

# 情報可視化

## LX8: 時間の表現

数理・計算科学系

脇田 建

# データ

```
import scipy as sci
import pandas as pd

pd.DataFrame(sci.random.randint(0, 100, size=(10, 4)), columns=list('ABCD'))
```

	A	B	C	D
0	87	20	79	57
1	31	59	30	95
2	72	68	66	82
3	40	26	38	55
4	14	63	45	20
5	56	5	77	45
6	21	40	12	86
7	53	72	36	65
8	43	30	84	2
9	5	87	72	98

# 時間変化するデータ

```
import scipy as sci
import pandas as pd

pd.DataFrame(sci.random.randint(0, 100, size=(10, 4)), columns=list('ABCD'))
```

	A	B	C	D
0	87	20	79	57
1	31	59	30	95
2	72	68	66	82
3	40	26	38	55
4	14	63	45	20
5	56	5	77	45
6	21	40	12	86
7	53	72	36	65
8	43	30	84	2
9	5	87	72	98

$t_0$

	A	B	C	D
0	81	36	83	64
1	35	63	56	43
2	64	90	86	35
3	60	62	68	0
4	72	27	92	37
5	84	2	88	46
6	37	19	17	9
7	26	55	95	87
8	7	70	35	15
9	42	37	12	33

$t_1$

	A	B	C	D
0	65	47	52	72
1	79	7	68	5
2	45	77	56	70
3	83	28	29	84
4	63	59	75	24
5	29	79	15	83
6	55	44	13	37
7	46	59	71	4
8	84	69	72	53
9	39	98	26	41

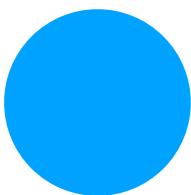
$t_2$

# 今日の話題

- 平面上に時間を描く方法
- 時間の切斷と集約
- 空間の切斷と集約
- データの操作 – インタラクション

# 平面上に時間を描く手法

## 静的な可視化手法



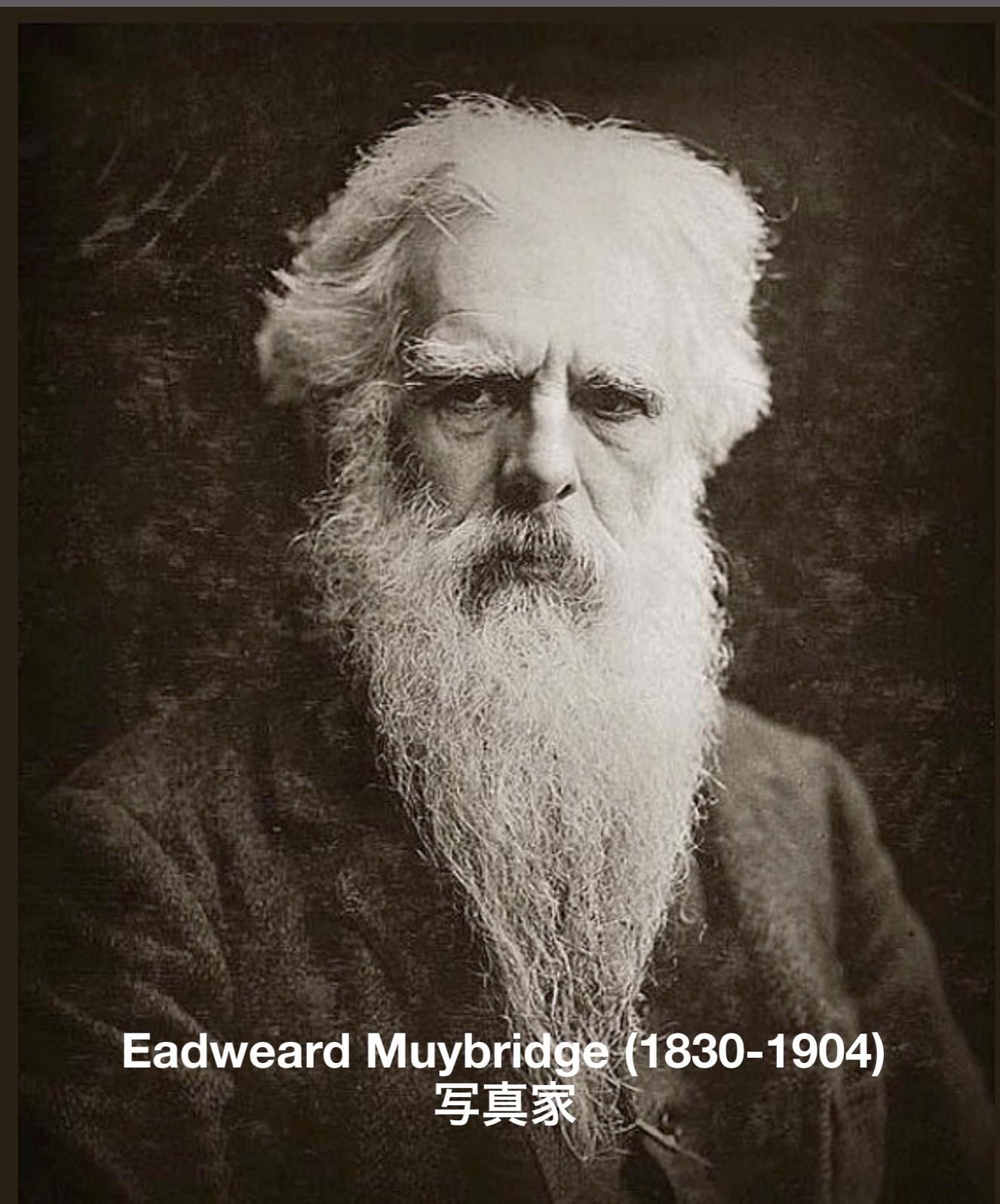
# 時間の切斷と集約

“駆歩（ギャロップ）の馬は一瞬でも  
四本の足を地面から離すのだろうか”

– 馬の駆歩問題 (1872)

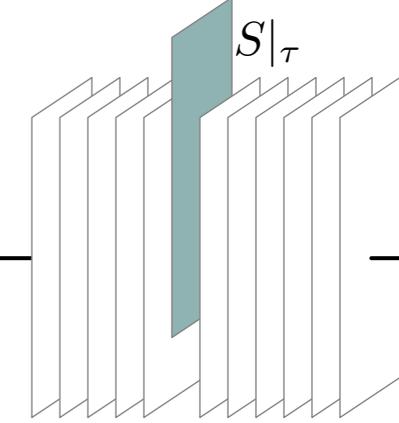


**Leland Stanford (1824-1893)**  
大立者、実業家、政治家



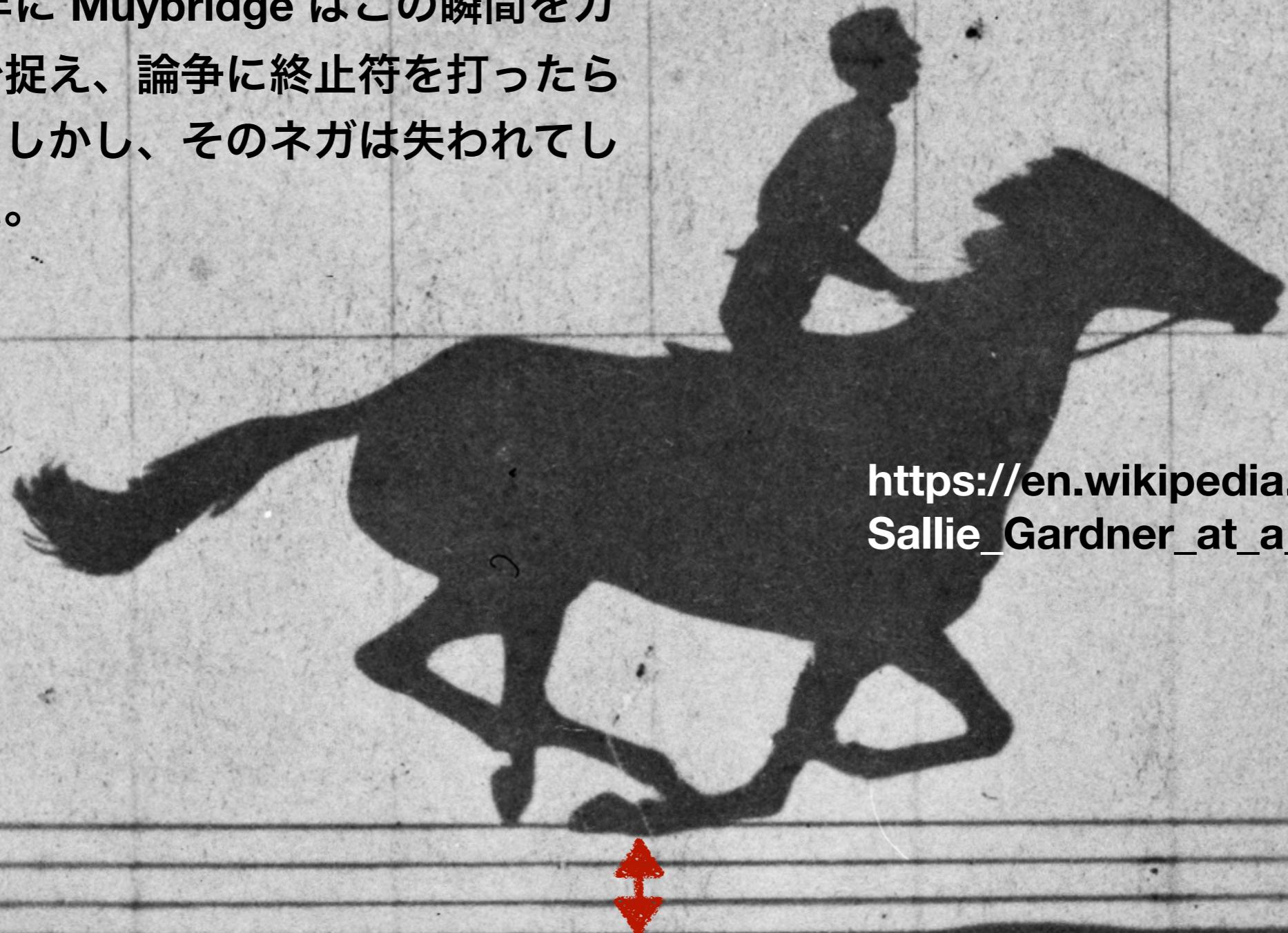
**Eadweard Muybridge (1830-1904)**  
写真家

