複数の巡査による
指定地点の警邏について

能城秀彬（東京大学）

警邏（けいら）

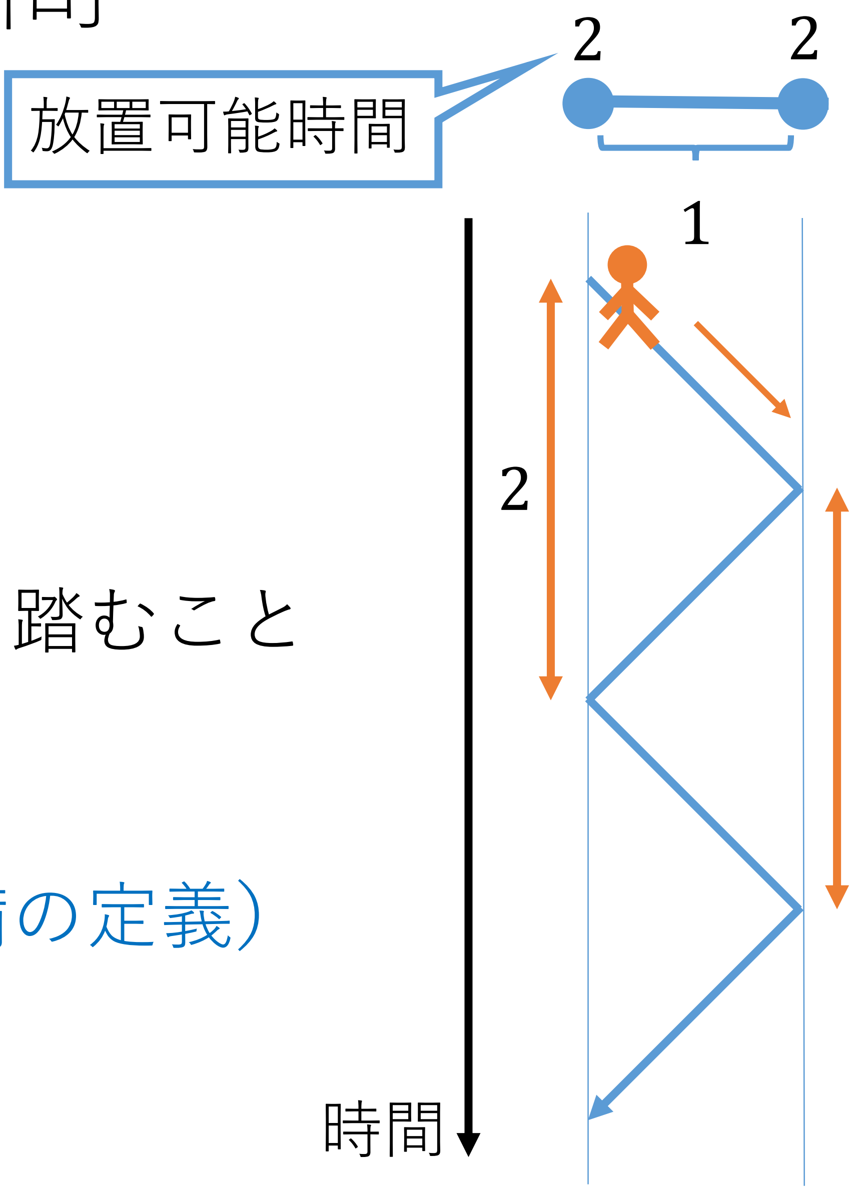
- 警邏(patrolling)とは
  - 1人または複数の巡査により
  - 領域内のあらゆる場所を<u>十分な頻度</u>で訪問すること
- 警邏する領域の例
  - 二次元の領域
  - 線分や閉路などの全体
  - グラフの頂点

警邏（けいら）

- 警邏(patrolling)とは
  - 1人または複数の巡査により
  - 領域内のあらゆる場所を<u>十分な頻度</u>で訪問すること
- 警邏する領域の例
  - 二次元の領域
  - 線分や閉路などの全体
  - グラフの頂点 ← 今回扱うもの

問題設定 – “放置可能時間”

- 頂点を警備するのに必要な
  訪問の頻度を定める
- 連続した2回の訪問時刻の差として
  許される最大値
- 訪問とは点で表される巡査が頂点を踏むこと
- 頂点を警備するには,
  放置可能時間を満たしながら
  訪問し続けなければならない（警備の定義）



問題設定

辺の長さ

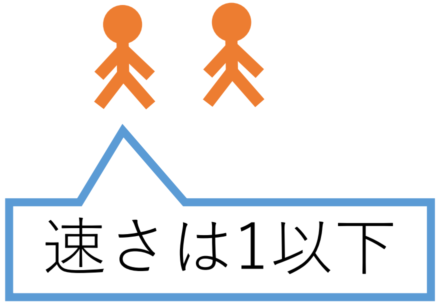
- 入力
  - 無向グラフ $G = (V, E, d)$ （警備する対象）
  - 各頂点の放置可能時間
  - 巡査の人数（どの巡査も速さ1以下で動く）
- 目的
  - DecisionPP : 全頂点を警備できるかどうかを判定
  - OptimizePP : 各頂点の利得も入力として与える．警備できる頂点部分集合のうち，利得の合計が最大のものを求める

