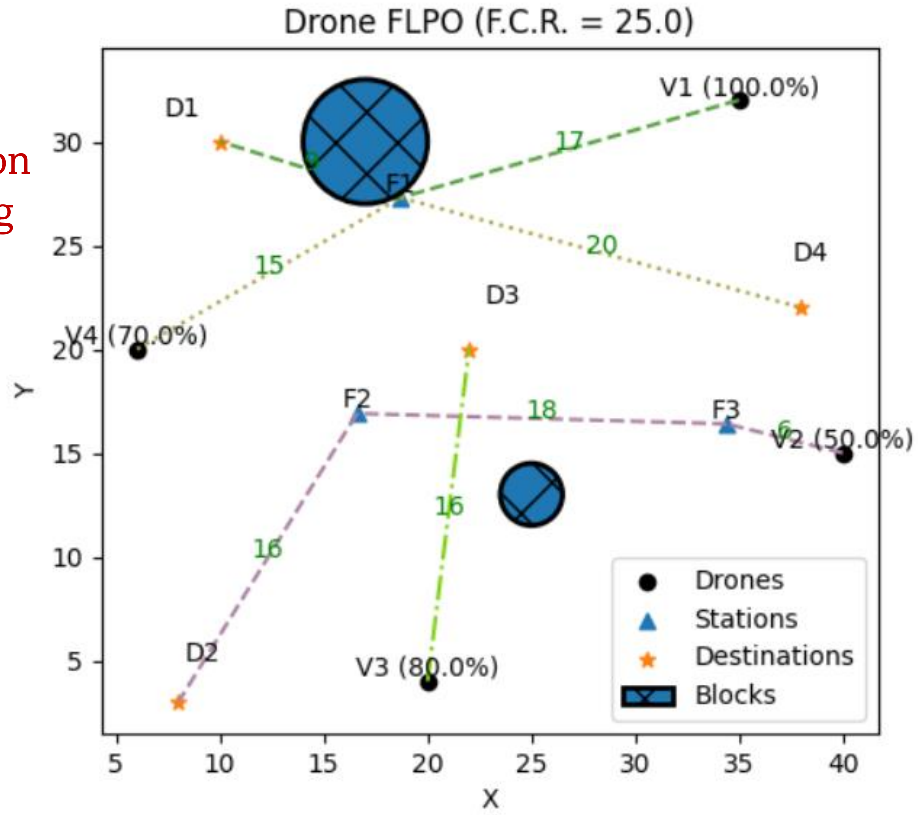


Simulations: scheduling + routing + congestion avoidance

Congestion happening at f1

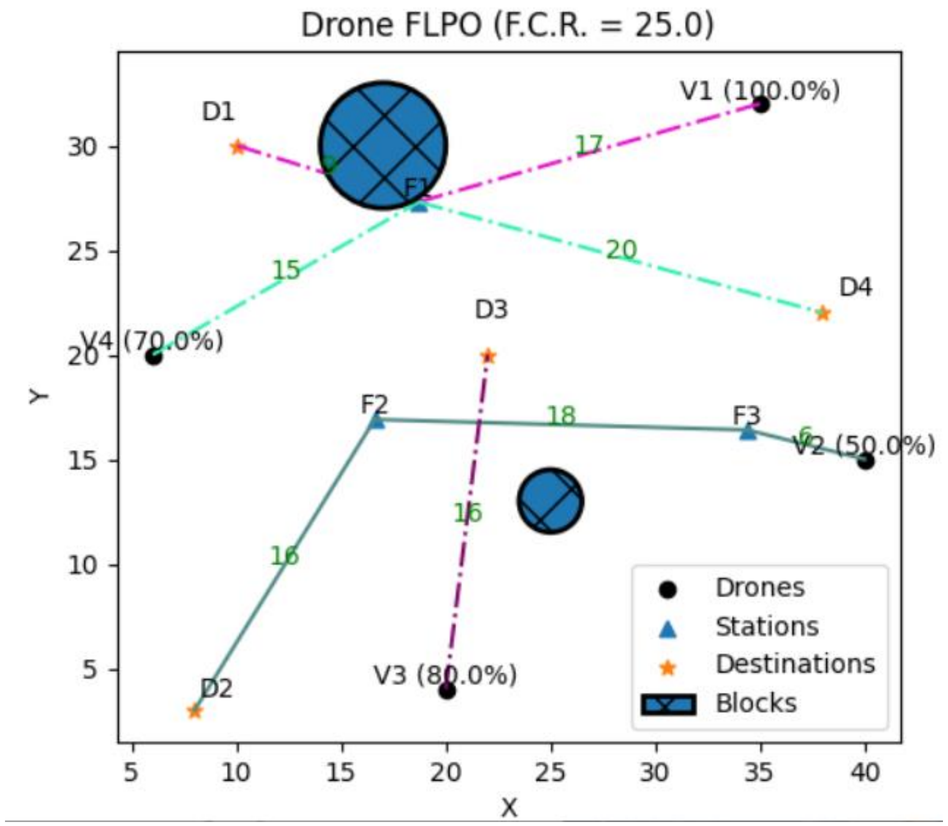
PTF = 0

```
Drone1(0.00) --->f1(3.16) --->[D1](5.40)
Drone2(0.00) --->f3(1.76) --->f2(5.02) --->[D2](8.10)
Drone3(0.00) --->[D3](3.05)
Drone4(0.00) --->f1(3.35) --->[D4](6.70)
```



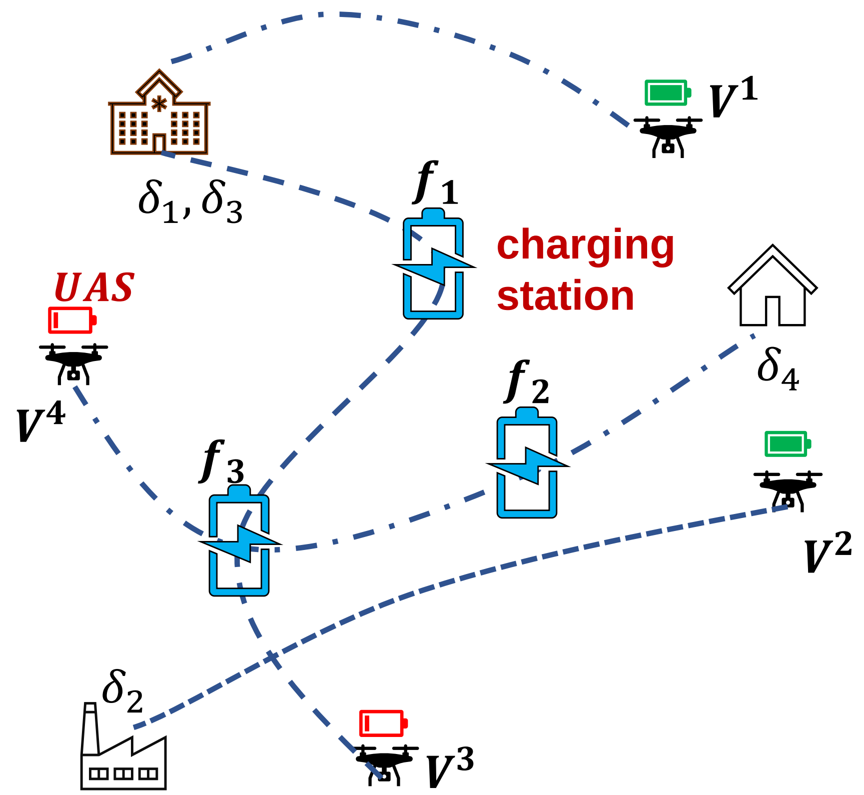
PTF = 4.0

```
Drone1(0.00) --->f1(8.77) --->[D1](11.18)
Drone2(0.00) --->f3(6.67) --->f2(12.37) --->[D2](20.99)
Drone3(0.00) --->[D3](4.95)
Drone4(0.00) --->f1(3.01) --->[D4](9.96)
```



Routing + Facility (Charging Stations) Locations

- **Scenario:**
  - ⋆ charging stations (fixed locations)
    - processing time for UAS
  - ⋆ n^{th}UAS $\boldsymbol{V^n} = (\boldsymbol{\ell_0^n}, \boldsymbol{\ell_d^n}, \boldsymbol{c_0^n}, \boldsymbol{T^n}(\boldsymbol{0}), \boldsymbol{S^n}, \boldsymbol{PTF^n}, \boldsymbol{F.C.R^n})$
    - $\boldsymbol{\ell_0^n}, \boldsymbol{\ell_d^n}$: entry and exit location
    - $\boldsymbol{c_0^n}$: initial charge
    - $\boldsymbol{T^n}(\boldsymbol{0})$: entry time
    - $\boldsymbol{S^n}$: maximum speed
    - $\boldsymbol{F.C.R^n}$: full-charge range



- **Objective: find simultaneously locations of charging stations and shortest paths (routes) of UAS**
- **Constraints: UAS are never without charge, avoid obstacles, threshold on number of UAS at a charging station at a given time.**

Problem Statement: Routing + Facility (Charging Stations) Locations

Decision Variables:

$\eta_j^n(k) \in \{0,1\}^{L\times N\times K}$:
1 if the n^{th} vehicle goes to the node j at the k^{th} step, otherwise 0.
$y_j \in \mathbb{R}^{L\times r}$:
The r dimensional coordinates of the nodes.