# 脚が地面を離れた瞬間



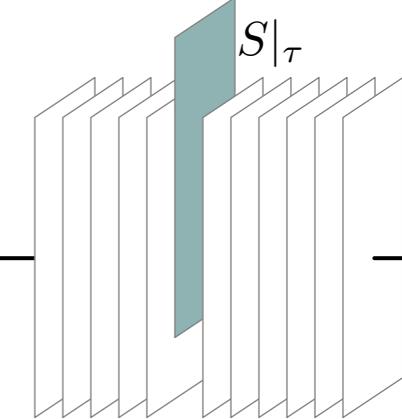
4 5 6 7 8 9 10

1872年に Muybridge はこの瞬間をカ  
メラで捉え、論争に終止符を打つたら  
しい。しかし、そのネガは失われてし  
まった。

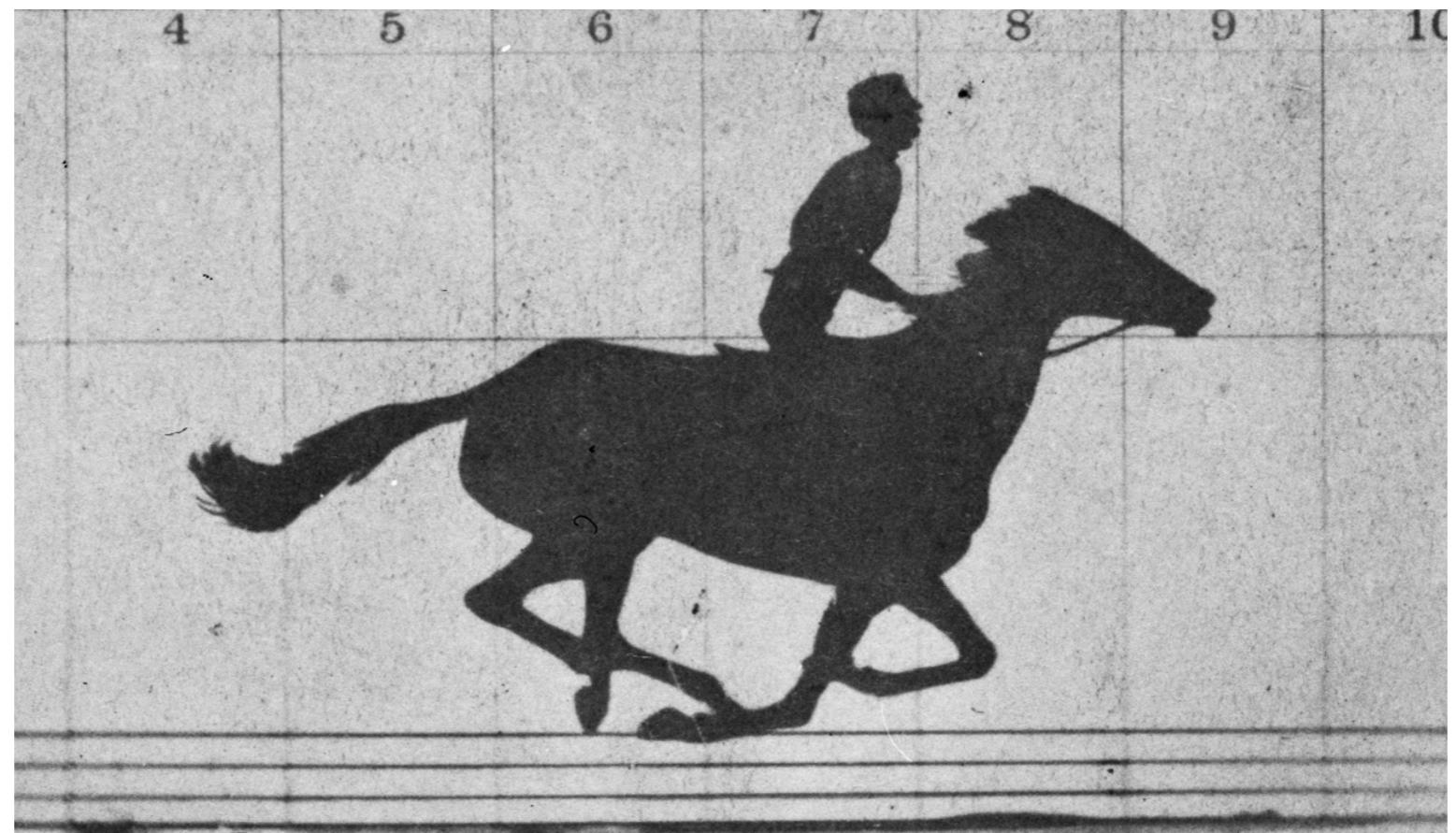


[https://en.wikipedia.org/wiki/  
Sallie\\_Gardner\\_at\\_a\\_Gallop](https://en.wikipedia.org/wiki/Sallie_Gardner_at_a_Gallop)

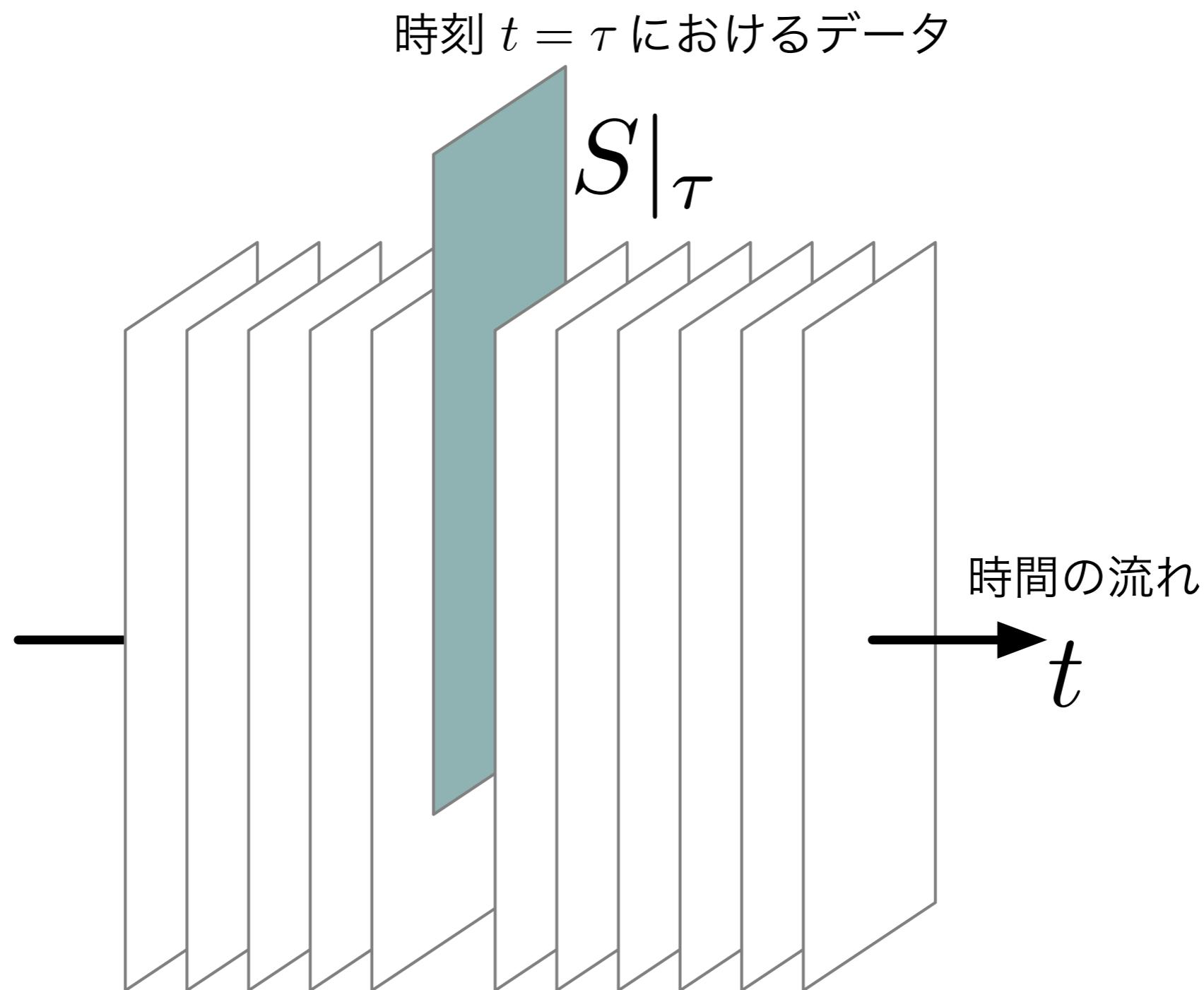
# 激写！



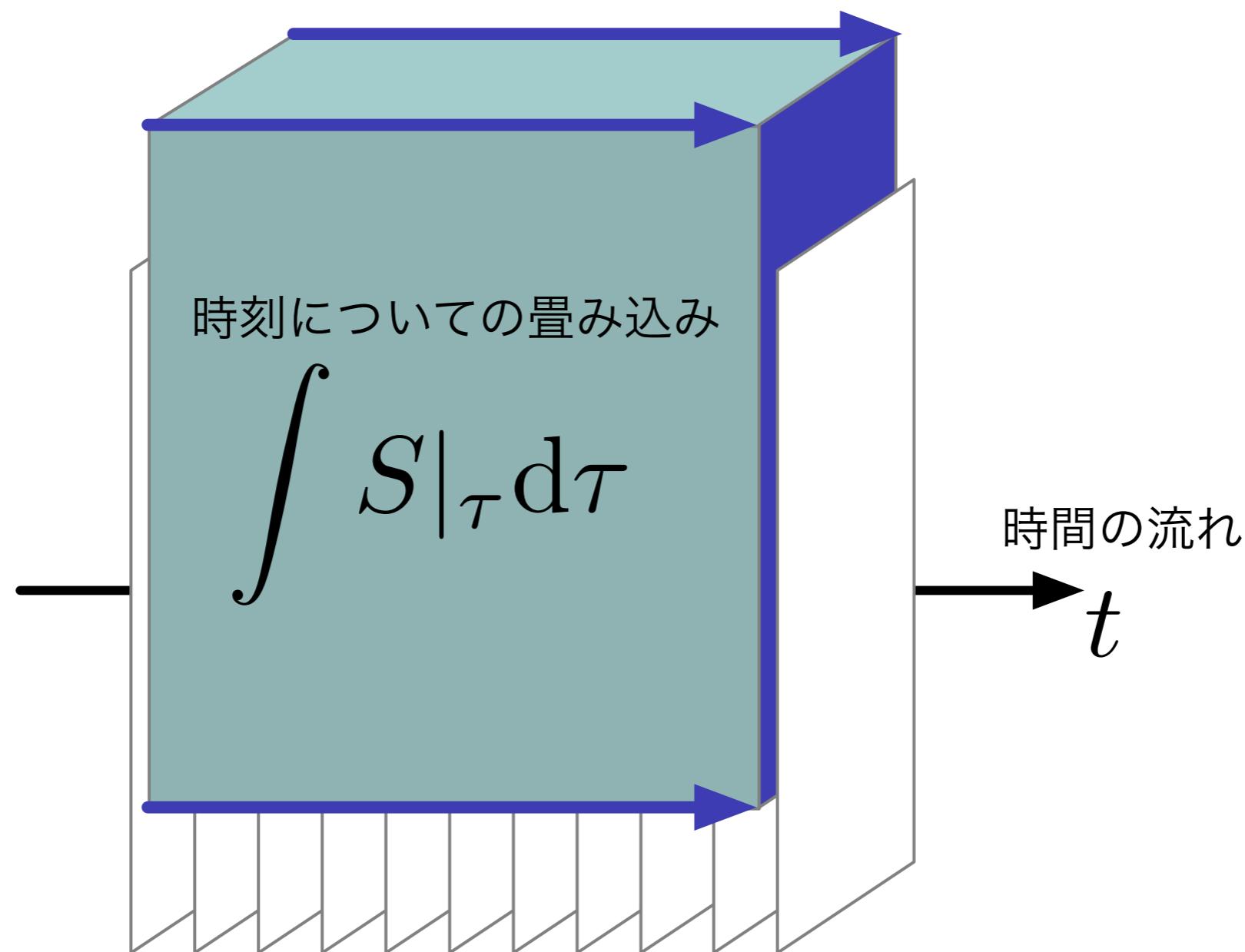
- Time:  $(0, T) - 0$ : レース開始時刻、 $T$ : レース終了時刻
- Space:  $\tau \rightarrow (\text{馬}_{t=\tau})$  - 時刻 $\tau$ における馬の様子
- 激写!: 馬 $\tau$



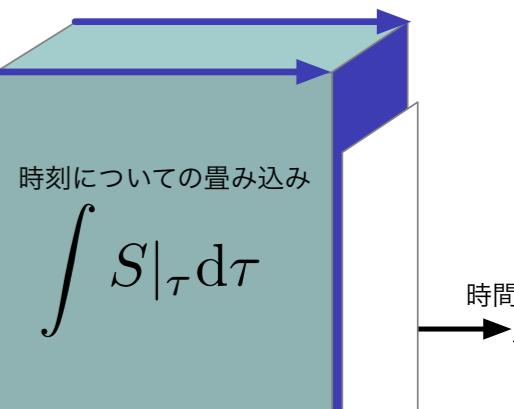
# 時間の切斷（激写！）



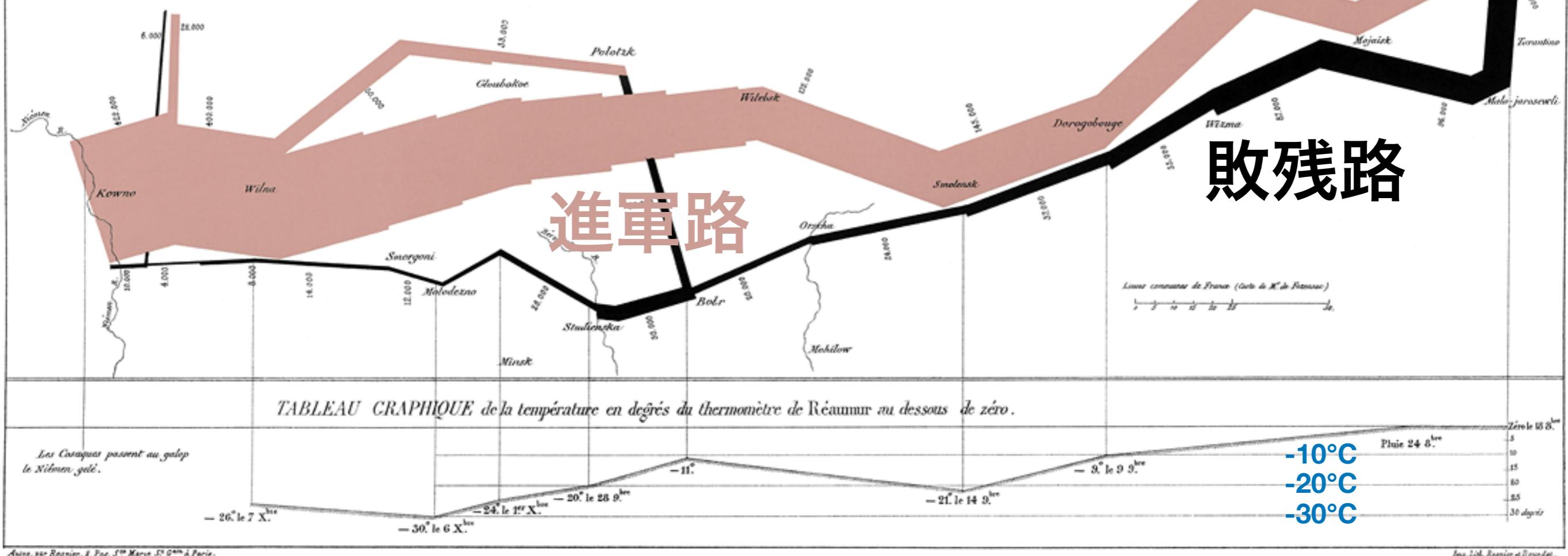
# 時間方向の集約（長時間露光）



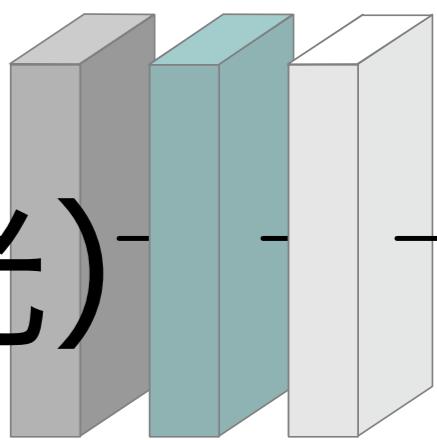
# 死のモスクワ遠征行



1812年ナポレオンは42万4千人の兵を率いて Niemen を渡河した。  
モスクワ攻略に失敗した兵を冬将軍が全滅に追い込んだ。再び  
Niemen を目についたものはわずか1万人だった。

