Polyamorous Scheduling

Sebastian Wild

joint work with Leszek Gąsieniec and Ben Smith

UNIVERSITY OF LIVERPOOL

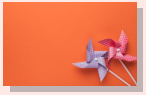
FUN with Algorithms 2024

Outline











1 Introduction

2 Hardness

3 Approximation

4 Density

5 Conclusion



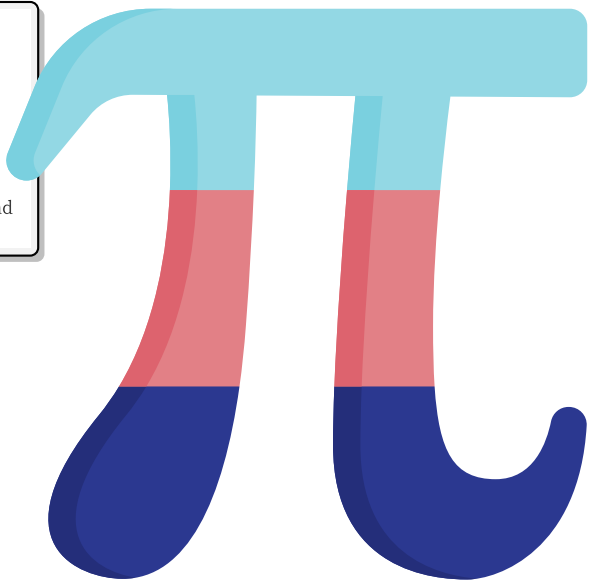
1 Introduction

Polyamory

polyamory
society

Written by Alison Eldridge
Fact-checked by The Editors of Encyclopaedia Britannica
Last Updated: Apr 30, 2024 • Article History

Polyamory, having or desiring multiple intimate relationships at the same time with the full knowledge and consent of all parties involved.

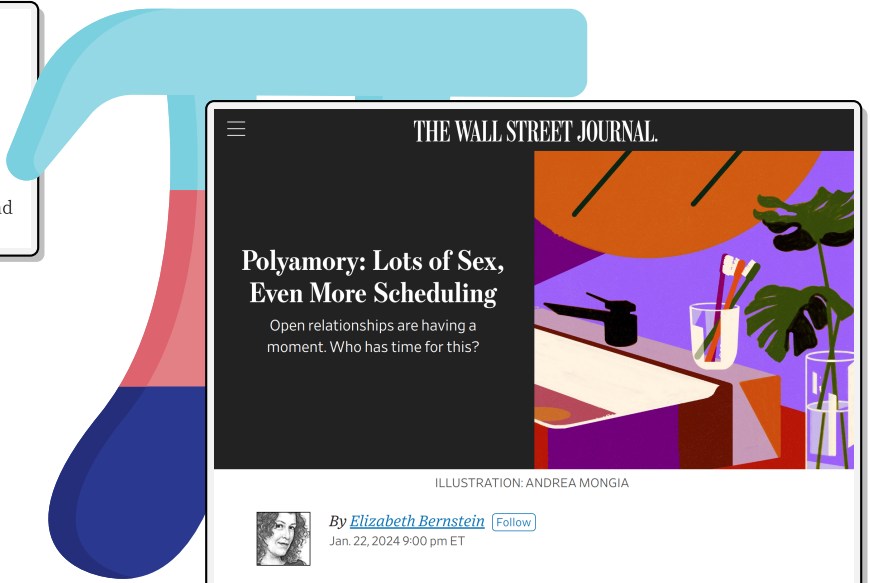


Polyamory

polyamory
society

Written by Alison Eldridge
Fact-checked by The Editors of Encyclopaedia Britannica
Last Updated: Apr 30, 2024 • Article History

Polyamory, having or desiring multiple intimate relationships at the same time with the full knowledge and consent of all parties involved.



THE WALL STREET JOURNAL.

Polyamory: Lots of Sex, Even More Scheduling

Open relationships are having a moment. Who has time for this?

ILLUSTRATION: ANDREA MONGIA

By Elizabeth Bernstein Follow
Jan. 22, 2024 9:00 pm ET

Polyamory

polyamory
society

Written by Alison Eldridge
Fact-checked by The Editors of Encyclopaedia Britannica
Last Updated: Apr 30, 2024 • Article History

Polyamory, having or desiring multiple intimate relationships at the same time with the full knowledge and consent of all parties involved.





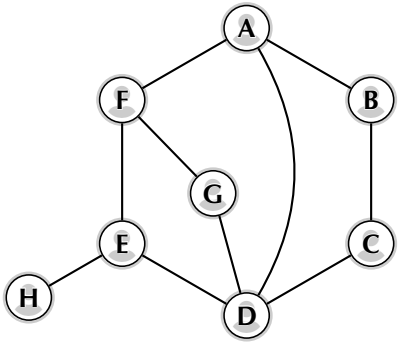
Poly Scheduling

Given:

- **People** and pairwise **Relationships**
  - ⇝ undirected graph (P, R)
- each relationship has a *"desire **growth rate**"*
  - ⇝ edge weights $g : R \to \mathbb{R}_{>0}$
    pain of separation grows by $g(e)$ each day

Goal:

- **perpetual schedule** $S : \mathbb{N}_0 \to 2^R$ that minimizes *"heat"*:
  maximal desire ever felt *(minimax objective)*
- each person can engage in **one relationship per day**
  - ⇝ every day t, schedule a **matching** $S(t)$



Example polycule with 8 persons:

Adam, Brady, Charlie, Daisy,
Eli, Frankie, Grace, and Holly

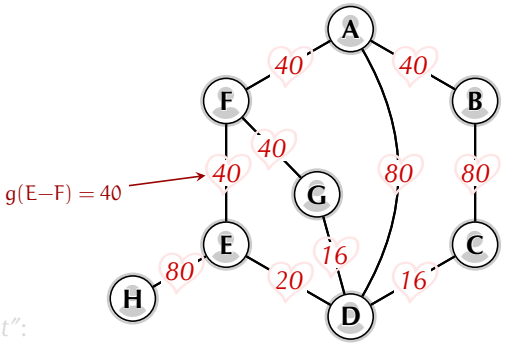
Poly Scheduling

Given:

- **People** and pairwise **Relationships**
  - ⇝ undirected graph (P, R)
- each relationship has a *"desire* ***growth rate****"*
  - ⇝ edge weights $g : R \to \mathbb{R}_{>0}$
    pain of separation grows by $g(e)$ each day

Goal:

- **perpetual schedule** $S : \mathbb{N}_0 \to 2^R$ that minimizes *"heat"*:
  maximal desire ever felt *(minimax objective)*
- each person can engage in **one relationship per day**
  - ⇝ every day t, schedule a **matching** $S(t)$



Example polycule with 8 persons:

Adam, Brady, Charlie, Daisy,
Eli, Frankie, Grace, and Holly

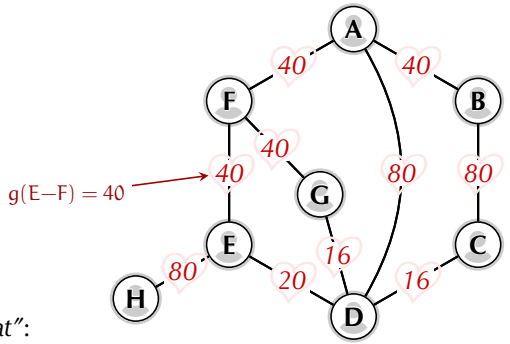
Poly Scheduling

Given:

- **People** and pairwise **Relationships**
  - ⇝ undirected graph (P, R)
- each relationship has a *"desire **growth rate**"*
  - ⇝ edge weights $g : R \to \mathbb{R}_{>0}$
    pain of separation grows by $g(e)$ each day

Goal:

- **perpetual schedule** $S : \mathbb{N}_0 \to 2^R$ that minimizes *"heat"*:
  maximal desire ever felt *(minimax objective)*
- each person can engage in **one relationship per day**
  - ⇝ every day t, schedule a **matching** $S(t)$



Example polycule with 8 persons:

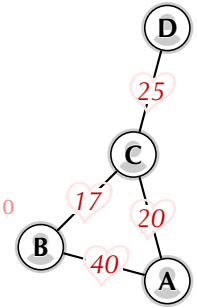
Adam, Brady, Charlie, Daisy,
Eli, Frankie, Grace, and Holly