Parameters:

$p_n \in [0 \quad 1]$: relative importance of the n^{th} vehicle.
$\bar{S}^n$: maximum speed of the n^{th} vehicle.
$l^n(k)$: location (node id) of the n^{th} vehicle at step k.
$D^n(k)$: Total distance traversed by the n^{th} vehicle at step k.
c_0^n : initial battery charge of the n^{th} vehicle.
$c^n(k)$: battery charge of the n^{th} vehicle at step k.
$R^n(c)$: range of the n^{th} vehicle with a battery charge c.
$d(\cdot,\cdot)$: distance function between two spatial nodes.
$\Gamma(\cdot,\cdot)$: obstacle-aware penalty function between two nodes.

$$V^n : (l_0^n, l_d^n, c_0^n)$$

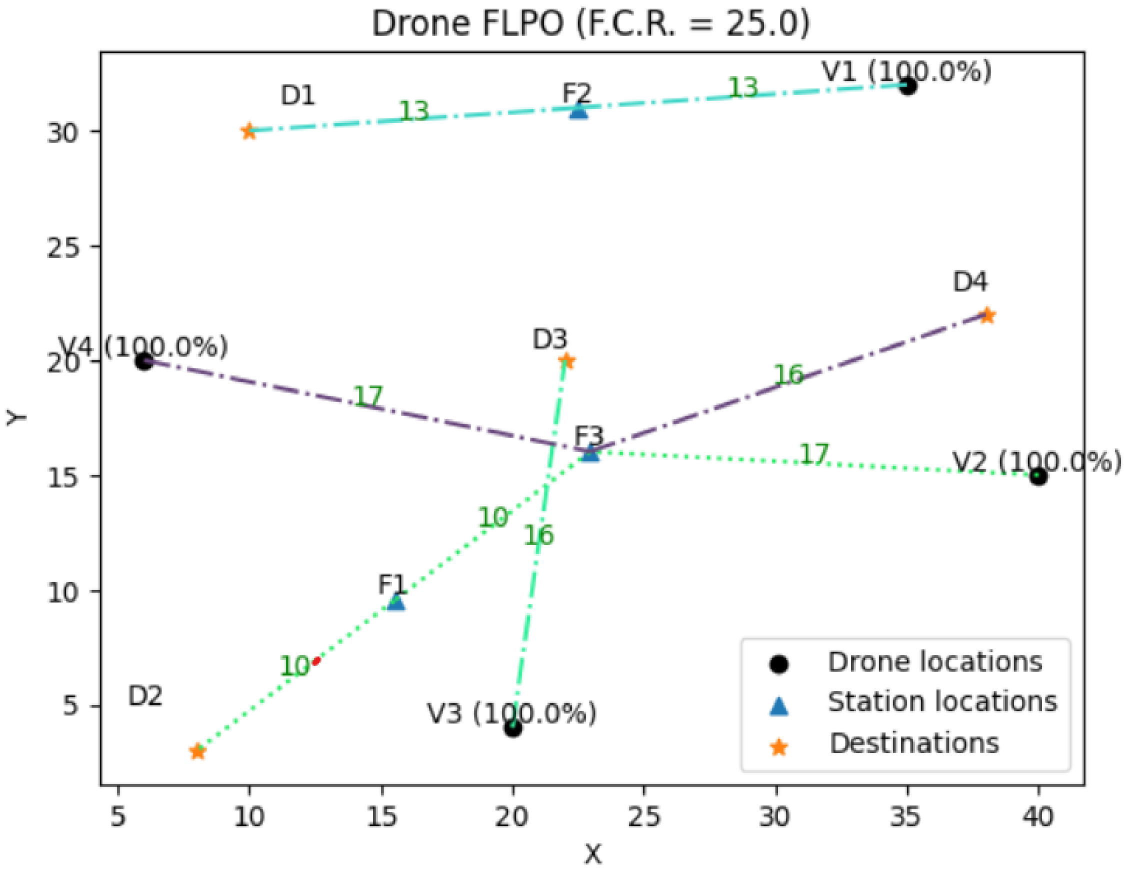
$$\min_{\eta,y} \sum_n p_n D^n(K)$$

$$\begin{aligned}
\text{s.t}\quad & l^n(0) = l_0^n \quad \forall n \\
& l^n(k) = l_d^n \quad \forall n, k \geq K \\
& D^n(0) = 0 \quad \forall n \\
& l^n(k+1) = \sum_{j=1}^{L} \eta_j^n(k) l_j \quad \forall n,k \quad l_j \in \{1,\dots,L\} \\
& D^n(k+1) = D^n(k) + d^*(l^n(k+1), l^n(k)) \quad \forall n,k \\
& R^n(c^n(k)) \geq d^*(l^n(k+1), l^n(k)) \quad \forall n,k \\
& \text{where:} \quad d^*(\cdot,\cdot) = d(\cdot,\cdot) + \Gamma(\cdot,\cdot) \\
& d(i,j) = \sqrt{(y_i - y_j)^\intercal (y_i - y_j)}
\end{aligned}$$

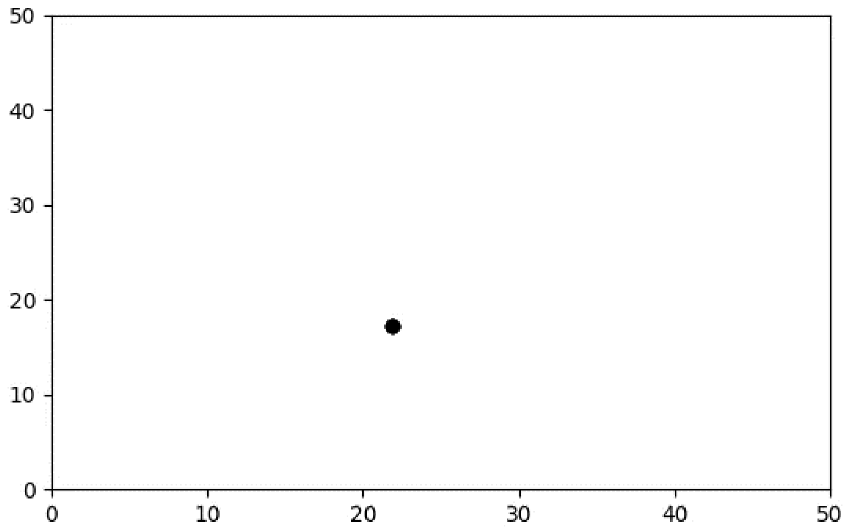
Battery constraint – obstacle constraint

Simulations – Routing + Facility Locations

```
Drone1 --->f2 --->[D1][D1][D1]
Drone2 --->f3 --->f1 --->[D2][D2]
Drone3 --->[D3][D3][D3][D3]
Drone4 --->f3 --->[D4][D4][D4]
```