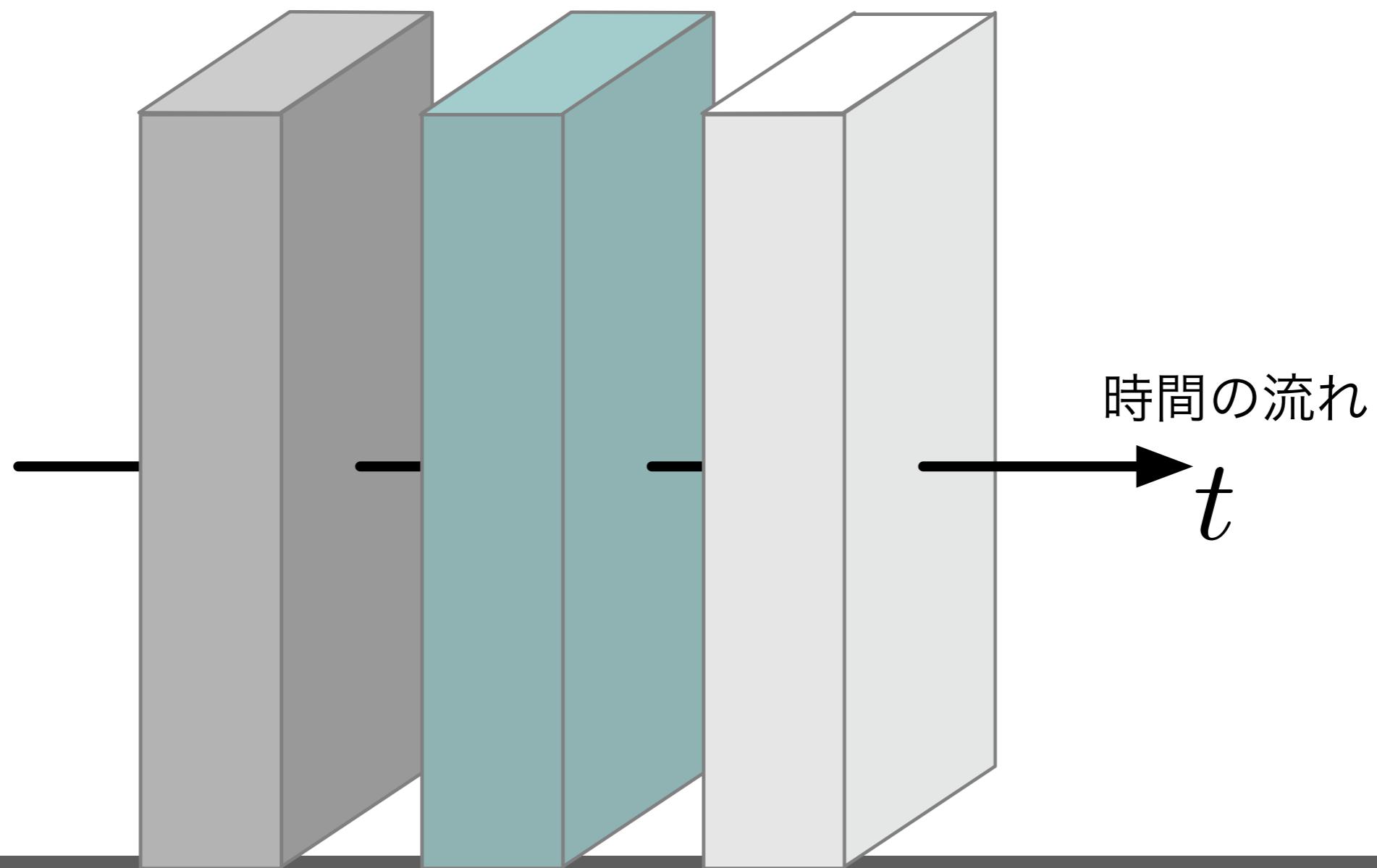


# 時刻片の切り出し（多重露光）

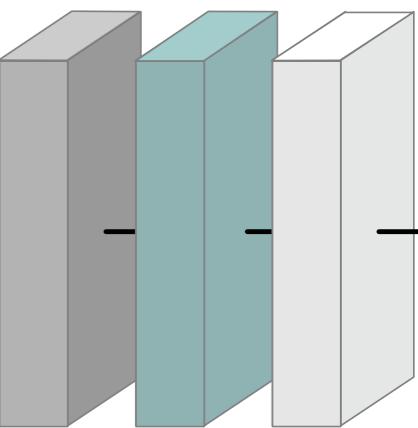


# 多重切断（多重露光）

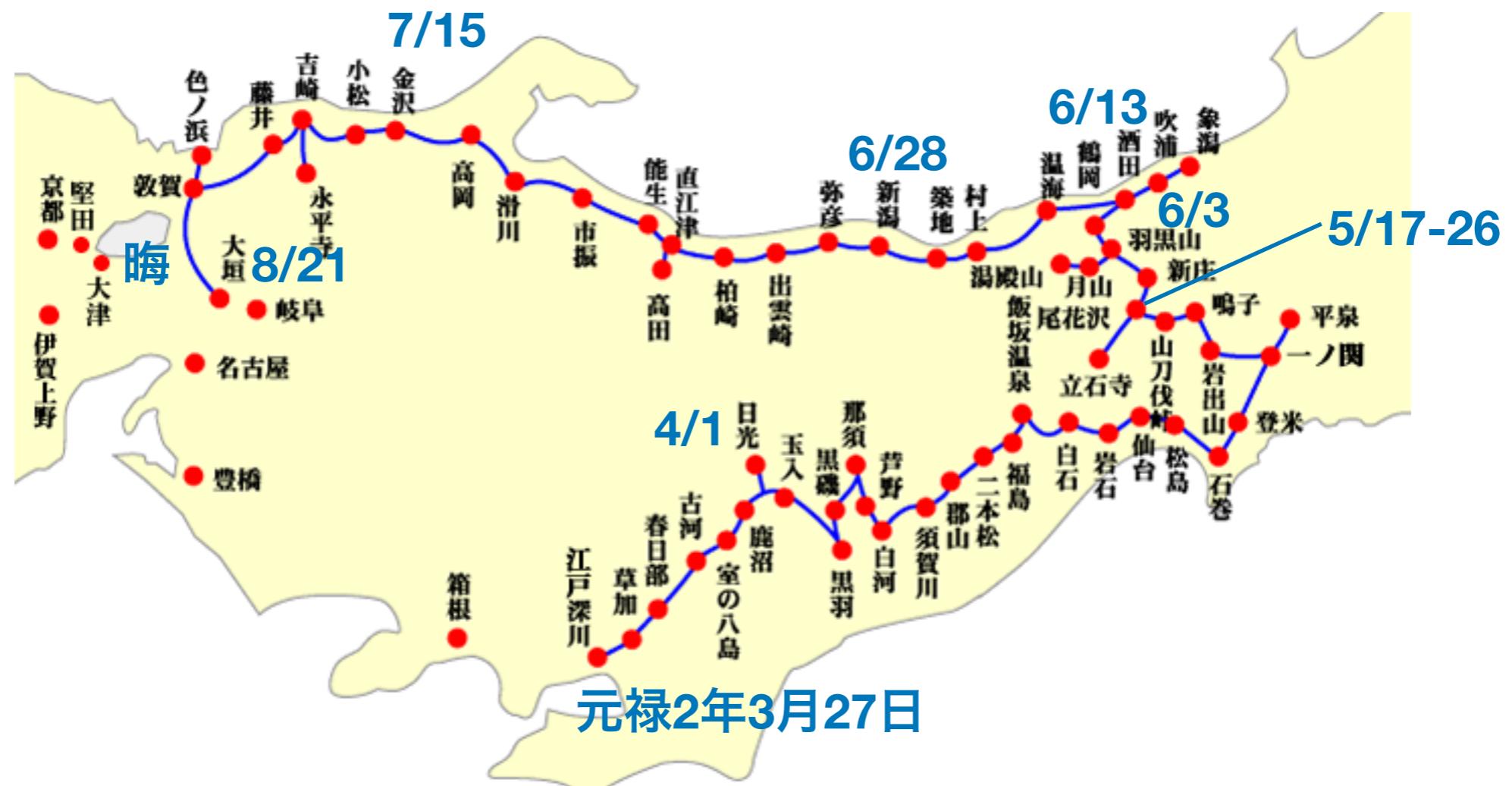
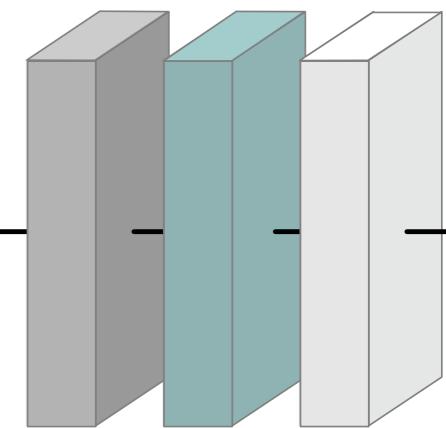
断続的な時刻片の切り出し