Poly Scheduling – Small Example

Day 0

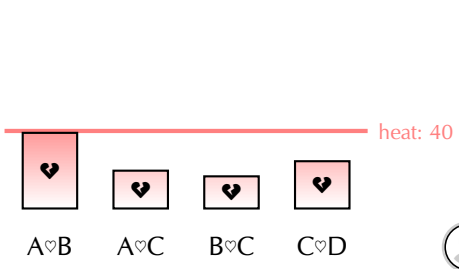
heat: 0

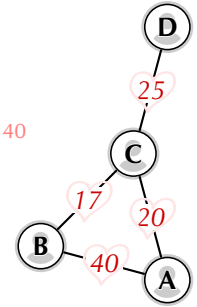
A♡B A♡C B♡C C♡D



Poly Scheduling – Small Example

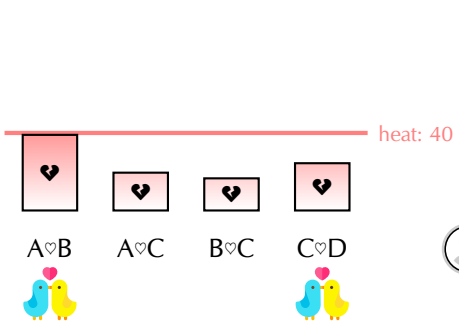
Day 0

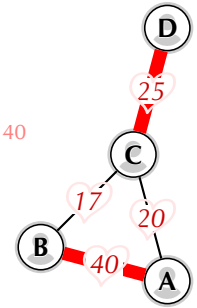




Poly Scheduling – Small Example

Day 0



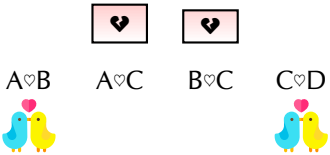


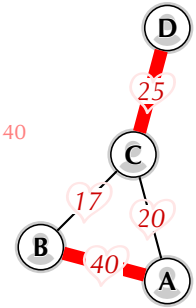
Day	A♡B	A♡C	B♡C	C♡D
0	❤			❤

Poly Scheduling – Small Example

Day 0

heat: 40

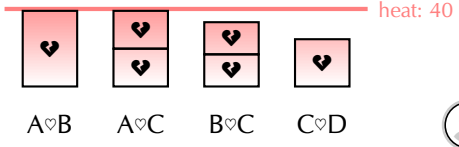


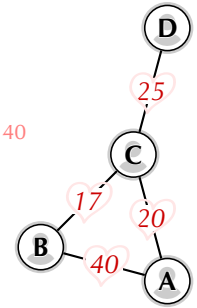


Day	A♡B	A♡C	B♡C	C♡D
0	❤			❤

Poly Scheduling – Small Example

Day 1

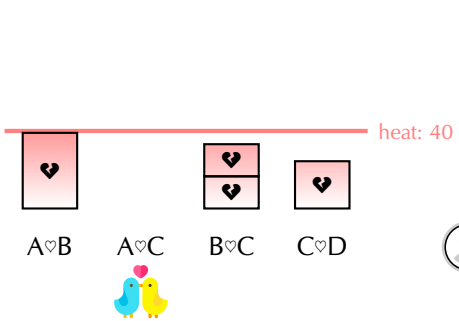


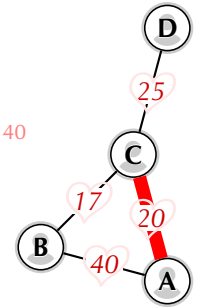


Day	A♡B	A♡C	B♡C	C♡D
0	❤			❤

Poly Scheduling – Small Example

Day 1

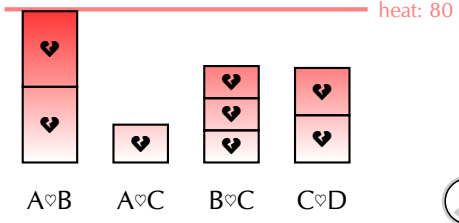


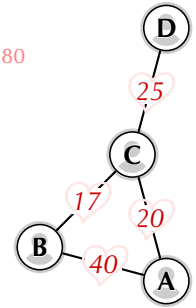


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		

Poly Scheduling – Small Example

Day 2

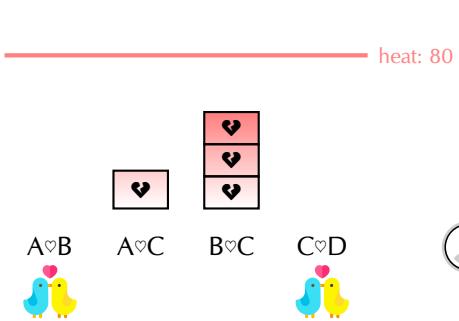


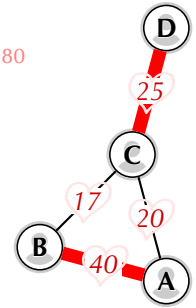


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		

Poly Scheduling – Small Example

Day 2

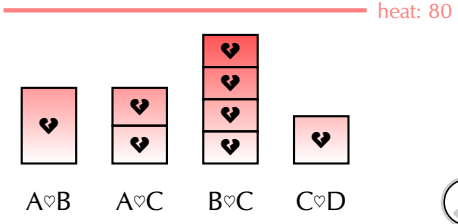


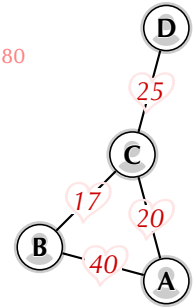


Day	A♡B	A♡C	B♡C	C♡D
0	❤			❤
1		❤		
2	❤			❤

Poly Scheduling – Small Example

Day 3

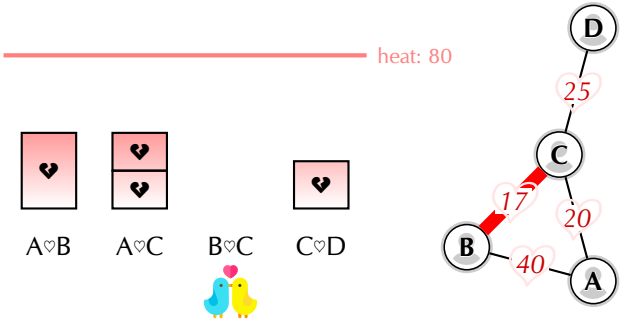




Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥

Poly Scheduling – Small Example

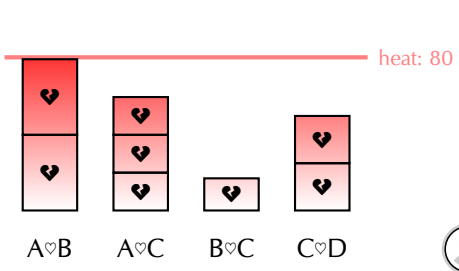
Day 3

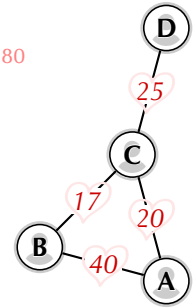


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	

Poly Scheduling – Small Example

Day 4

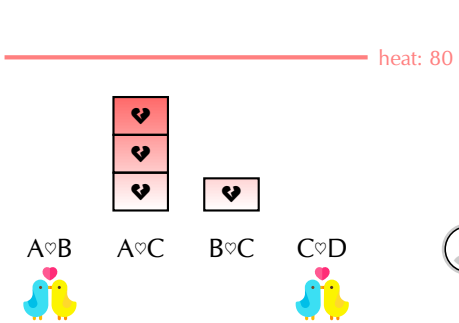


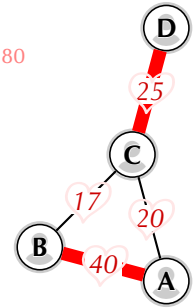


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	

Poly Scheduling – Small Example

Day 4

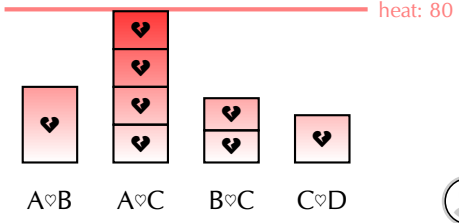


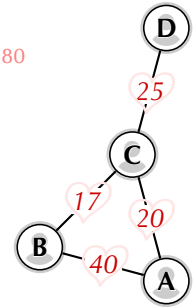


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥

Poly Scheduling – Small Example

Day 5

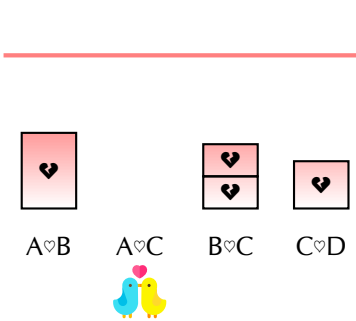


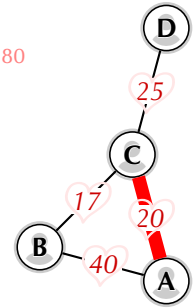


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥

Poly Scheduling – Small Example

Day 5

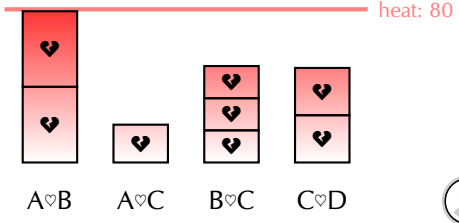


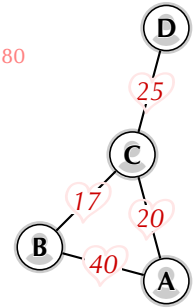


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥
5		♥		

Poly Scheduling – Small Example

Day 6

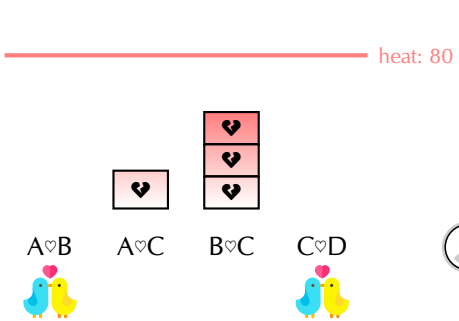


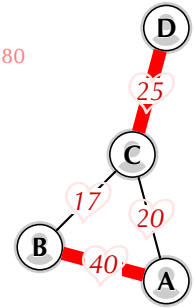


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥
5		♥		

Poly Scheduling – Small Example

Day 6

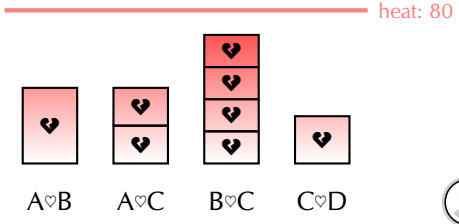


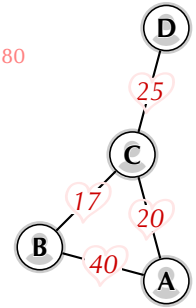


Day	A♡B	A♡C	B♡C	C♡D
0	❤			❤
1		❤		
2	❤			❤
3			❤	
4	❤			❤
5		❤		
6	❤			❤

Poly Scheduling – Small Example

Day 7

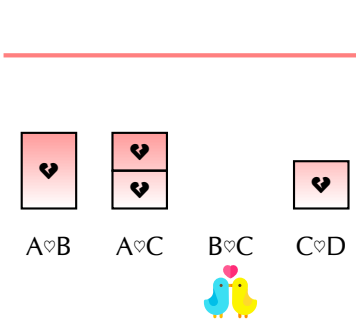


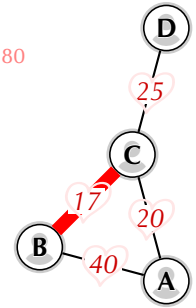


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥
5		♥		
6	♥			♥

Poly Scheduling – Small Example

Day 7

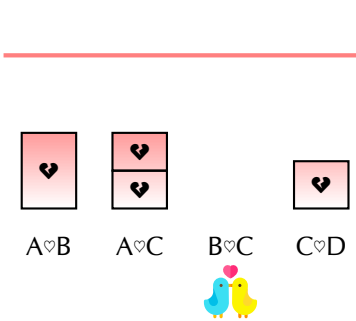


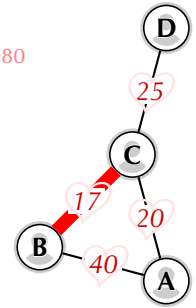


Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥
5		♥		
6	♥			♥
7			♥	

Poly Scheduling – Small Example

Day 7





Day	A♡B	A♡C	B♡C	C♡D
0	♥			♥
1		♥		
2	♥			♥
3			♥	
4	♥			♥
5		♥		
6	♥			♥
7			♥	

$S^* = (\{AB, CD\}, \{AC\}, \{AB, CD\}, \{BC\})$
with heat $h(S) = 80$

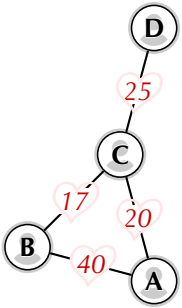
follows from finite state space

Observation: schedules without loss of generality **periodic**

$\exists T \forall t\ S(t) = S(t+T)$

Decision problem

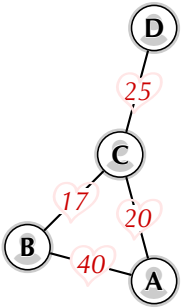
Optimization input:



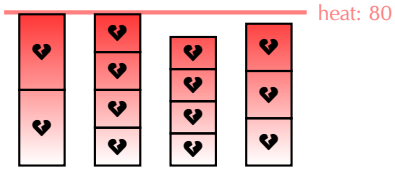
Optimization Poly Scheduling

Decision problem

Optimization input:

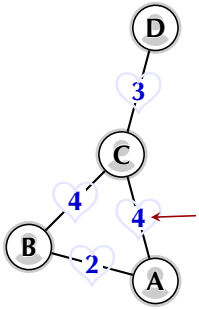


Maximal frequencies
to remain below **heat 80**:



$$\text{frequency} = \left\lfloor \frac{\text{heat}}{\text{growth rate}} \right\rfloor$$

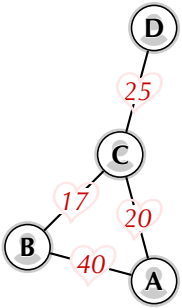
Decision instance:



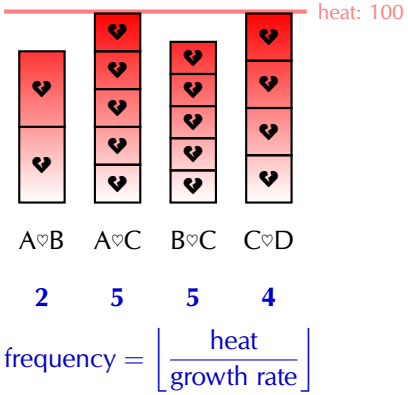
A and C must meet (at least) once in every 4-day period

Decision problem

Optimization input:

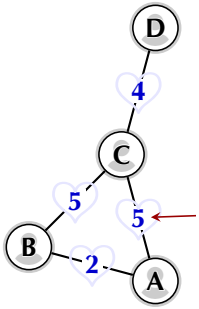


Maximal frequencies
to remain below **heat 100**:



$$\text{frequency} = \left\lfloor \frac{\text{heat}}{\text{growth rate}} \right\rfloor$$

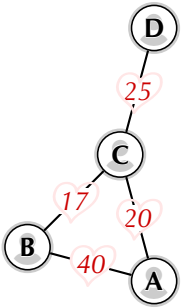
Decision instance:



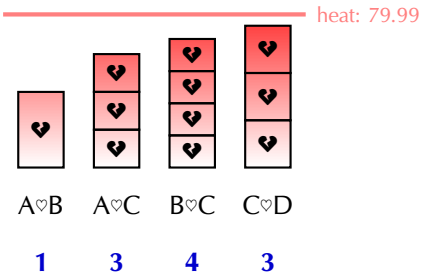
A and C must meet (at least) once in every 5-day period

Decision problem

Optimization input:

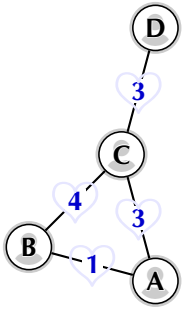


Maximal frequencies
to remain below **heat 79.99**:



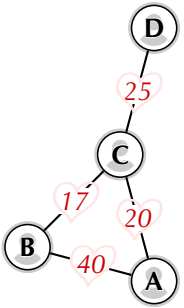
$$\text{frequency} = \left\lfloor \frac{\text{heat}}{\text{growth rate}} \right\rfloor$$

Decision instance:



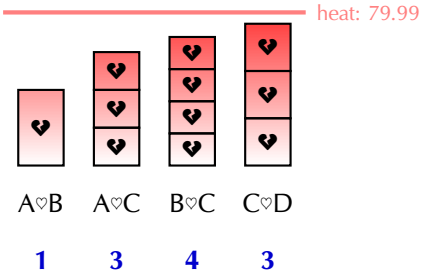
Decision problem

Optimization input:



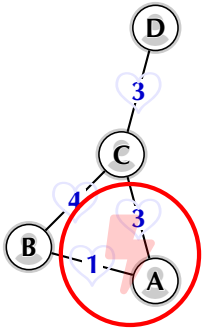
Optimization Poly Scheduling

Maximal frequencies
to remain below **heat 79.99**:



$$\text{frequency} = \left\lfloor \frac{\text{heat}}{\text{growth rate}} \right\rfloor$$