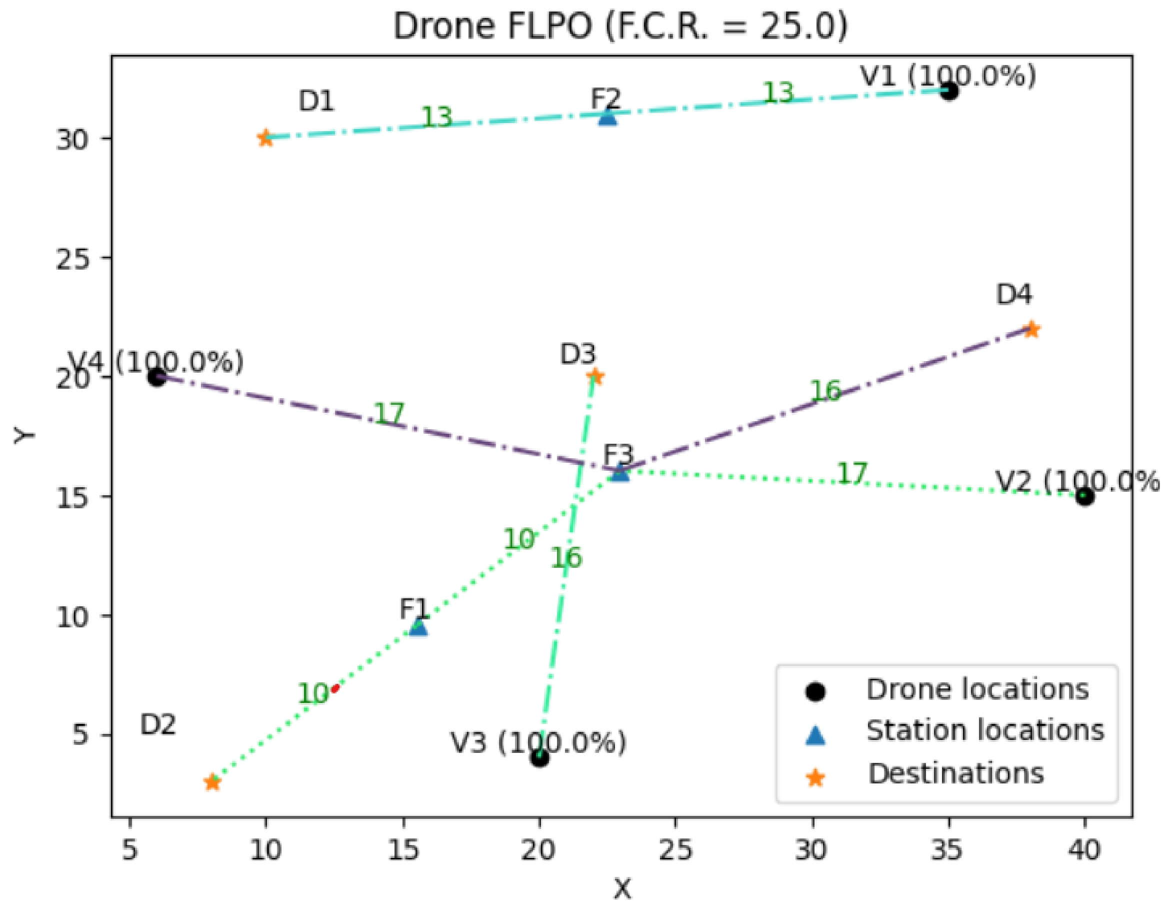


Phase transition

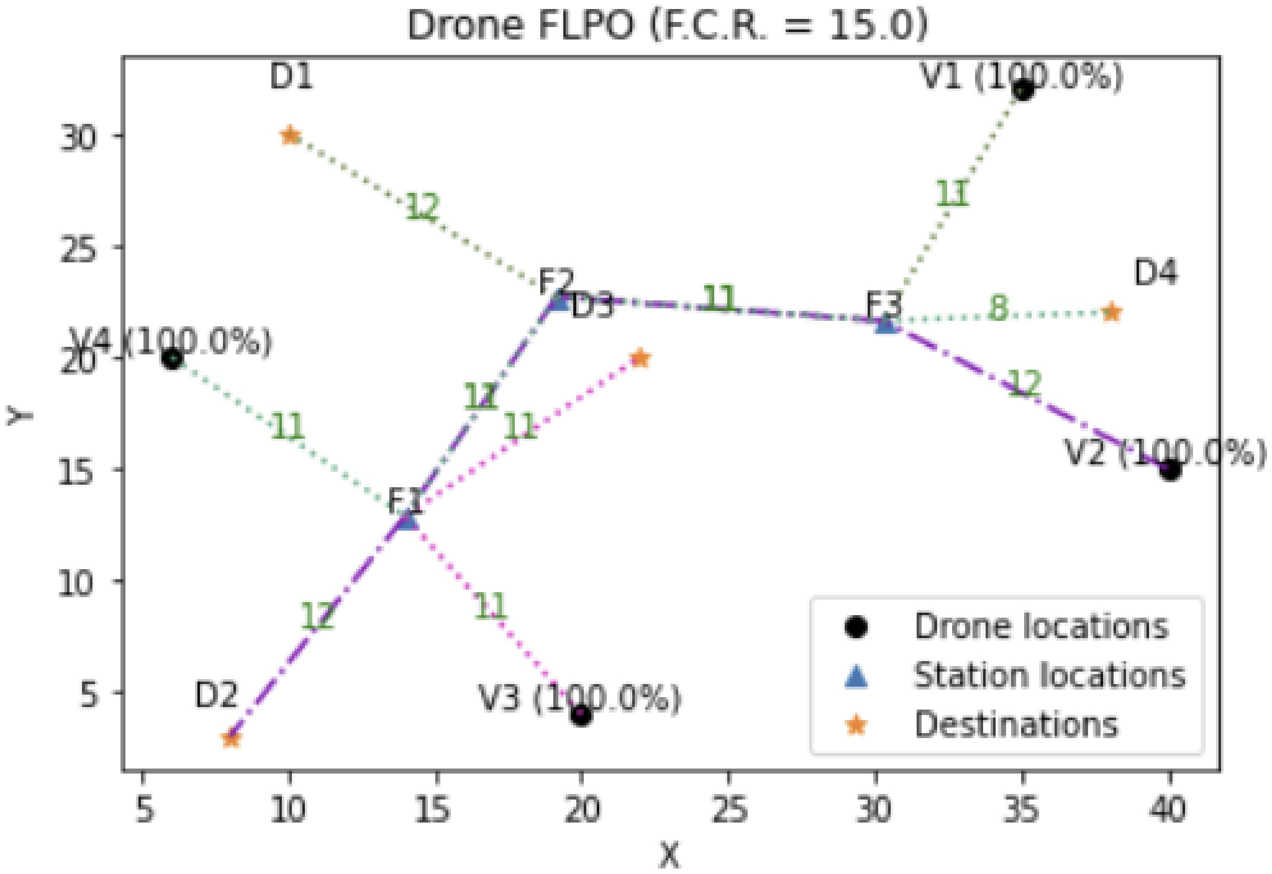


Simulations: (changing F.C.R.)

```
Drone1 --->f2 --->[D1][D1][D1]
Drone2 --->f3 --->f1 --->[D2][D2]
Drone3 --->[D3][D3][D3][D3]
Drone4 --->f3 --->[D4][D4][D4]
```