# 多重切斷 (多重露光)



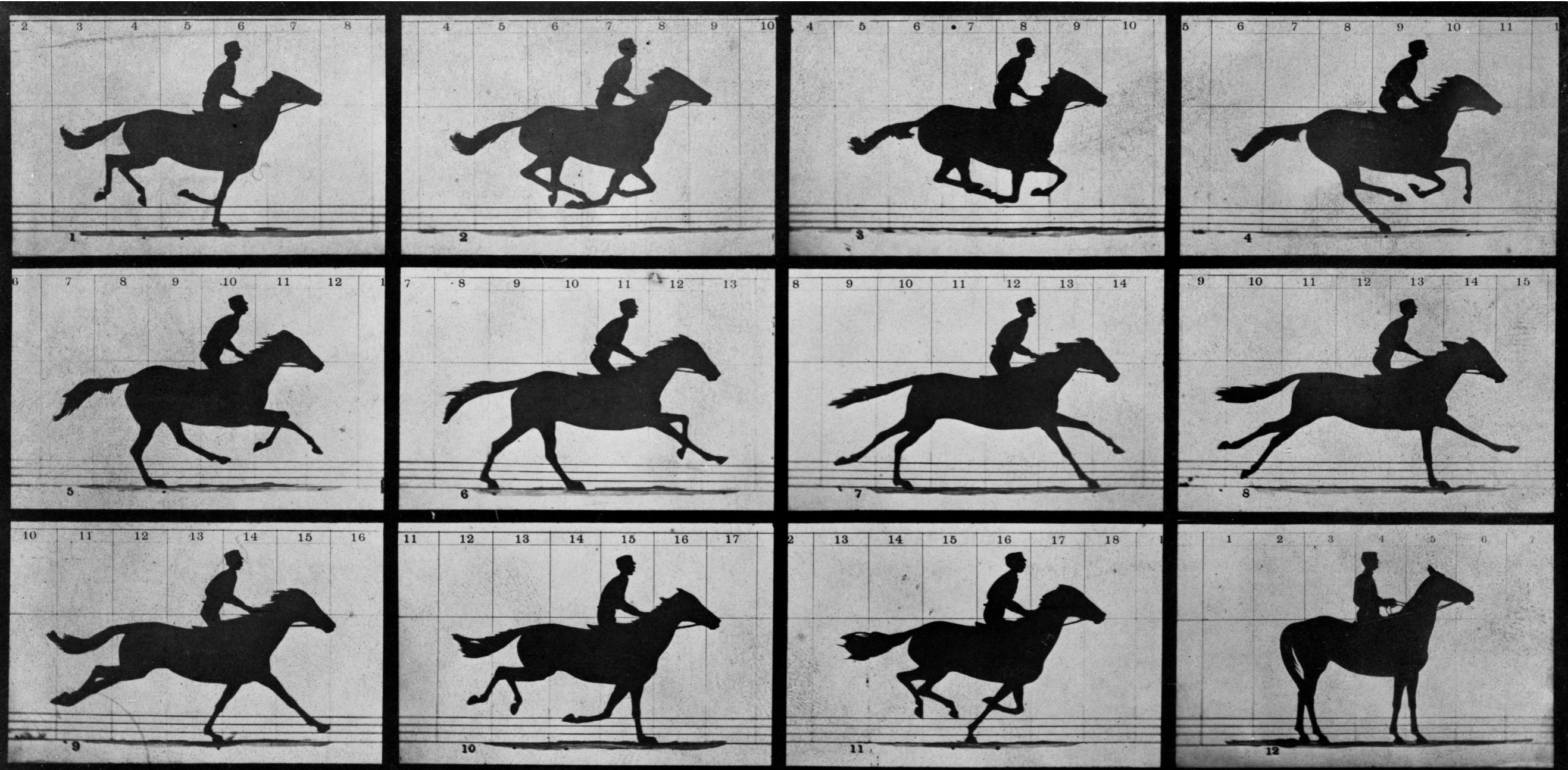
# 奥の細道



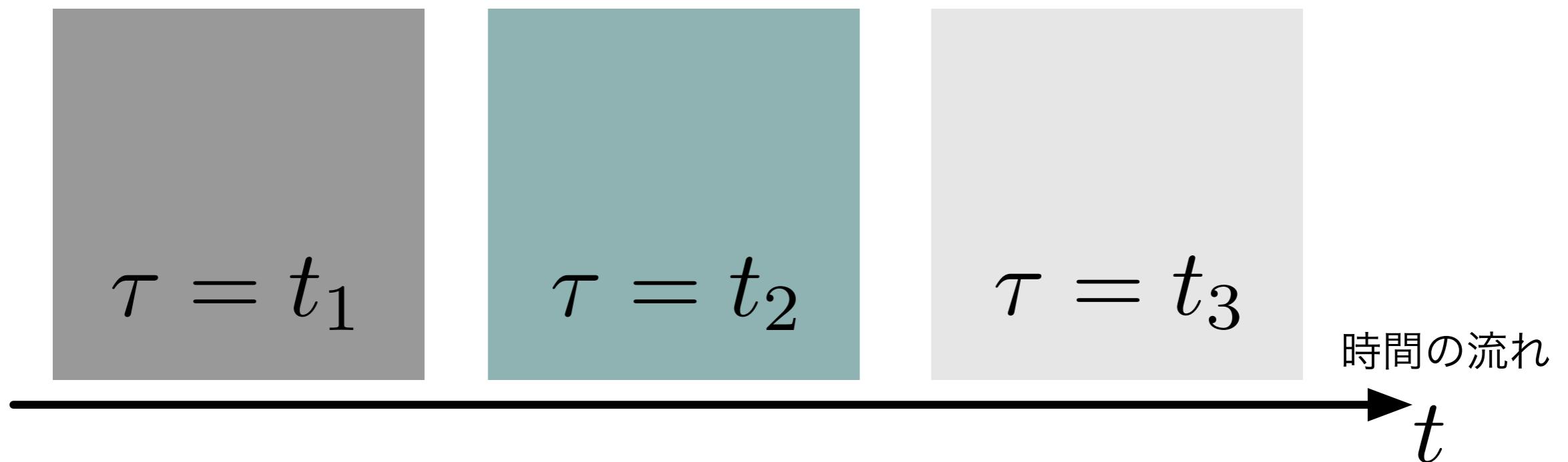
芭蕉の行動を一日を代表する地点にプロット（時間方向の集約）し、重畠表  
示したものと考えられる。モスクワ遠征行における時間は連続的表現だった  
が、この可視化では離散的。（連続的な動きは青い線で描かれている）

# 時分割並置

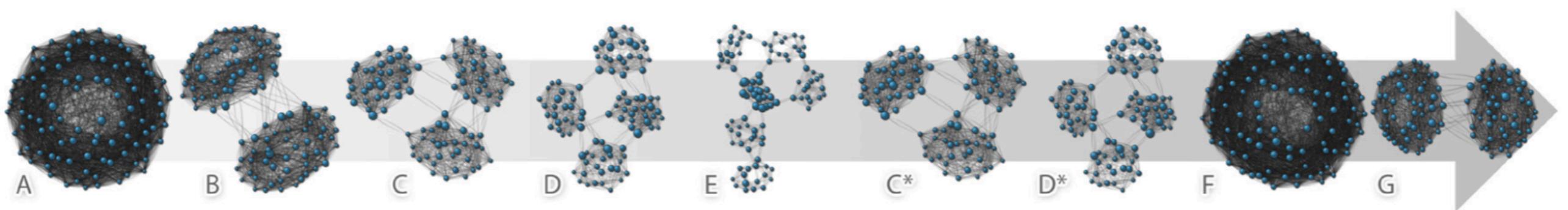
$\tau = t_1$     $\tau = t_2$     $\tau = t_3$



# 時分割並置



# 時分割並置

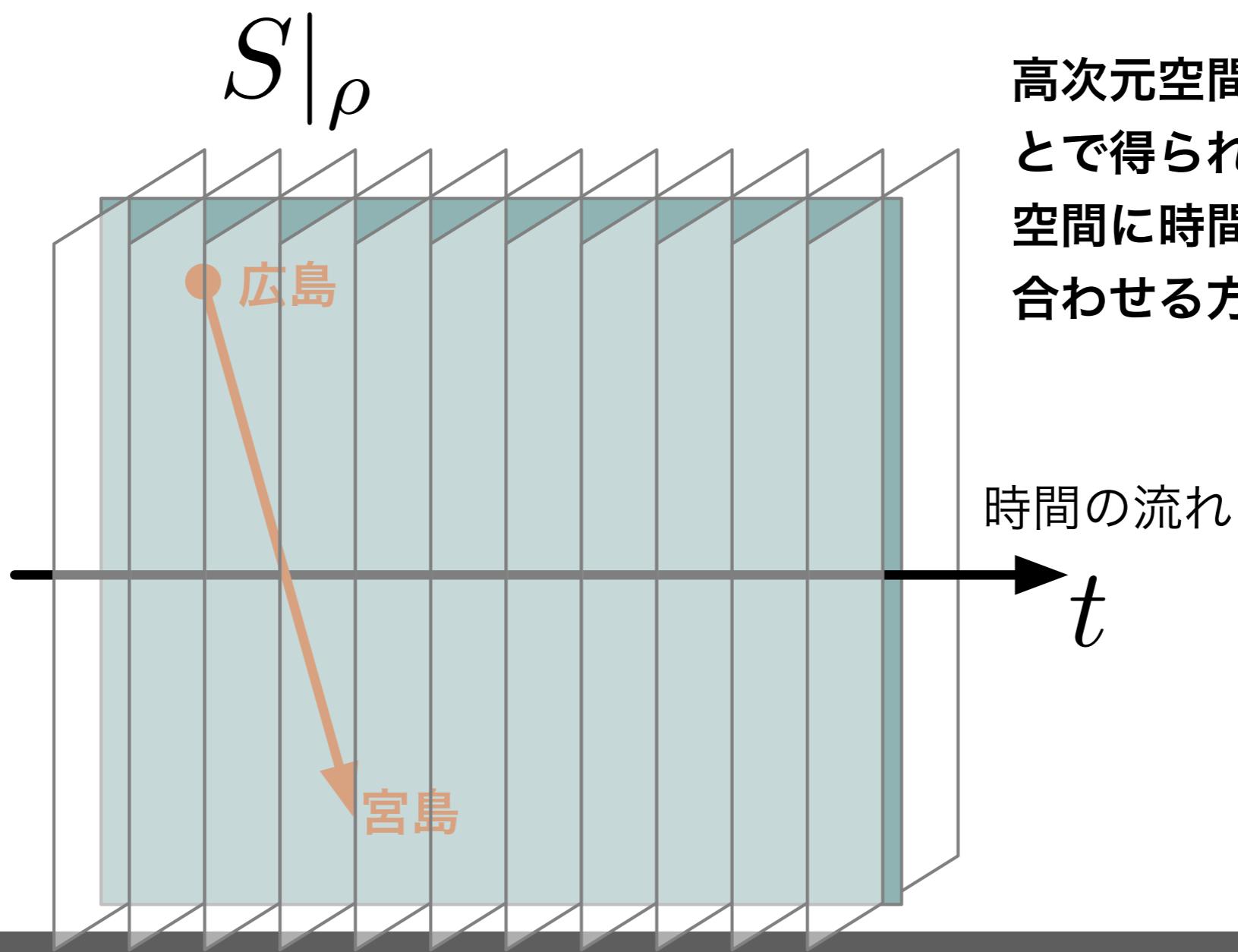


- ネットワーク**構造の変化**を可視化。
- この研究の本当の貢献は、**構造変化のパターン**を可視化した点（後述）

# 空間の切断と集約

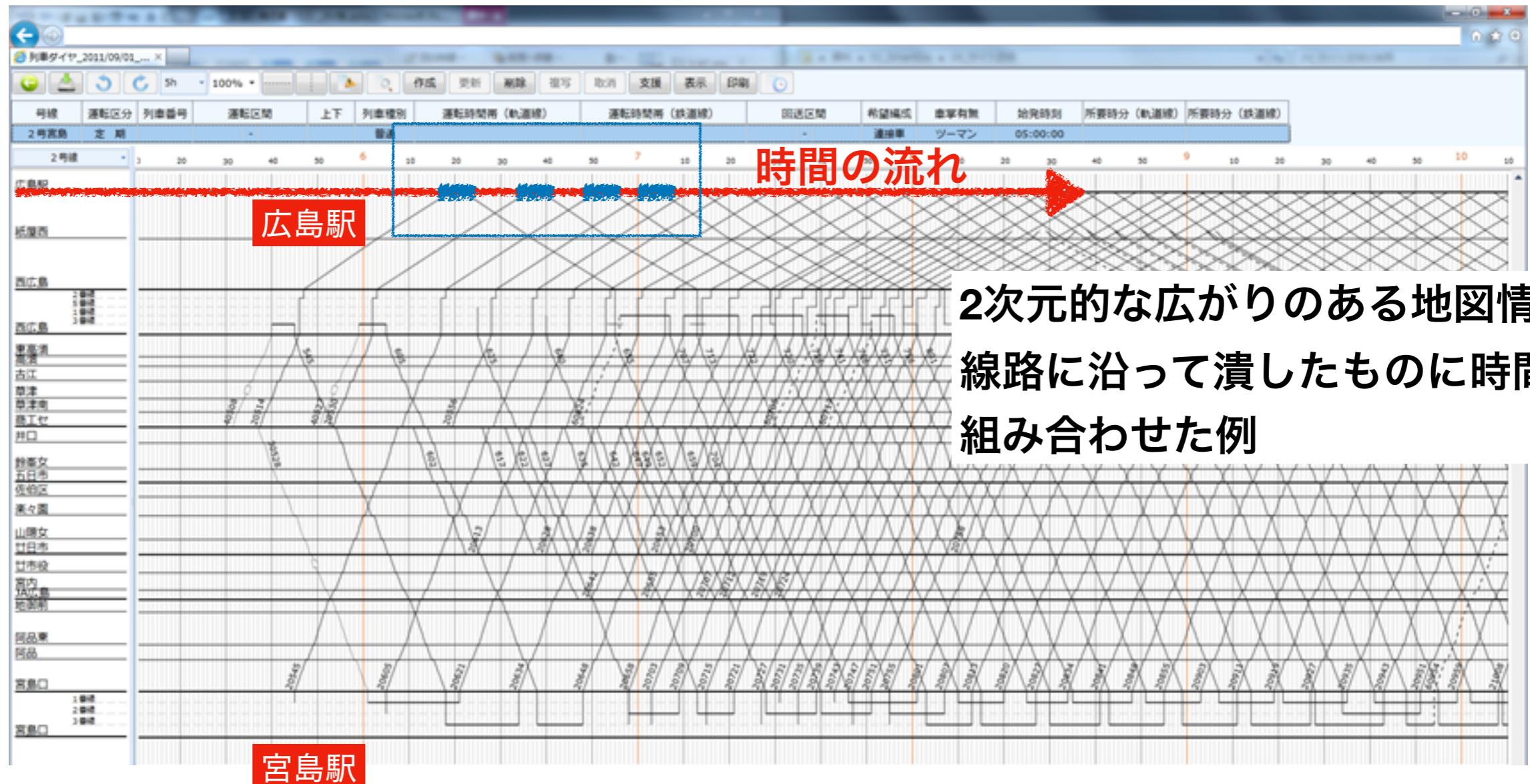
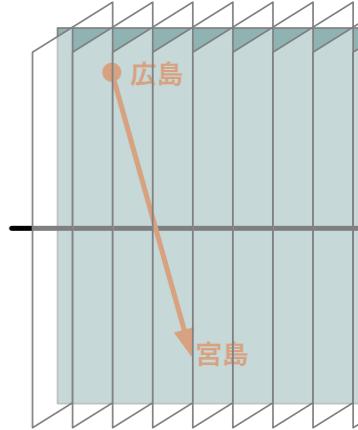
# 空間についての切斷