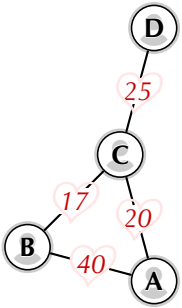
Decision instance:



Decision Poly Scheduling

Decision problem

Optimization input:



Maximal frequencies

For this instance:

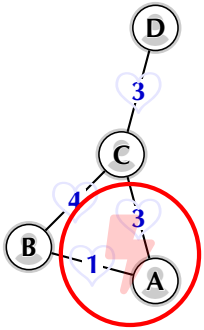
- schedule S^* has heat 80
  $S^* = (\{AB, CD\}, \{AC\}, \{AB, CD\}, \{BC\})$
- heat $80 - \varepsilon$ leaves A unschedulable

$\rightsquigarrow$ optimal heat $h^* = 80$

1 **3** **4** **3**

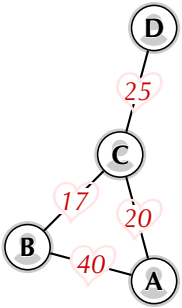
$$\text{frequency} = \left\lfloor \frac{\text{heat}}{\text{growth rate}} \right\rfloor$$

Decision instance:



Decision problem

Optimization input:



Maximal frequencies

For this instance:

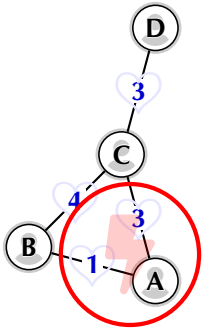
- schedule S^* has heat 80
  $S^* = (\{AB, CD\}, \{AC\}, \{AB, CD\}, \{BC\})$
- heat $80 - \varepsilon$ leaves A unschedulable

$\rightsquigarrow$ optimal heat $\hbar^* = 80$

1 **3** **4** **3**

$$\text{frequency} = \left\lfloor \frac{\text{heat}}{\text{growth rate}} \right\rfloor$$

Decision instance:



Outline



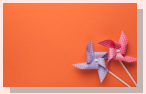
1 Introduction



2 Hardness



3 Approximation



4 Density



5 Conclusion



2 Hardness

Unweighted case

Do poly people just make their life complicated – or are they facing a ↓ *computationally hard problem?*

- Let's consider a simple special case: **Non-hierarchical Poly Scheduling**
  - (optimization): all edges have weight 1
  - (decision): all edges have the same frequency **f**

⇝ feasible schedule for decision problem $\iff$ schedule of heat $\leqslant$ f in optimization problem

⇝ Equivalent to **Edge Coloring** (a. k. a. **Chromatic Index**)

minimal #colors χ_1 to color all edges s.t. no 2 incident to same vertex have same color

- **Edge Coloring** $\Longrightarrow$ **Poly Schedule**:
  Schedule = Round Robin of χ_1 color classes
  heat of schedule χ_1.
- **Edge Coloring** $\Longleftarrow$ **Poly Schedule**:
  Color = first day edge is scheduled
  Heat h ⇝ at most h colors

Unweighted case

Do poly people just make their life complicated – or are they facing a computationally hard problem?

- Let's consider a simple special case: **Non-hierarchical Poly Scheduling**
  - (optimization): all edges have weight 1
  - (decision): all edges have the same frequency **f**
  - ⇝ feasible schedule for decision problem $\iff$ schedule of heat $\leqslant$ f in optimization problem

⇝ Equivalent to **Edge Coloring** (a. k. a. **Chromatic Index**)

minimal #colors χ_1 to color all edges s.t. no 2 incident to same vertex have same color

- **Edge Coloring $\Longrightarrow$ Poly Schedule**:
  Schedule = Round Robin of χ_1 color classes
  heat of schedule χ_1.
- **Edge Coloring $\Longleftarrow$ Poly Schedule**:
  Color = first day edge is scheduled
  Heat h ⇝ at most h colors

Unweighted case

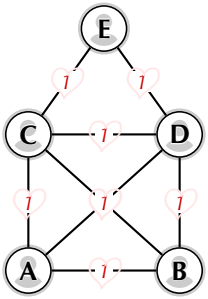
Do poly people just make their life complicated – or are they facing a computationally hard problem?

- Let's consider a simple special case: **Non-hierarchical Poly Scheduling**
  - (optimization): all edges have weight 1
  - (decision): all edges have the same frequency **f**

  ⇝ feasible schedule for decision problem ⟺ schedule of heat ⩽ f in optimization problem

⇝ Equivalent to **Edge Coloring** (a. k. a. **Chromatic Index**)

minimal #colors χ_1 to color all edges s.t. no 2 incident to same vertex have same color



- **Edge Coloring ⟹ Poly Schedule**:
  Schedule = Round Robin of χ_1 color classes
  heat of schedule χ_1.
- **Edge Coloring ⟸ Poly Schedule**:
  Color = first day edge is scheduled
  Heat h ⇝ at most h colors

Unweighted case

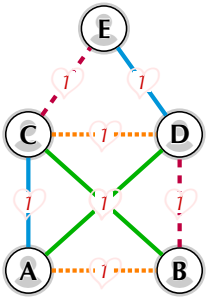
Do poly people just make their life complicated – or are they facing a computationally hard problem?

- Let's consider a simple special case: **Non-hierarchical Poly Scheduling**
  - (optimization): all edges have weight 1
  - (decision): all edges have the same frequency **f**

  ⇝ feasible schedule for decision problem ⟺ schedule of heat ⩽ f in optimization problem

⇝ Equivalent to **Edge Coloring** (a. k. a. **Chromatic Index**)

minimal #colors χ_1 to color all edges s.t. no 2 incident to same vertex have same color



- **Edge Coloring ⟹ Poly Schedule**:
  Schedule = Round Robin of χ_1 color classes
  heat of schedule χ_1.
- **Edge Coloring ⟸ Poly Schedule**:
  Color = first day edge is scheduled
  Heat h ⇝ at most h colors

Unweighted case

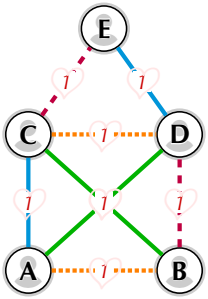
Do poly people just make their life complicated – or are they facing a ↓ *computationally hard problem?*

- Let's consider a simple special case: **Non-hierarchical Poly Scheduling**
  - (optimization): all edges have weight 1
  - (decision): all edges have the same frequency **f**

  ⇝ feasible schedule for decision problem ⟺ schedule of heat ⩽ f in optimization problem

⇝ Equivalent to **Edge Coloring** (a. k. a. **Chromatic Index**)

↖ minimal #colors χ_1 to color all edges s.t. no 2 incident to same vertex have same color



- **Edge Coloring** ⟹ **Poly Schedule**:
  Schedule = Round Robin of χ_1 color classes
  heat of schedule χ_1.
- **Edge Coloring** ⟸ **Poly Schedule**:
  Color = first day edge is scheduled
  Heat h ⇝ at most h colors

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. $\Delta + 1$ **coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (arxiv:2405.15449)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. $\Delta + 1$ **coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (arxiv:2405.15449)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. **$\Delta + 1$ coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (`arxiv:2405.15449`)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. $\Delta + 1$ **coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (`arxiv:2405.15449`)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. **$\Delta + 1$ coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (`arxiv:2405.15449`)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Implications for
(Non-hierarchical) Poly Scheduling

1. Decision **Poly Scheduling** is **NP-hard**.
2. Optimization **Poly Scheduling** does **not** admit poly-time $(1+\delta)$**-approx** for $\delta < 1/3$ unless P = NP.
   $\rightsquigarrow$ such an approximation would decide Δ vs $\Delta + 1$ for $\Delta = 3$.

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. **$\Delta + 1$ coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (`arxiv:2405.15449`)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Implications for
(Non-hierarchical) Poly Scheduling

1. Decision **Poly Scheduling** is **NP-hard**.
2. Optimization **Poly Scheduling** does **not** admit poly-time $(1+\delta)$-**approx** for $\delta < 1/3$ unless P = NP.
   $\rightsquigarrow$ such an approximation would decide Δ vs $\Delta + 1$ for $\Delta = 3$.

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. **$\Delta + 1$ coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (`arxiv:2405.15449`)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

Implications for
(Non-hierarchical) Poly Scheduling

1. Decision **Poly Scheduling** is **NP-hard**.
2. Optimization **Poly Scheduling** does **not** admit poly-time $(1+\delta)$**-approx** for $\delta < 1/3$ unless P = NP.
   $\rightsquigarrow$ such an approximation would decide Δ vs $\Delta + 1$ for $\Delta = 3$.

