

内航ケミカルタンカー船隊に対する 配船計画問題

小林和博*

久保幹雄**

加納敏幸*

*海上技術安全研究所

**東京海洋大学海洋工学部

配船計画問題とは？

配船計画とは？

荷主がオペレーター（海運会社）に貨物輸送を依頼

オペレーター（海運会社）はこれらの貨物を輸送して
運賃収入を得る

引き受けた貨物を輸送するための
各船舶の運航計画

オーダー表



オーダー表

No.	発港	発日	着港	着日	油量
1	門司	1	岡山	2	72
2	大洗	2	和歌山	3	501
3	門司	1	名古屋	3	703
4	大洗	2	坂出	4	1005
5



オーダー表

No.	発港	発日	着港	着日	油量
1	門司	1	岡山	2	72
2	大洗	2	和歌山	3	501
3	門司	1	名古屋	3	703
4	大洗	2	坂出	4	1005
5



オーダー表

No.	発港	発日	着港	着日	油量
1	門司	1	岡山	2	72
2	大洗	2	和歌山	3	501
3	門司	1	名古屋	3	703
4	大洗	2	坂出	4	1005
5



オーダー表

No.	発港	発日	着港	着日	油量
1	門司	1	岡山	2	72
2	大洗	2	和歌山	3	501
3	門司	1	名古屋	3	703
4	大洗	2	坂出	4	1005
5



オーダー表

No.	発港	発日	着港	着日	油量
1	門司	1	岡山	2	72
2	大洗	2	和歌山	3	501
3	門司	1	名古屋	3	703
4	大洗	2	坂出	4	1005
5



港間距離表

	門司	大洗	岡山	和歌山	名古屋	坂出
門司	0	405	552	534	657	791
大洗	405	0	560	501	220	184
岡山	552	560	0	366	127	74
和歌山	534	501	366	0	498	652
名古屋	657	220	127	498	0	738
坂出	791	184	74	652	738	0



船舶仕様

	総積載量	ホールド 1	ホールド 2	ホールド 3	ホールド 4	ホールド 5	空船速力	満船速力
A丸	1700	370	370	360	360	240	13.8	13.2
B丸	1550	344	341	336	292	237	13.8	13.5
C丸	1514	462	423	384	245	-	13.8	13.5
D丸	1301	352	350	335	264	-	12.5	11.7
E丸	1190	322	321	274	273	-	11.5	11
F丸	1549	393	356	290	262	248	12.4	11.5

ホールド：船上の区切られた貨物積載区域

異なる製品を同時に載せられる

ホールド 1

ホールド 2

ホールド 3

ホールド 4

ホールド 5

配船計画

オーダー表

港間距離表

船舶仕様

配船計画

オーダー表

港間距離表

船舶仕様

船舶割当

ホールド割当

訪問順序

スポット用船計画

配船計画

オーダー表

港間距離表

船舶仕様

船舶割当

ホールド割当

訪問順序

スポット用船計画

各船舶の向こう一ヶ月のスケジュール決定

配船表

各船舶のスケジュール運航スケジュールを表にまとめたもの

	A丸		B丸		C丸	
1日					鹿島	+1200
2日	大洗	+500	下関	+600		
3日	東京	-500	相生	+700	坂出	-1200
4日						
5日	倉敷	+1200	田原	-1300	千葉	+600
6日						
7日	鹿島	-1200	和歌山	+700	仙台	-600
8日			大阪	-700		

定式化

二段階に分割

入力

二段階に分割

入力

第一段階：ペアリング問題を解く
「ペア」を得る

二段階に分割

入力

第一段階：ペアリング問題を解く
「ペア」を得る

第二段階：ルーティング問題を解く

二段階に分割

入力

第一段階：ペアリング問題を解く
「ペア」を得る

第二段階：ルーティング問題を解く

答え（配船計画）

定式化

ペアリング問題

ペアリング問題：ペアとは？

	A丸		B丸		C丸	
1日					鹿島	+1200
2日	大洗	+500	下関	+600		
3日	東京	-500	相生	+700	坂出	-1200
4日						
5日	倉敷	+1200	田原	-600 -700	千葉	+600
6日	大分	-1200			名古屋	-600



2つのオーダーをB丸に同時に積載している

オーダー 1 : (下関, 600積) → (田原, 600揚)

オーダー 2 : (相生, 700積) → (田原, 700揚)

ペアリング問題：ペアとは？

	A丸		B丸		C丸	
1日					鹿島	+1200
2日	大洗	+500	下関	+600		
3日	東京	-500	相生	+700	坂出	-1200
4日						
5日	倉敷	+1200	田原	-600 -700	千葉	+600
6日	大分	-1200			名古屋	-600

↓ 下関積み

600

相生積み ↓

600

700

↑ 田原揚げ ↑

600

700

ペアリング問題：ペアとは？

ペアとは？

同時に一隻の船に積み合わせるオーダーの組

ペアの例

オーダー 1	：	(下関,600積)		(田原,600揚)
オーダー 2	：	(相生,700積)		(田原,700揚)

ペアリング問題：タスクとは？

タスクとは？

ペアか、どのペアにも含まれないオーダーのこと

タスク集合とは？

ペアの集合と、どのペアにも含まれないオーダーの和集合

ペアリング問題：タスクとは？

	A丸		B丸		C丸	
1日					鹿島	+1200
2日	大洗	+500	下関	+600		
3日	東京	-500	相生	+700	坂出	-1200
4日						
5日	倉敷	+1200	田原	-600 -700	千葉	+600
6日	大分	-1200			名古屋	-600

↓ 下関積み

600

相生積み ↓

600

700

↑ 田原揚げ ↑

600

700

ペアリング問題：タスクとは？

オーダー 番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	600
4	相生	3	田原	5	700
5	鹿島	1	坂出	3	1200
6	千葉	5	名古屋	6	600