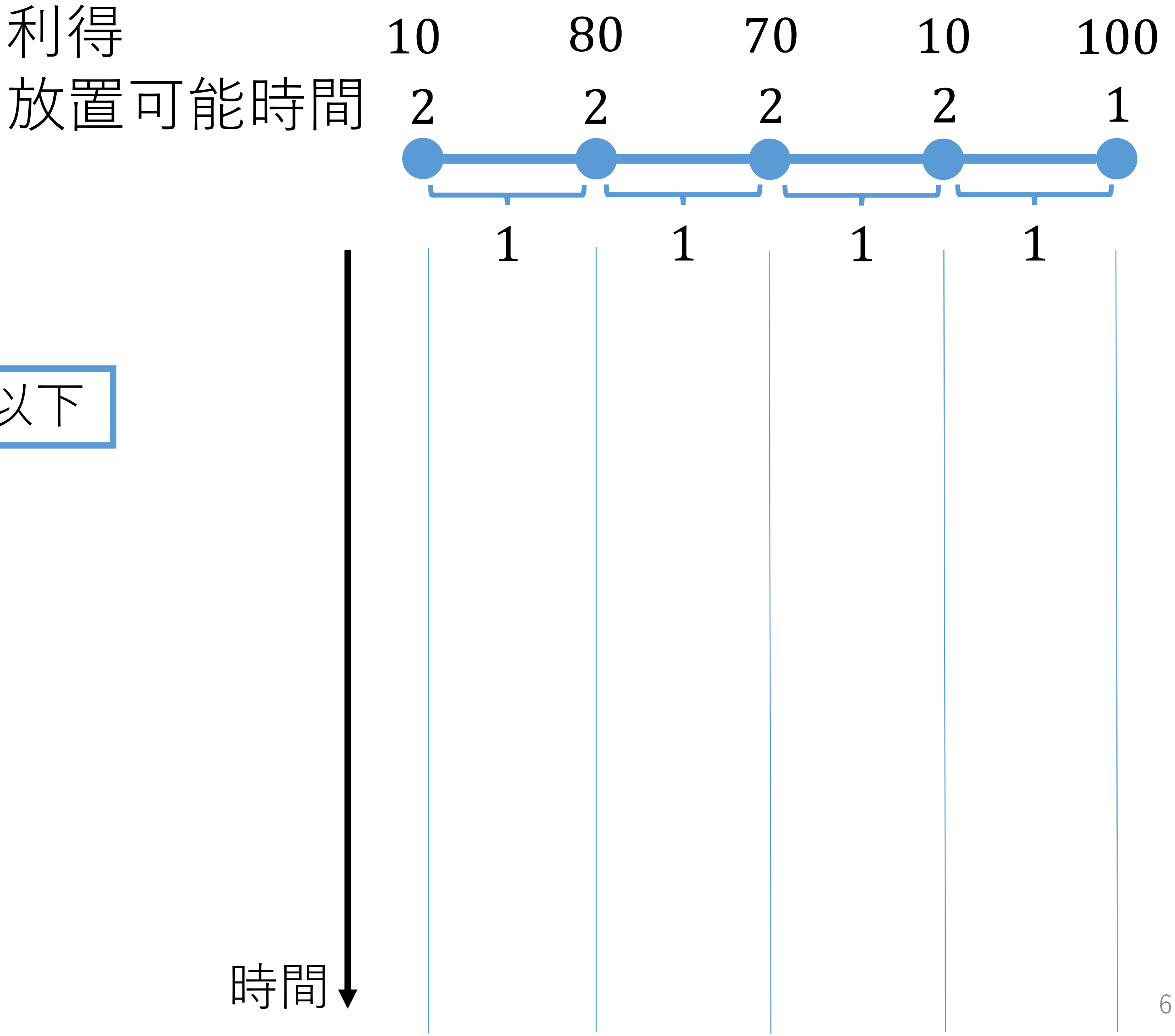
DecisionPPの一般化

この2つの問題についてそれぞれ計算量を調べる

例

- 巡査が2人

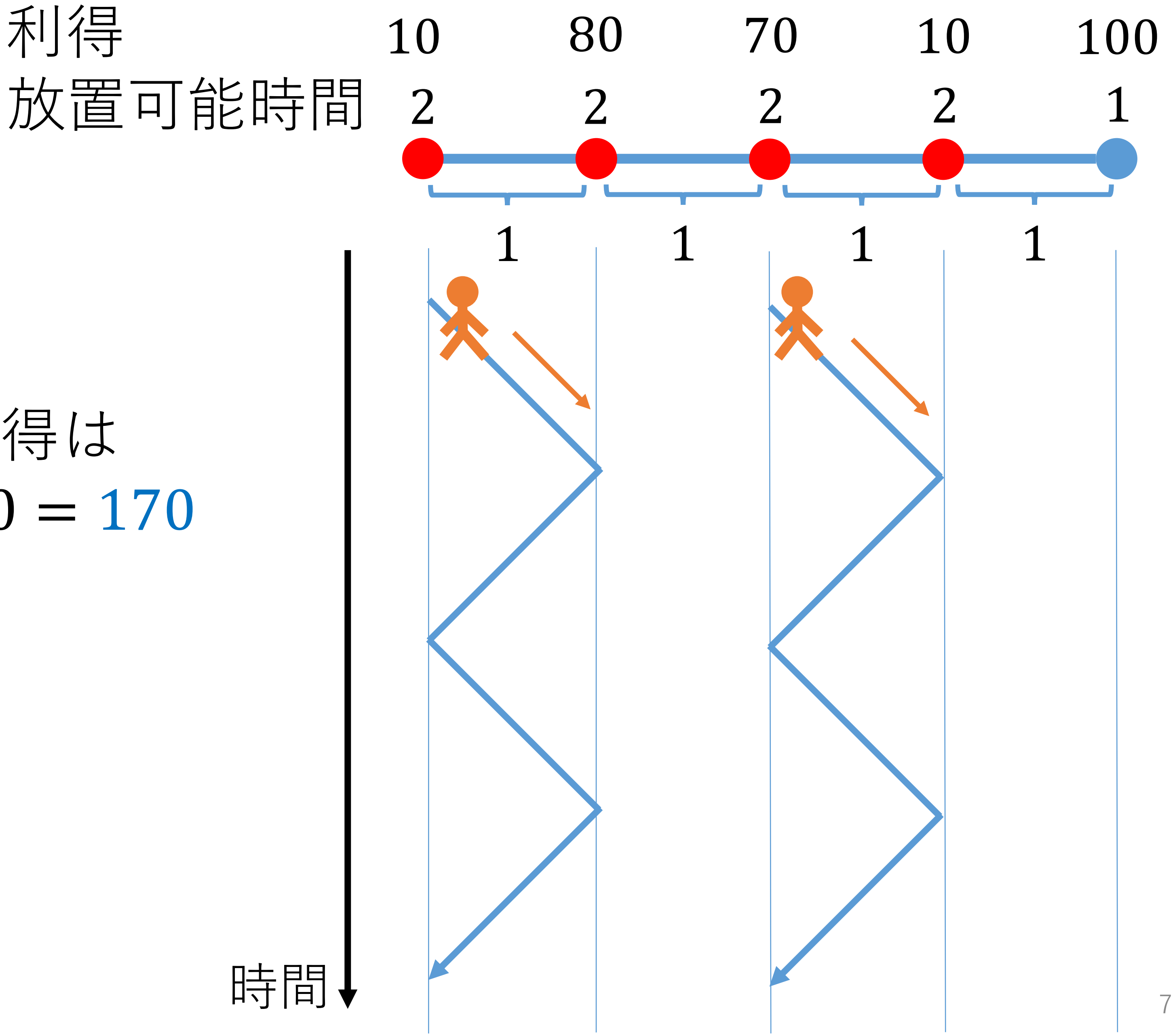




例

- 巡査が2人

- 青の動きを選ぶと利得は
  $10 + 80 + 70 + 10 = 170$

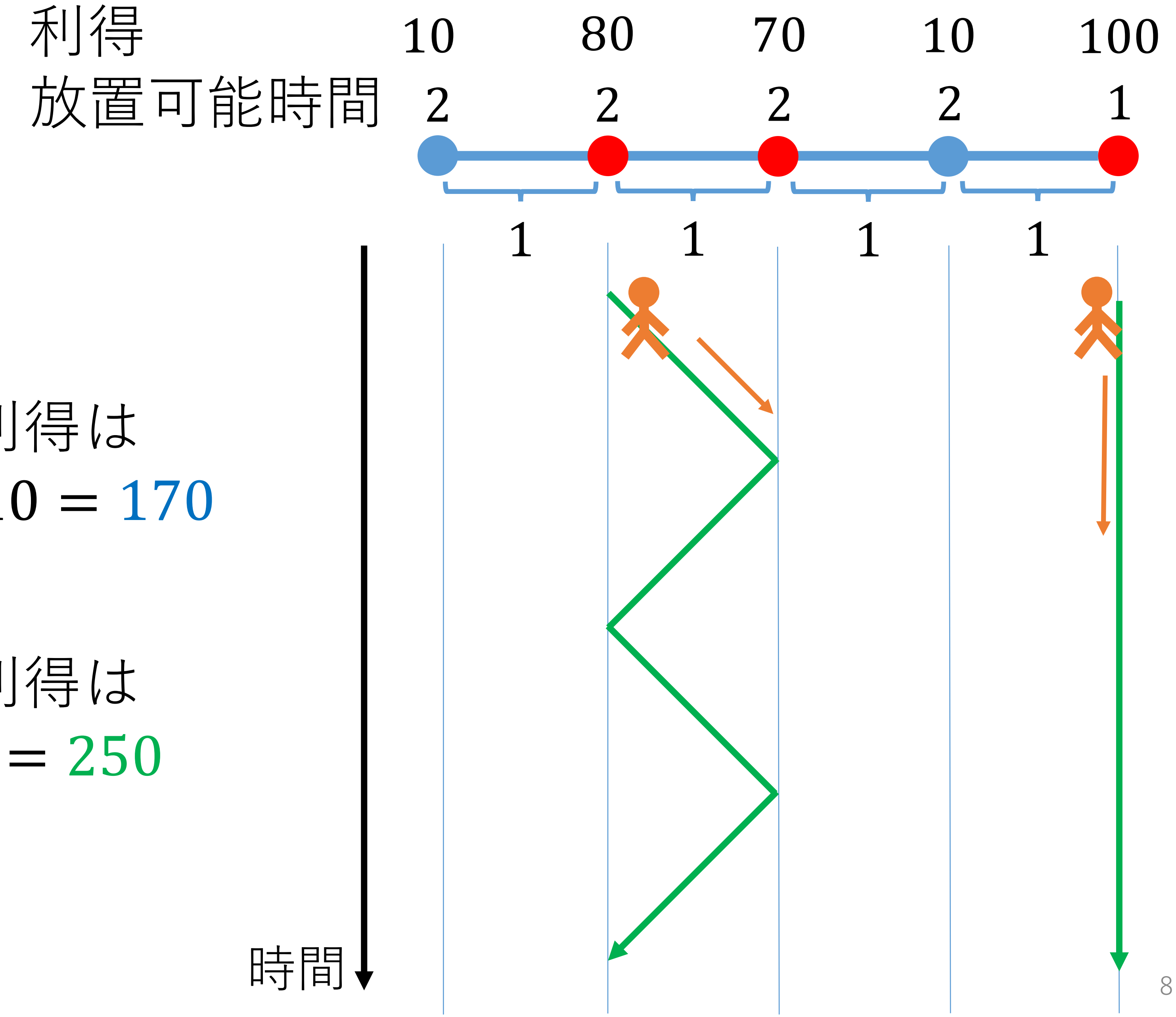


例

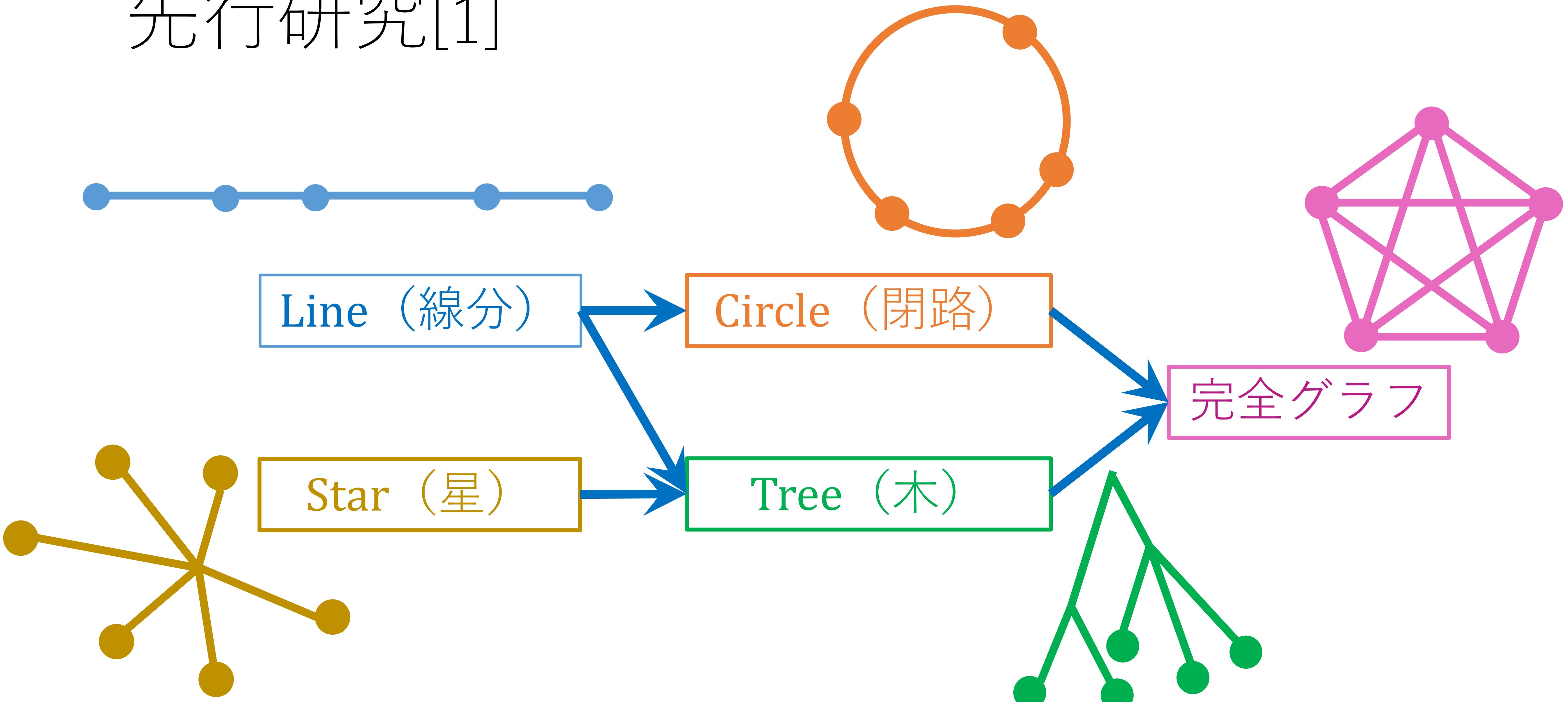
- 巡査が2人

- 青の動きを選ぶと利得は
  $10 + 80 + 70 + 10 = 170$

- 緑の動きを選ぶと利得は
  $80 + 70 + 100 = 250$

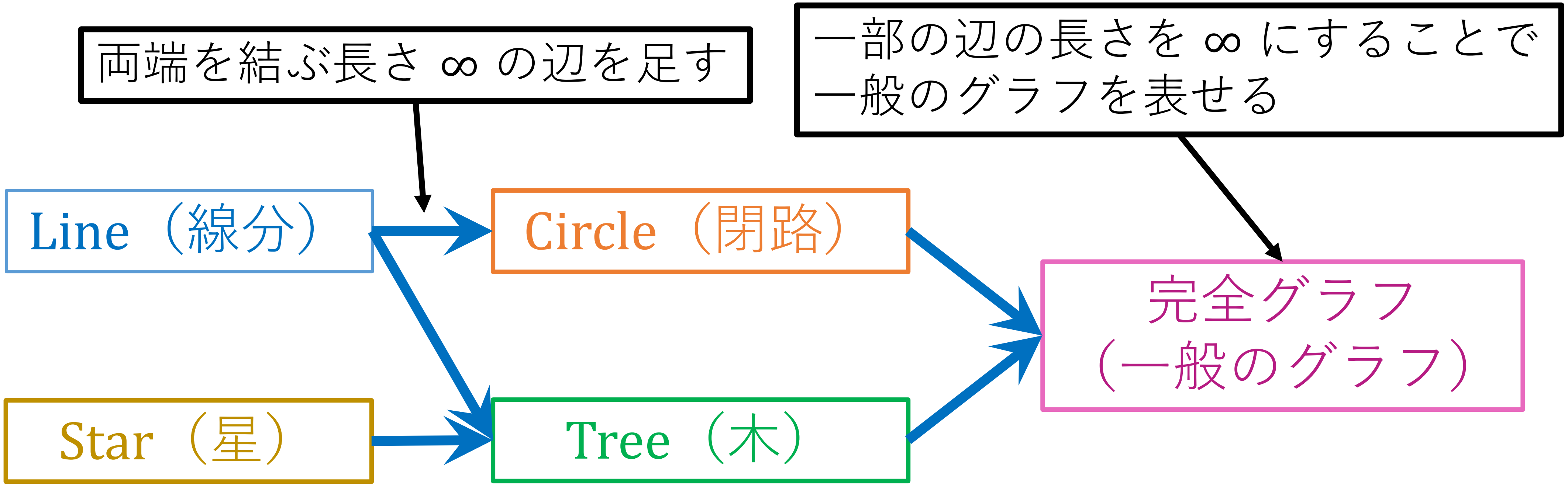


先行研究[1]



[1] : S. Coene, F.C.R. Spieksma, and G.J. Woeginger. (2011). Charlemagne's challenge: the periodic latency problem. *Operations Research,* 59(3), pp. 674–683.

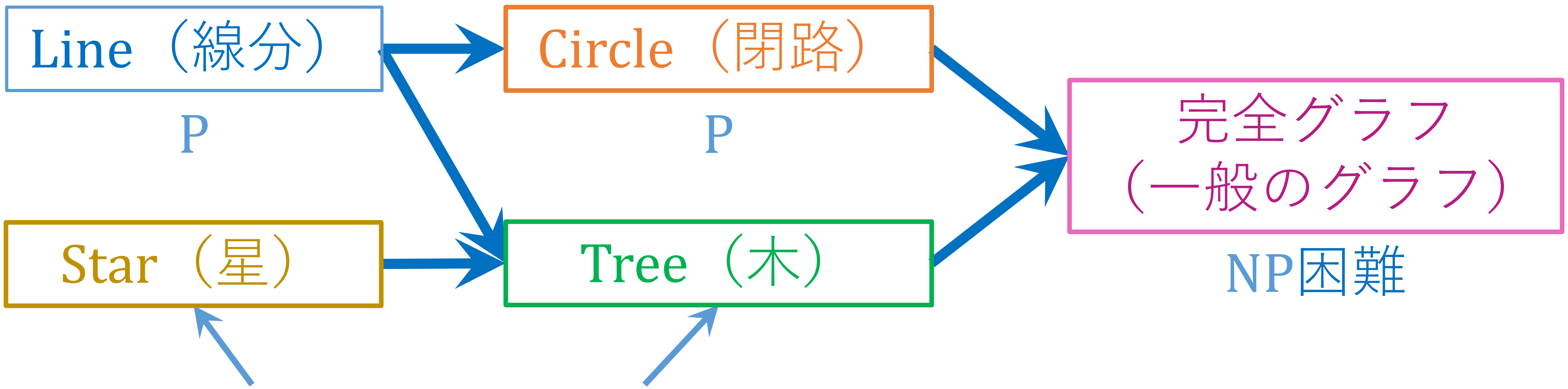
先行研究[1]



[1] : S. Coene, F.C.R. Spieksma, and G.J. Woeginger. (2011). Charlemagne's challenge: the periodic latency problem. *Operations Research*, 59(3), pp. 674–683.