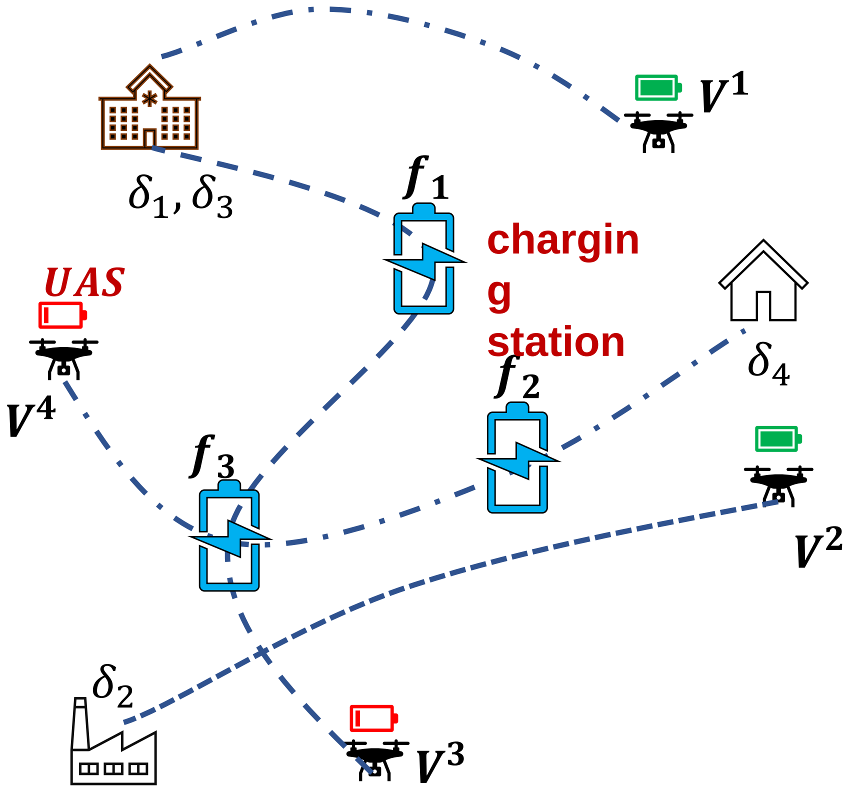



```
Drone1 --->f3 --->f2 --->[D1][D1]
Drone2 --->f3 --->f2 --->f1 --->[D2]
Drone3 --->f1 --->[D3][D3][D3]
Drone4 --->f1 --->f2 --->f3 --->[D4]
```



Scheduling on pre-defined routes

- **Scenario:**
  - ⋆ charging stations (fixed locations)
  - ⋆ n^{th}UAS $\boldsymbol{V^n} = (\boldsymbol{T^n}(\boldsymbol{0}), \boldsymbol{S^n}, \boldsymbol{L^n})$
    - $\boldsymbol{L^n} = \{\ell_0^n \prec \ell_1^n \prec \cdots \prec \ell_K^n = \ell_d^n\}$



- **Objective: schedule drones to minimize the total weighted travel time, to reach their destination**

- **Constraints:**
  - ⋆ **precedence order**
  - ⋆ **congestion constraint**
  - ⋆ **processing time constraint**
  - ⋆ **drone speed limits**

Problem Statement: Scheduling on pre-defined routes

Decision Variables:

$\delta\theta^n(k) \in \mathbb{R}^{N\times K}$:

The time interval for the transition of the n^{th} vehicle at the k^{th} step.

Parameters:

$p_n \in [0 \quad 1]$: relative importance of the n^{th} vehicle.

$\bar{S}^n$: maximum speed of the n^{th} vehicle.

$l^n(k)$: location (node id) of the n^{th} vehicle at step k.

$L^n = [l^n(k)] \quad \forall k \le K$: given route for each vehicle.

T_0^n : clock time of deployment for the n^{th} vehicle.

$T^n(k)$: clock time of the n^{th} vehicle at step k.

$d(\cdot,\cdot)$: distance function between two spatial nodes.

$\Gamma(\cdot,\cdot)$: obstacle-aware penalty function between two nodes.

$f^n(\cdot)$: processing time function of the n^{th} vehicle on a given node.

$$V^n : (L^n, T_0^n, \bar{S}^n)$$

$$\min_{\eta,\delta t} \sum_n p_n T^n(K)$$

s.t $\quad T^n(k+1) = T^n(k) + \delta t^n(k) \quad \forall n,k$

1

$$|T^n(k) - T^m(k)| > T^* \quad \text{if:} \quad l^n(k) = l^m(k) = l^* \quad \forall n,m,k$$

$$\text{where} \quad T^* = \begin{cases} f^n(l^*) & \text{if} \quad T^n(k) \le T^m(k), \\ f^m(l^*) & \text{else} \end{cases}$$

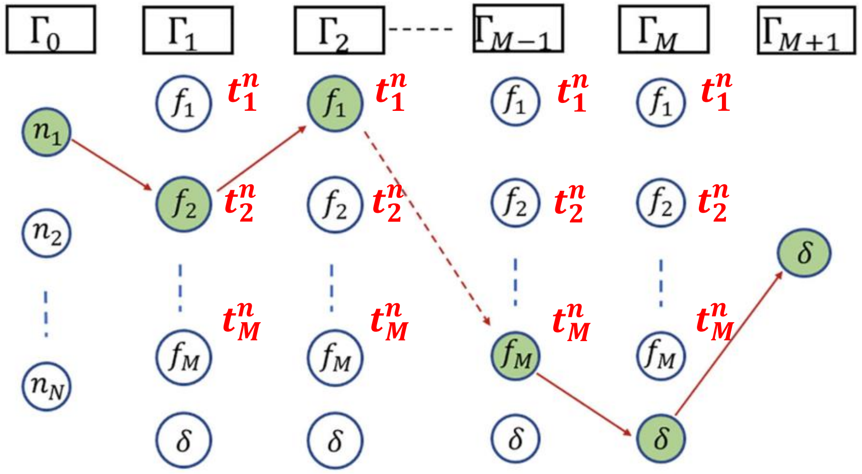
2

$$\delta t^n(k) \ge \frac{1}{\bar{S}^n}\left(d(l^n(k+1), l^n(k)) + \Gamma(l^n(k+1), l^n(k))\right) \quad \forall n,k$$

1- congestion constraint
2- max-speed constraint

Problem Statement: Scheduling on Pre-defined routes in FLPO form

Stage Illustration for drone n



Decision Variables

- t_j^n – Time taken to reach node j, $1 \le j \le M$
- $\eta_{j|i}^n(k) \in \{0, 1\}$ – i to j transition in stage k (Given)

Total cost for a drone : $D^n = T^n(K) + C^n(K)$

- Time taken : $T^n(K) = \Pi_{k=0}^{K} \eta_{j|i}^n(k) \sum_{k=0}^{K}(t_{j(k+1)}^n - t_{i(k)}^n)$
- Penalty incurred : $C^n(K) = \Pi_{k=0}^{K} \eta_{j|i}^n(k) \sum_{k=0}^{K} c_k^n(i,j)$
- $c_k^n(i,j)$ – penalty due to inequality constraints

Solution Approach Using the MEP Framework

$$D = \min_{t_j^n} \sum_n p_n \Pi_{k=0}^{K} \eta_{j|i}^n(k) \sum_{k=0}^{K} (t_{j(k+1)}^n - t_{i(k)}^n) + c_k^n$$

- Replace binary associations with a probability associations

$$\{0,1\} \ni \eta_{j|i}^n \to p_{j|i}^n(k) \in [0,1], \forall k$$

- Minimize free energy iteratively for $\beta > 0,\ \beta = \epsilon \to \infty,\ 0 < \epsilon \ll 1$

$$F = \min_{\{p_{j|i}^n(k), t_j^n\}} D - {}^{1}/_{\beta} H,$$

$$\text{where } H = \Pi_{k=0}^{K} p_{j|i}^n(k) \sum_{k=0}^{K} \log p_{j|i}^n(k)$$