↓ 下関積み

600

相生積み ↓

600

700

↑ 田原揚げ ↑

600

700

ペアリング問題：タスクとは？

オーダー 番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	600
4	相生	3	田原	5	700
5	鹿島	1	坂出	3	1200
6	千葉	5	名古屋	6	600

↓ 下関積み

600

相生積み ↓

600

700

↑ 田原揚げ ↑

600

700

ペアリング問題：タスクとは？

オーダー 番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	600
4	相生	3	田原	5	700
5	鹿島	1	坂出	3	1200
6	千葉	5	名古屋	6	600

ペアリング問題：タスクとは？

タスク 番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	600 700
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ペアリング問題：タスクとは？

オーダー 番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	600
4	相生	3	田原	5	700
5	鹿島	1	坂出	3	1200
6	千葉	5	名古屋	6	600

ペアの集合：

$\{\{3,4\}\}$

どのペアにも含まれない

オーダーの集合：

$\{1,2,5,6\}$

タスクの集合：

$\{\{3,4\},1,2,5,6\}$

ペアリング問題：タスクとは？

オーダー 番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	600
4	相生	3	田原	5	700
5	鹿島	1	坂出	3	1200
6	千葉	5	名古屋	6	600

ペアの集合：

$\{\{3,4\}\}$

どのペアにも含まれない

オーダーの集合：

$\{1,2,5,6\}$

タスクの集合：

$\{\{3,4\}, 1, 2, 5, 6\}$

定式化

ルーティング問題

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

タスク番号	発港	発日	着港	着日	油量
1	大洗	2	東京	3	500
2	倉敷	5	大分	6	1200
3	下関	2	田原	5	1300
4	鹿島	1	坂出	3	1200
5	千葉	5	名古屋	6	600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

	A丸		B丸		C丸	
A丸					鹿島	+1200
タスク 1	2日	大洗 +500	下関	+600		
	3日	東京 -500	相生	+700	坂出	-1200
	4日					
タスク 2	5日	倉敷 +1200	田原	-600 -700	千葉	+600
	6日	大分 -1200			名古屋	-600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

B丸

タスク3

	A丸		B丸		C丸	
1日					鹿島	+1200
2日	大洗	+500	下関	+600		
3日	東京	-500	相生	+700	坂出	-1200
4日						
5日	倉敷	+1200	田原	-600 -700	千葉	+600
6日	大分	-1200			名古屋	-600

ルーティング問題とは？

タスクを順に並べて船のスケジュールを作成する

C丸	A丸		B丸		C丸		
タスク 4	1日				鹿島	+1200	
	2日	大洗	+500	下関	+600		
	3日	東京	-500	相生	+700	坂出	-1200
	4日						
タスク 5	5日	倉敷	+1200	田原	-600 -700	千葉	+600
	6日	大分	-1200			名古屋	-600