先行研究[1]

巡査が1人の場合

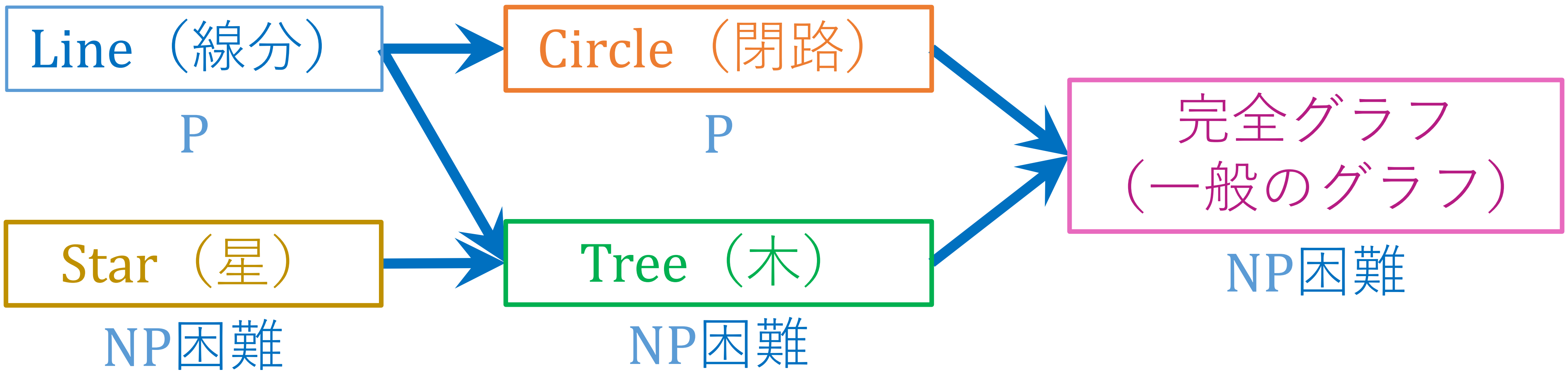


全頂点の利得・放置可能時間が等しいときのみ P，
そうでなければ NP困難

[1] : S. Coene, F.C.R. Spieksma, and G.J. Woeginger. (2011). Charlemagne's challenge: the periodic latency problem. *Operations Research,* 59(3), pp. 674–683.

先行研究[1]

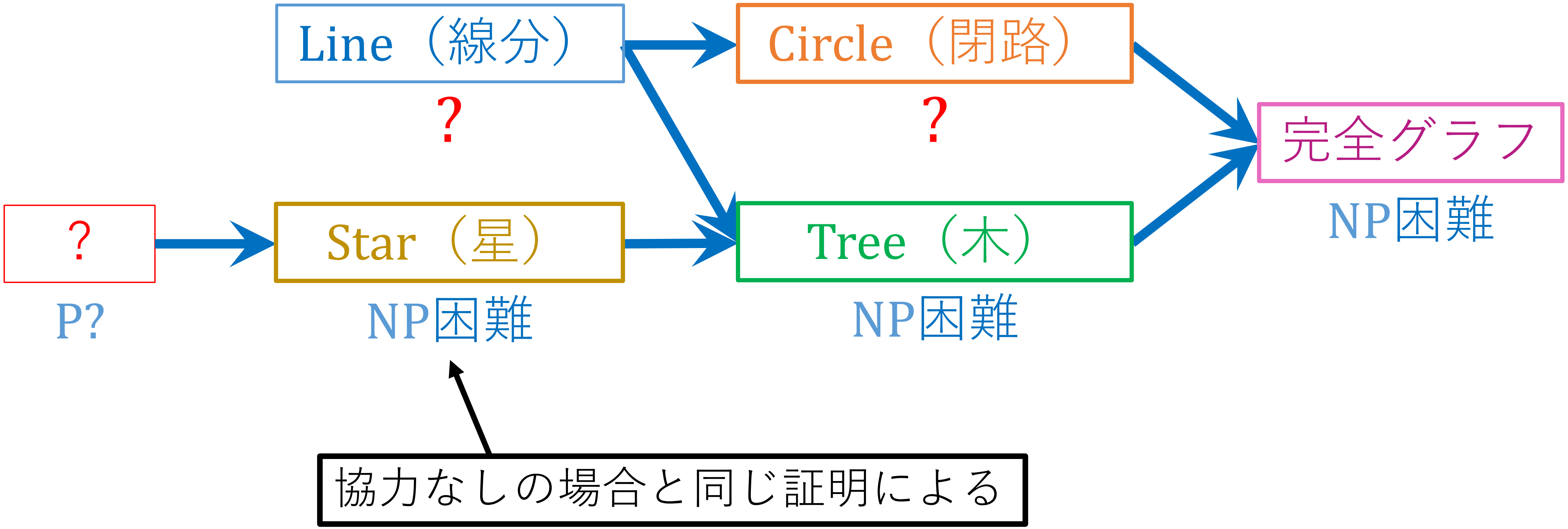
巡査が複数人の場合（※協力して警備はしない問題設定）



[1] : S. Coene, F.C.R. Spieksma, and G.J. Woeginger. (2011). Charlemagne's challenge: the periodic latency problem. *Operations Research,* 59(3), pp. 674–683.

本研究の目的

巡査が複数人の場合（**複数の巡査の協力を許す場合**）



本研究の目

巡査が複数人の場

す場合）

Line

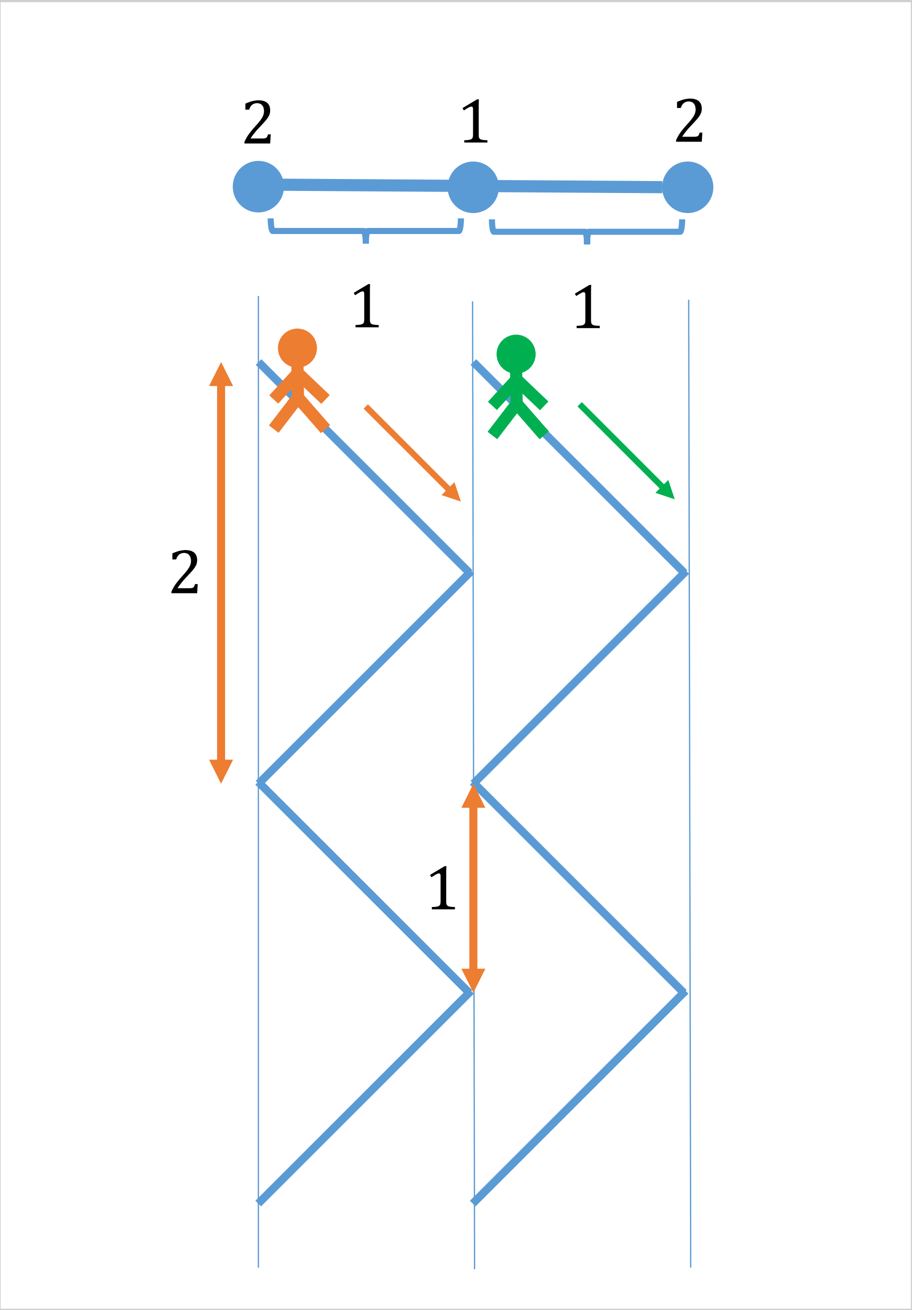
？

P?

