Dynamic Inter-day and Intra-day Scheduling

Christos Zacharias
Miami Herbert Business School
University of Miami
czacharias@bus.miami.edu

Nan Liu
Carroll School of Management
Boston College
nan.liu@bc.edu

Mehmet A. Begen
Ivey Business School
Western University
mbegen@ivey.uwo.ca

Scheduling Seminar, Sep 14 2022

wait room

long backlogs of appointments

- Patients experience difficulties in accessing medical care.

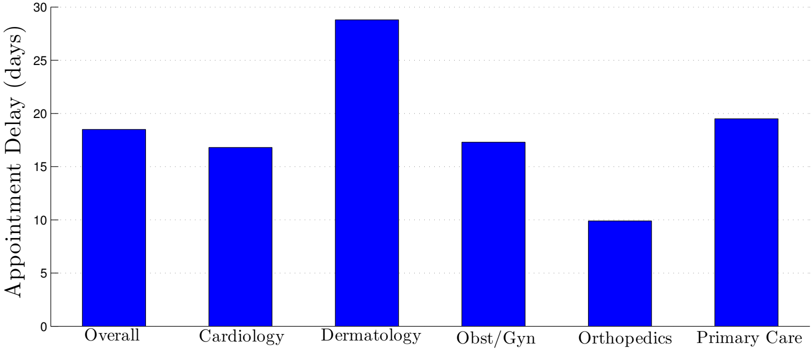
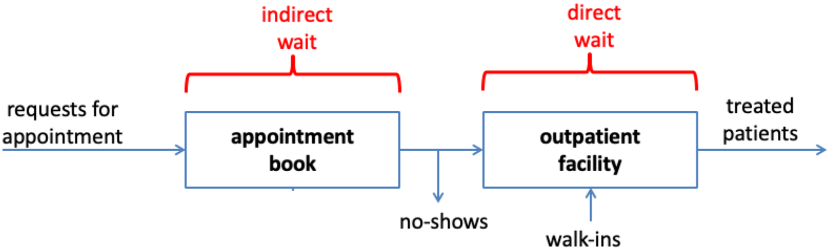


Figure: Merrit et al. (2015): Average appointment delay across 1399 medical offices in 15 US metropolitan areas

time scales

1. **Indirect delay**
   - Out-of-clinic wait
   - Virtual
   - Order of days, weeks
   - Negative health outcomes
2. **Direct delay**
   - On-site wait
   - Physical
   - Order of minutes, hours
   - Discomfort, frustration



outpatient appointment scheduling problem

- **Outpatient Appointment Scheduling**
  - optimize **intra-day** and/or **inter-day** operations
  - **manage patient arrivals**
    across work-days **inter-day scheduling**
    within a work-day **intra-day scheduling**
  - sources of **uncertainty**
    1. no-shows
    2. non-punctuality
    3. emergency walk-ins
    4. stochastic consultation times
    5. stochastic demand for outpatient services
    6. patient heterogeneity/preferences
    7. seasonality

    ⋮
  - computationally complex **combinatorial** problem
  - curse of dimensionality (in dynamic settings)

literature

	intra-day scheduling	inter-day scheduling	joint intra/inter-day
static models	Zacharias and Yunes (2020) Wang et al. (2020) Kong et al. (2018) Qi (2017) Jiang et al. (2017) Zacharias and Pinedo (2017) Cayirli et al. (2012) Luo et al. (2012) Begen and Queyranne (2011) Robinson and Chen (2010) ...	Liu (2016) Liu and Ziya (2014) Green and Savin (2008) ...	Zacharias and Armony (2017) Luo et al. (2015) ...
dynamic models	Green et al. (2006) ...	Truong (2015) Feldman et al. (2014) Truong and Shapiro (2013) Liu et al. (2010) Patrick et al. (2008) ...	uncharted territory

literature

	intra-day scheduling	inter-day scheduling	joint intra/inter-day
static models	Zacharias and Yunes (2020) Wang et al. (2020) Kong et al. (2018) Qi (2017) Jiang et al. (2017) Zacharias and Pinedo (2017) Cayirli et al. (2012) Luo et al. (2012) Begen and Queyranne (2011) Robinson and Chen (2010) ...	Liu (2016) Liu and Ziya (2014) Green and Savin (2008) ...	Zacharias and Armony (2017) Luo et al. (2015) ...
dynamic models	Green et al. (2006) ...	Truong (2015) Feldman et al. (2014) Truong and Shapiro (2013) Liu et al. (2010) Patrick et al. (2008) ...	uncharted territory

research question

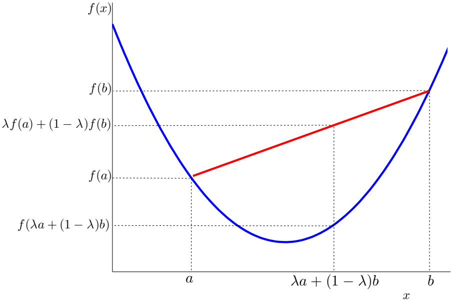
Research Question

- **Dynamic** inter-day and intra-day scheduling model
- Mathematically and computationally tractable
- Realistic enough to be useful for practice