Chromatic Index Problem

- Recall: χ_1 = chromatic index of graph G
  = #colors in proper edge coloring

$\rightsquigarrow$ Obivous: $\chi_1 \geqslant \Delta$ ← maximum degree in G

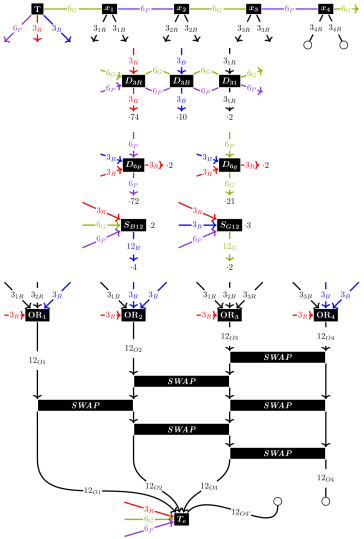
- Vizing's Theorem (1965): $\chi_1 \leqslant \Delta + 1$

1. **$\Delta + 1$ coloring** can always be found in **efficiently**.
   running time very recently improved to $\tilde{O}(mn^{1/3})$
   (`arxiv:2405.15449`)
2. Computing χ_1 **exactly** (Δ or $\Delta + 1$!) is **NP-hard**
   - even for **3-regular** graphs! (Holyer 1981)

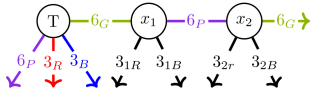
Implications for
(Non-hierarchical) Poly Scheduling

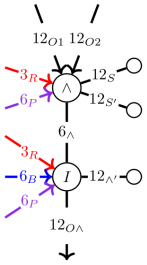
1. Decision **Poly Scheduling** is **NP-hard**.
2. Optimization **Poly Scheduling** does **not** admit poly-time **$(1+\delta)$-approx** for $\delta < 1/3$ unless P = NP.
   $\rightsquigarrow$ such an approximation would decide Δ vs $\Delta + 1$ for $\Delta = 3$.

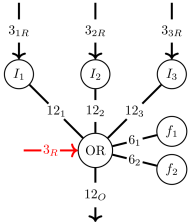
Bonus: SAT-Reduction

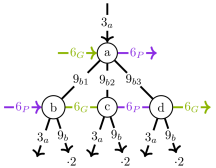


- Alternative hardness of approximation: Reduction from (decision version of) MAX-3SAT
- Given formula φ over $\mathfrak{m}$ clauses and $\mathfrak{n}$ variables, can $\geqslant k$ clauses be satisfied simultaneously?
- Key idea: relation scheduled on odd (even) days = True (False)
- doesn't quite work,
  but with slots module 6 it does.









Outline



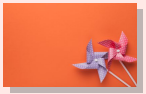
1 Introduction



2 Hardness



3 Approximation



4 Density



5 Conclusion



3 Approximation

Edge-Coloring Approximation

- **Recall:** Can compute $(\Delta+1)$-coloring efficiently

$\rightsquigarrow$ $\dfrac{\Delta+1}{\Delta}$-approx for ***unweighted*** instances

- any valid schedule must reach heat $h \geqslant \Delta$ on a degree-Δ vertex

$\rightsquigarrow$ in general: $\dfrac{\Delta+1}{\Delta} \cdot \dfrac{g_{\max}}{g_{\min}}$-approx and (simultaneously) $(\Delta+1)$-approx

unbounded

could be $\Omega(n)$ ...

- any schedule reaches heat $h \geqslant \Delta \cdot g_{\min}$ on a degree-Δ vertex
- any schedule reaches heat $h \geqslant g_{\max}$
- Colors Round Robin schedule achieves $h \leqslant (\Delta+1) g_{\max}$

Edge-Coloring Approximation

- **Recall:** Can compute $(\Delta+1)$-coloring efficiently

$\rightsquigarrow$ $\dfrac{\Delta+1}{\Delta}$-approx for ***unweighted*** instances

- any valid schedule must reach heat $h \geqslant \Delta$ on a degree-Δ vertex

$\rightsquigarrow$ in general: $\dfrac{\Delta+1}{\Delta} \cdot \dfrac{g_{\max}}{g_{\min}}$-approx and (simultaneously) $(\Delta+1)$-approx

unbounded

could be $\Omega(n)$ ...

- any schedule reaches heat $h \geqslant \Delta \cdot g_{\min}$ on a degree-Δ vertex
- any schedule reaches heat $h \geqslant g_{\max}$
- Colors Round Robin schedule achieves $h \leqslant (\Delta+1) g_{\max}$

Edge-Coloring Approximation

- **Recall:** Can compute $(\Delta+1)$-coloring efficiently

$\rightsquigarrow$ $\dfrac{\Delta+1}{\Delta}$-approx for ***unweighted*** instances

- any valid schedule must reach heat $h \geqslant \Delta$ on a degree-Δ vertex



$\rightsquigarrow$ in general: $\dfrac{\Delta+1}{\Delta} \cdot \dfrac{g_{\max}}{g_{\min}}$-approx and (simultaneously) $(\Delta+1)$-approx

unbounded

could be $\Omega(n)$ ...

- any schedule reaches heat $h \geqslant \Delta \cdot g_{\min}$ on a degree-Δ vertex
- any schedule reaches heat $h \geqslant g_{\max}$
- Colors Round Robin schedule achieves $h \leqslant (\Delta+1) g_{\max}$

Edge-Coloring Approximation

- **Recall:** Can compute $(\Delta+1)$-coloring efficiently

$\rightsquigarrow$ $\frac{\Delta+1}{\Delta}$-approx for ***unweighted*** instances

- any valid schedule must reach heat $h \geqslant \Delta$ on a degree-Δ vertex



$\rightsquigarrow$ in general: $\frac{\Delta+1}{\Delta} \cdot \frac{g_{max}}{g_{min}}$-approx and (simultaneously) $(\Delta+1)$-approx

unbounded

could be $\Omega(n)$ ...

- any schedule reaches heat $h \geqslant \Delta \cdot g_{min}$ on a degree-Δ vertex
- any schedule reaches heat $h \geqslant g_{max}$
- Colors Round Robin schedule achieves $h \leqslant (\Delta+1) g_{max}$

Edge-Coloring Approximation

- **Recall:** Can compute $(\Delta+1)$-coloring efficiently

$\rightsquigarrow$ $\frac{\Delta+1}{\Delta}$-approx for ***unweighted*** instances

- any valid schedule must reach heat $h \geqslant \Delta$ on a degree-Δ vertex



$\rightsquigarrow$ in general: $\frac{\Delta+1}{\Delta} \cdot \frac{g_{max}}{g_{min}}$-approx and (simultaneously) $(\Delta+1)$-approx

unbounded

could be $\Omega(n)$ ...

- any schedule reaches heat $h \geqslant \Delta \cdot g_{min}$ on a degree-Δ vertex
- any schedule reaches heat $h \geqslant g_{max}$
- Colors Round Robin schedule achieves $h \leqslant (\Delta+1) g_{max}$

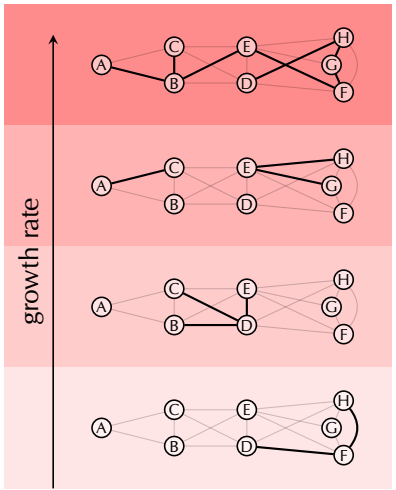
Layering approximation

- Partition edges into exponential **"layers"** by growth rate
  - Layers $i = 0, \dots L-1$ contain edges with $\frac{g_{\max}}{2^{i+1}} \leqslant g(e) \leqslant \frac{g_{\max}}{2^i}$
    Layer L the rest, i.e., $g(e) \leqslant \frac{g_{\max}}{2^L}$
  - ⇝ within one layer $g_{\max}/g_{\min} \leqslant 2$
  - ⇝ Color Round Robin **within layer** gives good approximation.
- Also schedule **layers** Round Robin
- ⇝ Heat of overall schedule $\leqslant \overline{h}$ with Δ_i = max degree in layer i

$$\overline{h} = \max_{i \in [0..L]} (L+1) \cdot (\Delta_i + 1)\frac{g_{\max}}{2^i}$$