Annealing

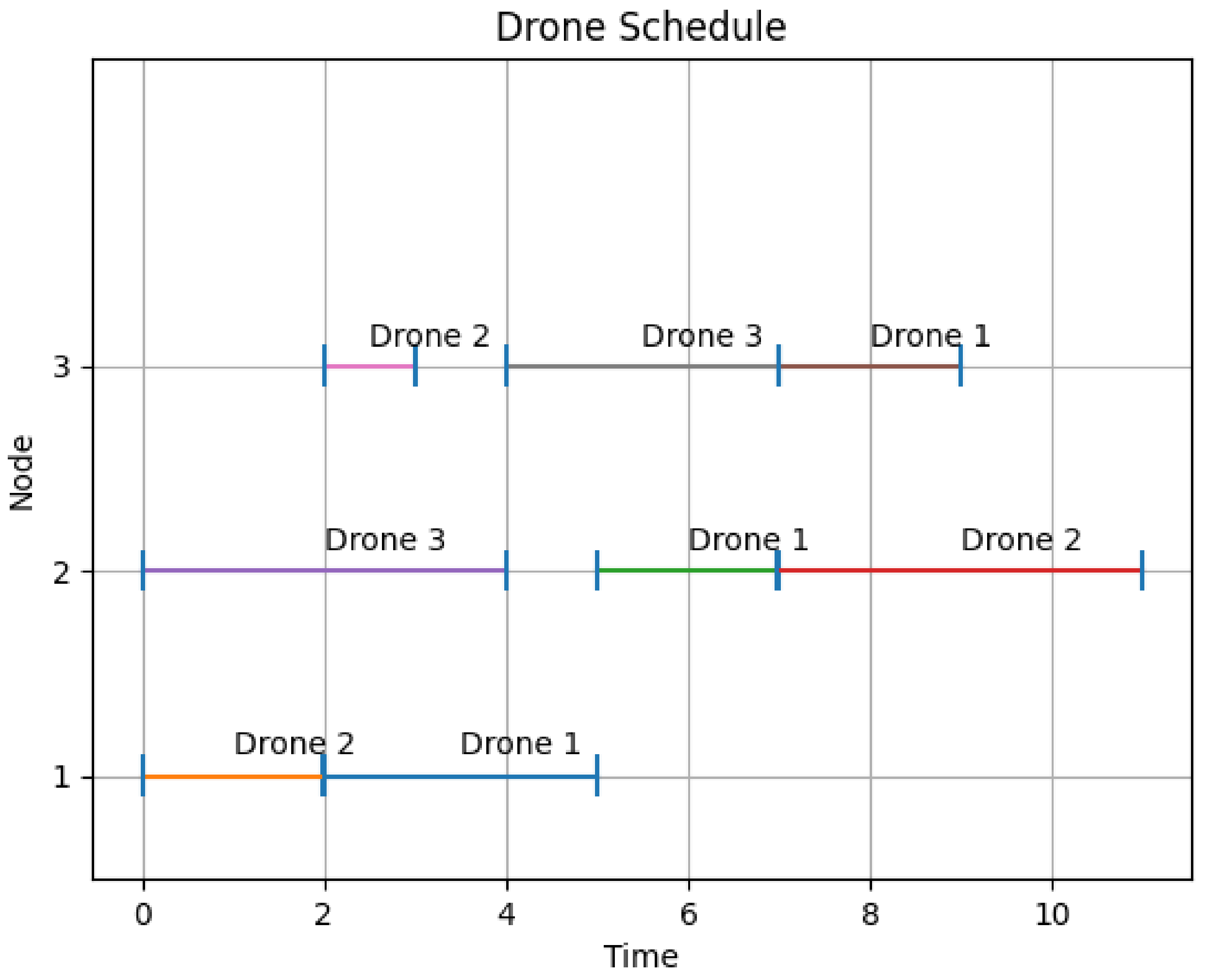
- Minimize for every β starting from 0 to ∞
- $\beta = 0 \Rightarrow$ convex free energy $\Rightarrow$ (Global minima = uniform distribution)
- $\beta \to \infty \Rightarrow [p(\gamma|\gamma_0) \to \eta(\gamma|\gamma_0)] \Rightarrow$ (hard associations = solution)

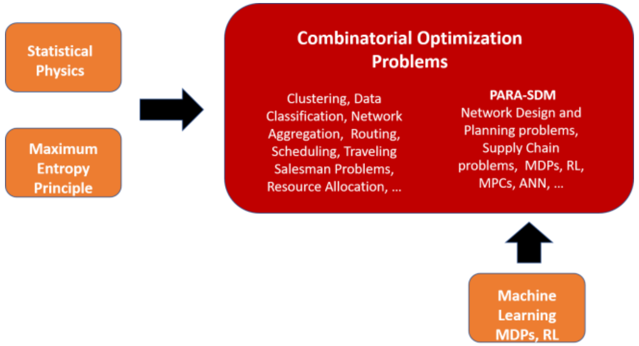
Problem Statement: Scheduling on pre-defined routs

```
Drone_list=[
```
$D1$: [(1, 3)≺(2, 2)≺ (3, 2)]
$D2$: [(1, 2)≺(3, 1)≺ (2, 4)],
$D3$: [(2, 4)≺(3, 3)]
]

facility node

dwell time



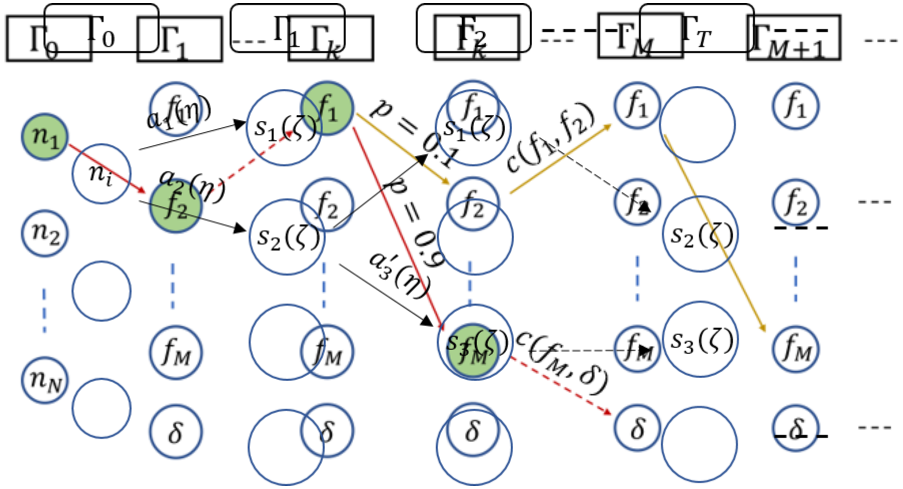


Parameterized Sequence-Decision Making Problems
Expanding the Framework

Algorithm

- **Sensitivity and Robustness**
- **Incorporating Constraints/Scalability**
- **Dynamic Constraints**

Infinite Horizon and Learning in PARA-SDMs



Infinite Horizon para-SDMs

SDM $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \zeta, \eta, c, p, \alpha \rangle$ with cost-free termination state δ

$$J^{\mu}_{\zeta\eta}(s) = \mathbb{E}_{p_\mu}\left[\sum_{t=0}^{\infty} \alpha^t c\big(x_t(\zeta), u_t(\eta), x_{t+1}(\zeta)\big) \,|x_0 = s\right] = \sum_{\omega\in\Omega} p_\mu(\omega|s)\ \bar{c}(s,\omega)$$

- $p_\mu : \omega \to [0,1]$ and $\omega = (a_0, x_1, a_1, x_2, \dots)$
  - $p_\mu(\omega|s) = \mu(a_0|s)p(x_1|a_0,s)\mu(a_1|x_1)p(x_2|x_1,a_1)\cdots$
- state, action parameters: $\zeta = \{\zeta_s\}, \eta = \{\eta_a\}$
- cost and dynamics: $c(s,a,s'), p(s'|s,a)$

- Generalizations
  - ★ **Parameterized states and actions**
  - ★ **Para-SDMs: Shortest Path + Facility Location Problems**
  - ★ **Stochastic Dynamics:**
    - transition **probability:** $p(s'|s,a)$
    - stochastic **policy:** $\mu(a|s)$
    - a realized path $\omega = (a_0, x_1, a_1, x_2, \cdots)$
  - ★ **Infinite horizon**
  - ★ **Para-RL:** cost $c(s,a,s')$ and dynamics $p(s'|s,a)$ not explicitly known

Infinite Horizon and Learning in para-SDMs

Lagrangian

$$V_\beta^\mu(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \alpha^t c(x_t, u_t, x_{t+1}) + \frac{1}{\beta}\log\mu(u_t|x_t)\right]$$

Assume $\mu_t = \mu$ and non-zero probability to reach terminal state $p_\mu(\omega|x_0) = \prod_{t=0}^{\infty} \mu(u_t|x_t) p(x_{t+1}|x_t, u_t)$

Theorem: *The Lagrangian $V_\beta^\mu(s)$ for the optimization problem (1) satisfies the following recursive Bellman equation*

$$V_\beta^\mu(s) = \sum_{a,s'} \mu(a|s)\, p(s'|s,a)\left(c(s',a,s') + \frac{\alpha}{\beta} log\, \mu(a|s) + \alpha V_\beta^\mu(s')\right) + \lambda_s$$