Approach

- Combine results from
  - Truong (2015) [Management Science]
    - **Dynamic** inter-day scheduling
    - Dimensionality reduction results
    - Analytical characterization
    - Computationally feasible exact solution
  - Zacharias and Yunes (2020) [Management Science]
    - **Static** intra-day scheduling
    - General stochastic service times, walk-ins, no-shows
    - Exact transient analysis, discrete convexity results
    - Computationally feasible exact solution

Convex Functions on Continuous Spaces

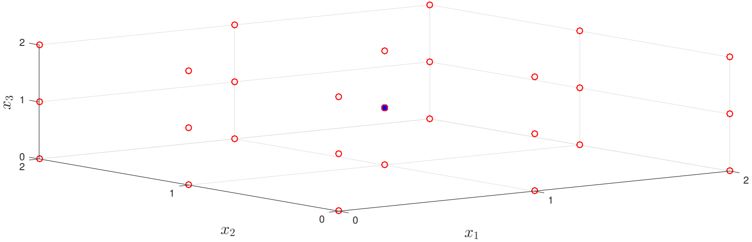


"A function f is convex if the line segment connecting any two points on the graph of the function lies above the graph."

Convex Functions on $\mathbb{R}^n$

- When f is twice differentiable, then $\nabla^2 f(x)$ is positive definite.
- $\begin{pmatrix} \text{local} \\ \text{minimum} \end{pmatrix} = \begin{pmatrix} \text{global} \\ \text{minimum} \end{pmatrix}$
- First order condition for local minimum: $\nabla f(x) = 0$

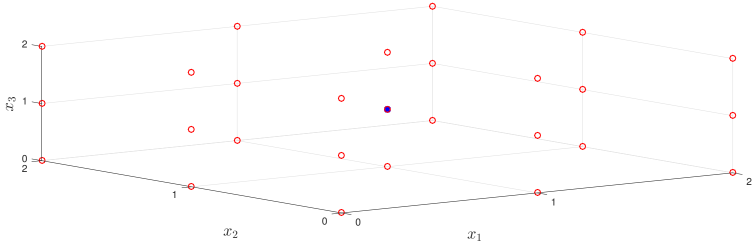
Convex Functions on Discrete Spaces



- Optimization on $\mathbb{Z}^n$ is in general computationally expensive
- It is crucial to identify structures that guarantee the success of local search algorithms, so that

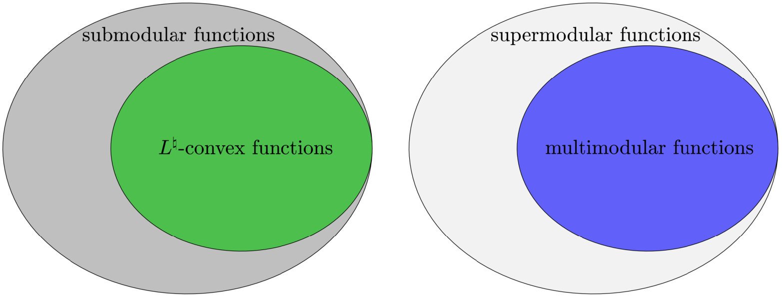
$$\begin{pmatrix} \text{local} \\ \text{optimum} \end{pmatrix} = \begin{pmatrix} \text{global} \\ \text{optimum} \end{pmatrix}$$

Convex Functions on $\mathbb{Z}^n$



- How do we define convexity in discrete spaces?
- How do we define a local neighborhood?
- Do we consider all possible combinations of unit directions?

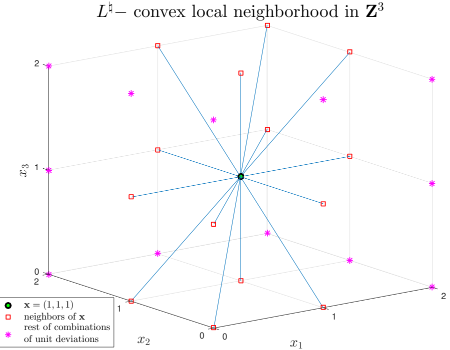
Multimodularity and $L^\natural$-convexity