どう解くか？

ルーティング問題

ルーティング問題：アプローチ

各船舶ごとに、運航可能なスケジュールを
あらかじめ列挙しておく

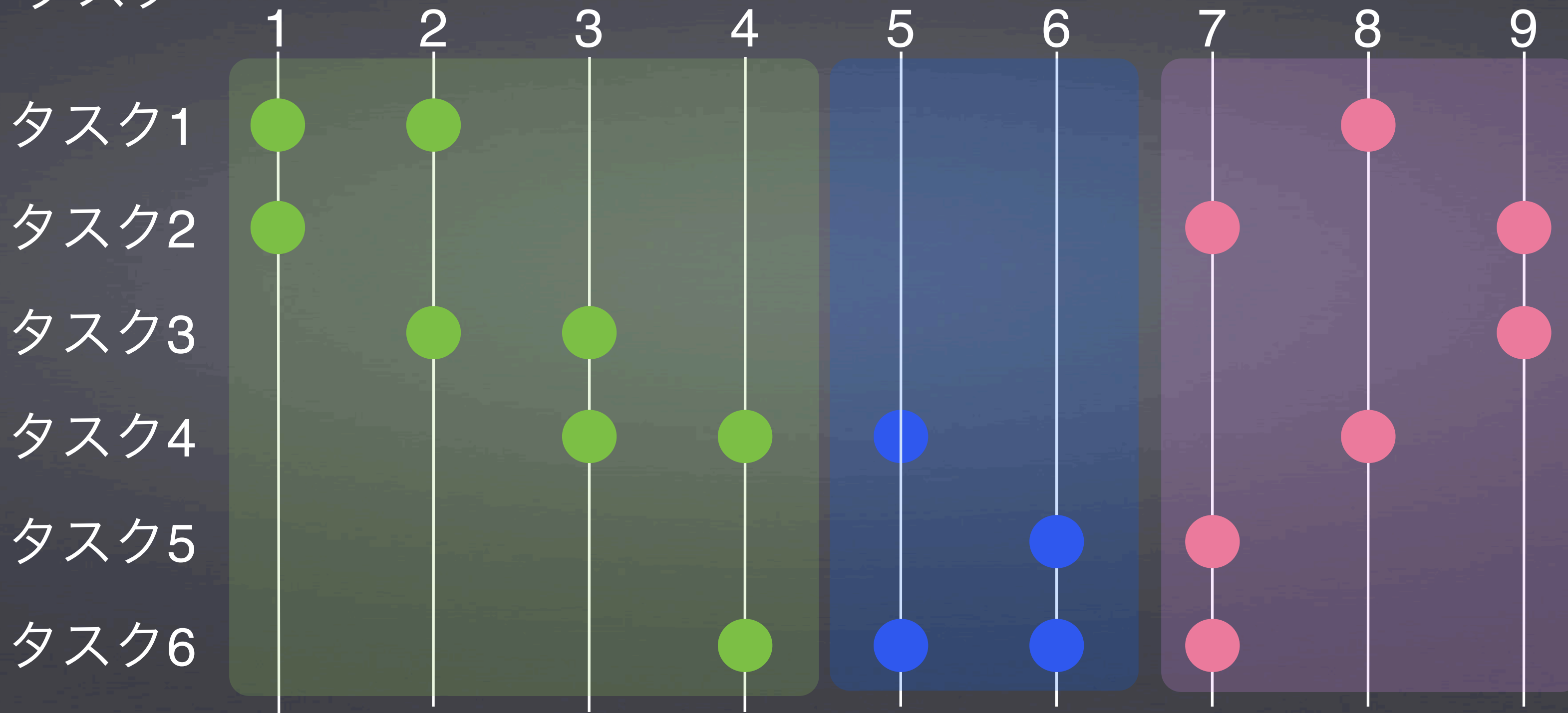
全てのタスクが処理されるように、
各船舶の運航スケジュールをうまく選んでくる

ルーティング問題：アプローチ

- 船1の候補
- 船2の候補
- 船3の候補

候補ルート

タスク

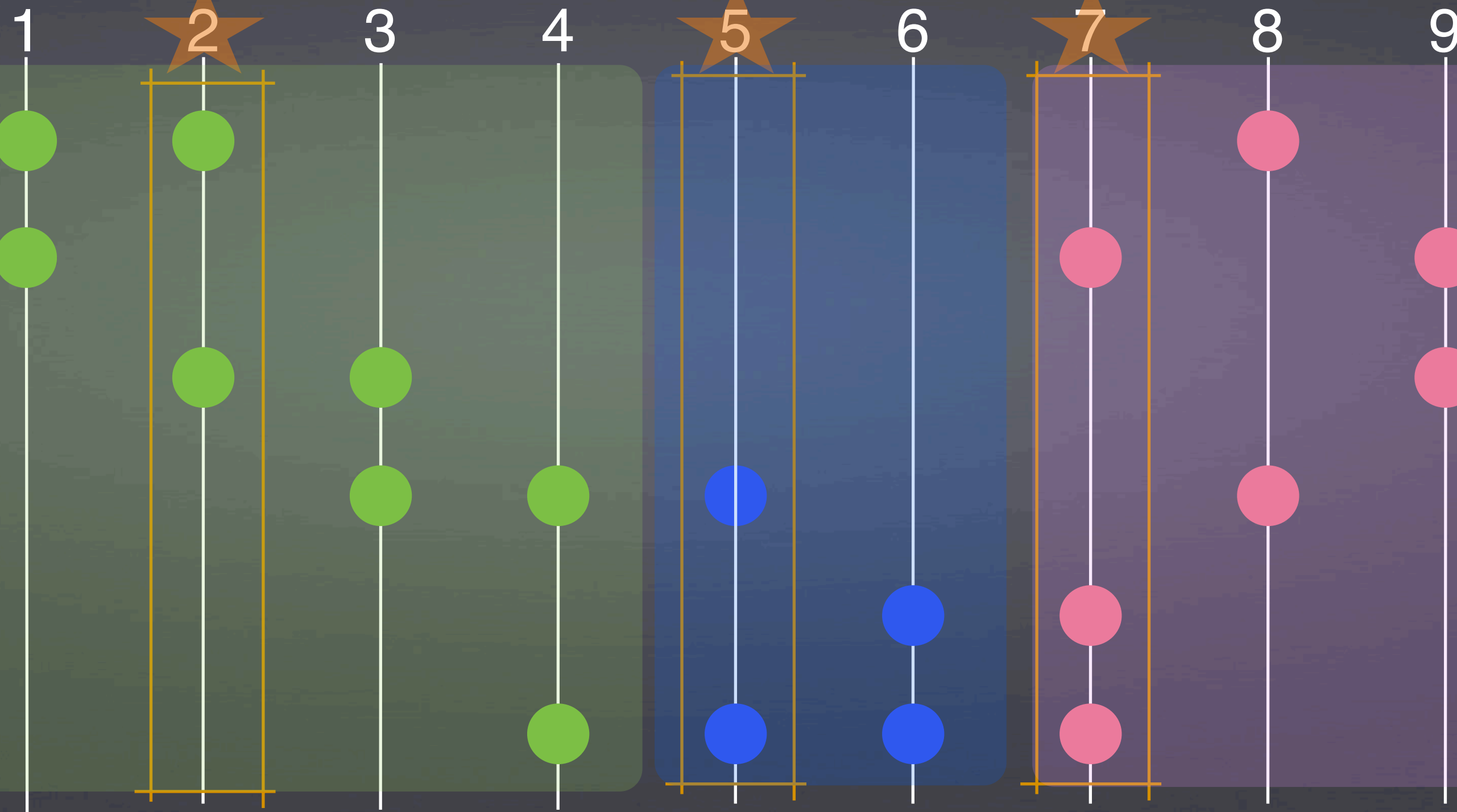