地点  $p = \rho$  における観測

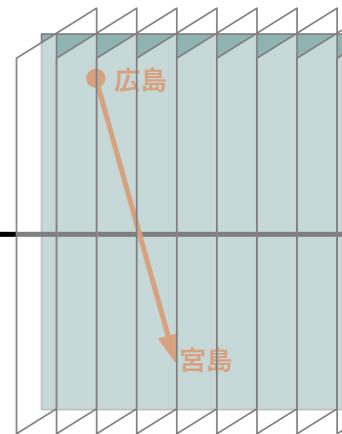


高次元空間を潰すことで得られた低次元空間に時間軸を組み合わせる方式

# 列車ダイヤ

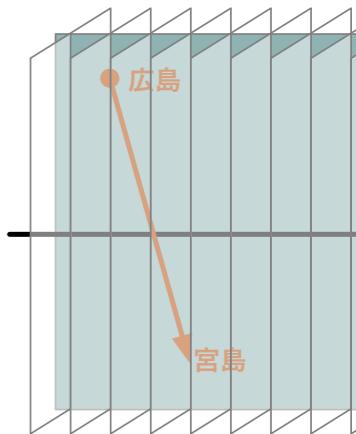


# クイズ – これはなに？



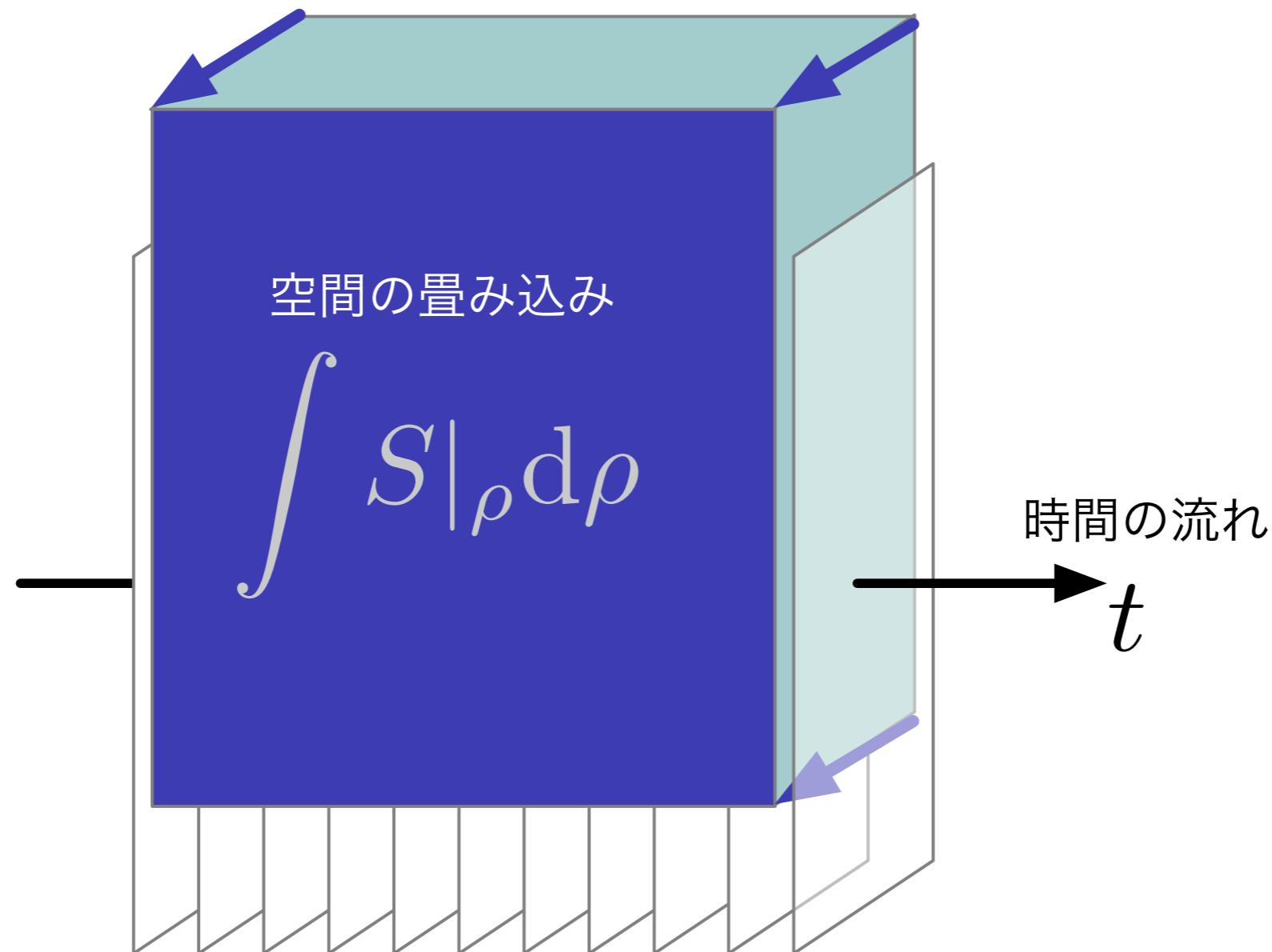
- クイズ – 右の画像（上下2つ）はなんでしょう？

# クイズ – これはなに？

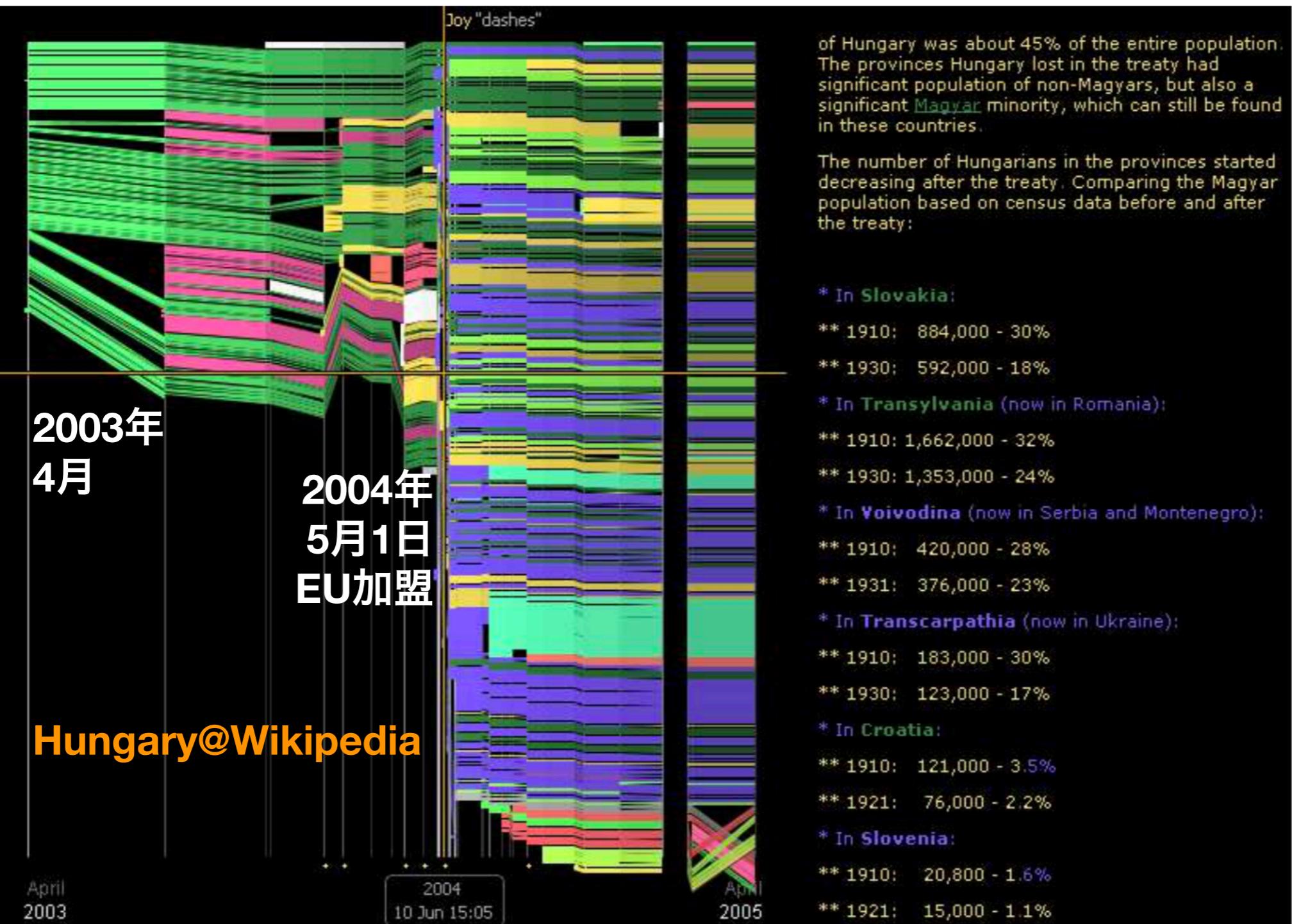
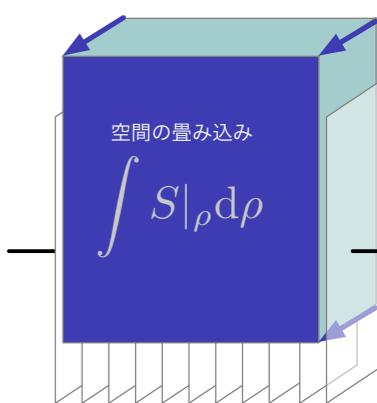


- クイズ – 右の画像（上下2つ）はなんでしょう？
- スリットスキャン

# 空間の畳み込み

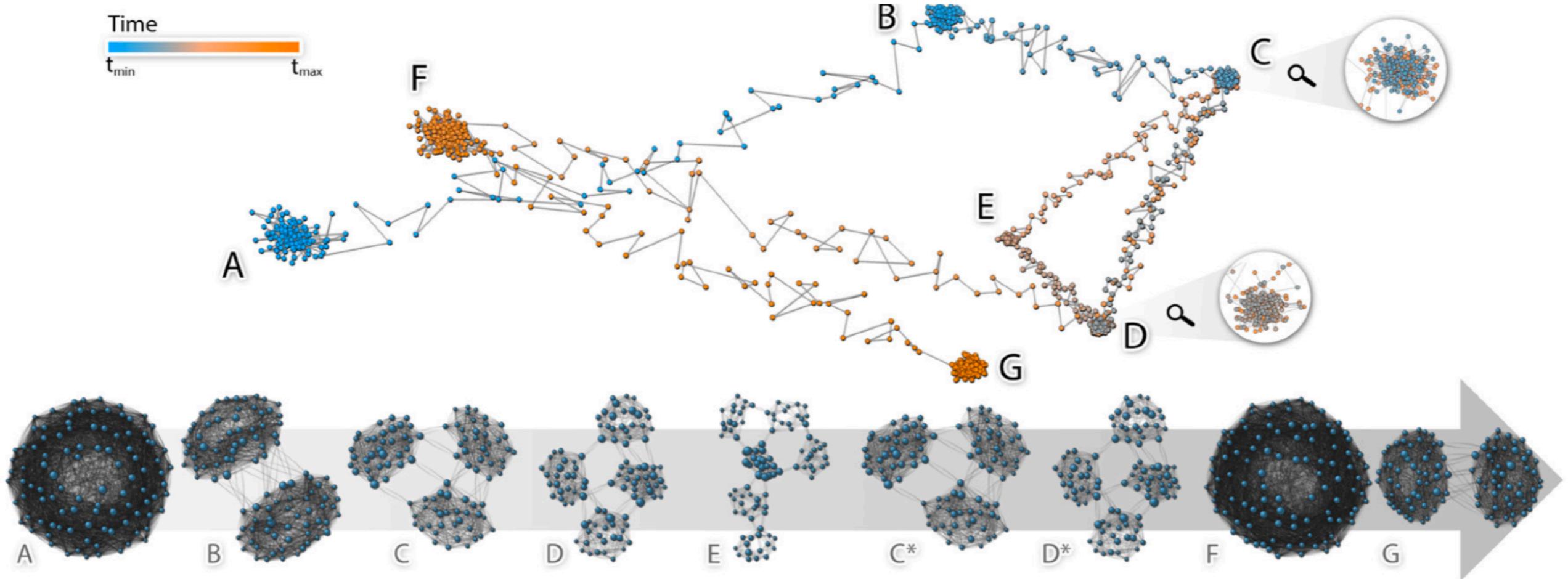
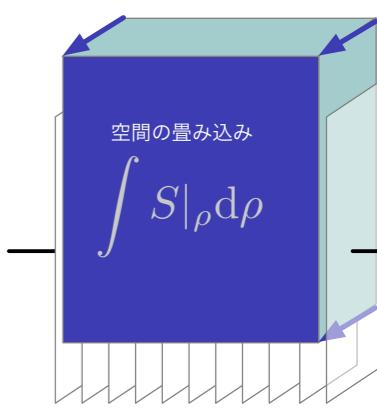