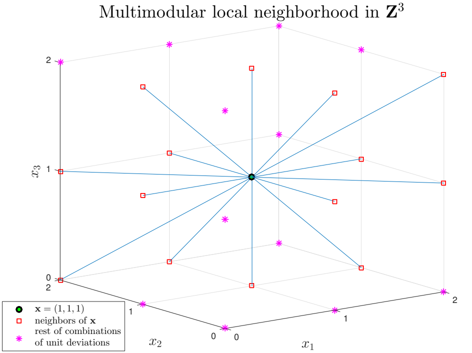


- $L^\natural$-convexity
  - Introduced by Murota (1998)
  - **Local optima also global**
- Multimodularity
  - Introduced by Hajek (1985)
  - **Local optima also global**

$$\begin{pmatrix}\text{local neighborhood}\\ \text{under}\\ L^\natural\text{-convexity}\end{pmatrix} \neq \begin{pmatrix}\text{local neighborhood}\\ \text{under}\\ \text{multimodularity}\end{pmatrix}$$

Multimodularity and $L^\natural$-convexity (cont'd)





$$\begin{pmatrix} \text{size of local} \\ \text{neighborhood} \\ \text{in } \mathbb{Z}^n \end{pmatrix} = 2^{n+1} - 2$$

Roadmap

state of the MDP

$$\mathrm{X}_t = \begin{bmatrix} \mathrm{x}_{t1} \\ \mathrm{x}_{t2} \\ \vdots \\ \mathrm{x}_{t\tau} \\ \vdots \end{bmatrix} = \begin{array}{c} \text{day } 1 \\ \text{day } 2 \\ \vdots \\ \text{day } \tau \\ \vdots \end{array} \overset{\begin{array}{cccc} \text{slot 1} & \text{slot 2} & \ldots & \text{slot n} \end{array}}{\begin{bmatrix} x_{t1}^1 & x_{t1}^2 & \ldots & x_{t1}^n \\ x_{t2}^1 & x_{t2}^2 & \ldots & x_{t2}^n \\ \vdots & \vdots & \ddots & \vdots \\ x_{t\tau}^1 & x_{t\tau}^2 & \ldots & x_{t\tau}^n \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}} \in \mathcal{X}$$

- X_t = state of the MDP on day t (int. matrix)
- $\mathrm{x}_{t\tau}$ = schedule in τ days from t (int. vector)
- $x_{t\tau}^i$ = # patients scheduled at slot i in τ days from t (int. scalar)
- curse of dimensionality:
    - dimensionality reduction results
    - heuristics/approximations based on theory

dynamic programming formulation

1. Observe current state (schedule) X_t → a matrix
2. Observe new demand d_t
3. Make a scheduling decision $\mathrm{B}_t \in \mathcal{B}(X_t, d_t)$ → a matrix
4. Update the schedule $\mathrm{Z}_t = \mathrm{X}_t + \mathrm{B}_t \in \mathcal{Z}(\mathrm{X}_t, d_t)$ → a matrix
5. Tomorrow's updated schedule is z_{t1} → a vector
6. Incur inter-day and intra-day costs
7. Update the state $\mathrm{X}_{t+1} = \zeta(\mathrm{Z}_t)$ and move to next period

$$V_t(\mathrm{X}_t, d_t) = \min_{\mathrm{Z}_t \in \mathcal{Z}(\mathrm{X}_t, d_t)} \Big\{ \mathtt{inter}(\mathrm{Z}_t) + \mathtt{intra}(\mathrm{z}_{t1}) + \gamma \mathbb{E}_{d_{t+1}} \left[V_{t+1}(\zeta(\mathrm{Z}_t), d_{t+1}) \right] \Big\}$$

where

$\mathtt{inter}(\mathrm{Z}_t) = c_a|\mathrm{Z}_t|$ = linear indirect waiting cost for all patients

$\mathtt{intra}(\mathrm{z}_{t1})$ = multimodular function

Roadmap

Truong (2015): Optimal Advance Scheduling

Truong (2015): Optimal Advance Scheduling [Mgmt Science]:

- **dynamic** inter-day scheduling with commitment
- Dynamically assign appointment day (inter-day decision) but not appointment time (intra-day decision).
- State of the MDP is a vector that captures the total number of patients scheduled for each day in the scheduling horizon.
  - $\mathrm{x}_t = (x_{t\tau})_\tau$ = state of the MDP on day t (vector)
  - $x_{t\tau}$ = # patients scheduled in τ days from t (scalar)

Truong (2015): Optimal Advance Scheduling

Truong (2015): Optimal Advance Scheduling [Mgmt Science]:

1. Observe current state (schedule) x_t → a vector
2. Observe new demand d_t
3. Make a booking decision $\mathrm{b}_t \in \mathcal{B}(\mathrm{x}_t, d_t)$ → a vector
4. Update the schedule $\mathrm{z}_t = \mathrm{x}_t + \mathrm{b}_t \in \mathcal{Z}(\mathrm{x}_t, d_t)$ → a vector
5. Tomorrow's updated schedule has z_{t1} patients → a scalar
6. Incur inter-day and intra-day costs
7. Update the state $\mathrm{x}_{t+1} = \eta(\mathrm{z}_t)$ and move to next period

$$\tilde{V}_t(\mathrm{x}_t, d_t) = \min_{\mathrm{z}_t \in \mathcal{Z}(\mathrm{x}_t, D_t)} \Big\{ \tilde{\mathtt{inter}}(\mathrm{z}_t) + \tilde{\mathtt{intra}}(z_{t1}) + \beta \mathbb{E}_{d_{t+1}} \Big[\tilde{V}_{t+1}(\eta(\mathrm{z}_t), d_{t+1}) \Big] \Big\}$$

where

$\tilde{\mathtt{inter}}(\mathrm{z}_t) = c_a|\mathrm{z}_t| =$ linear indirect waiting cost for all patients

$\tilde{\mathtt{intra}}(z_{t1}) =$ convex function of one viariable

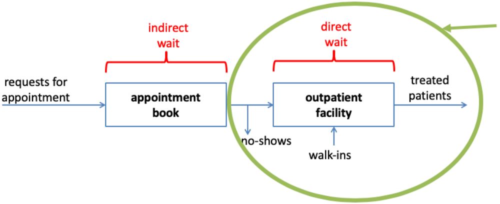
Truong (2015): Optimal Advance Scheduling

Reduction of the problem to a single dimension due to successive refinability property

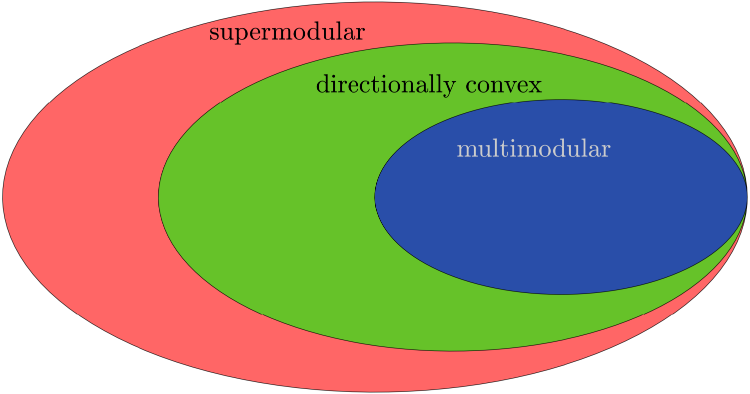
- Relax the constraint that prior commitments are binding
- Solve the unconstrained problem
- Well defined and easily computable solution
- The solution is a refinement of the existing schedule
- Any changes in the schedule can be made with new requests

Zacharias and Yunes (2020)

- Zacharias and Yunes (2020) Multimodularity in the Stochastic Appointment Scheduling Problem with Discrete Arrival Epochs [Management Science]
  - **Static** intra-day scheduling
  - General stochastic service times, walk-ins, no-shows
  - Exact transient analysis, discrete convexity results
  - Computationally feasible exact solution



Supermodularity & Multimodularity



$$\min_{x \in \mathbb{Z}_+^n} f(x)$$

Supermodularity & Directional Convexity

Definition: Supermodularity & Directional Convexity

(a) A function $g : \mathbb{Z}_+^n \to \mathbb{R}$ is **supermodular** if

$$g(x + e_i + e_j) - g(x + e_i) \geq g(x + e_j) - g(x) \qquad (1)$$

for all $x \in \mathbb{Z}_+^n$ and for all $1 \leq i < j \leq n$.

(b) A function $g : \mathbb{Z}_+^n \to \mathbb{R}$ is **directionally convex** if inequality (1) holds for all $1 \leq i \leq j \leq n$.

(directionally convex) = (supermodular) + (componentwise convex)

note: directional convexity alone does not guarantee optimality of local optima, under any definition of locality in the literature.

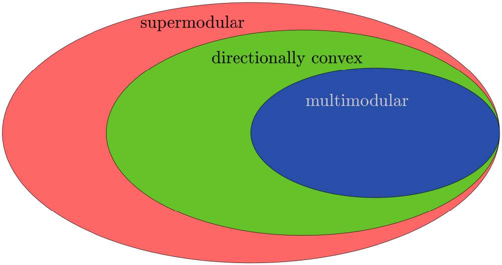
Multimodularity

Definition: Multimodularity

A function $g : \mathbb{Z}_+^n \rightarrow \mathbb{R}$ is **multimodular** if for all $\mathrm{x} \in \mathbb{Z}_+^n$