ルーティング問題：アプローチ

- 船1の候補
- 船2の候補
- 船3の候補

候補ルート

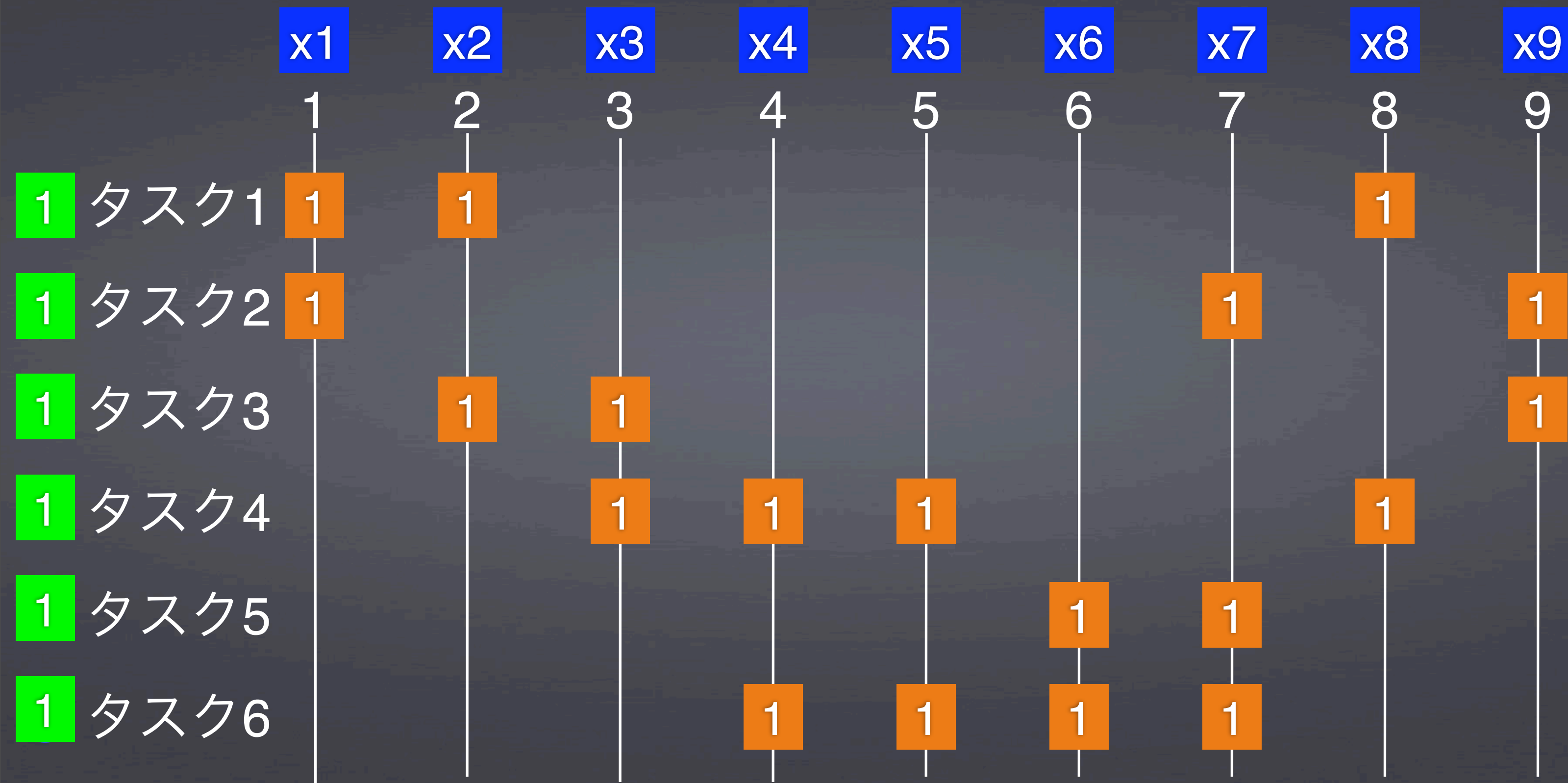
タスク



- タスク1
- タスク2
- タスク3
- タスク4
- タスク5
- タスク6

ルーティング問題：アプローチ

候補ルート



整数計画問題への定式化

整数計画問題への定式化

最小化 $c^T x$

制約条件 $Ax = b$