- ⇝ each layer (in isolation) induces lower bound, so $h^* \geqslant \underline{h}$ with

$$\underline{h} = \max\left\{ \max_{i \in [0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max} \right\}$$



Layering approximation

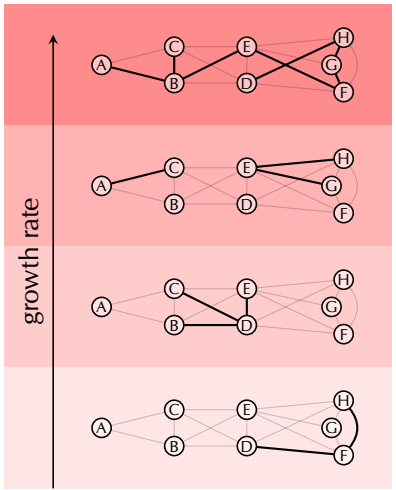
- Partition edges into exponential **"layers"** by growth rate
  - Layers $i = 0, \dots L-1$ contain edges with $\frac{g_{max}}{2^{i+1}} \leqslant g(e) \leqslant \frac{g_{max}}{2^i}$
    Layer L the rest, i.e., $g(e) \leqslant \frac{g_{max}}{2^L}$
  - ⇝ within one layer $g_{max}/g_{min} \leqslant 2$
  - ⇝ Color Round Robin **within layer** gives good approximation.
- Also schedule **layers** Round Robin

⇝ Heat of overall schedule $\leqslant \overline{h}$ with Δ_i = max degree in layer i

$$\overline{h} = \max_{i \in [0..L]} (L+1) \cdot (\Delta_i + 1)\frac{g_{max}}{2^i}$$

⇝ each layer (in isolation) induces lower bound, so $h^* \geqslant \underline{h}$ with

$$\underline{h} = \max\left\{ \max_{i \in [0..L]} \Delta_i \cdot \frac{g_{max}}{2^{i+1}},\ g_{max} \right\}$$



g_{max}

$g_{max}/2$

$g_{max}/2^2$

$g_{max}/2^3$

$g_{max}/2^4$

Layering approximation

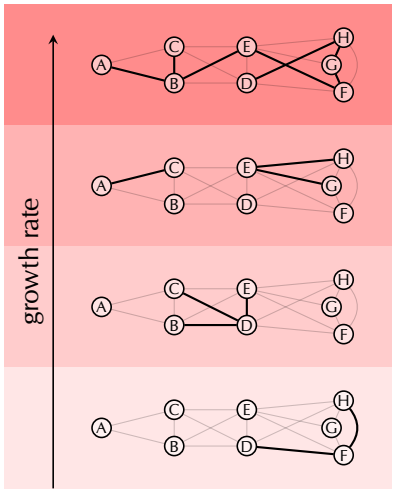
- Partition edges into exponential **"layers"** by growth rate
  - Layers $i = 0, \dots L-1$ contain edges with $\frac{g_{\max}}{2^{i+1}} \leqslant g(e) \leqslant \frac{g_{\max}}{2^i}$
    Layer L the rest, i.e., $g(e) \leqslant \frac{g_{\max}}{2^L}$
  - $\rightsquigarrow$ within one layer $g_{\max}/g_{\min} \leqslant 2$
  - $\rightsquigarrow$ Color Round Robin **within layer** gives good approximation.
- Also schedule **layers** Round Robin

$\rightsquigarrow$ Heat of overall schedule $\leqslant \overline{h}$ with $\quad \Delta_i$ = max degree in layer i

$$\overline{h} = \max_{i \in [0..L]} (L+1) \cdot (\Delta_i + 1)\frac{g_{\max}}{2^i}$$

$\rightsquigarrow$ each layer (in isolation) induces lower bound, so $h^* \geqslant \underline{h}$ with

$$\underline{h} = \max\left\{ \max_{i \in [0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max} \right\}$$



$g_{\max}$
$g_{\max}/2$
$g_{\max}/2^2$
$g_{\max}/2^3$
$g_{\max}/2^4$

Layering approximation

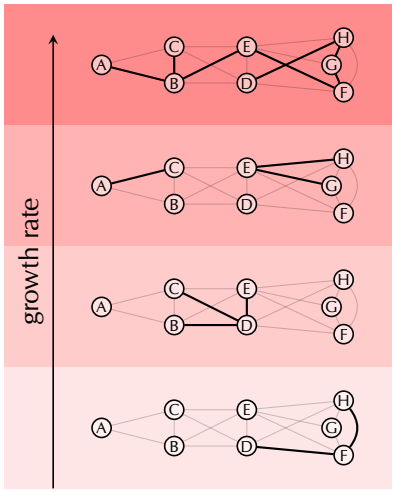
- Partition edges into exponential **"layers"** by growth rate
  - Layers $i = 0, \dots L-1$ contain edges with $\frac{g_{\max}}{2^{i+1}} \leqslant g(e) \leqslant \frac{g_{\max}}{2^i}$
    Layer L the rest, i.e., $g(e) \leqslant \frac{g_{\max}}{2^L}$
  - $\rightsquigarrow$ within one layer $g_{\max}/g_{\min} \leqslant 2$
  - $\rightsquigarrow$ Color Round Robin **within layer** gives good approximation.
- Also schedule **layers** Round Robin

$\rightsquigarrow$ Heat of overall schedule $\leqslant \overline{h}$ with $\quad \Delta_i$ = max degree in layer i

$$\overline{h} = \max_{i \in [0..L]} (L+1) \cdot (\Delta_i + 1)\frac{g_{\max}}{2^i}$$

$\rightsquigarrow$ each layer (in isolation) induces lower bound, so $h^* \geqslant \underline{h}$ with

$$\underline{h} = \max\left\{ \max_{i \in [0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max} \right\}$$



$g_{\max}$
$g_{\max}/2$
$g_{\max}/2^2$
$g_{\max}/2^3$
$g_{\max}/2^4$

Layering approximation [2]

- Distinguish cases where max in $\overline{h}$ is attained.

1. **Layer $i < L$ dominates**: $\overline{h} = (L+1)(\Delta_i)\frac{g_{\max}}{2^i}$
   - we by definition: $\underline{h} \geqslant \Delta_i \cdot \frac{g_{\max}}{2^{i+1}}$
   - $\rightsquigarrow$ $3(L+1)$-approximation (for $\Delta_i \geqslant 2$; otherwise trivial)
2. **Layer L dominates** (a.k.a. otherwise)
   - $\rightsquigarrow$ $\overline{h} = (L+1)(\Delta_L+1)\frac{g_{\max}}{2^L}$
   - by definition: $\underline{h} \geqslant g_{\max}$
   - $\rightsquigarrow$ $(L+1)(\Delta_L+1)2^{-L}$-approximation

- Equating both cases suggests $L = \log_2(\Delta_L+1) - O(1)$

$\rightsquigarrow$ $3\lceil\log_2(\Delta+1)\rceil$-approximation

Recall:

- $\overline{h} = \max_{i\in[0..L]} (L+1)\cdot(\Delta_i+1)\frac{g_{\max}}{2^i}$
- $\underline{h} = \max\left\{\max_{i\in[0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max}\right\}$

Layering approximation [2]

- Distinguish cases where max in $\overline{h}$ is attained.

1. **Layer $i < L$ dominates**: $\overline{h} = (L+1)(\Delta_i)\frac{g_{\max}}{2^i}$
   - we by definition: $\underline{h} \geqslant \Delta_i \cdot \frac{g_{\max}}{2^{i+1}}$
   - $\rightsquigarrow$ $3(L+1)$-approximation (for $\Delta_i \geqslant 2$; otherwise trivial)
2. **Layer L dominates** (a.k.a. otherwise)
   - $\rightsquigarrow$ $\overline{h} = (L+1)(\Delta_L+1)\frac{g_{\max}}{2^L}$
   - by definition: $\underline{h} \geqslant g_{\max}$
   - $\rightsquigarrow$ $(L+1)(\Delta_L+1)2^{-L}$-approximation

- Equating both cases suggests $L = \log_2(\Delta_L+1) - O(1)$

$\rightsquigarrow$ $3\lceil\log_2(\Delta+1)\rceil$-approximation

Recall:

- $\overline{h} = \max_{i\in[0..L]} (L+1)\cdot(\Delta_i+1)\frac{g_{\max}}{2^i}$
- $\underline{h} = \max\left\{\max_{i\in[0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max}\right\}$

Layering approximation [2]

- Distinguish cases where max in $\overline{h}$ is attained.