$$
\begin{array}{lrl}
\text{(m1)} & g(\mathrm{x}+\mathrm{e}_1+\mathrm{e}_n) - g(\mathrm{x}+\mathrm{e}_1) \geq & g(\mathrm{x}+\mathrm{e}_n) - g(\mathrm{x}) \\
\text{(m2)} & g(\mathrm{x}+\mathrm{e}_{i+1}+\mathrm{e}_j) - g(\mathrm{x}+\mathrm{e}_{i+1}+\mathrm{e}_{j+1}) \geq & g(\mathrm{x}+\mathrm{e}_i+\mathrm{e}_j) - g(\mathrm{x}+\mathrm{e}_i+\mathrm{e}_{j+1}) \ \ \forall i \neq j \\
\text{(m3)} & g(\mathrm{x}+\mathrm{e}_1+\mathrm{e}_j) - g(\mathrm{x}+\mathrm{e}_1+\mathrm{e}_{j+1}) \geq & g(\mathrm{x}+\mathrm{e}_j) - g(\mathrm{x}+\mathrm{e}_{j+1}) \ \ \forall j \\
\text{(m4)} & g(\mathrm{x}+\mathrm{e}_{i+1}+\mathrm{e}_n) - g(\mathrm{x}+\mathrm{e}_{i+1}) \geq & g(\mathrm{x}+\mathrm{e}_i+\mathrm{e}_n) - g(\mathrm{x}+\mathrm{e}_n) \ \ \forall i
\end{array}
$$

- introduced by Hajek (1985) - optimal admission control to queues



Local Neighborhood

Theorem (Murota 2004): local optima are global optima

For a multimodular function $g : \mathbb{Z}^n \to \mathbb{R}$ we have

$$g(\mathrm{x}) \leq g(\mathrm{y}) \text{ for all } \mathrm{y} \in \mathbb{Z}^n \iff g(\mathrm{x}) \leq g(\mathrm{x} \pm \mathrm{d}) \text{ for all } \mathrm{d} \in \mathcal{D},$$

where $\mathcal{D}$ is the set of vectors of the form $\mathrm{e}_{i_1} - \mathrm{e}_{i_2} + \ldots + (-1)^{k-1}\mathrm{e}_{i_k}$ for some increasing sequence of indices $1 \leq i_1 < i_2 < \ldots < i_k \leq n$.

$$2|\mathcal{D}| = 2^{n+1} - 2$$

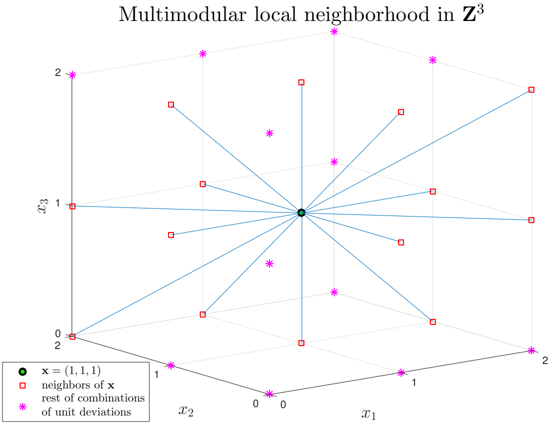
Local Neighborhood in 3D

- $$\mathcal{D} = \begin{Bmatrix} (1, & 0, & 0), \\ (0, & 1, & 0), \\ (0, & 0, & 1), \\ (1, & -1, & 0), \\ (1, & 0, & -1), \\ (0, & 1, & -1), \\ (1, & -1, & 1) \end{Bmatrix}$$
- neighbors of x

$$\mathsf{x} \pm \mathsf{d}, \mathsf{d} \in \mathcal{D}$$

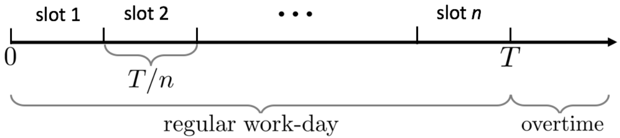
- size of neighborhood

$$14 = 2|\mathcal{D}| = 2^4 - 2$$

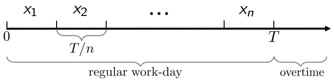


Queueing Model

- **Queueing System:**
  - $GI_t/GI/1$ in $[0, T]$
  - Work conserving
  - FIFO
- **Time Scale:**
  - time is measured in minutes
  - T = length of regular work-day (e.g., $T = 480\text{mins} = 8\text{hrs}$)
  - time is continuous, but
  - work-day is partitioned into n discrete slots of equal duration
  - $d = T/n$ = slot duration (e.g., 30 mins, 10 mins, 5 mins, 1min)
  - larger $n \Rightarrow$ more refined scheduling decisions
  - larger $n \Rightarrow$ increased computational complexity



Optimization Problem



- **Decision Variables:**

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

x_1: # of patients assigned to slot 1
x_2: # of patients assigned to slot 2
⋮
x_n: # of patients assigned to slot n