 x は0または1の変数

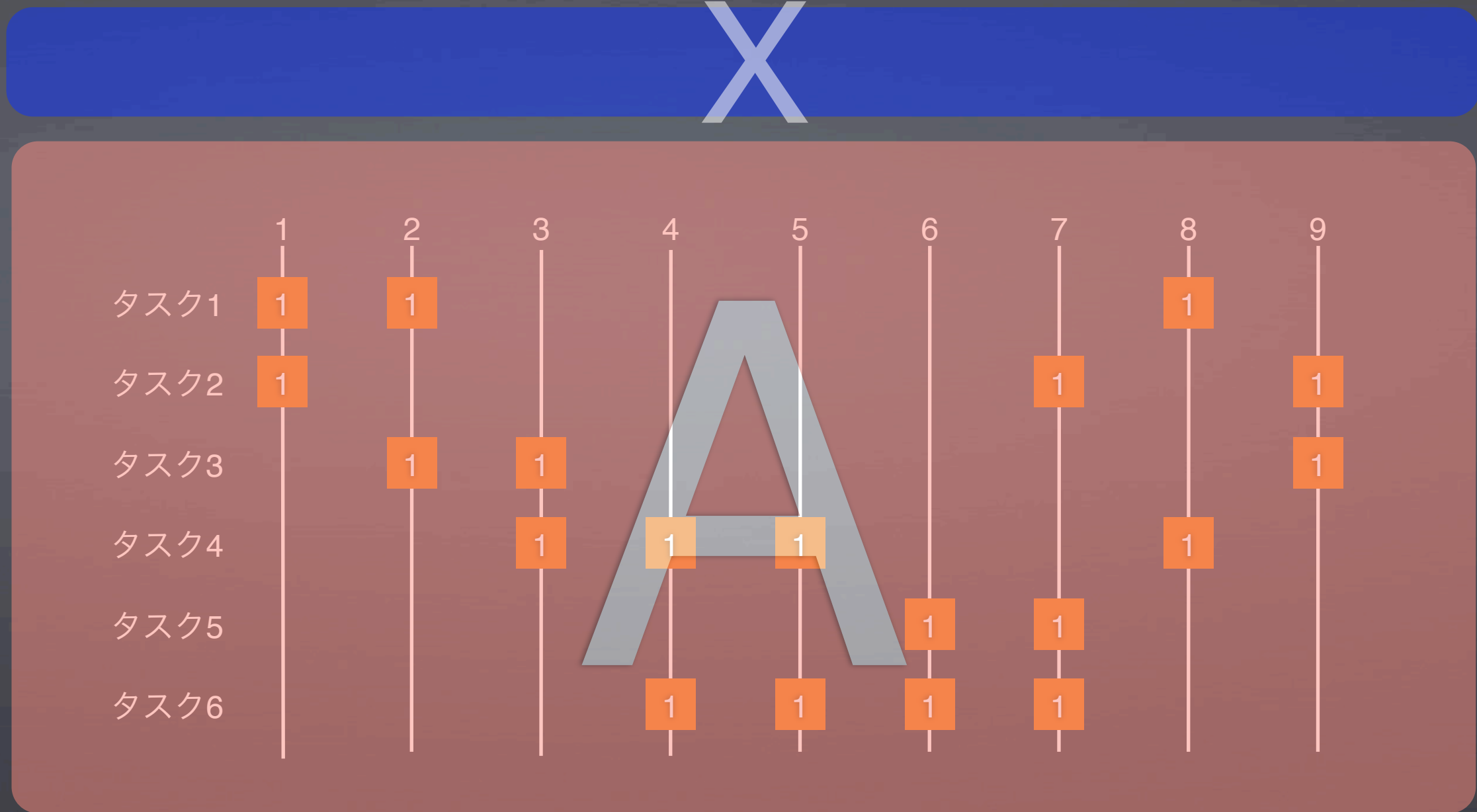
c, x : n 次元ベクトル, A : $m \times n$ の行列

整数計画問題への定式化

制約条件 $Ax = b$ $c, x: n$ 次元ベクトル, $A: m \times n$ の行列

b

=



整数計画問題としての定式化

目的関数

船舶の総航行距離の最小化

各候補ルートでの航行距離をコストと定義する