# 空間方向の集約



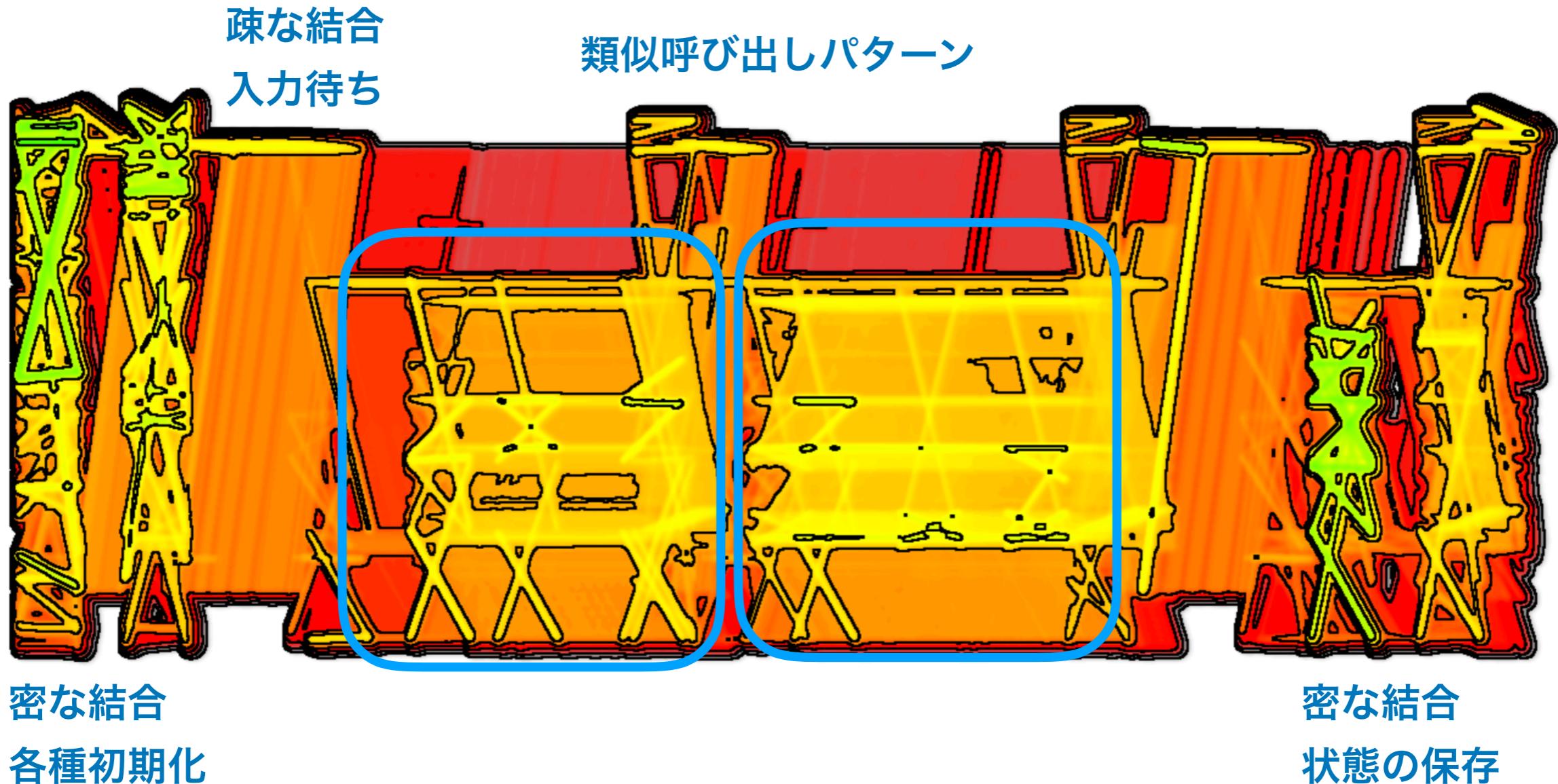
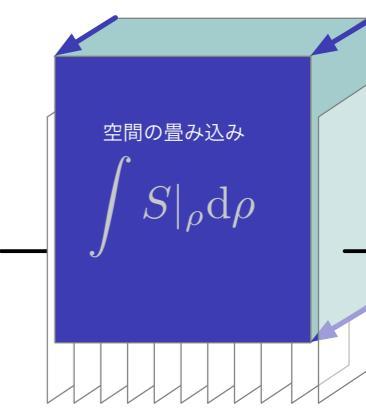
2次元にレイアウトされる Wikipedia の記事を先頭からの文字数を用いた1次元に集約したものと組み合わせていて。

# 空間の畳み込み



下に表示された各ネットワークの画像を畳み込んで点で代表させることで、トポロジーの遷移のトポロジーの表示を可能にしている。

# 空間の畳み込み



某ソフトウェア実行時の関数呼び出しの様子。色は呼び出し関係の密度分布。

#関数定義 = 982個、#関数呼び出し = 32,259個。

# 変化するデータを操作する インタラクション手法

# 操作の体系

- 時空間からの
  - 情報の抽出
  - 情報の縮約
  - 情報の補間
  - 幾何変換
  - 内容の変換

# 情報の抽出

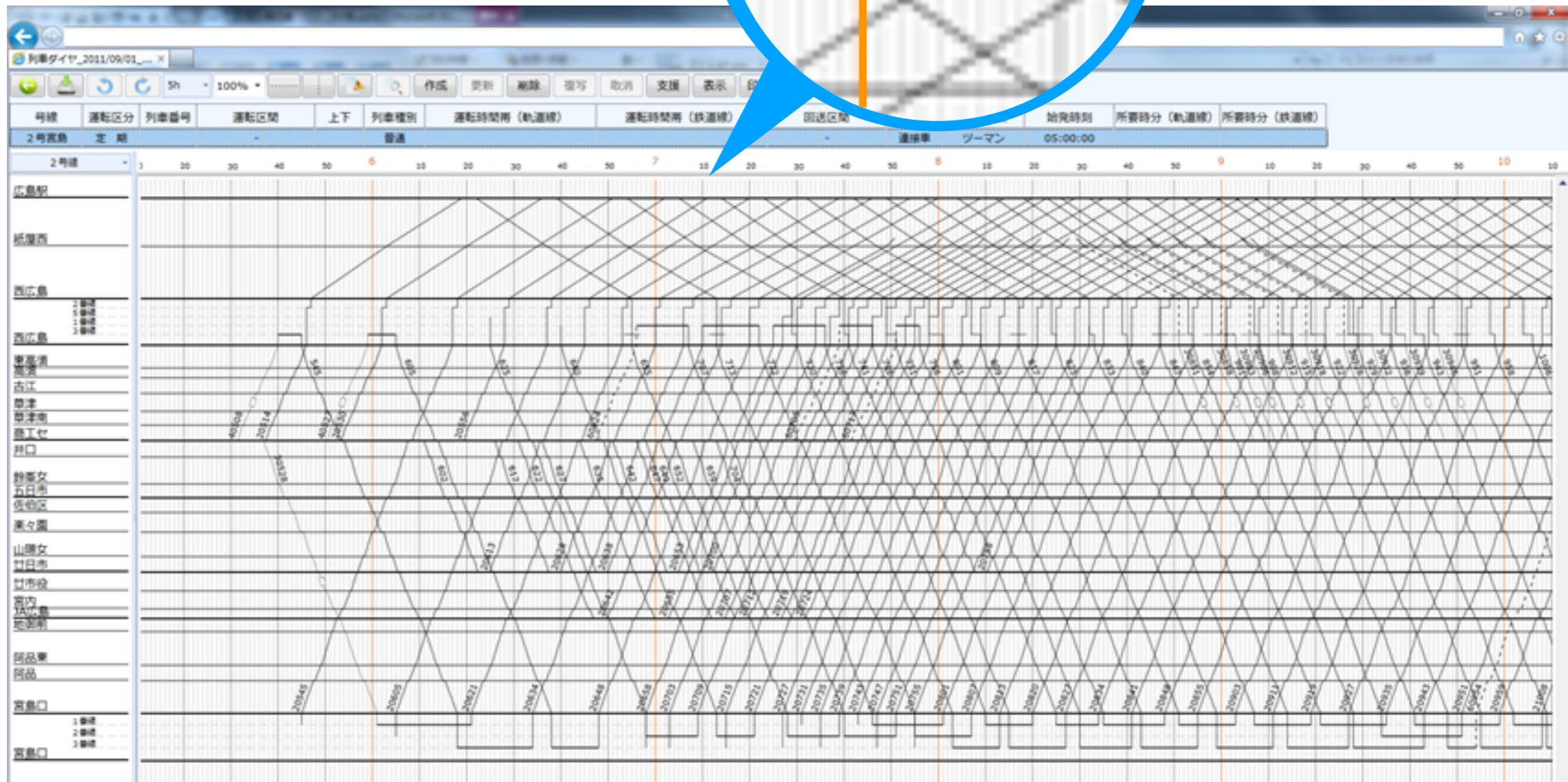
操作			時間	空間
頂点	狙った点を			
線分	まっすぐに ストロー	Time Drilling	Space Drilling	
空間 領域	まっすぐに チーズ	Time Cutting	Linear Space Cutting	
	まっすぐに 羊羹	Time Chopping	Linear Space Chopping	

The diagram illustrates four types of operations on a 3D cube, separated by a vertical dashed line into Time (left) and Space (right) dimensions.

- 頂点 (Vertex):** Drills a hole at a specific vertex. The Time dimension shows a point being drilled, and the Space dimension shows the resulting hole.
- 線分 (Line):** Drills a hole along a straight line (straw). The Time dimension shows a line being drilled, and the Space dimension shows the resulting slot.
- 空間領域 (Space Domain):** Cuts a rectangular block from a straight line (cheese). The Time dimension shows a line being cut, and the Space dimension shows the resulting rectangular block removed.
- 領域 (Area):** Cuts a rectangular block from a plane (jam). The Time dimension shows a plane being cut, and the Space dimension shows the resulting rectangular block removed.

# 33 ダイヤを例に

## 点の抽出

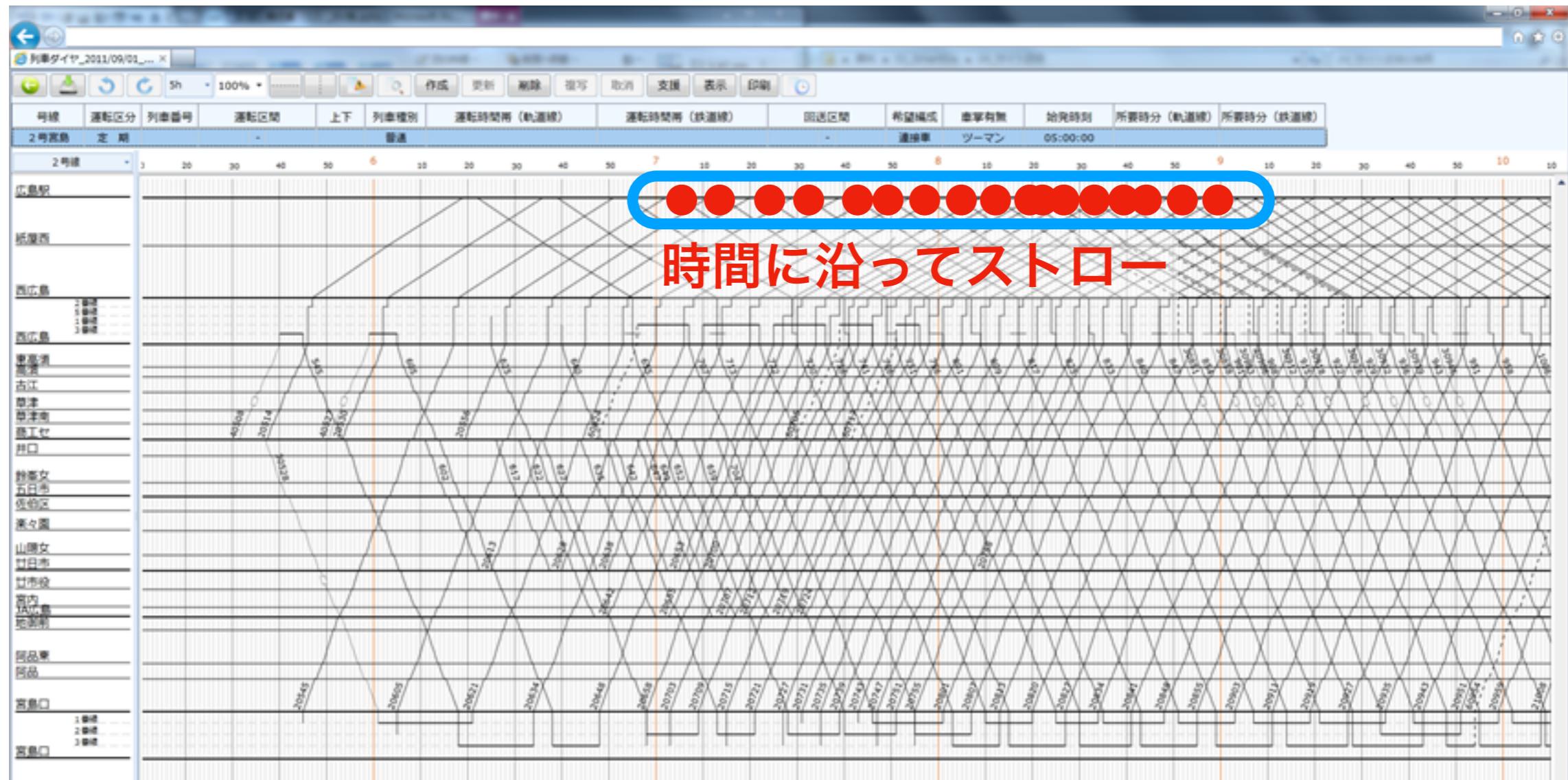


# 34 ダイヤを例に

7:00-9:00 の広島駅  
の便数は？

## 点の抽出

D[広島, 0700:0900]