- **Optimization Problem:**

• idle time • overtime • direct wait

$$\min_{x \in \mathbb{Z}_+^n} f(x) = c_i \, \mathbb{E}[I(x)] + c_o \, \mathbb{E}[O(x)] + c_w \, \mathbb{E}[W(x)]$$

Multimodular Function Minimization

Theorem

The objective function $f : \mathbb{Z}_+^n \to \mathbb{R}$ is multimodular.

$$\min_{x \in \mathbb{Z}_+^n} f(x)$$

- let $f : \mathbb{Z}_+^n \to \mathbb{R}$ be a multimodular function
- $\begin{pmatrix}\text{local}\\ \text{min}\end{pmatrix} = \begin{pmatrix}\text{global}\\ \text{min}\end{pmatrix}$
- $\begin{pmatrix}\text{size of local}\\ \text{neighborhood}\end{pmatrix} \sim 2^{n+1}$
- **can we solve the problem in polynomial time?**

Multimodular Function Minimization

Theorem: local search in polynomial time

Let $f : \mathbb{Z}^n_+ \to \mathbb{R}$ be a multimodular function. Then the local neighborhood search can be performed in polynomial time.

proof sketch:

1. $f(\mathrm{x})$ multimodular
2. $g(\mathrm{x}) = f(D\mathrm{x})$ is $L^\natural$-convex, for some bidiagonal matrix D ← Murota (2005)
3. $\min\{f(\mathrm{x}) : \mathrm{x} \in \mathbb{Z}^n_+\} \iff$
   $\min\{g(\mathrm{x}) : D\mathrm{x} \in \mathbb{Z}^n_+\} \iff$
   $\min\{g(\mathrm{x}) : 0 \le x_1 \le x_2 ... \le x_n, \mathrm{x} \in \mathbb{Z}^n_+\}$
4. translate the problem to **constrained** submodular set-function minimization ← NP-hard in general
5. constraint set is a **ring family** & identify its **join-irreducible** members
6. translate the problem to **unconstrained** submodular set-function minimization ← Schrijver (2000)
7. problem solved in polynomial time ← $\mathcal{O}(n^5\gamma + n^6)$ Orlin (2009)

Roadmap

1. introduction ✓
2. DP framework ✓
3. Truong (2015) inter-day scheduling ✓
4. Zacharias and Yunes (2020) intra-day scheduling ✓
5. joint inter-day & intra-day scheduling
6. conclusion

Connecting Truong (2015) & Zacharias and Yunes (2020)

- Truong (2015):
  - The intra-day cost is a function of a **scalar** (# of patients in the schedule)
  - Assumed to be general convex.
- Zacharias and Yunes (2020):
  - The intra-day cost is a function of a **vector** (a detailed schedule)
  - The # of patients in the schedule is the outcome of **unconstrained** optimization
  - The unconstrained problem can be solved in polynomial time
- Connecting link:
  - The intra-day cost is a function of a **scalar** (# of patients in the schedule)
    It is the outcome of **constrained** optimization.
  - Computable efficiently?
  - Convex?

Unconstrained Intra-day Scheduling (UIS)

- Consider the dynamic joint inter-day and intra-day problem defined by $V(X_t, d_t)$
- Allow last-minute rearrangements of patients within tomorrow's schedule
- Only the # of patients in an intra-day schedule affects the intra-day cost function $\mathtt{in\bar{t}ra}^u(\cdot)$
- Denote this relaxation of the problem as $V^u(X_t, d_t)$

$$\mathtt{in\bar{t}ra}^u(y) = \min_{x \in \mathbb{Z}_+^n} f(x) \quad \textbf{s.t.} \quad \sum_{i=1}^{n} x_i = y$$

$$f(\mathbf{x}) = c_i \, \mathbb{E}[I(\mathbf{x})] + c_o \, \mathbb{E}[O(\mathbf{x})] + c_w \, \mathbb{E}[W(\mathbf{x})]$$

- idle time ($\mathbb{E}[I(\mathbf{x})]$)
- overtime ($\mathbb{E}[O(\mathbf{x})]$)
- direct wait ($\mathbb{E}[W(\mathbf{x})]$)

Unconstrained Intra-day Scheduling (UIS)

- idle time
- overtime
- direct wait

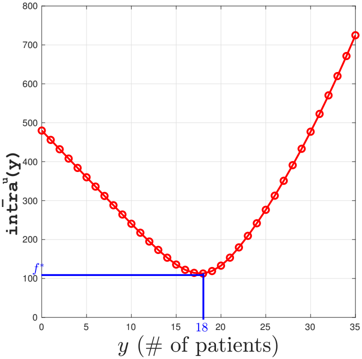
$$f(\mathrm{x}) = c_i \, \mathbb{E}[I(\mathrm{x})] + c_o \, \mathbb{E}[O(\mathrm{x})] + c_w \, \mathbb{E}[W(\mathrm{x})]$$

$$\overline{\mathtt{intra}}^u(y) = \min_{\mathrm{x} \in \mathbb{Z}_+^n} \quad f(\mathrm{x})$$

$$\text{s.t.} \quad \sum_{i=1}^{n} x_i = y$$

Theorem

If $f : \mathbb{Z}_+^n \to \mathbb{R}$ is multimodular, then $\overline{\mathtt{intra}}^u(y)$ can be computed in polynomial time for all y. Moreover, $\overline{\mathtt{intra}}^u(\cdot)$ is convex and $V^u(\cdot,\cdot)$ is computationally tractable.