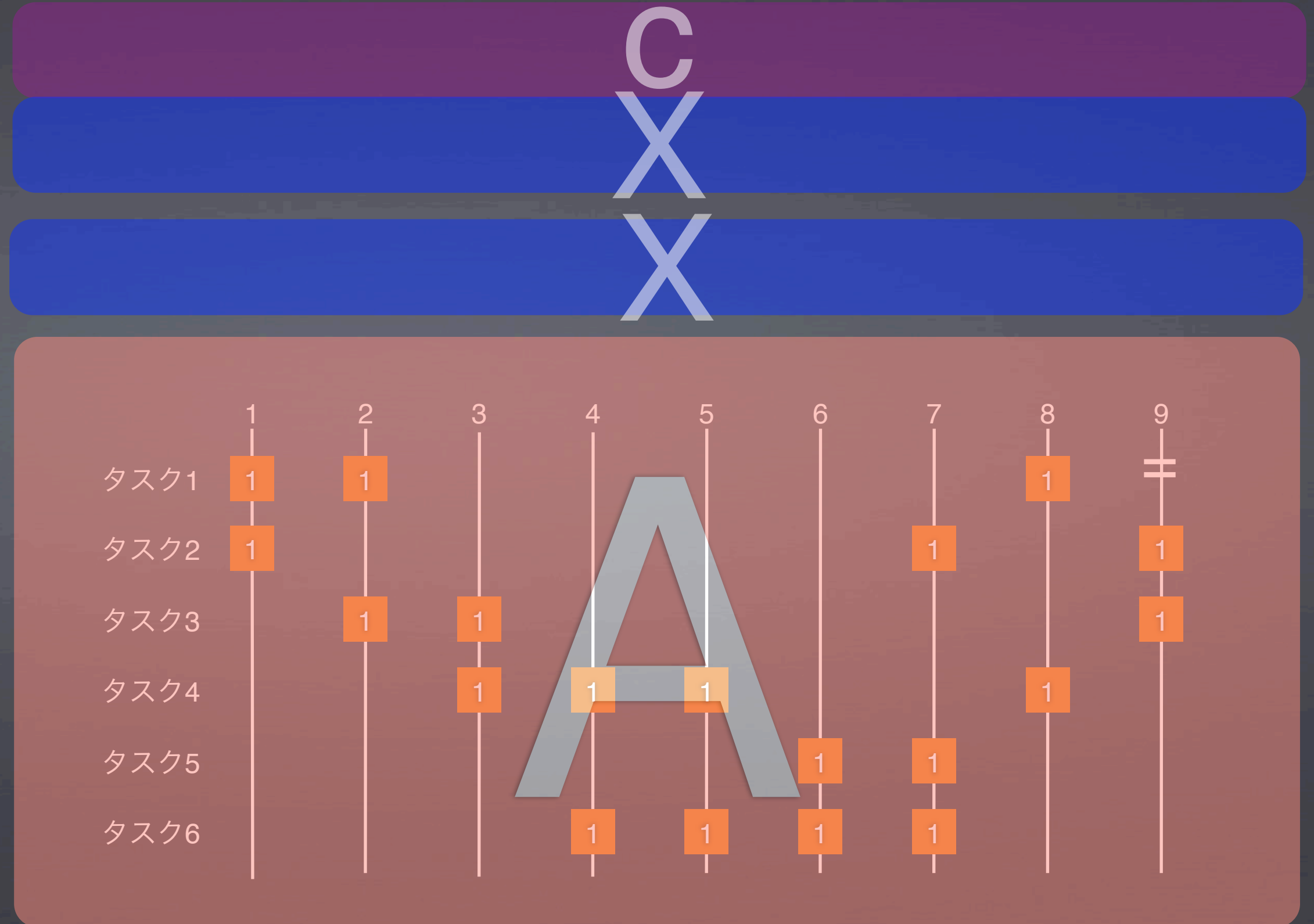
整数計画問題への定式化

最小化

制約条件

b

=



整数計画問題としての定式化

制約条件

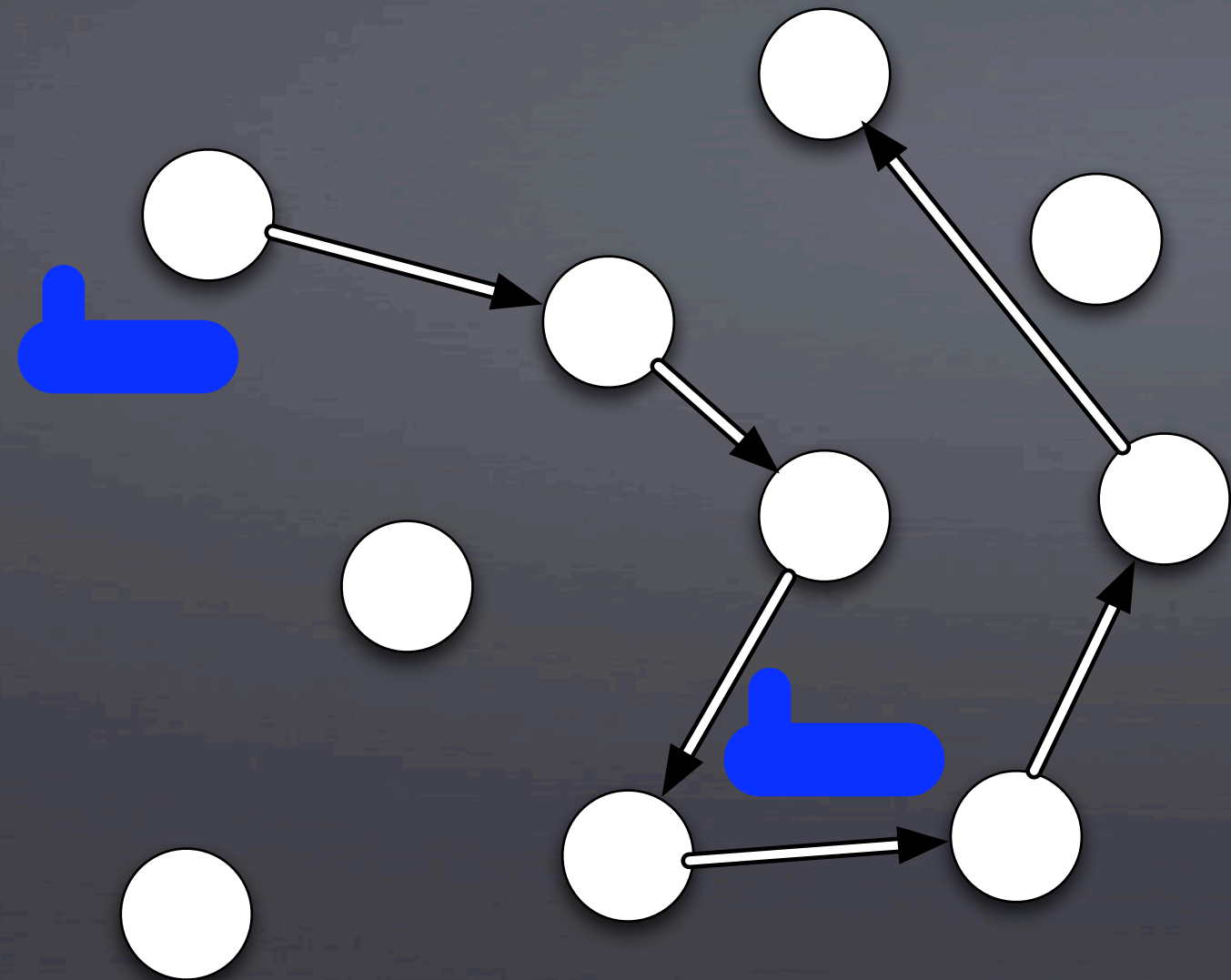
1. 全てのオーダーを処理する
2. 船への積載の制約：積載量、ホールド割当
3. 時間枠制約：積日、揚日までに運ぶ
4. 荷役時刻制約：港での荷役可能時刻の制約
(例:8:00-15:00)