Star

N

協力

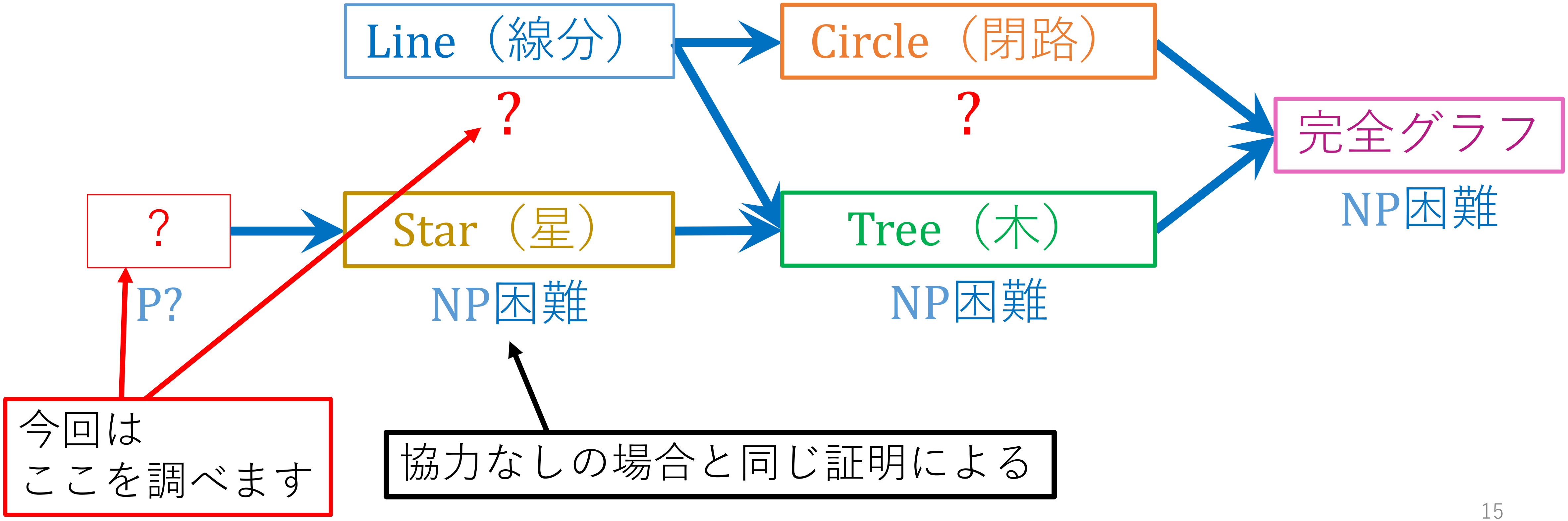


完全グラフ

NP困難

本研究の目的

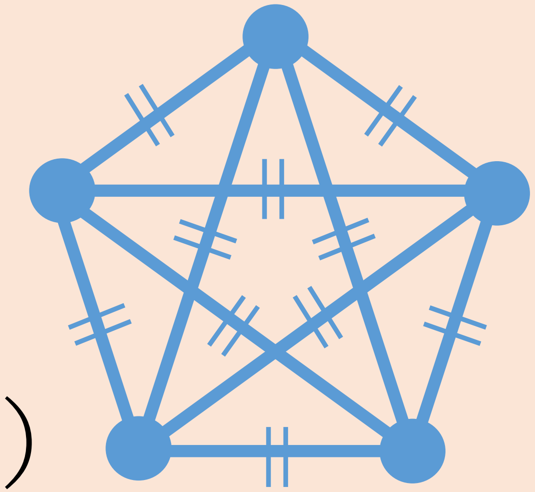
巡査が複数人の場合（**複数の巡査の協力を許す場合**）



今回扱うケース

1. Line
   巡査が複数の場合

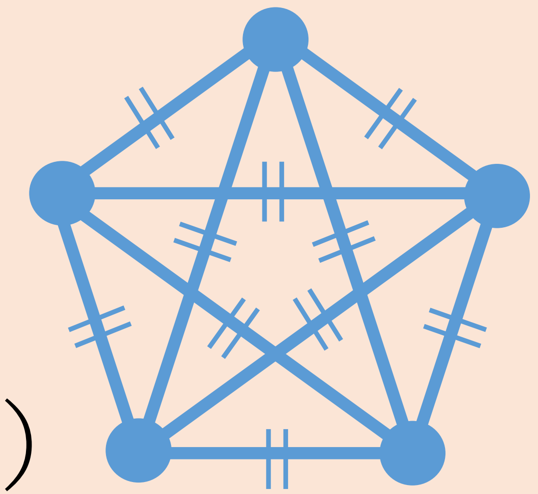
2. Comp
   （辺の長さが全て等しい完全グラフ）



今回扱うケース

1. Line
   巡査が複数の場合

2. Comp
   （辺の長さが全て等しい完全グラフ）



Lineの場合の概要

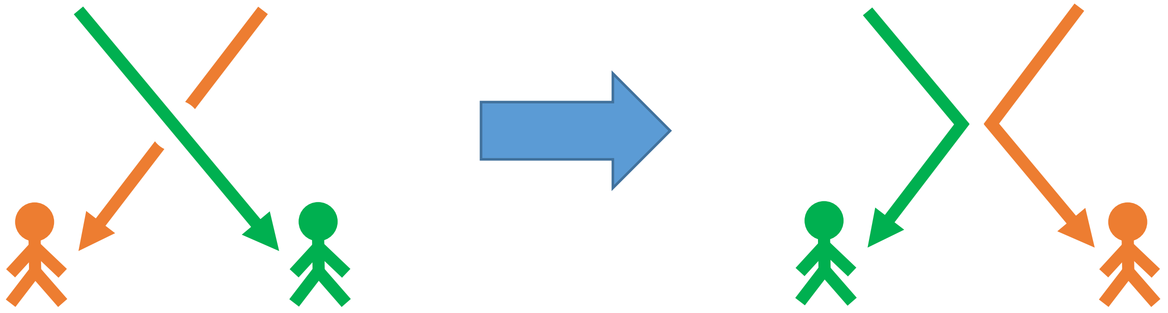
- 巡査が1人の場合（既知）
  → OptimizePPに多項式時間アルゴリズムあり
- 巡査が複数の場合（本研究）
  - 放置可能時間が全て同じ場合
    → OptimizePPに多項式時間アルゴリズムあり
  - 放置可能時間が一般の場合 → 未解決
    - 複雑な動きの例
    - 別の問題設定について

Lineの場合の概要

- 巡査が1人の場合（既知）
  → OptimizePPに多項式時間アルゴリズムあり
- 巡査が複数の場合（本研究）
  - 放置可能時間が全て同じ場合
    → OptimizePPに多項式時間アルゴリズムあり
  - 放置可能時間が一般の場合 → 未解決
    - 複雑な動きの例
    - 別の問題設定について

Line：巡査の位置関係について

- 巡査は線分上を右か左に動く（か停止）
- 巡査の能力は全員同じなので,
  すれ違う代わりに互いに引き返してもよい



→ 巡査は初期配置の順番を保って動くとしてよい

Line：巡査が複数，放置可能時間が全て同じ場合

定理1

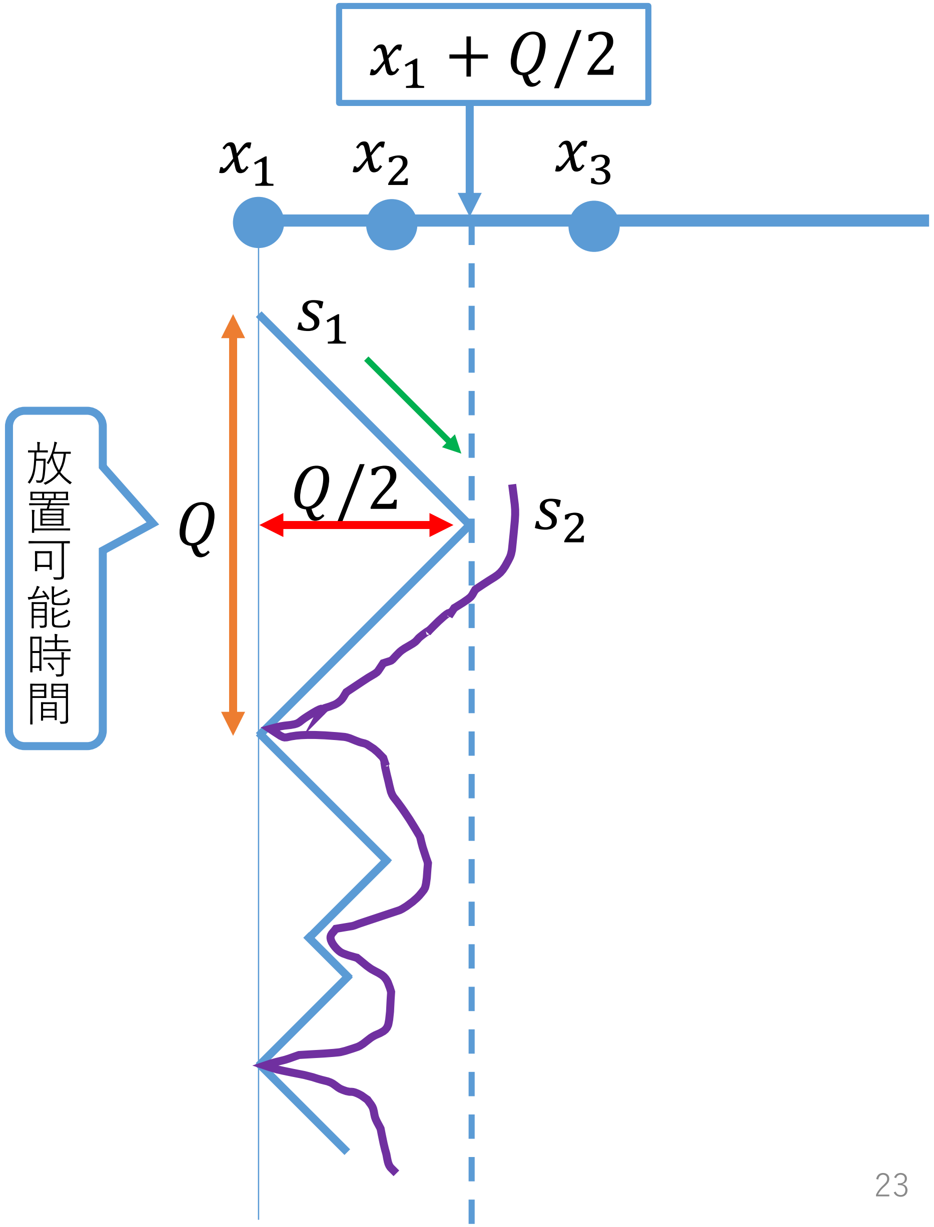
Lineで放置可能時間が全て等しい場合，巡査が複数でもOptimizePPに多項式時間アルゴリズムが存在する．

定理1の証明手順

1. 任意の実行可能解の変換
   頂点部分集合 $V_s \subset V$ を警備できる巡査の動き方が存在するならば，ある特別な動き方（後述）でも V_s を警備できることを示す
2. 特別な動き方のなかで最適解を求める
   このような動き方での最適解が存在するので
   それを探す多項式時間アルゴリズムを示す

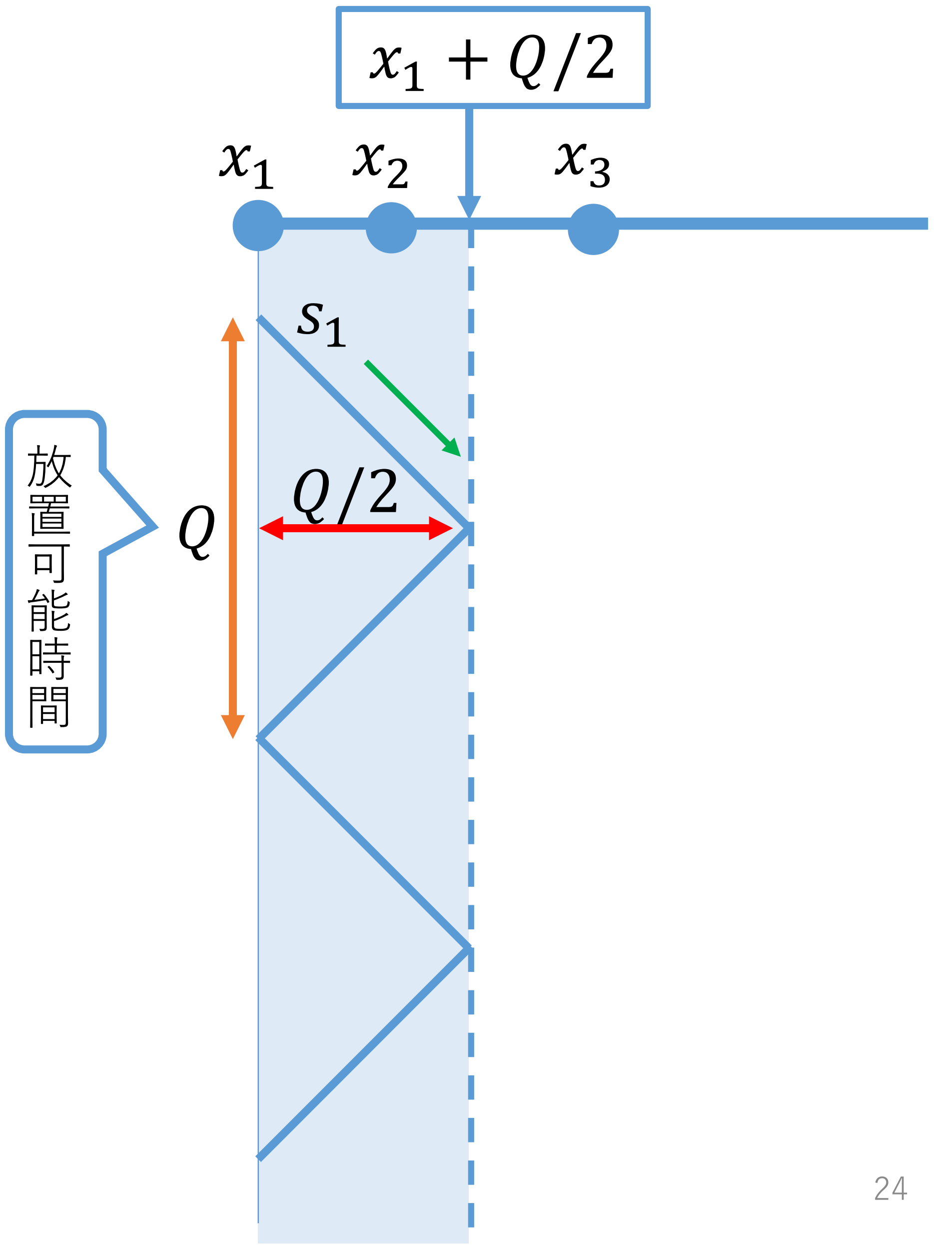
定理1の証明：step 1

- ある頂点部分集合 V_S を警備できる巡査の動きがあったとする
- V_S の点の座標を $x_1 \leq x_2 \leq \cdots \leq x_k$ とする
- 初期順序を保つとき，最も左の巡査 s_1 以外が x_1 にいるならば s_1 も x_1 にいる
  → x_1 はs_1 のみにより警備される