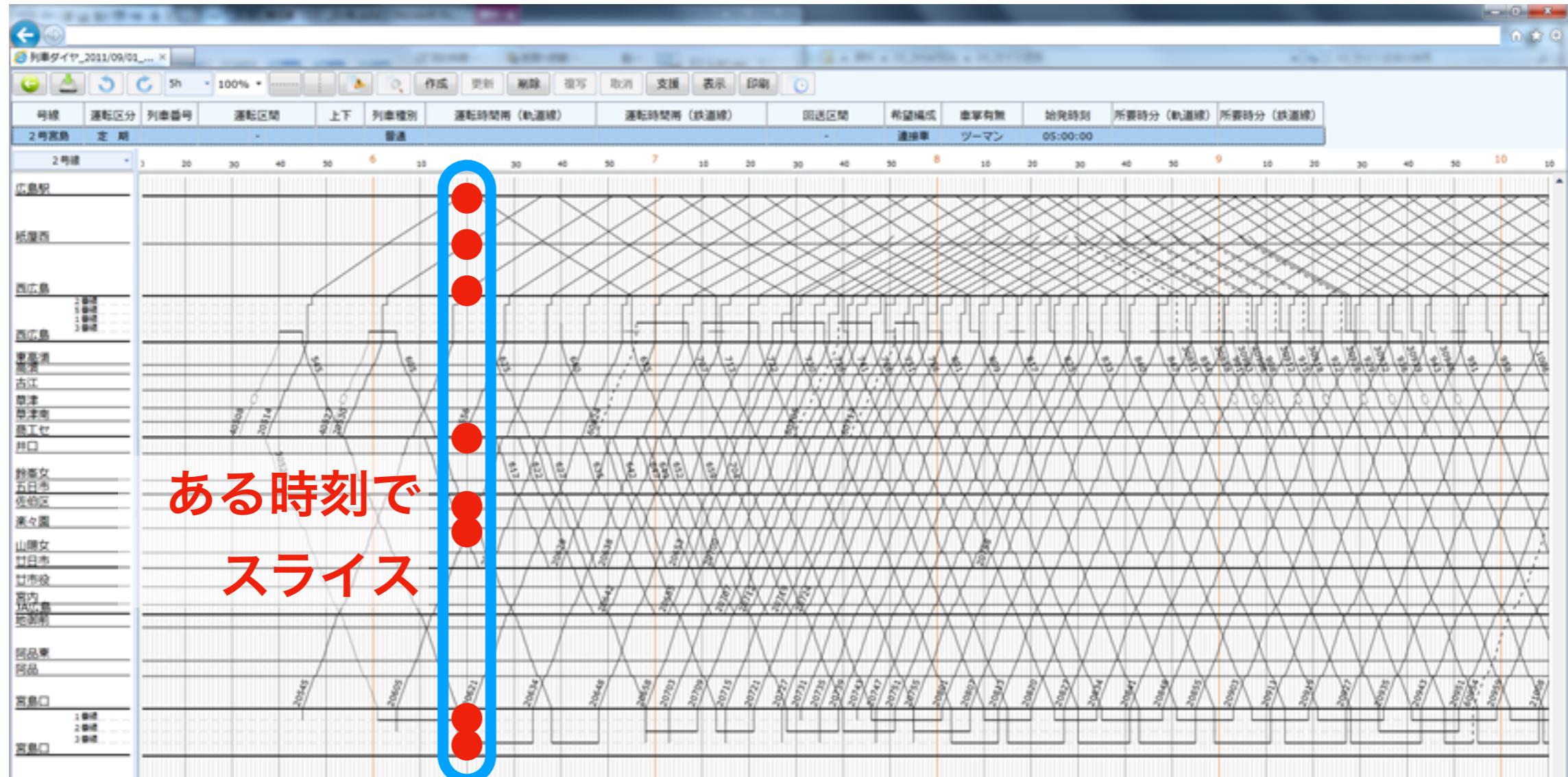


# 35 ダイヤを例に

6:20に運行している電車は？

D[:, 0620]

## 点の抽出

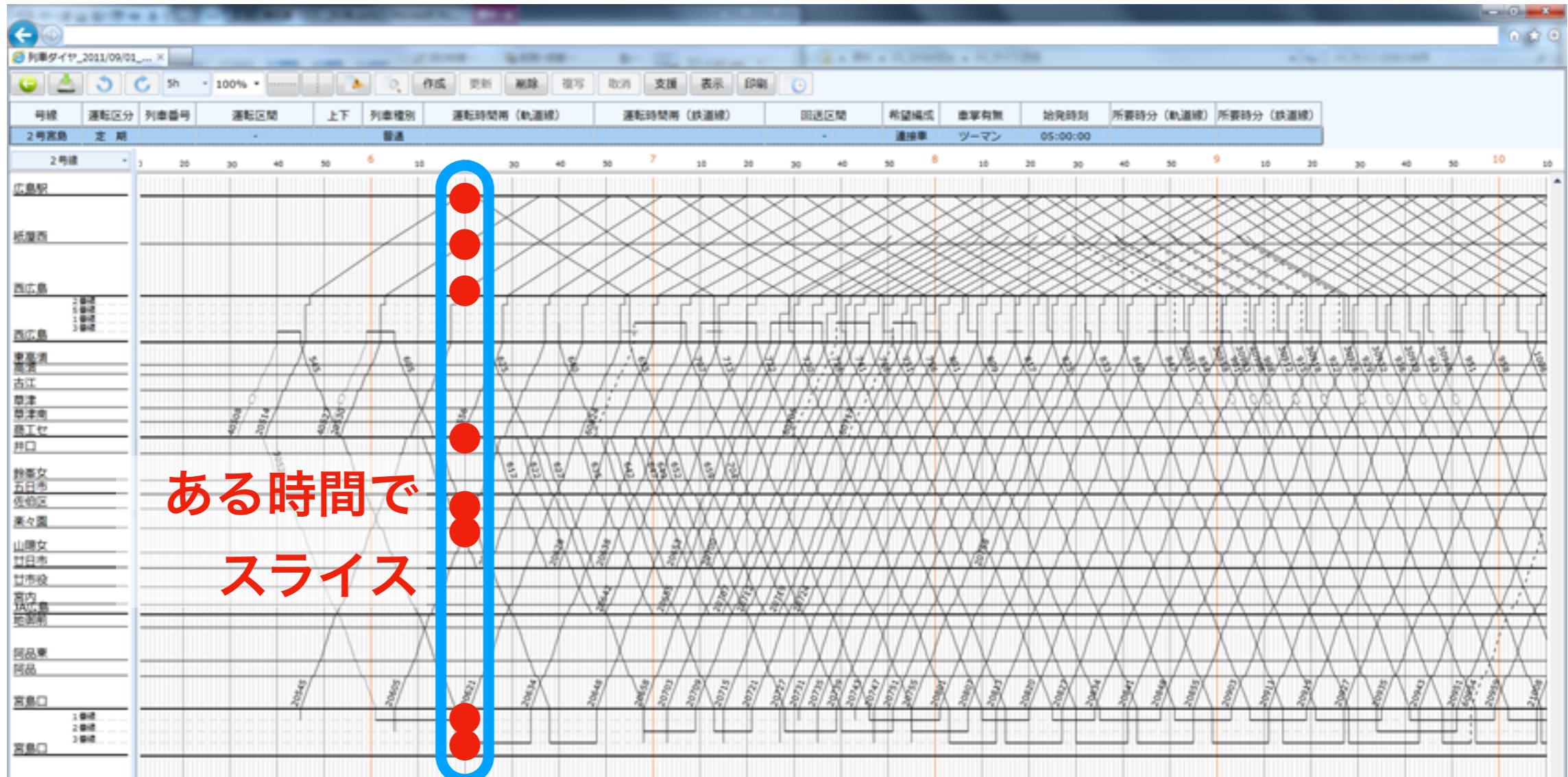


# 36 ダイヤを例に

6:20に運行している電車は？

D[:, 0620]

## 点の抽出

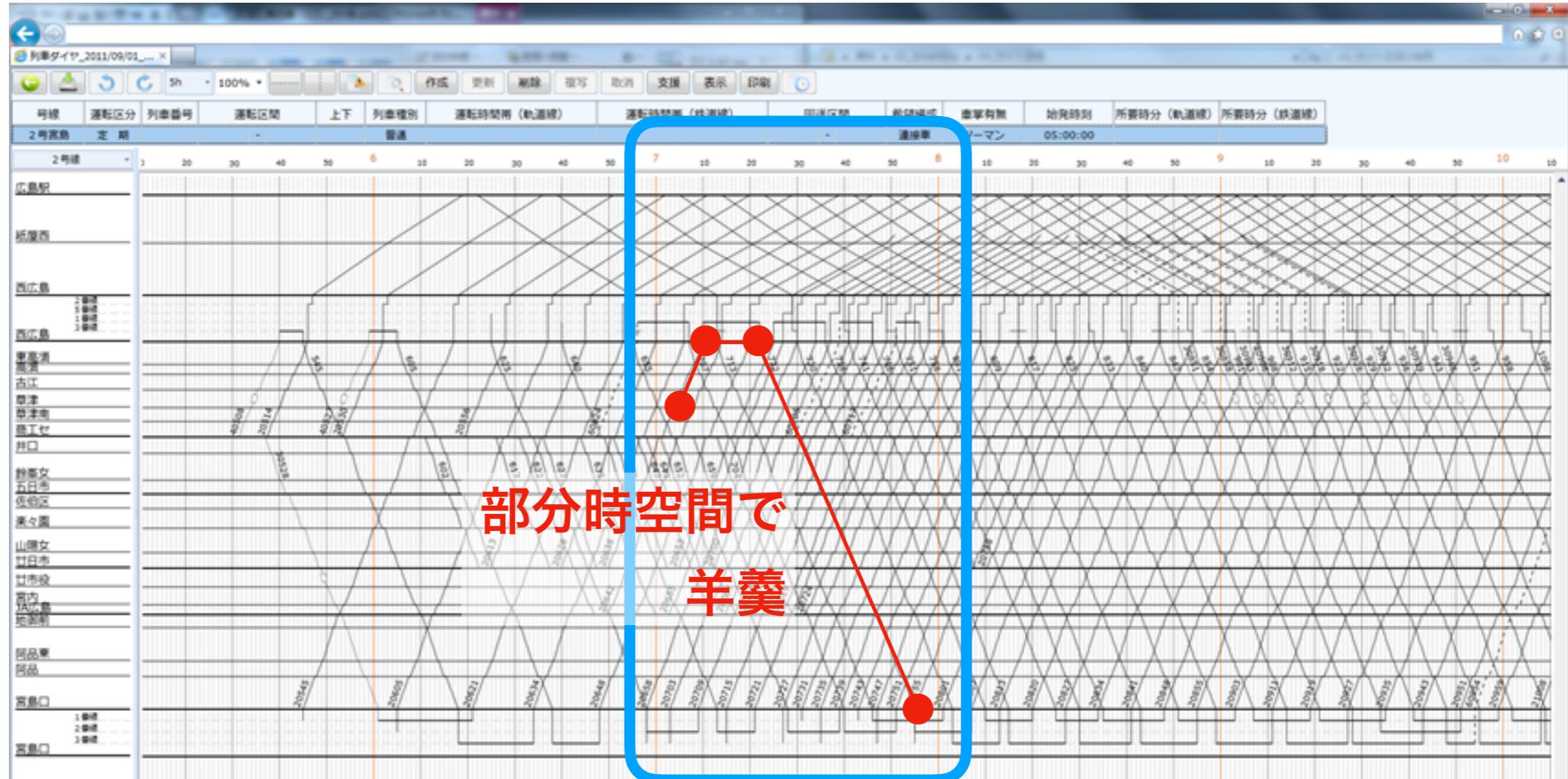


# 37 ダイヤを例に

## 点の抽出

友人は7時すぎに草津から逆向きの電車に乗り、折り返して8時前に宮島口に着いたという

D[広島:宮島, 0700:0800]