整数計画問題としての定式化

制約条件を満たす候補ルートを見つける



各船舶ごとの時刻依存最短路問題を解けば見つかる



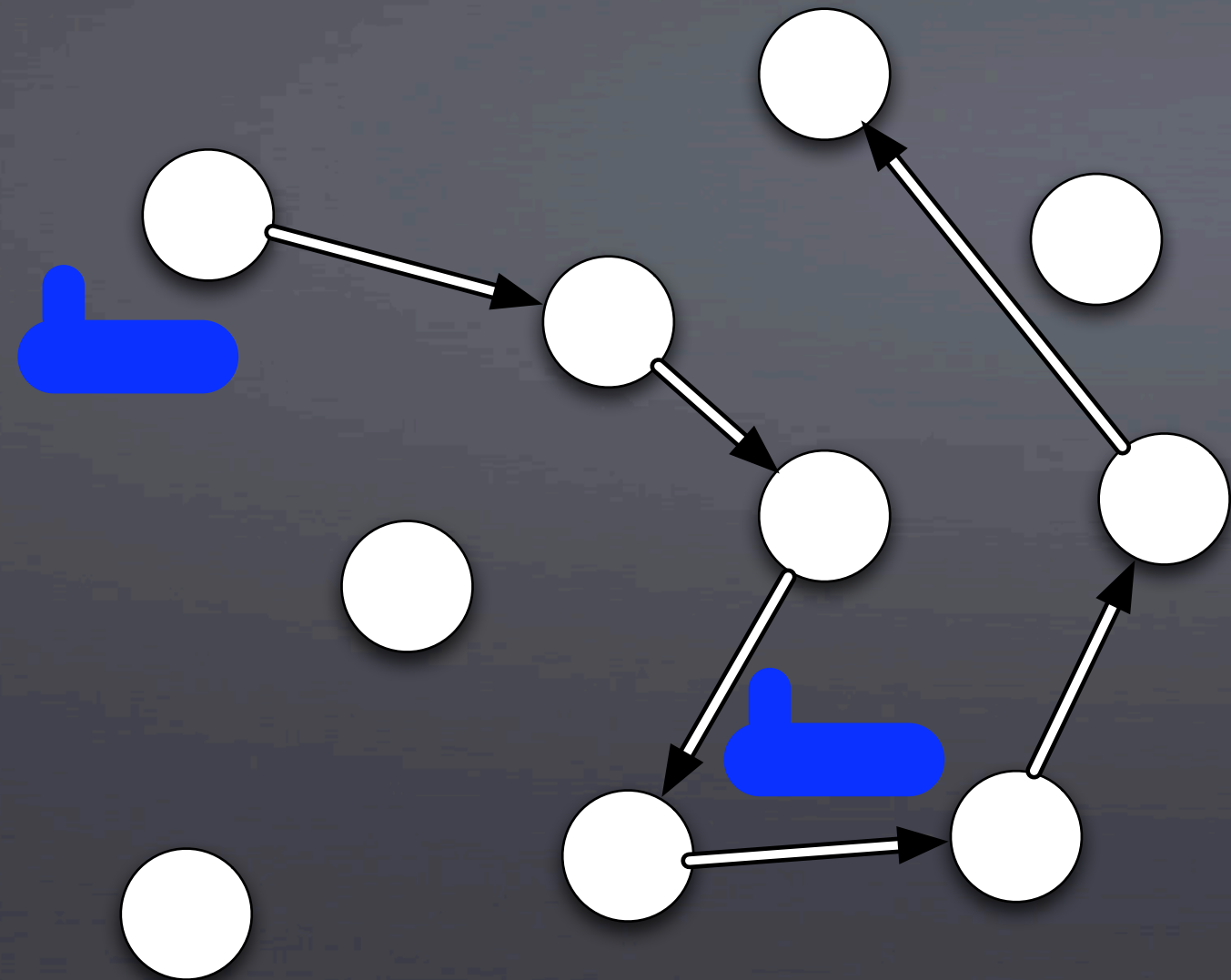
得られた最短路を
制約条件 $Ax=b$ の
行列 A の列に加える

整数計画問題としての定式化

制約条件を満たす候補ルートを見つける



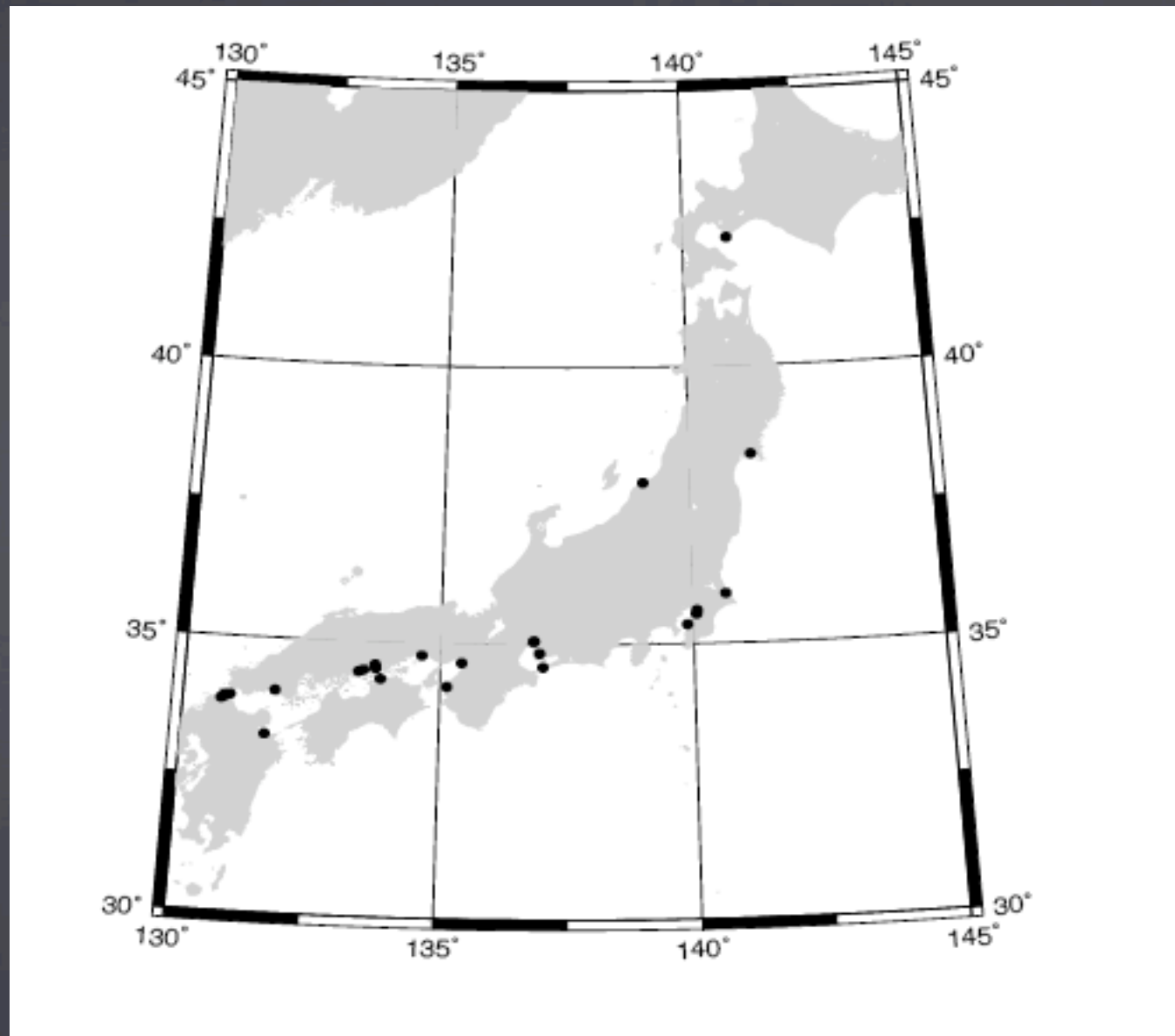
各船舶ごとの時刻依存最短路問題を解けば見つかる



得られた最短路を
制約条件 $Ax=b$ の
行列 A の列に加える

数值実験

事例：内航ケミカルタンカー船社1



寄港地

36日分128オーダー、25港、7隻

スポット傭船処理オーダー数：10

総航行距離：38973海里

数値実験環境

演算装置：Intel Core 2 6700 2.66GHz

メモリ 2GB RAM

OS Windows XP

モデリング言語：Xpress Mosel Version 1.6.3

Dash Optimization社

線形計画及び整数計画ソルバー：Xpress Optimizer Version

17.10.04