1. **Layer $i < L$ dominates**: $\overline{h} = (L+1)(\Delta_i)\frac{g_{\max}}{2^i}$
   - we by definition: $\underline{h} \geqslant \Delta_i \cdot \frac{g_{\max}}{2^{i+1}}$
   - $\rightsquigarrow$ $\boxed{3(L+1)\text{-approximation}}$ (for $\Delta_i \geqslant 2$; otherwise trivial)
2. **Layer L dominates** (a.k.a. otherwise)
   - $\rightsquigarrow$ $\overline{h} = (L+1)(\Delta_L+1)\frac{g_{\max}}{2^L}$
   - by definition: $\underline{h} \geqslant g_{\max}$
   - $\rightsquigarrow$ $\boxed{(L+1)(\Delta_L+1)2^{-L}\text{-approximation}}$

- Equating both cases suggests $L = \log_2(\Delta_L+1) - O(1)$

$\rightsquigarrow$ $3\lceil\log_2(\Delta+1)\rceil$-approximation

Recall:

- $\overline{h} = \max_{i\in[0..L]} (L+1)\cdot(\Delta_i+1)\frac{g_{\max}}{2^i}$
- $\underline{h} = \max\left\{\max_{i\in[0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max}\right\}$

Layering approximation [2]

- Distinguish cases where max in $\overline{h}$ is attained.

1. **Layer $i < L$ dominates**: $\overline{h} = (L+1)(\Delta_i)\frac{g_{\max}}{2^i}$
   - we by definition: $\underline{h} \geqslant \Delta_i \cdot \frac{g_{\max}}{2^{i+1}}$
   - $\rightsquigarrow$ $\boxed{3(L+1)\text{-approximation}}$ (for $\Delta_i \geqslant 2$; otherwise trivial)
2. **Layer L dominates** (a.k.a. otherwise)
   - $\rightsquigarrow$ $\overline{h} = (L+1)(\Delta_L+1)\frac{g_{\max}}{2^L}$
   - by definition: $\underline{h} \geqslant g_{\max}$
   - $\rightsquigarrow$ $\boxed{(L+1)(\Delta_L+1)2^{-L}\text{-approximation}}$

- Equating both cases suggests $L = \log_2(\Delta_L+1) - O(1)$

$\rightsquigarrow$ $3\lceil\log_2(\Delta+1)\rceil$-approximation

Recall:

- $\overline{h} = \max_{i\in[0..L]} (L+1)\cdot(\Delta_i+1)\frac{g_{\max}}{2^i}$
- $\underline{h} = \max\left\{\max_{i\in[0..L]} \Delta_i \cdot \frac{g_{\max}}{2^{i+1}},\ g_{\max}\right\}$

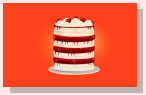
Outline



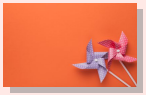
1 Introduction



2 Hardness



3 Approximation



4 Density



5 Conclusion

4 Density



Special Case: Pinwheel Scheduling

- The **Pinwheel Scheduling Problem (Holte et.al., 1989)**:
  - Given k tasks (each with **frequency** $a_i \in \mathbb{N}$)
    Is there a perpetual schedule for a (single) agent who can perform one task per day such that every section of length a_i contains at least one instance of task i?
- Example:
  - $A = \underbrace{[3, 4, 5, 8]}_{k=4}$

Task	1	2	3	4
Frequency	3	4	5	8

  - $S = 1, 2, 4, 1, 3, 2, 1, 3$

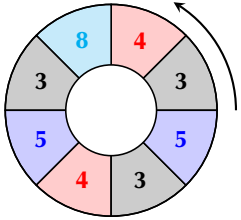
Special Case: Pinwheel Scheduling

- The **Pinwheel Scheduling Problem (Holte et.al., 1989)**:
  - Given k tasks (each with **frequency** $a_i \in \mathbb{N}$)
    Is there a perpetual schedule for a (single) agent who can perform one task per day such that every section of length a_i contains at least one instance of task i?

- Example:
  - $A = \underbrace{[3, 4, 5, 8]}_{k=4}$

Task	1	2	3	4
Frequency	3	4	5	8

  - $S = 1, 2, 4, 1, 3, 2, 1, 3$



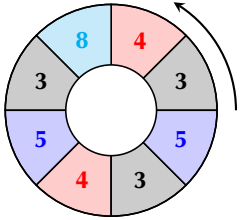
Special Case: Pinwheel Scheduling

- The **Pinwheel Scheduling Problem (Holte et.al., 1989)**:
  - Given k tasks (each with **frequency** $a_i \in \mathbb{N}$)
    Is there a perpetual schedule for a (single) agent who can perform one task per day such that every section of length a_i contains at least one instance of task i?

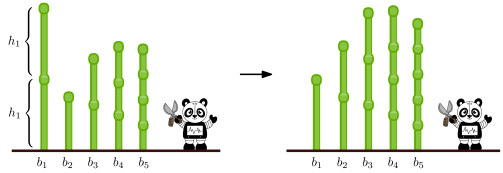
- Example:
  - $A = \underbrace{[3, 4, 5, 8]}_{k=4}$

Task	1	2	3	4
Frequency	3	4	5	8

  - $S = 1, 2, 4, 1, 3, 2, 1, 3$

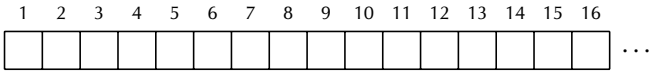


Decision version of
Bamboo Garden Trimming
(Bilò et al., FUN 2020/21)



The 5/6 Density Conjecture

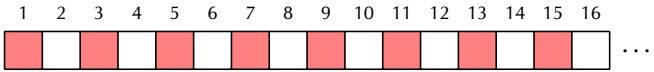
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

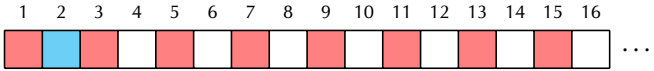
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

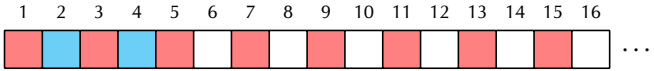
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

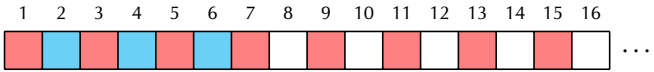
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

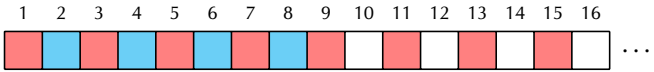
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

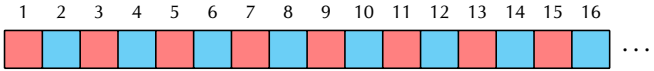
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

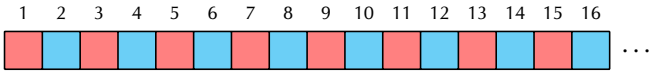
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density Conjecture

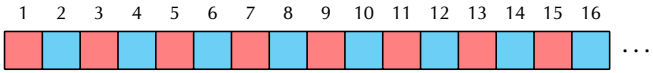
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.

The 5/6 Density ~~Conjecture~~ Theorem

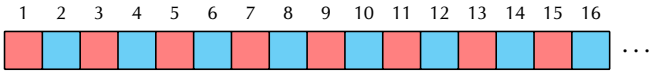
- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.
- BREAKING NEWS Conjecture is proven $\rightsquigarrow$ upcoming STOC (Akitoshi Kawamura, 2024)

The 5/6 Density ~~Conjecture~~ Theorem

- Density $d(A) = \sum_{i=1}^{k} 1/a_i$
- $[2, 3, M]$



- **5/6 Density Conjecture (Chan and Chin, 1993):**
  All Pinwheel scheduling instances where $d \leqslant 5/6$ are schedulable.
- BREAKING NEWS Conjecture is proven $\rightsquigarrow$ upcoming STOC (Akitoshi Kawamura, 2024)

$\rightsquigarrow$ Similar result possible for Poly Scheduling?

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - ⇝ we can spend a fraction y_M of each day on matching M
  - ⇝ without loss of generality: all days the same!

⇝ Best fractional solution given by LP:

$\mathcal{M}$ = set of maximal matchings

$$\begin{aligned} \min \quad & \bar{h} \\ \text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\ & \frac{1}{\sum_{M \in \mathcal{M}: e \in M} y_M} \cdot g_e \leqslant \bar{h} \qquad \forall e \in R \\ & y_M \in [0,1] \qquad \forall M \in \mathcal{M} \end{aligned}$$

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - ⇝ we can spend a fraction y_M of each day on matching M
  - ⇝ without loss of generality: all days the same!