Control policy $$\mu_\beta^*(a|s) = \frac{\exp\left\{-\left({}^{\beta}/_{\alpha}\right)\boldsymbol{\Lambda_\beta^*(s,a)}\right\}}{\sum_{a'} \exp\left\{-\left({}^{\beta}/_{\alpha}\right)\Lambda_\beta^*(s,a')\right\}}$$

$$V_\beta^*(s) = -\frac{\alpha}{\beta}\log\left(\sum_{a\in\mathcal{A}} \exp\left\{-\frac{\beta}{\alpha}\boldsymbol{\Lambda_\beta^*(s,a)}\right\}\right)$$

$$\boldsymbol{\Lambda_\beta(s,a)} = \sum_{s'\in\mathcal{S}} p_{ss'}^a \left[\bar{c}_{ss'}^a - \frac{\alpha^2}{\beta}\log\left(\sum_{a\in\mathcal{A}} \exp\left\{-\frac{\beta}{\alpha}\boldsymbol{\Lambda_\beta(s',a)}\right\}\right)\right]$$

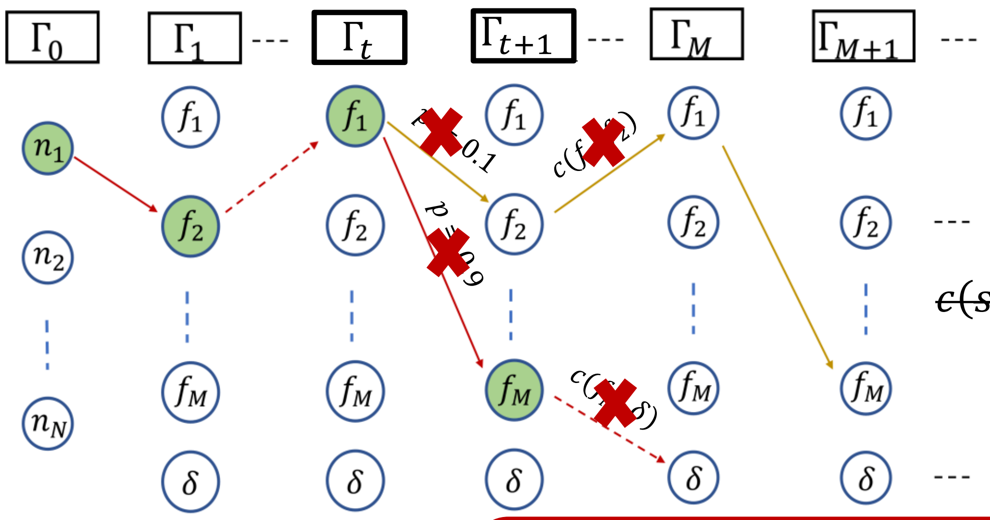
[Theorem: Contraction Map]

Parameters ζ^*, η^*

$$\sum_{s'\in\mathcal{S}} \frac{\partial V_\beta^*(s')}{\partial \zeta_s} = 0 \qquad \sum_{s'\in\mathcal{S}} \frac{\partial V_\beta^*(s')}{\partial \eta_a} = 0$$

Infinite Horizon and Learning in para-SDMs (para-RL)

Unknown dynamics and cost functions

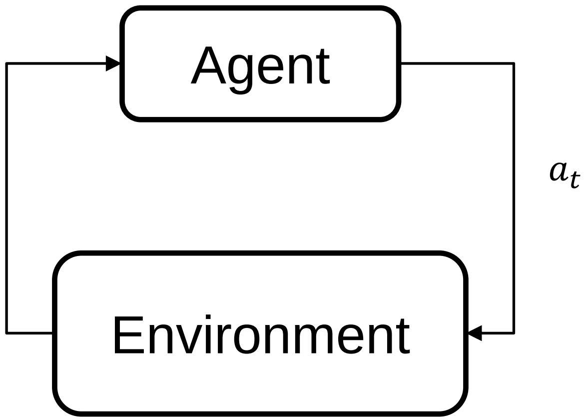


Observations:

$c(x_t, a_t, x_{t+1}),\ x_{t+1}$

~~$c(s, a, s'), p(s'|s, a)$~~

At time instant t



Model-free updates

$$\boldsymbol{\Psi}_{t+1}(x_t, a_t) = (1 - \nu_t)\boldsymbol{\Psi}_t(x_t, u_t) + \nu_t \left[c^{a_t}_{x_t x_{t+1}} - \frac{\alpha^2}{\beta} \log \sum_{a' \in \mathcal{A}} exp\left\{ \frac{-\beta}{\alpha} \boldsymbol{\Psi}_t(x_{t+1}, a') \right\} \right]$$

Policy: Converges to state−action value function : $\Psi_t \to \Lambda^*_\beta$ at (ζ, η)

Parameters: $\sum_{s' \in \mathcal{S}} \frac{\partial V^*_\beta(s')}{\partial \zeta_s} = 0$ $\qquad \frac{\partial V^*_\beta}{\partial \zeta} \approx \frac{V^*_\beta(\zeta') - V^*_\beta(\zeta)}{\zeta' - \zeta}$

Sensitivity and Robustness

- **Sensitivity:**
  - ★ Free-energy $V_\beta(s)$: smooth approximation of non-smooth $J(s)$
    - $V_\beta(s,\theta) = -\frac{\alpha}{\beta}\log\left(\sum_a \exp(-\frac{\beta}{\alpha}\Lambda_\beta(s,a,\theta)\right)$
    - θ : parameter; Sensitivity analysis $\frac{\partial V_\beta}{\partial\theta}$

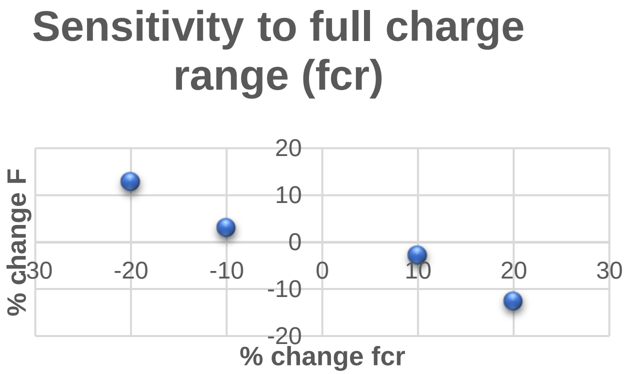
- **Robustness:**
  - ★ e.g., uncertainty in coordinates of UAS given $v(x|i)$;replace the distance by modified distance

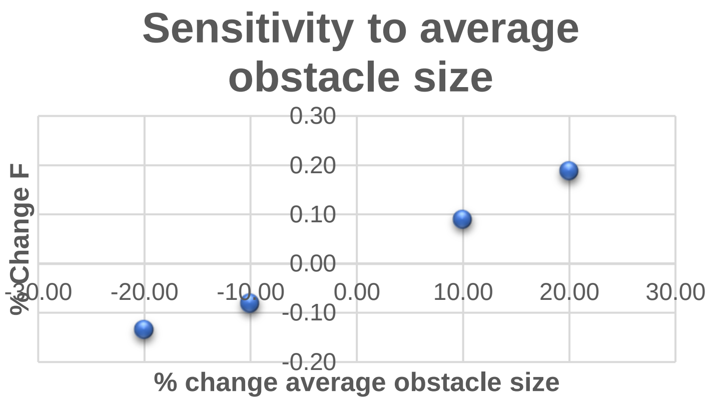
$$d'\left(x_i, y_j\right) = \sum_x v(x|i)d(x, y_j)$$

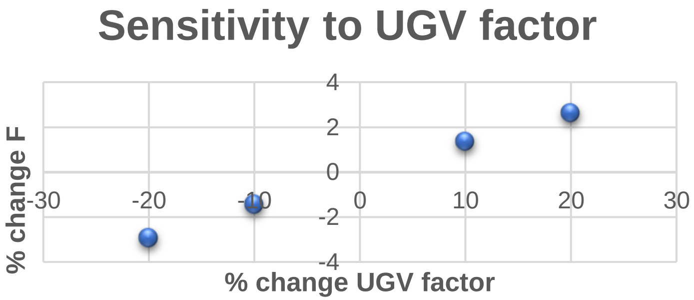
  - ★ number of UAS required for adequate coverage
    - exploit phase transition property