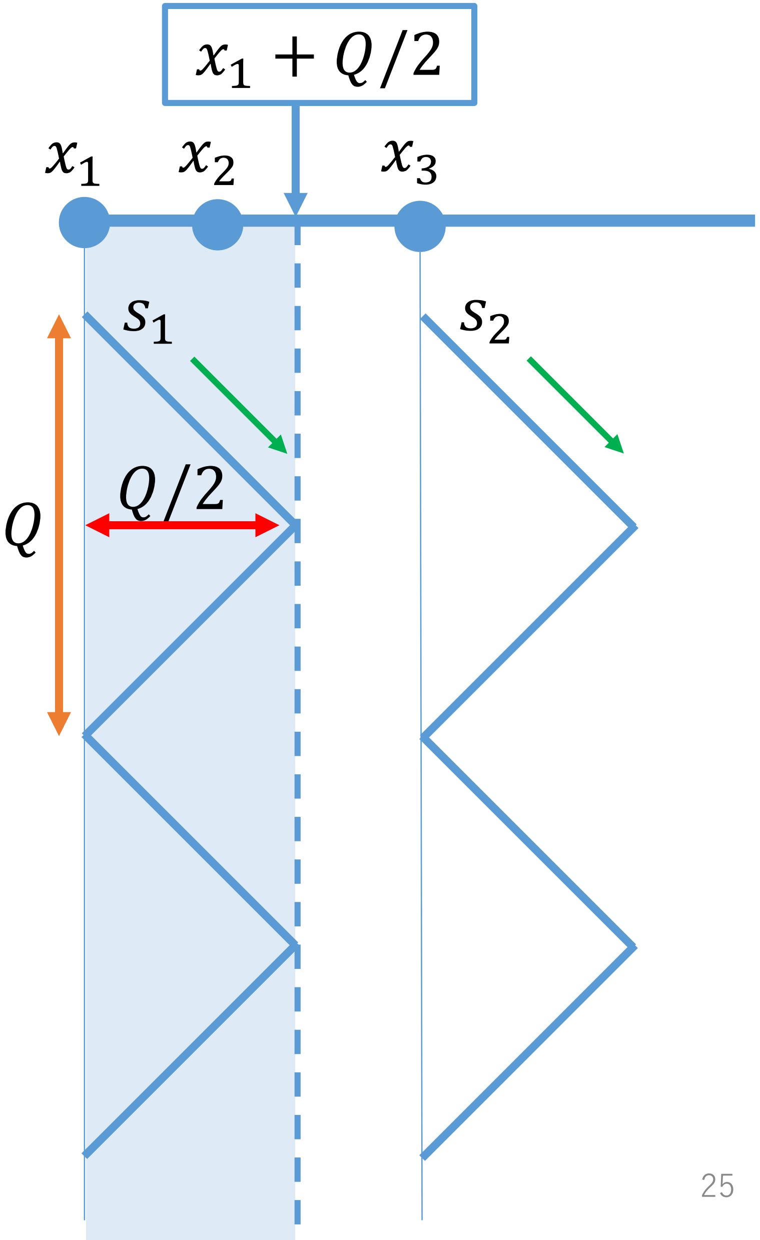
定理1の証明：step 1

- s_1 は $x_1 + Q/2$ までしか動けない
- s_1 は $[x_1, x_1 + Q/2]$ を往復すれば, この区間に含まれる全点を警備できる
- これ以上は警備できないのでこれが最適



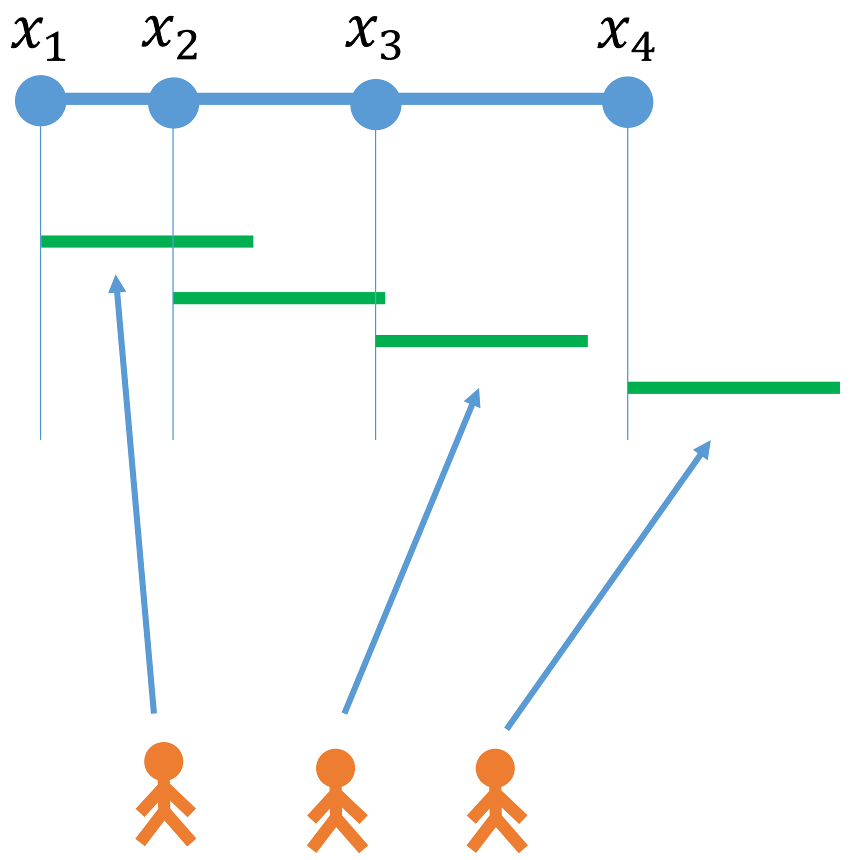
定理1の証明：step 1

- $x_1 + Q/2$ より右側は元々 s_1 以外の巡査達により警備可能
- 残りの点と巡査で同じ変換を繰り返す
- 変換後の動き
  $= m$ 人の巡査それぞれが，n 個の区間
  $$\left[x_1, x_1 + \frac{Q}{2}\right], \dots, \left[x_n, x_n + \frac{Q}{2}\right]$$
  のうち互いに交わりのない m 個以下の区間を1つずつ担当し往復する動き



定理1の証明：step 2

- n個の利得付きの区間から，利得の合計が最大となる交わりの無い$m(< n)$個の区間を選べばよい
  - あとはそれらの区間を巡査が往復するだけ
- 動的計画法により $O(n\log n+nm)$ で計算できる（省略）



Lineの場合の概要

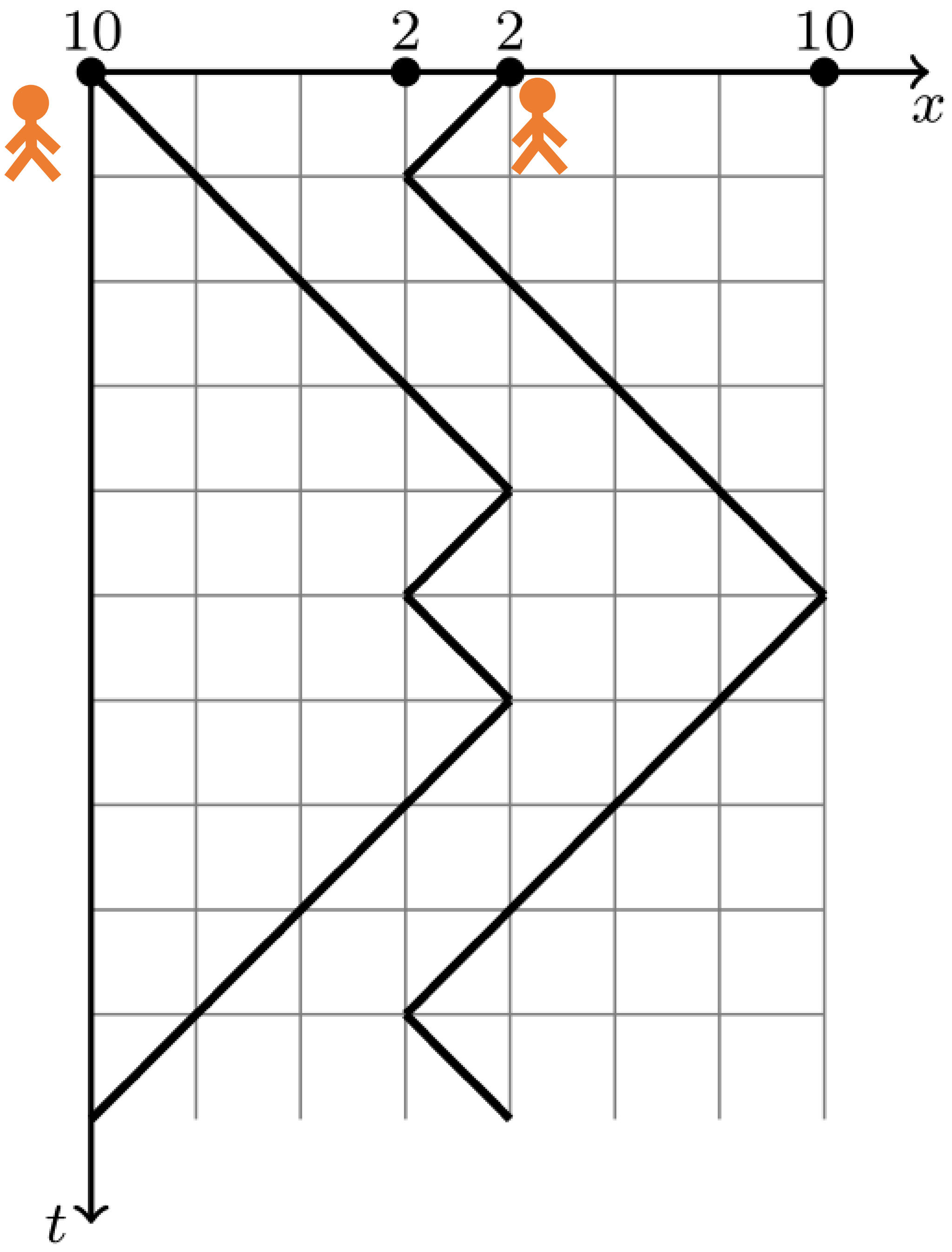
- 巡査が1人の場合（既知）
  → OptimizePPに多項式時間アルゴリズムあり
- 巡査が複数の場合（本研究）
  - 放置可能時間が全て同じ場合
    → OptimizePPに多項式時間アルゴリズムあり
  - 放置可能時間が一般の場合 → 未解決
    - 複雑な動きの例
    - 別の問題設定について