Sequentially Refinable Intra-day Scheduling (SRIS)

- We say that the sequence of vectors $\{x^s_y, \ y = 1, 2, ...\}$ is **sequentially refinable** if

$$0 \leq x^s_1 \leq x^s_2 \leq x^s_3 .... \text{ and } |x^s(y)| = y \text{ for all } y.$$

- Any changes in the schedule, as the number of patients grows, can be made with new requests.
- Intra-day daily schedules are constructed to form a sequentially refinable sequence by construction.

$$x_y \triangleq \begin{cases} \text{argmin}\{f(x) : x \in \mathbb{Z}^n_+\} & \text{if } y = y^* = |\text{argmin}\{f(x) : x \in \mathbb{Z}^n_+\}| \\ \text{argmin}\{f(x) : x \in \mathbb{Z}^n_+, |x| = y, x \leq x_{y+1}\} & \text{for } b = y^* - 1, y^* - 2, \ldots, 2, 1, 0 \\ \text{argmin}\{f(x) : x \in \mathbb{Z}^n_+, |x| = y, x \geq x_{y-1}\} & \text{for } y = y^* + 1, y^* + 2, \ldots. \end{cases}$$

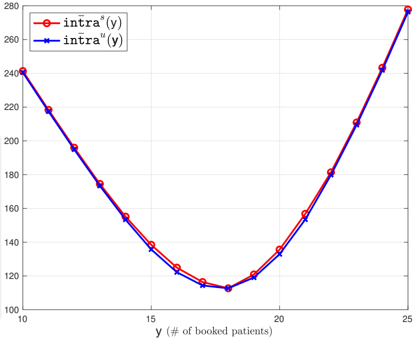
- Only the # of patients in an intra-day schedule affects the intra-day cost function $\texttt{intra}^s(y) = f(x^s_y)$
- Denote this special case of the problem as $V^s(X_t, d_t)$

Sequentially Refinable Intra-day Scheduling (SRIS)

$$x_y \triangleq \begin{cases} \operatorname{argmin}\{f(x) : x \in \mathbb{Z}_+^n\} & \text{if } y = y^* = |\operatorname{argmin}\{f(x) : x \in \mathbb{Z}_+^n\}| \\ \operatorname{argmin}\{f(x) : x \in \mathbb{Z}_+^n, |x| = y, x \leq x_{y+1}\} & \text{for } b = y^* - 1, y^* - 2, \ldots, 2, 1, 0 \\ \operatorname{argmin}\{f(x) : x \in \mathbb{Z}_+^n, |x| = y, x \geq x_{y-1}\} & \text{for } y = y^* + 1, y^* + 2, \ldots. \end{cases}$$

$\mathtt{intra}^s(y) = f(x_y^s)$

$\mathtt{intra}^u(y) = \min\{f(x) : x \in \mathbb{Z}_+^n, |x| = y\}$



Theorem

If $f : \mathbb{Z}_+^n \to \mathbb{R}$ is multimodular, then $\mathtt{intra}^s(y)$ can be computed in polynomial time for all y. Moreover, $\mathtt{intra}^s(\cdot)$ is convex and $V^s(\cdot,\cdot)$ is computationally tractable.

Sequentially Refinable Intra-day Scheduling (SRIS)

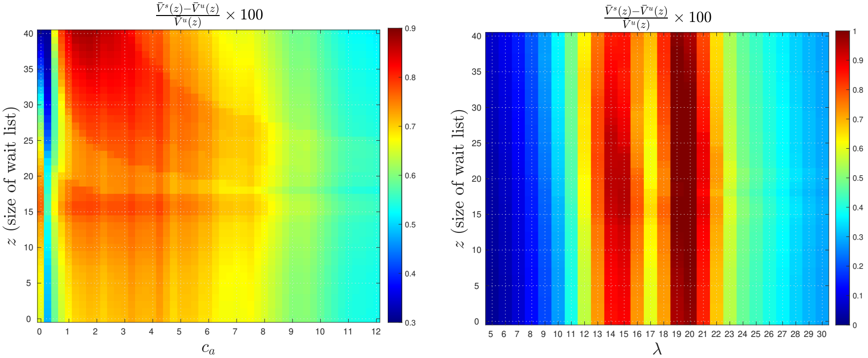
Theorem

$V^u(\cdot,\cdot) \leq V(\cdot,\cdot) \leq V^s(\cdot,\cdot)$

- UIS provides a lower bound through $V^u(\cdot,\cdot)$
- UIS is infeasible for the original, but computationally tractable
- SRIS provides an upper bound through $V^s(\cdot,\cdot)$
- SRIS is feasible and thus a heuristic for the original & computationally tractable
- % optimality gap for SRIS is bounded from above by