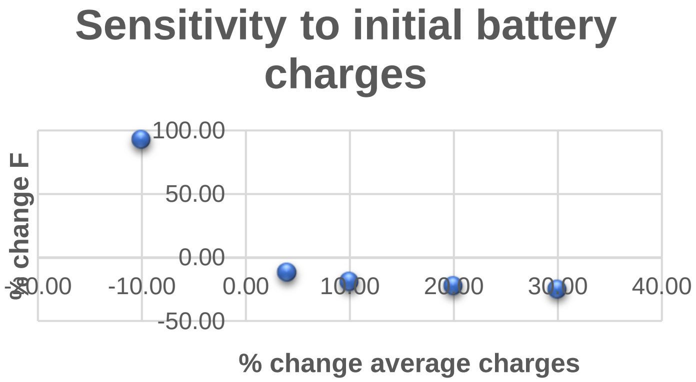
Srivastava, Amber, and Srinivasa M. Salapaka. "Robustness Analysis for Simultaneous Resource Allocation and Route Optimization Problems." *Dynamic Systems and Control Conference*. Vol. 58288. American Society of Mechanical Engineers, 2017.

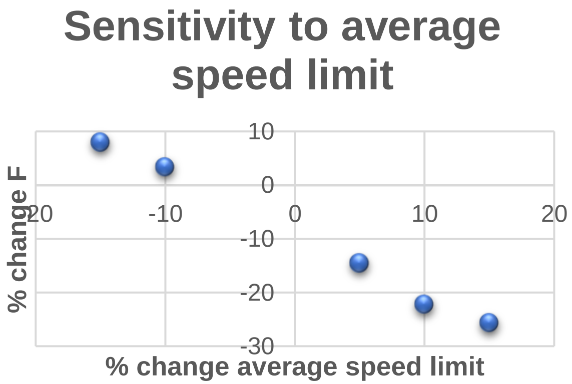
Sensitivity plots with respect to different parameters

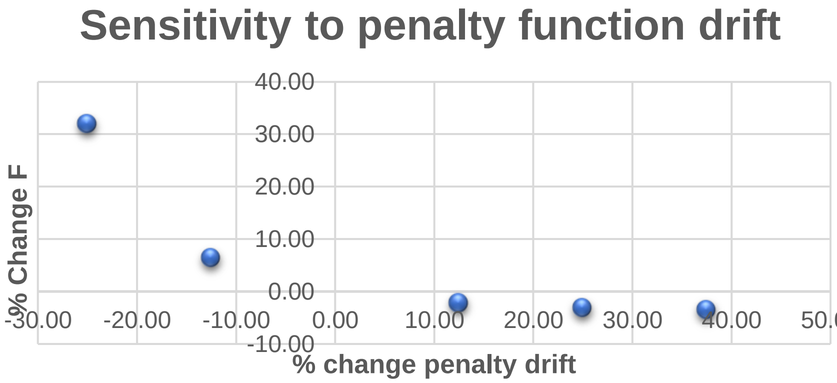








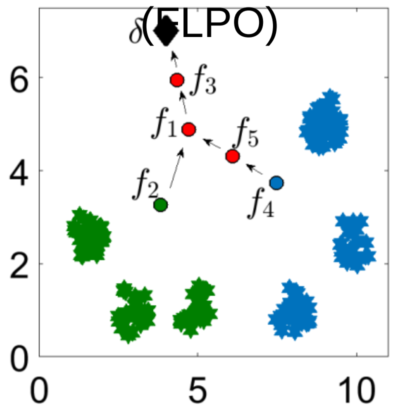




Incorporating Capacity and Exclusion-Inclusion Constraints

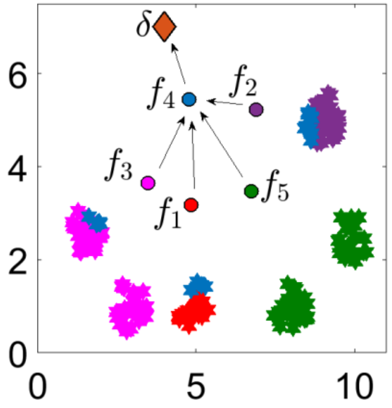
- Constraints in Network Design
- Partially Connected Network
- Capacity constraints on Facilities
- Link capacity constraints

Unconstrained (FLPO)



Link capacity constraints
$f_3f_4, f_1f_4, f_5f_4, f_2f_4$

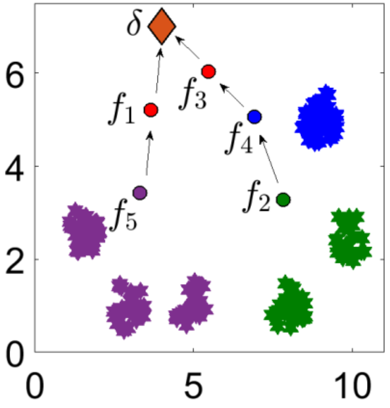
Capacity of transitions s to s'



$$C(s,s') = \sum_a p_{ss'}^a \mu(a|s)$$

Partially connected network FLPO
$f_1f_2,\ f_4f_1,\ f_2f_3,\ f_4f_5,\ f_5f_3$

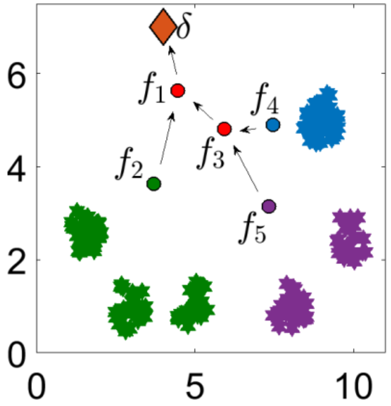
Restricted action space $\mathcal{A}_s$



$$\mu(a|s) = \frac{w_{as} \exp\left\{-\frac{\beta}{\alpha}\Lambda_\beta(s,a)\right\}}{\sum_{a'} w_{a's} \exp\left\{-\frac{\beta}{\alpha}\Lambda_\beta(s,a')\right\}}$$

Facility capacity constraints
$f_1 : f_2 : f_3 : f_4 : f_5$

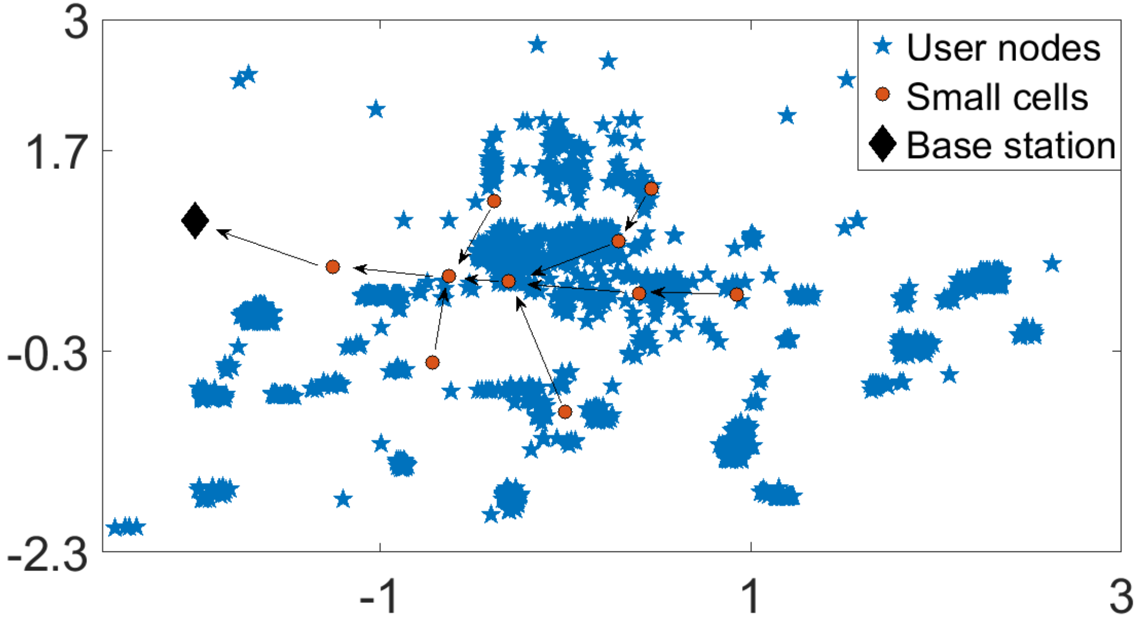
Capacity of a state $s \in \mathcal{S}$.