⇝ Best fractional solution given by LP:

$\mathcal{M}$ = set of maximal matchings

$$\begin{aligned} \min \quad & \bar{h} \\ \text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\ & \frac{1}{\sum_{M \in \mathcal{M}: e \in M} y_M} \cdot g_e \leqslant \bar{h} \qquad \forall e \in R \\ & y_M \in [0,1] \qquad \forall M \in \mathcal{M} \end{aligned}$$

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - $\rightsquigarrow$ we can spend a fraction y_M of each day on matching M
  - $\rightsquigarrow$ without loss of generality: all days the same!

$\rightsquigarrow$ Best fractional solution given by LP:

$\mathcal{M}$ = set of maximal matchings

$$
\begin{aligned}
\min \quad & \bar{h} \\
\text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\
& \frac{1}{\sum_{M \in \mathcal{M}: e \in M} y_M} \cdot g_e \leqslant \bar{h} && \forall e \in R \\
& y_M \in [0,1] && \forall M \in \mathcal{M}
\end{aligned}
$$

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - ⇝ we can spend a fraction y_M of each day on matching M
  - ⇝ without loss of generality: all days the same!

⇝ Best fractional solution given by LP:

$\mathcal{M}$ = set of maximal matchings

$$
\begin{aligned}
\min \quad & \bar{h} \\
\text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\
& \frac{1}{\sum_{M \in \mathcal{M}: e \in M} y_M} \cdot g_e \leqslant \bar{h} && \forall e \in R \\
& y_M \in [0,1] && \forall M \in \mathcal{M}
\end{aligned}
$$

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - ⇝ we can spend a fraction y_M of each day on matching M
  - ⇝ without loss of generality: all days the same!

⇝ Best fractional solution given by LP: after change of variable $\boxed{\bar{h} = \frac{1}{\ell}}$ and slight massaging

$\mathcal{M}$ = set of maximal matchings

$$
\begin{aligned}
\min \quad & \frac{1}{\ell} \\
\text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\
& \frac{1}{\sum_{M \in \mathcal{M}: e \in M} y_M} \cdot g_e \leqslant \frac{1}{\ell} && \forall e \in R \\
& y_M \in [0,1] && \forall M \in \mathcal{M}
\end{aligned}
$$

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - ⇝ we can spend a fraction y_M of each day on matching M
  - ⇝ without loss of generality: all days the same!

⇝ Best fractional solution given by LP: after change of variable $\boxed{\bar{h} = \frac{1}{\ell}}$ and slight massaging

$\mathcal{M}$ = set of maximal matchings

$$
\begin{aligned}
\min \quad & \frac{1}{\ell} \\
\text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\
& \sum_{M \in \mathcal{M}: e \in M} y_M \cdot \frac{1}{g_e} \geqslant \ell \qquad \forall e \in R \\
& y_M \in [0,1] \qquad \forall M \in \mathcal{M}
\end{aligned}
$$

Fractional Scheduling

- lower-bound tool: **fractional relaxation**
  - ⇝ we can spend a fraction y_M of each day on matching M
  - ⇝ without loss of generality: all days the same!

⇝ Best fractional solution given by LP: after change of variable $\boxed{\bar{h} = \frac{1}{\ell}}$ and slight massaging

$\mathcal{M}$ = set of maximal matchings

$$
\begin{aligned}
\max \quad & \ell \\
\text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\
& \sum_{M \in \mathcal{M}: e \in M} y_M \cdot \frac{1}{g_e} \geqslant \ell && \forall e \in R \\
& y_M \in [0,1] && \forall M \in \mathcal{M}
\end{aligned}
$$

Dual LP $\rightarrow$ Density

- LP has exponentially many variables ... but few constraints

$\rightsquigarrow$ Consider the **dual LP**:

$$\begin{aligned} \min \quad & x \\ \text{s.t.} \quad & \sum_{e \in R} z_e \geqslant 1 \\ & \sum_{e \in M} \frac{z_e}{g_e} \leqslant x \quad \forall M \in \mathcal{M} \\ & z_e \geqslant 0 \quad \forall e \in R \end{aligned}$$

$\rightsquigarrow$ any feasible solution $(x, \vec{z})$ lower bounds optimal heat h^*!

Fractional Poly Scheduling LP

$$\begin{aligned} \max \quad & \ell \\ \text{s.t.} \quad & \sum_{M \in \mathcal{M}} y_M \leqslant 1 \\ & \frac{1}{g_e} \sum_{M \in \mathcal{M}: e \in M} y_M \geqslant \ell \quad \forall e \in R \\ & y_M \geqslant 0 \quad \forall M \in \mathcal{M} \end{aligned}$$

Theorem (Fractional lower bound)

For any $z_e \in [0,1]$, for $e \in R$, with $\sum_{e \in R} z_e = 1$, we have

$$h^* \geqslant \bar{h}(z) = \frac{1}{\max_{M \in \mathcal{M}} \sum_{e \in M} \frac{z_e}{g_e}}.$$

Outline



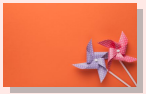
1 Introduction



2 Hardness



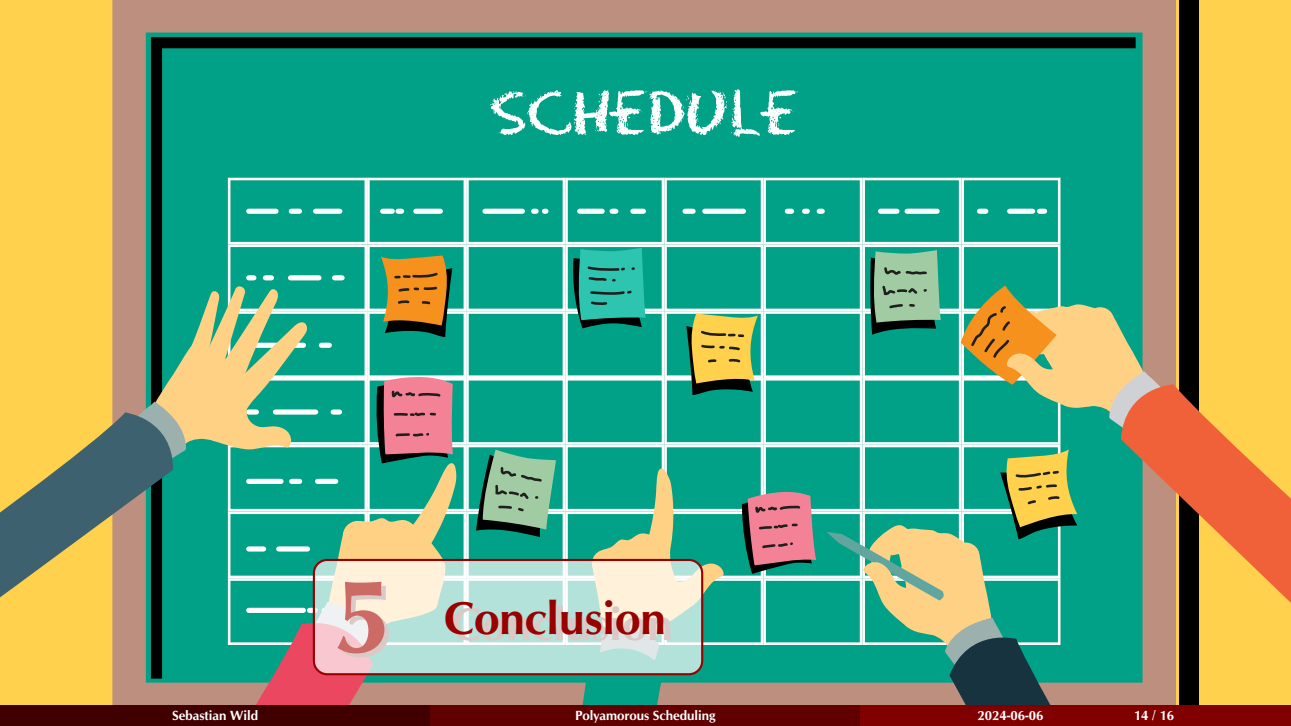
3 Approximation



4 Density



5 Conclusion



5 Conclusion

Conclusion

What you've seen:

- **Poly Scheduling** is a highly desired practical problem
- but it's hard
  - NP-hard
  - hard to approximate better than $4/3$
- best known polytime approximation only $O(\log \Delta) = O(\log n)$

Some questions:

1. Best possible approximation ratio?
2. Better results for restricted versions (e.g., special types of graphs)
3. Density threshold that guarantees schedule?


M T W T F S S

Icons made by *Freepik*, *Gregor Cresnar*, *Those Icons*, *Smashicons*, *Good Ware*, *Pause08*, and *Madebyoliver* from `www.flaticon.com`.

Vector graphics from Pressfoto, brgfx, macrovector and Jannoon028 on freepik.com
Other photos from `www.pixabay.com`.