Line：巡査が複数，放置可能時間が一般の場合

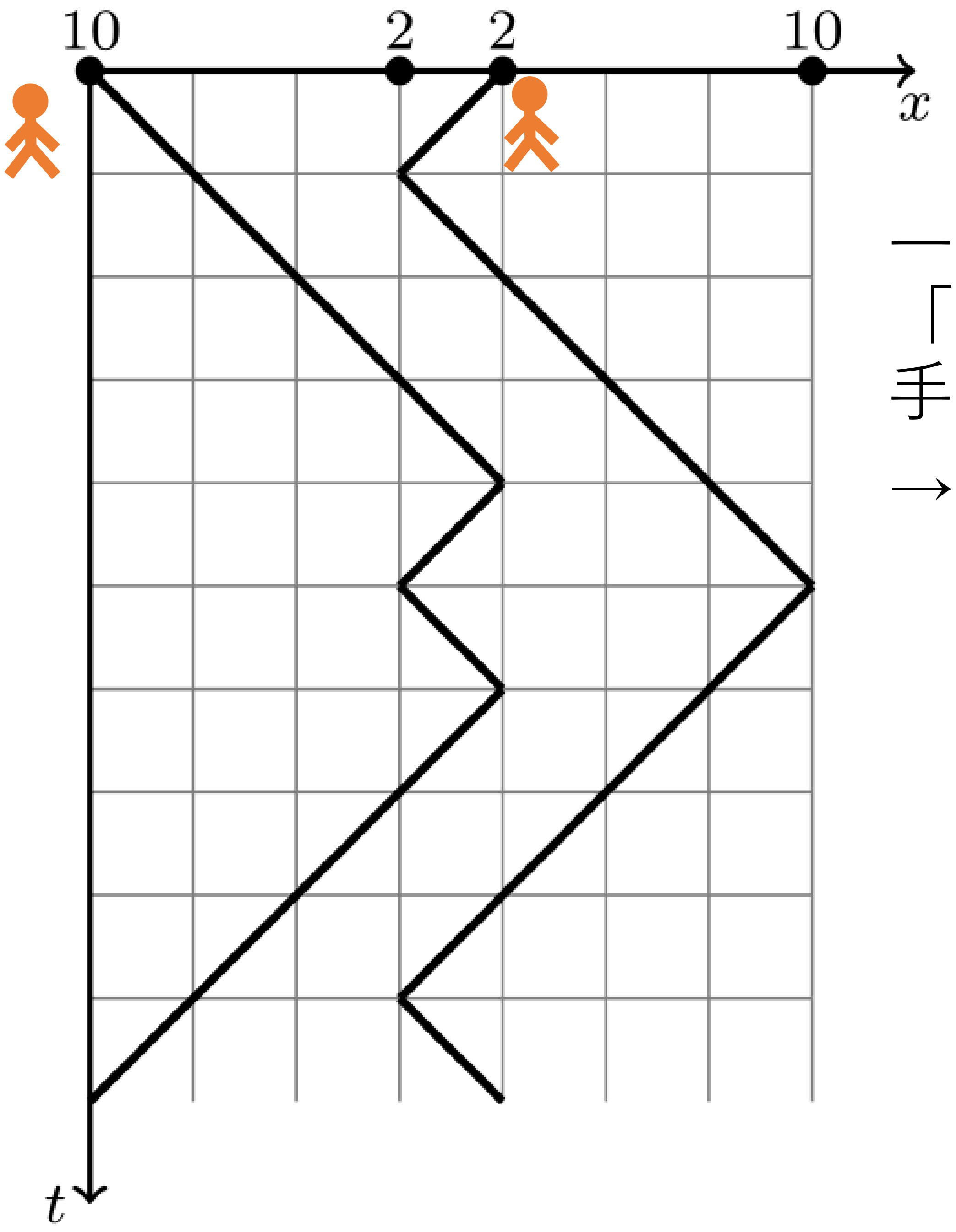
- 放置可能時間が全て同じならば,
  互いに交わりのない区間を往復する動きのみ考えればよかった
  →複数の巡査が協力しなくてよいので単純になっている

- 放置可能時間が一般の場合は，そうでない動きが最適となる例が存在

放置可能時間

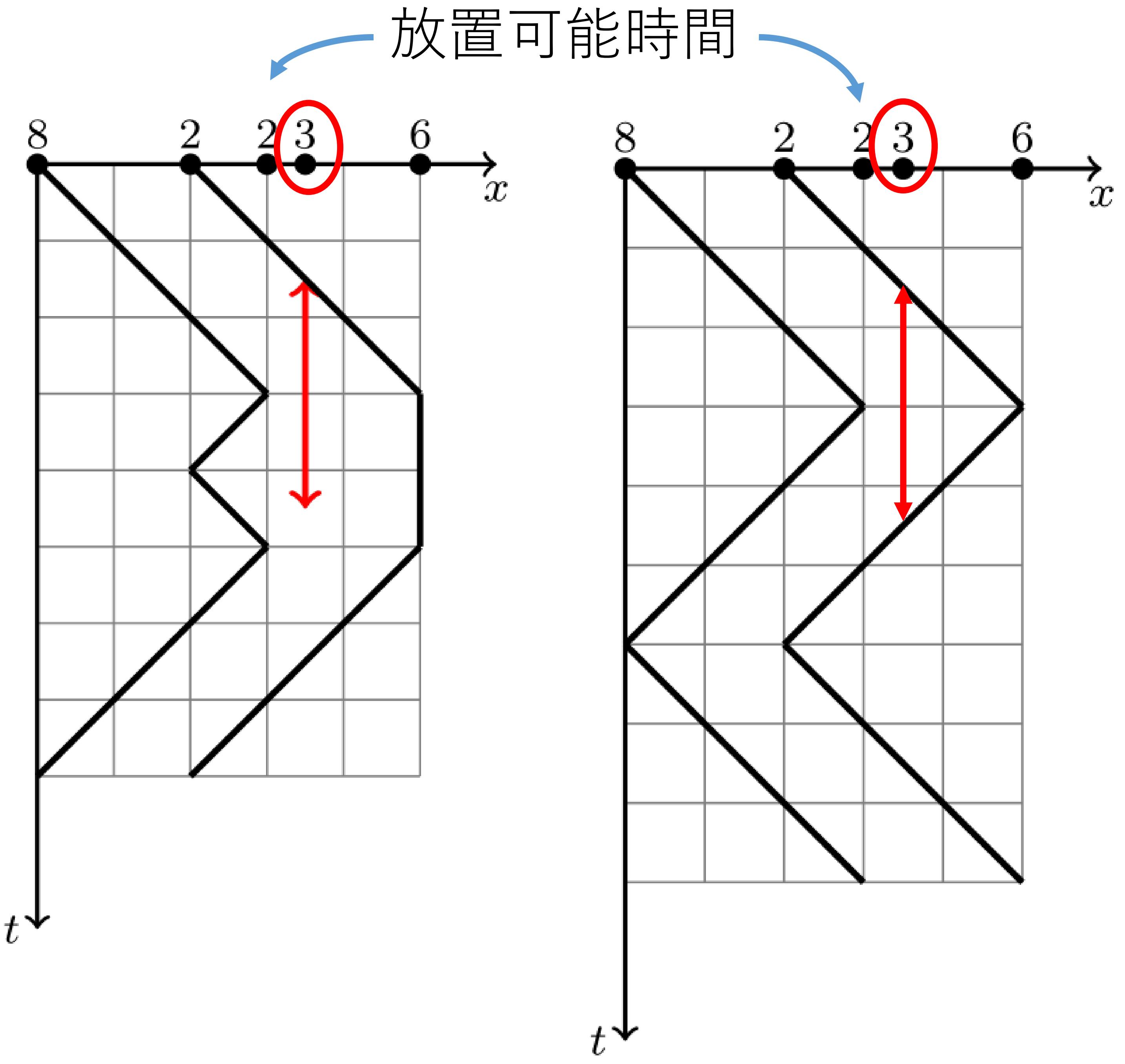


放置可能時間



一番左の巡査はいつも
「可能な限り」右に
手伝いに行ってよいか？

→ No.



一番左の巡査はいつも
「可能な限り」右に
手伝いに行ってよいか？

→ No.