$$\frac{V_t^s(\cdot,\cdot) - V_t^u(\cdot,\cdot)}{V_t^u(\cdot,\cdot)} \times 100\%.$$

computational experiments

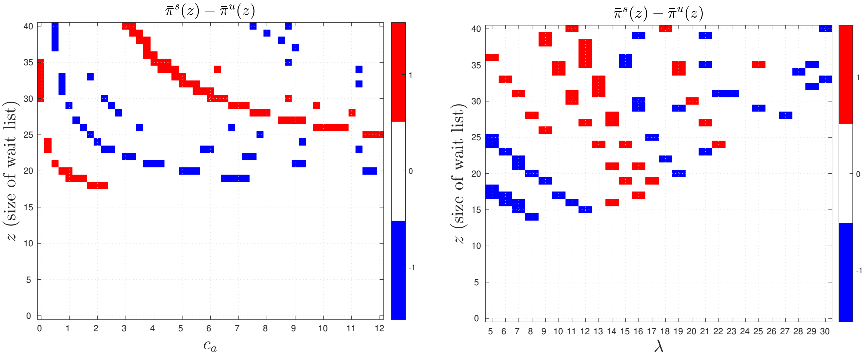


(a) $\lambda = 18$, c_a varies   (b) $c_a = 3$, λ varies

Figure: upper-bound on the % optimality gap of SRIS

input: $d \sim$ Poisson(λ), $N = 8$ hrs, $k = 15$ mins, $p = 0.8$, $c_i = 1$, $c_o = 1$, $c_w = 0.1$, $R \sim$ BetaBin($90, \alpha, \beta$) with α & β such that $m = 30$ mins & $\sigma m^{-1} = 0.4$, U $= 0$, $\gamma = 0.975$.

computational experiments

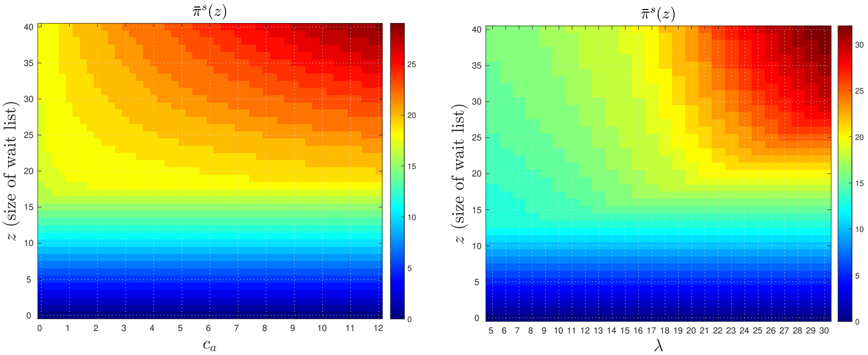


(a) $\lambda = 18$, c_a varies (b) $c_a = 3$, λ varies

Figure: Difference between optimal controls for SRIS and UIS

input: $d \sim \text{Poisson}(\lambda)$, $N = 8$ hrs, $k = 15$ mins, $p = 0.8$, $c_i = 1$, $c_o = 1$, $c_w = 0.1$, $R \sim \text{BetaBin}(90, \alpha, \beta)$ with α & β such that $m = 30$ mins & $\sigma m^{-1} = 0.4$, U = 0, $\gamma = 0.975$.

computational experiments



(a) $\lambda = 18$, c_a varies

(b) $c_a = 3$, λ varies

Figure: Optimal controls for SRIS

input: $d \sim$ Poisson(λ), $N = 8$ hrs, $k = 15$ mins, $p = 0.8$, $c_i = 1$, $c_o = 1$, $c_w = 0.1$, $R \sim$ BetaBin(90, α, β) with α & β such that $m = 30$ mins & $\sigma m^{-1} = 0.4$, U $= 0$, $\gamma = 0.975$.

conclusion

- Dynamic inter-day and intra-day scheduling problem had been a complex open problem, analytically and computationally intractable.
- We developed a dynamic programming framework
- We proved novel theoretical results in discrete convex analysis
- Theoretical lower and upper bounds
- Computationally efficient heuristic solution with a theoretically guaranteed optimality gap
- Optimality gap numerically less than 1%

thank you

- thank you