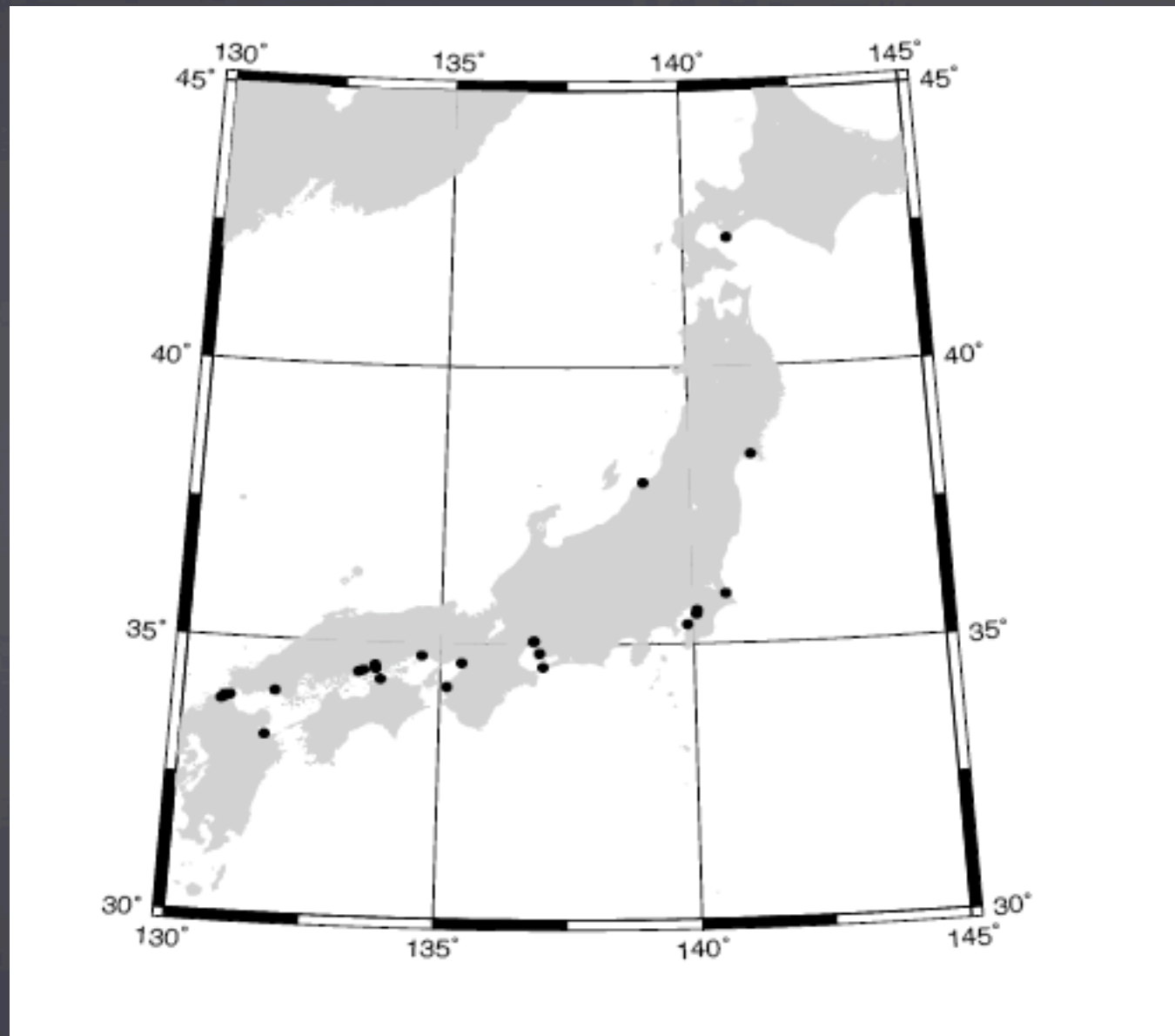
Dash Optimization社

実験結果(36日分)

提案手法によって得られた配船計画のコスト (実績のペアを使用)

	傭船処理	総航行距離	実行時間(秒)	距離減少率
傭船処理最小化	9	38708	547	0.7%
傭船処理 ≤ 10 で距離最小化	10	36105	558	7.9%
傭船処理 ≤ 11 で距離最小化	11	35846	539	8%

事例：内航ケミカルタンカー船社2



寄港地

26日分95オーダー、25港、7隻

スポット傭船処理オーダー数：9

総航行距離：27086海里

実験結果(26日分)

提案手法によって得られた配船計画のコスト（実績のペアを使用）

	傭船処理	総航行距離	実行時間(秒)	距離減少率
傭船処理最小化	6	27000	240	0.3%
傭船処理 ≤ 10 で距離最小化	9	25617	249	5.4%
傭船処理 ≤ 11 で距離最小化	10	25325	238	6.5%

謝辞

本研究は、
独立行政法人新エネルギー・産業技術総合開発機構(NEDO)
先導研究開発
「内航船の環境調和型運航計画支援システムの研究開発」
の一部として行った。