あえて早めに引き返すと
協力しやすくなることもある

→「あえて早めに戻る」
が許されない問題設定
にしたらどうか？

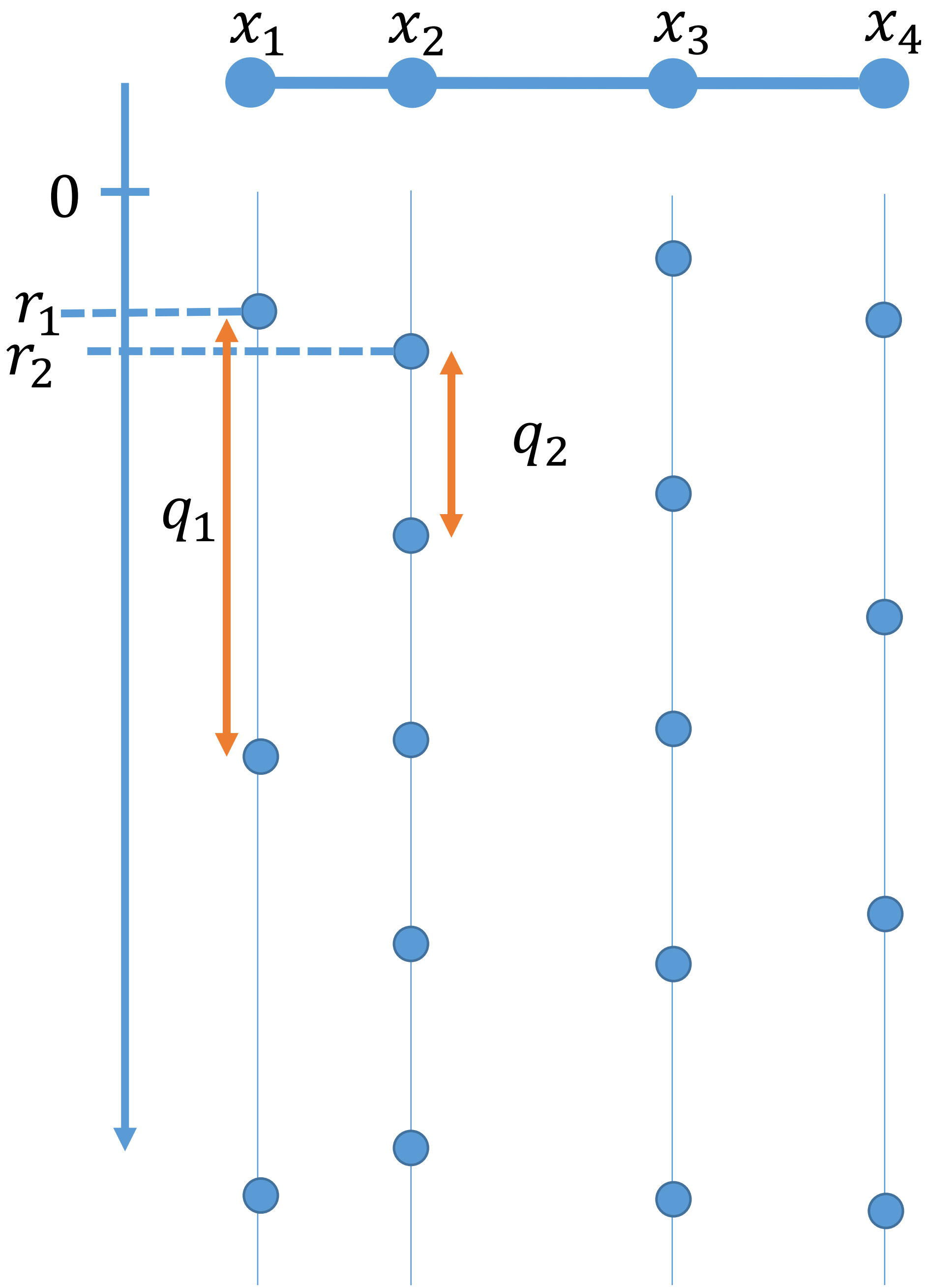
別の問題設定

- 放置可能時間　… ある訪問後この時間以内にまた訪問

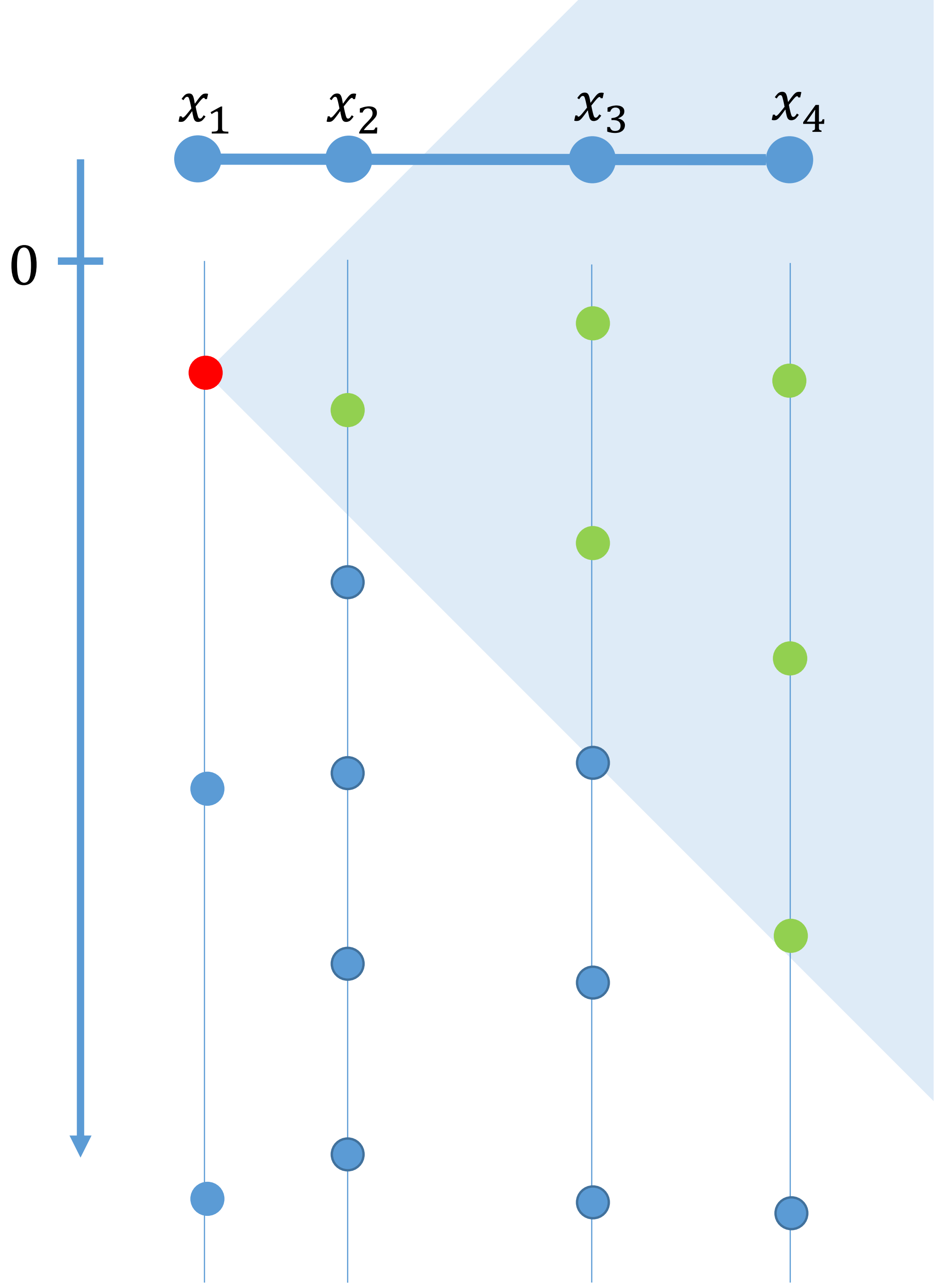
↓

- 周期　… ある訪問後この時間ちょうどにまた訪問
  - 先ほどの例のように「あえて早めに戻る」ができないように
  - さらに最初の訪問時刻も指定
    - 全頂点警備を考えるDecisionPPでは、可能な限り右側を動く戦略が最適に

- 最初の訪問時刻と
そこからの訪問間隔（周期）
が指定される問題
- $t - x$ 平面に訪問すべき時刻
と位置の組 (t, x) を表す点が
全て与えられる

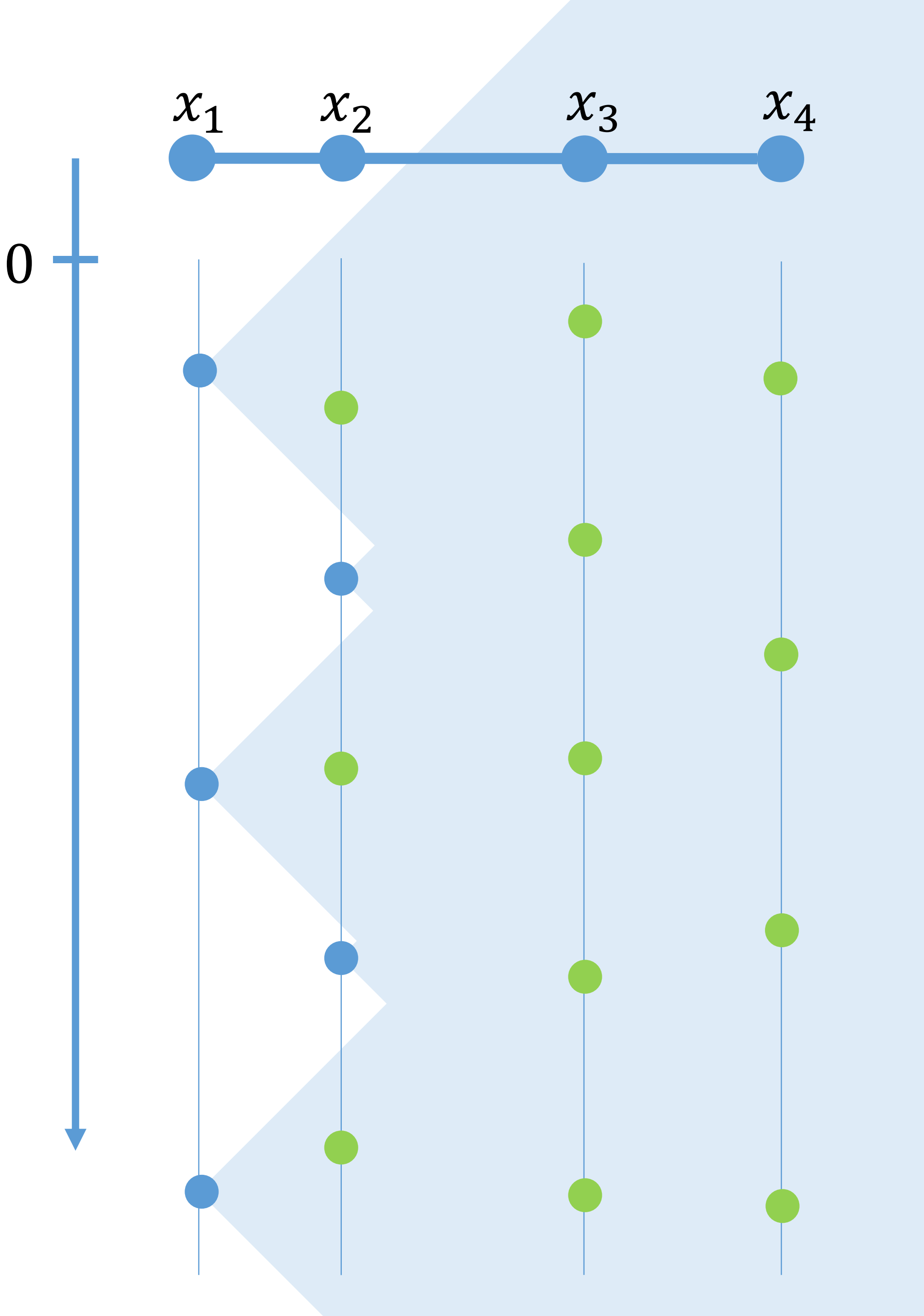


- 巡査 s_1 がある点を訪問する→その右に広がる直角三角形の領域（境界含まず）に含まれる点は s_1 は訪問することができない

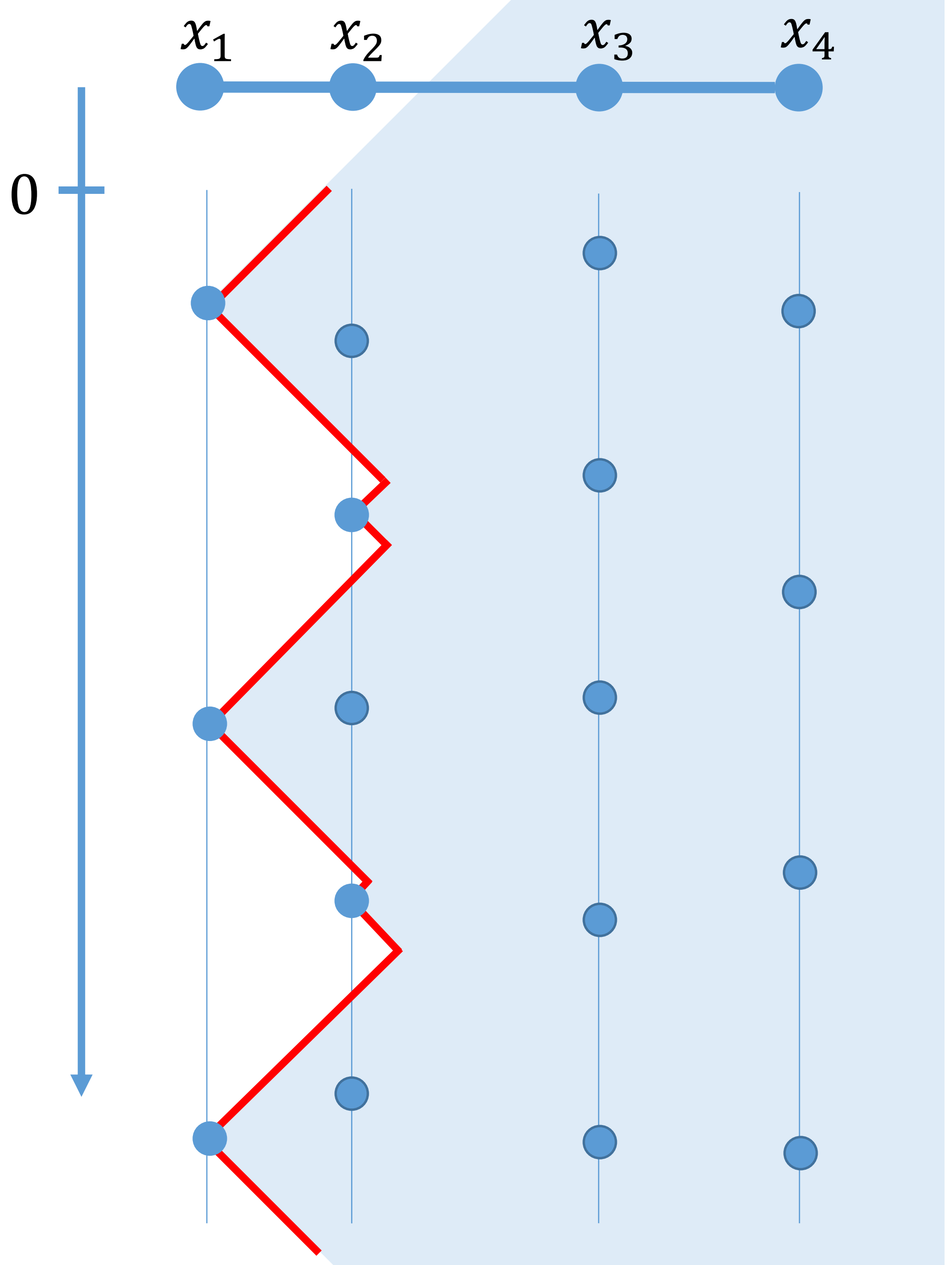


- 平面上の全ての点で
このような直角三角形領域の
和集合をとる
- この領域に含まれる点は
s_1が訪問できない

※最も左側を動く巡査



- その境界を s_1 が動けばよい
→ 訪問できない点以外を全て訪問できているのでこれが最適
- 可能な限り右側を動く戦略が最適になっている
- あとは s_1 が訪問した点を除いてこれを繰り返せばよい
- 必要な巡査の数が分かる



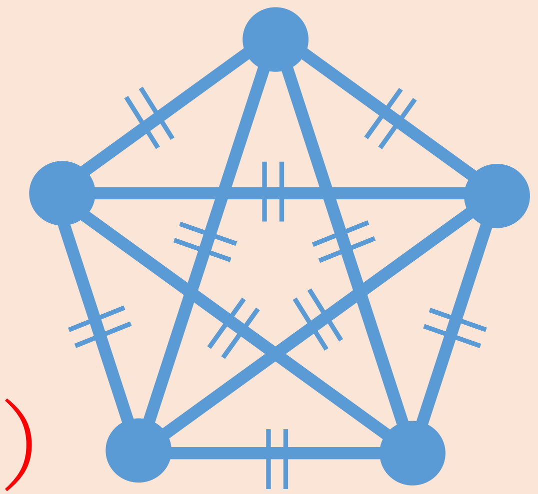
Lineの場合（巡査複数）のまとめ

- 放置可能時間が全て同じであるとき
  → 多項式時間アルゴリズムあり($O(n\log n + nm)$)
- 放置可能時間が一般のとき → 未解決
  - 複雑な動きが最適となる例が存在
  - 別の問題設定 …
    放置可能時間のかわりに周期と最初の訪問時刻を与える
    → DecisionPP ならば可能な限り右側を動く戦略が最適に
  - 周期のみ与えられる場合 → 未解決
    （全点警備できるように最初の訪問時刻を設定できるか）