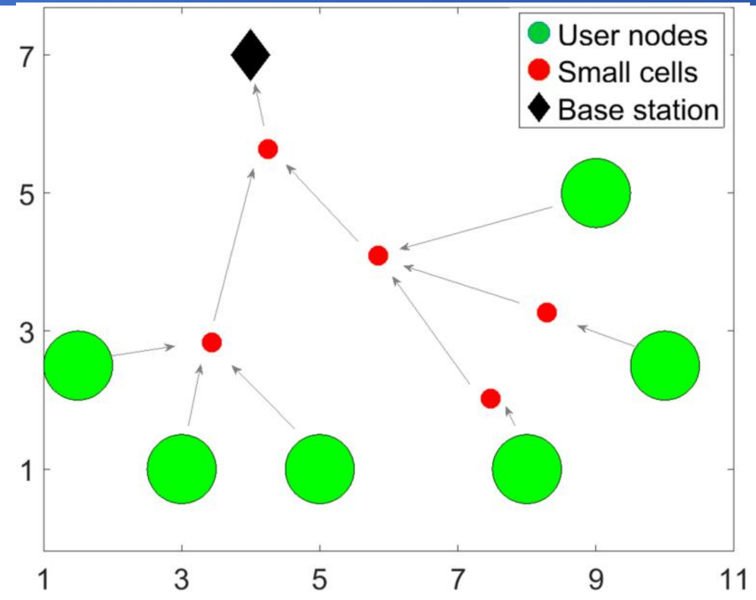
Quantify # of visits to s'
$C(s') = \sum_{s,a} p_{ss'}^a \mu(a|s)$
$V_\beta^\mu(s') + \nu_{s'}(C(s') - c_{s'})$

Scalability

- Function Approximator (ANN) $\widehat{\Lambda}_\beta(s,a;\mathbf{w}) \approx \Lambda^*_\beta(s,a)$
- Address **para-SDM in large state and action spaces.**
- Feature: para-SDM algorithm is parallelizable



Urbana city data : 2200 user nodes
Allocated : 10 small cells

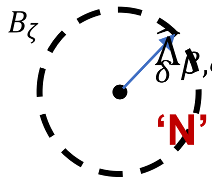


$$\zeta^+ = \zeta - \tau \frac{\partial V^*_\beta}{\partial \zeta}; \text{ where } \frac{\partial V^*_\beta}{\partial \zeta} \approx \frac{V^*_\beta(\zeta') - V_\beta(\zeta)}{\zeta' - \zeta}$$

$V^*_\beta(\zeta') \rightarrow \widehat{\Lambda}_{\beta,\zeta'}(s,a;\mathbf{w}')$

$V^*_\beta(\zeta) \rightarrow \widehat{\Lambda}_{\beta,\zeta}(s,a;\mathbf{w})$

2 parallel agents



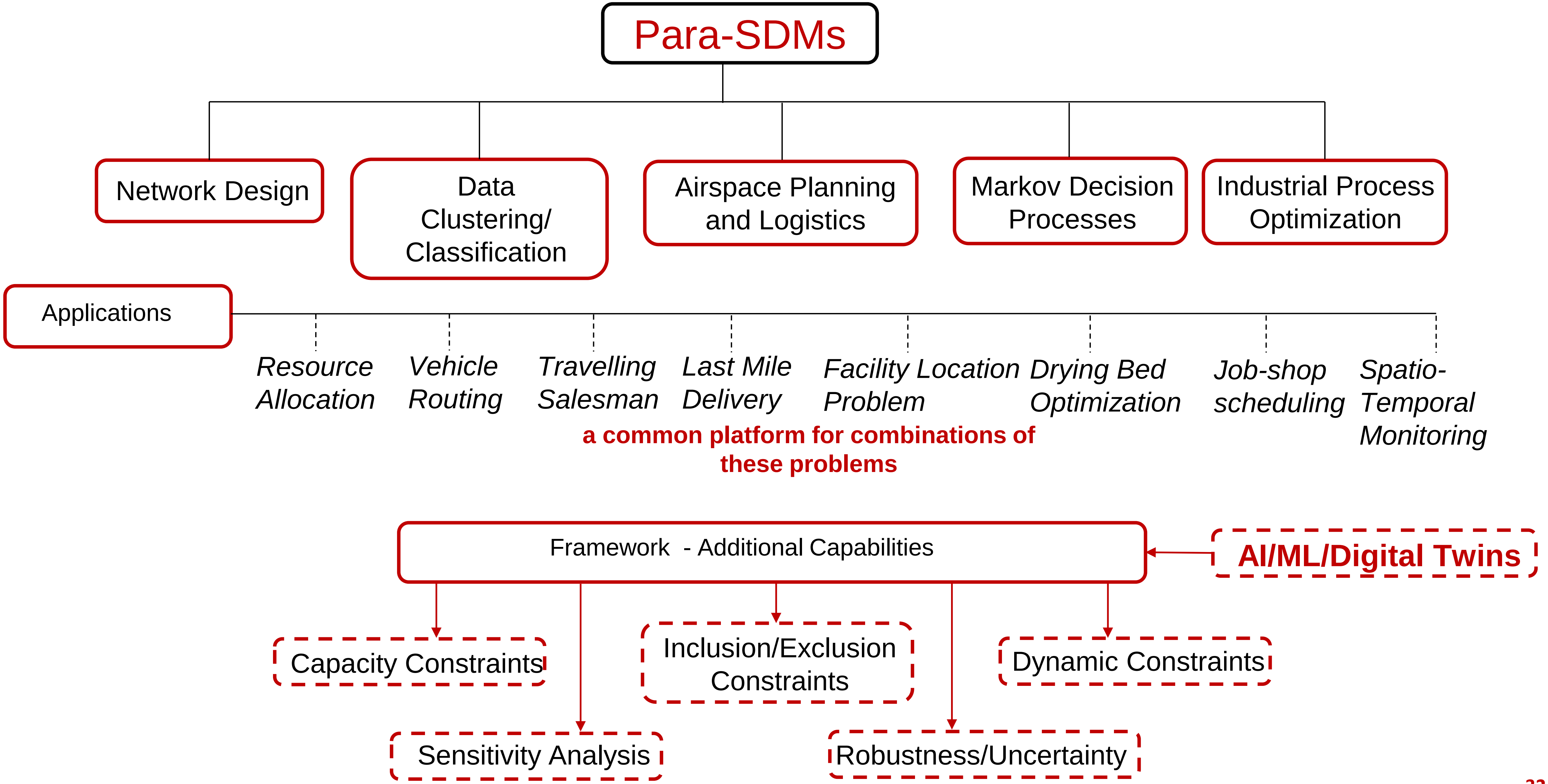
$\widehat{\Lambda}_{\beta,\zeta}$ in B_ζ

'N' parallel agents

Optimize ζ

$\min_{\zeta \in B_\zeta} V_\beta(\zeta)$

Conclusion


Para-SDMs
Network Design
Data Clustering/ Classification
Airspace Planning and Logistics
Markov Decision Processes
Industrial Process Optimization
Applications
Resource Allocation
Vehicle Routing
Travelling Salesman
Last Mile Delivery
Facility Location Problem
Drying Bed Optimization
Job-shop scheduling
Spatio-Temporal Monitoring
a common platform for combinations of these problems
Framework - Additional Capabilities
AI/ML/Digital Twins
Capacity Constraints
Inclusion/Exclusion Constraints
Dynamic Constraints
Sensitivity Analysis
Robustness/Uncertainty

Salar Basiri
UIUC

Dhananjay Tiwari
UIUC

Mustafa Kapadia
Cummins Inc

Amber
Srivastava
IIT Delhi, India

Thank You