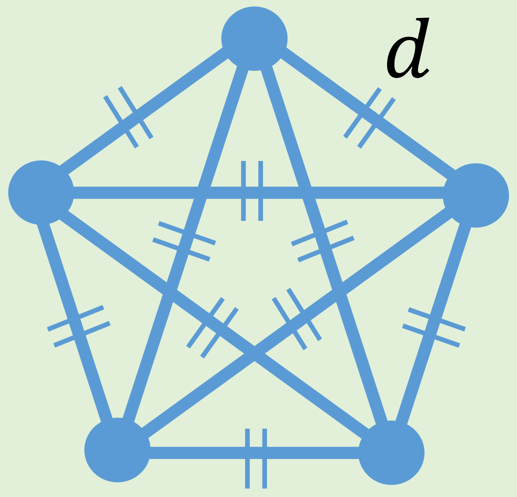
今回扱うケース

1. Line
   巡査が複数の場合

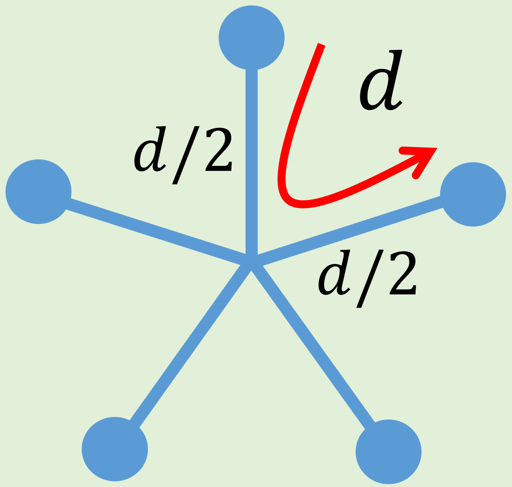
2. Comp
   （辺の長さが全て等しい完全グラフ）



Compの場合の概要



- 辺の長さが全て d（定数）である完全グラフ
  … 任意の2頂点間の移動コストが互いに等しい
- 一般の完全グラフではNP困難
- 全ての辺の長さが $d/2$ のStar（星）と同等
  <u>Star で P の場合</u>は Comp でも P



- 巡査1人
- 全頂点の利得・放置可能時間が同じ

辺の長さが全て等しいと簡単な場合がある

Compの場合の概要

- Comp（本研究）
  - 放置可能時間が全て等しい→ 多項式時間アルゴリズム存在
  - 放置可能時間が一般の場合とき → 未解決
    - 別の問題設定を2つ考える

Comp：放置可能時間が全て同じとき

定理2

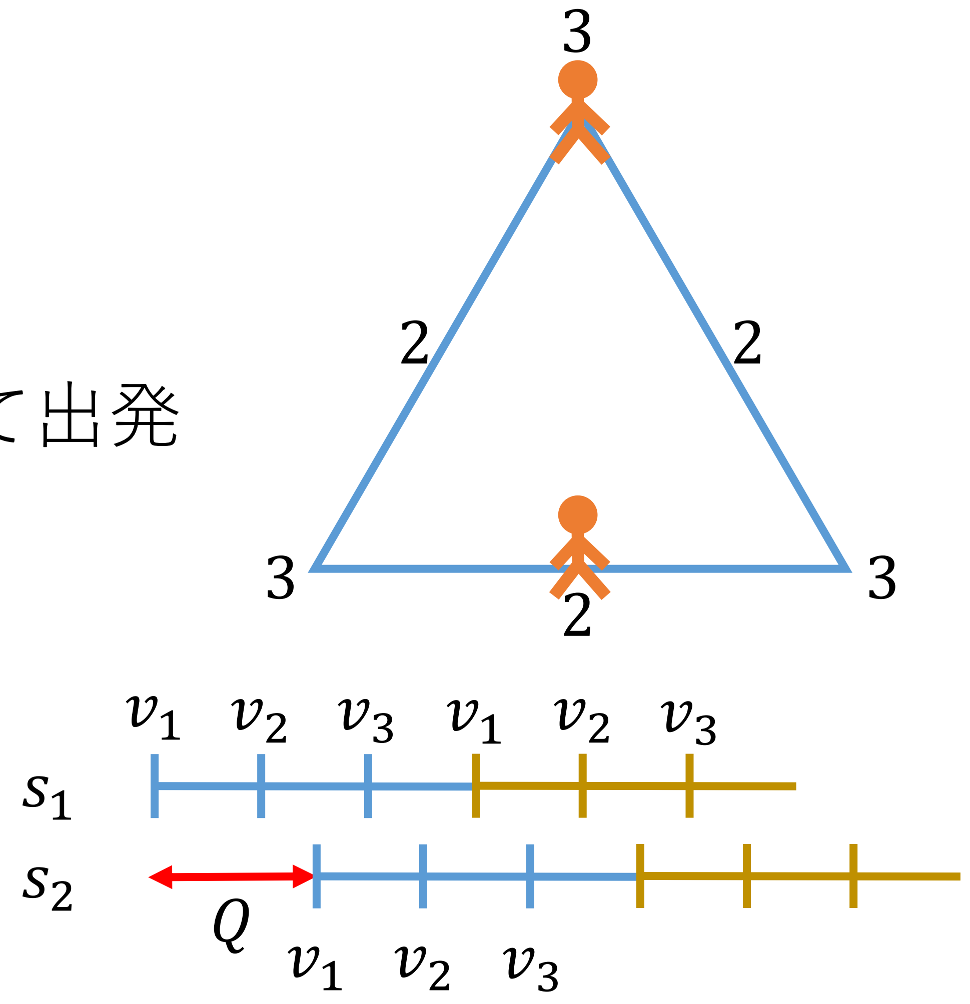
Compで放置可能時間が全て等しい場合，巡査が複数でもOptimizePPに多項式時間アルゴリズムが存在する．

- 訪問のコストが全頂点で等しい
- 利得の大きいものから $\left\lfloor \frac{mQ}{d} \right\rfloor$ 個の頂点を警備できる

m : 巡査の数
Q : 放置可能時間
d : 辺の長さ

定理2の証明

1. 巡査 $s_1, s_2, \ldots, s_m$ が時間 Q ずつ遅れて出発
   利得の多い頂点から順に訪問
   $\rightarrow \left\lfloor \frac{mQ}{d} \right\rfloor$点警備できる



2. $\left\lfloor \frac{mQ}{d} \right\rfloor$個より多くの頂点は警備できない（証明略）
   のでこれが最適解

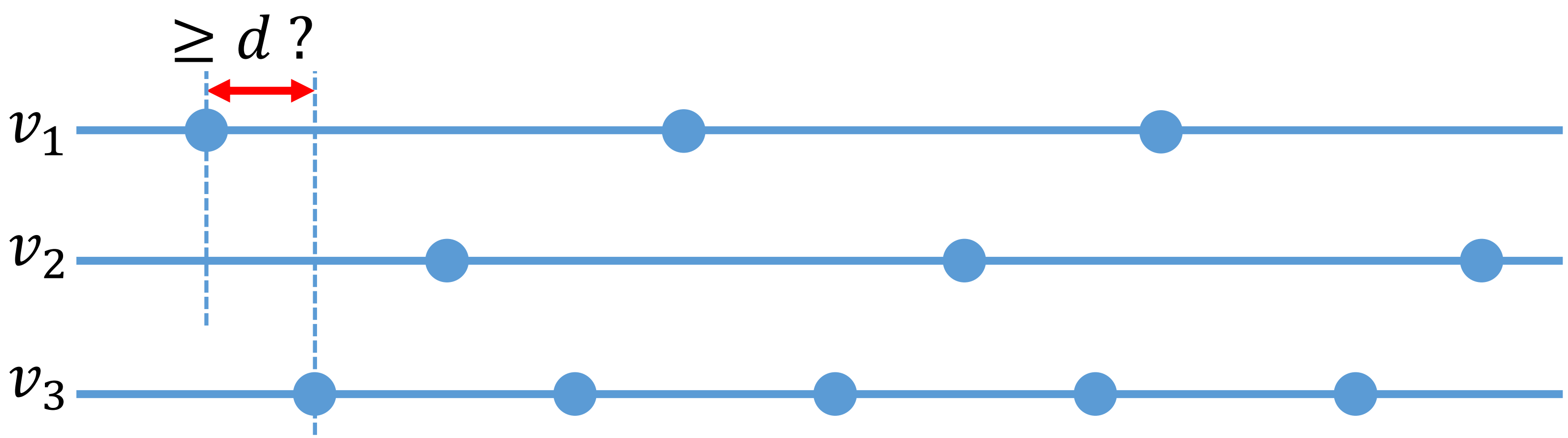
Comp：放置可能時間が一般のとき

- 放置可能時間が全て等しい → 多項式時間アルゴリズムあり
- 放置可能時間が一般の場合 → 未解決
  → ここでも，最初の訪問時刻と訪問の周期が与えられる問題を代わりに考えてみる
  - DecisionPP
    - 巡査が1人　→ 簡単に解ける
  - OptimizePP
    - 巡査が1人で利得が全て等しくてもNP困難

Comp：最初の訪問時刻と訪問の周期指定

DecisionPP, 巡査1人

- 任意の異なる2頂点間の移動に最短 d の時間がかかる
- どの2点についても，訪問しなければならない時刻の間隔が d 以上になっているか調べればよい（詳細略）



Comp：最初の訪問時刻と訪問の周期指定

OptimizePP → NP困難（最大独立点集合問題から帰着）

- 最大独立点集合問題（NP完全問題）
  n点からなる無向グラフが与えられたときに
  独立点集合のうちサイズが最大のものを求める

帰着

- OptimizePP
  n点からなるComp が与えられたときに
  警備できる頂点の数の最大値を求める

独立点集合という制約をどう表現するか？

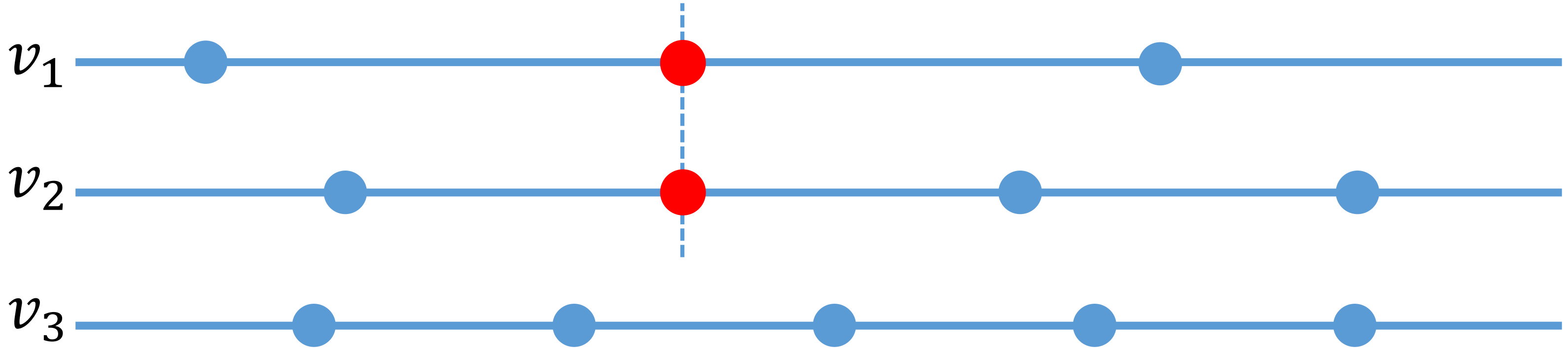
利得を全て1としたときの利得の合計の最大値

Comp：最初の訪問時刻と訪問の周期指定

- 最大独立点集合問題
  「間に辺がある2頂点の両方は選べない」

- OptimizePP
  「2頂点の訪問しなければならない時刻が重複しているので両方は警備できない」

こうなるように
最初の訪問時刻と周期を
設定すれば良い（詳細略）



Comp

- 最初の訪問時刻と訪問の周期が与えられる問題
  - DecisionPP
    - 巡査が1人　→ 簡単に解ける
    - 巡査が複数 → ？
  - OptimizePP
    - 巡査が1人で利得が全て等しくてもNP困難
- 訪問の周期のみ与える問題
  - 「うまく最初の訪問時刻を設定すれば全点を警備できるか？」
  - DecisionPPで巡査が1人で利得が全て等しくてもNP困難
    （“Disjoint Residue Class Problem”[2]と同等）

[2] A. Kawamura and M. Soejima. Simple strategies versus optimal schedules in multi-agent patrolling. In International Conference on Algorithms and Complexity, pp.261–273. Springer, 2015.

まとめ

複数の巡査の協力を考えるときは難しい

- Lineは放置可能時間が全て同じならば P
  - 巡査の協力を考えなくてよいので単純
- Compも放置可能時間が全て同じならば P
- 放置可能時間が一般の場合はLineもCompも未解決
  - 放置可能時間を周期にした場合
    - CompはDecisionPPで巡査1人でもNP困難
  - 最初の訪問時刻も与えられる場合
    - Compで巡査1人だとDecisionPPならP, OptimizePPならNP困難