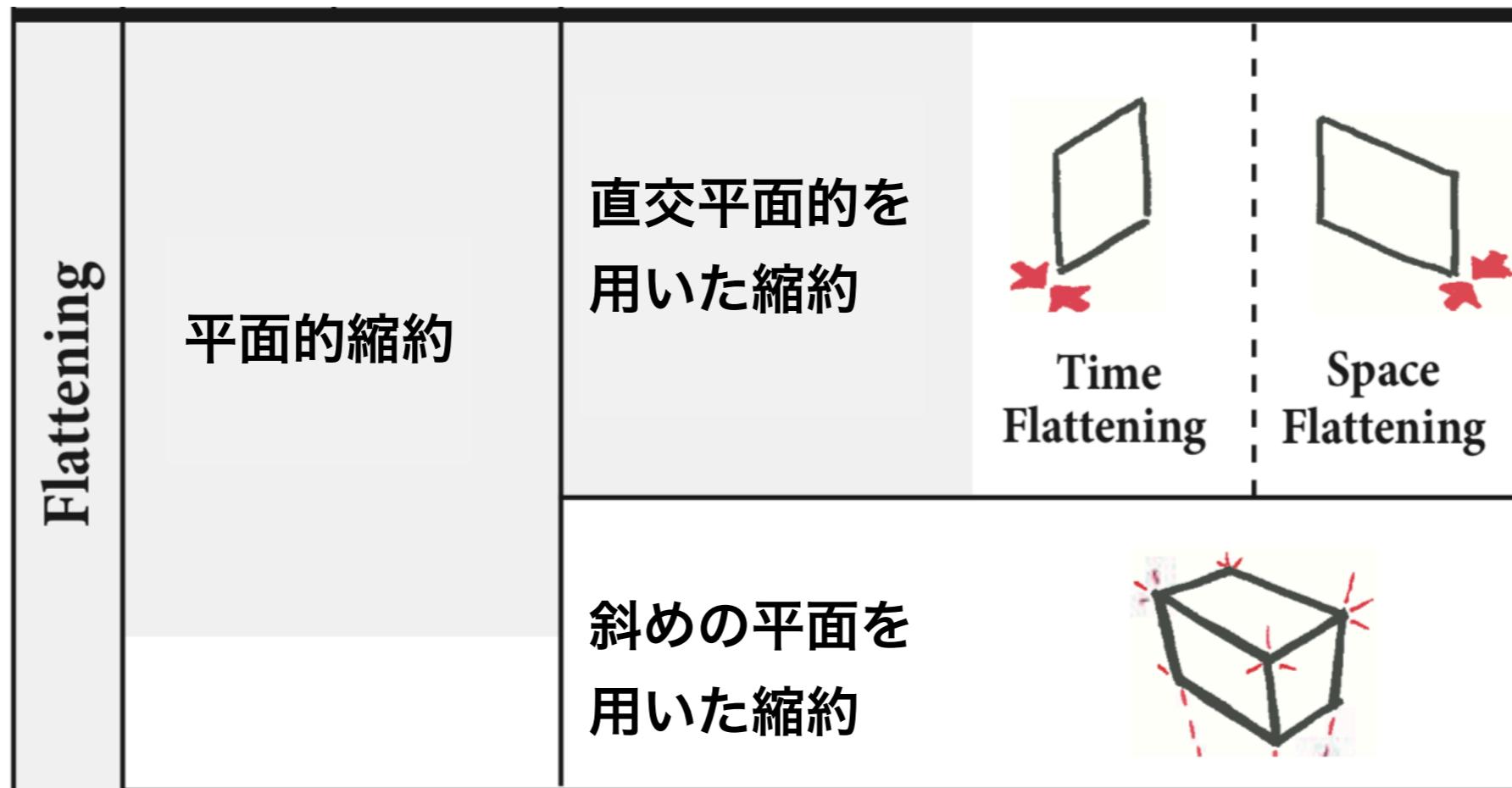


# データ項目の抽出

Operations		Time	Space
Extraction	Point	Point Extraction	
	Curve	Orthogonal Drilling	
		Time Drilling	
		Space Drilling	
		Oblique Drilling	
		Planar Curvilinear Drilling	
		Non-Planar Drilling	
	Surface	Orthogonal Cutting	
		Time Cutting	
		Linear Space Cutting	
	Non-Planar Cutting	Oblique Cutting	
		Curvilinear Space Cutting	

Volume	Planar Chopping	Orthogonal Chopping	
		Time Chopping	
Non-Planar Chopping		Linear Space Chopping	
		Oblique Chopping	
Other		Curvilinear Space Chopping	
		Other	

# 情報の集約



# 時刻についての集約

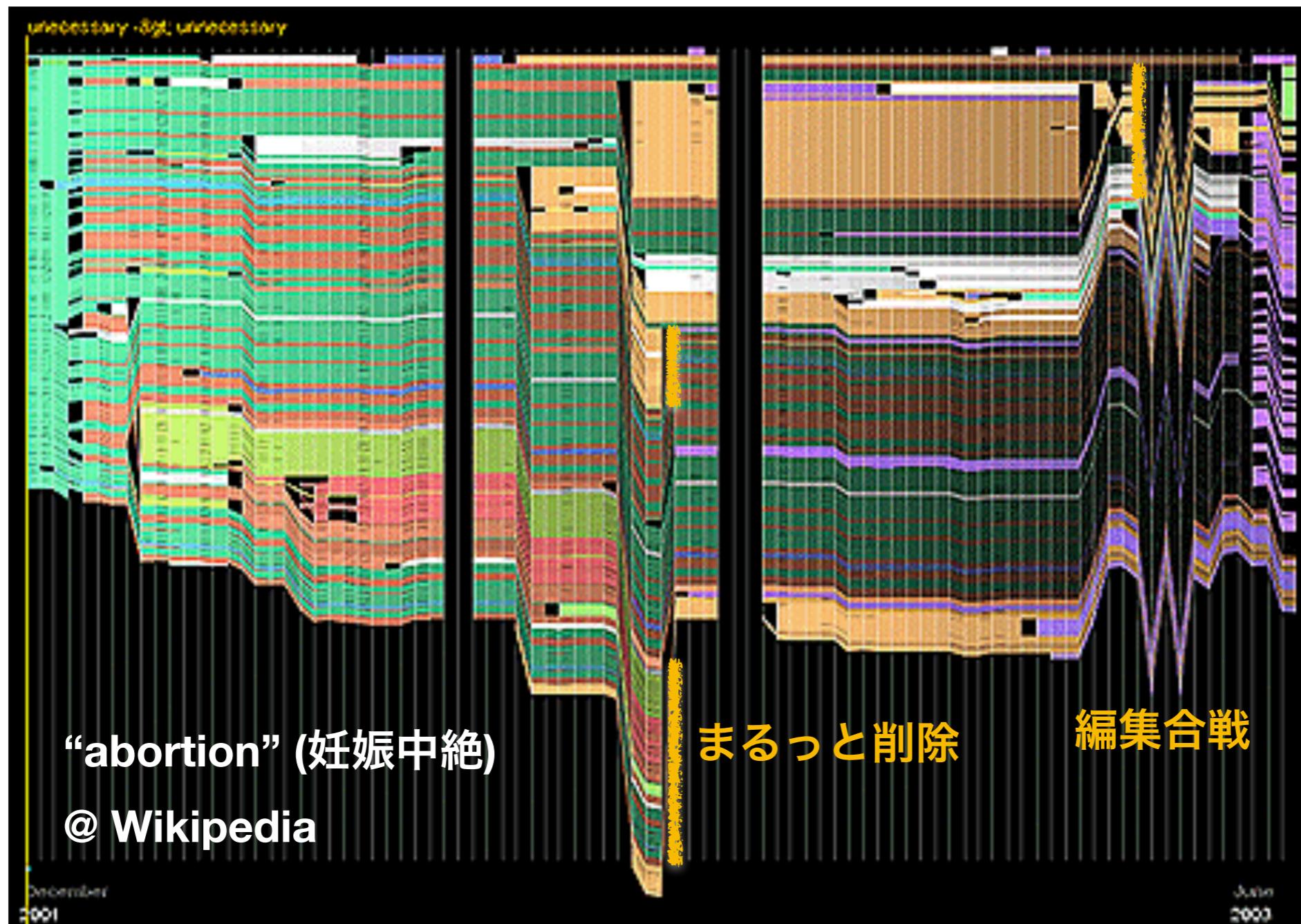
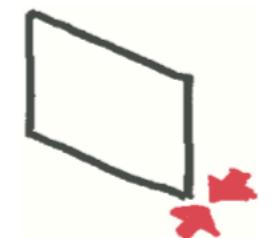


ぼくのご先祖さまはモスクワ遠征に参加して、奇跡的に生還したんだよ。行軍中の手紙を取ってあるんだ。

D[兵士=“ぼくのご先祖さま”, \*]

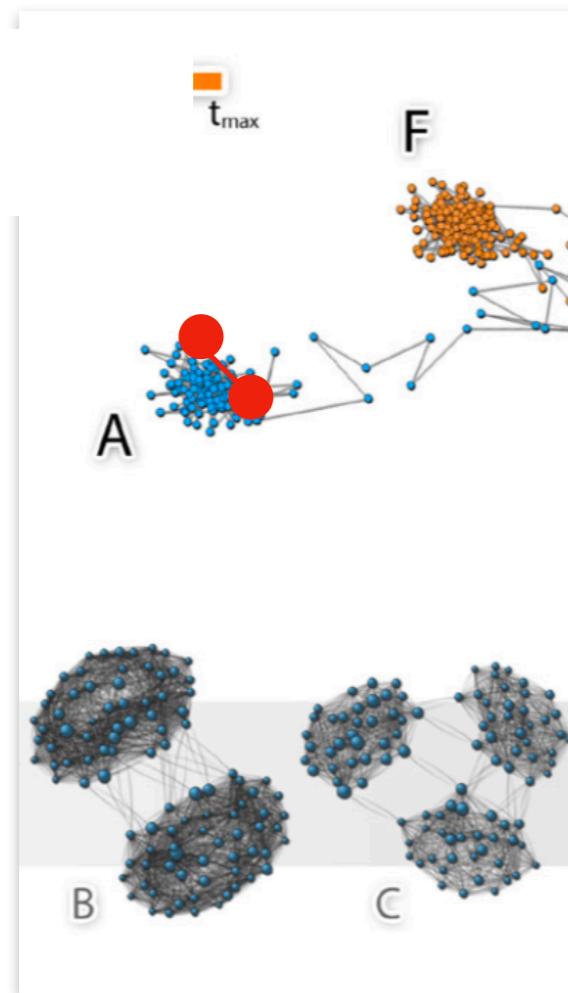


# 空間についての集約



# 情報の補間

離散値を線で繋いだり、  
集合を平均的な値で代表したり、  
結線を平滑化したする

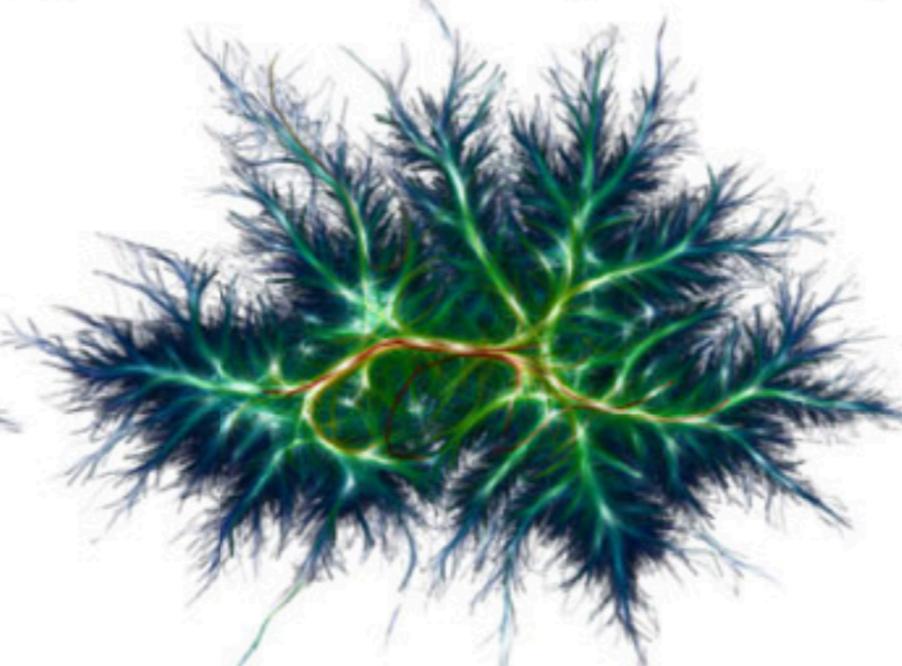
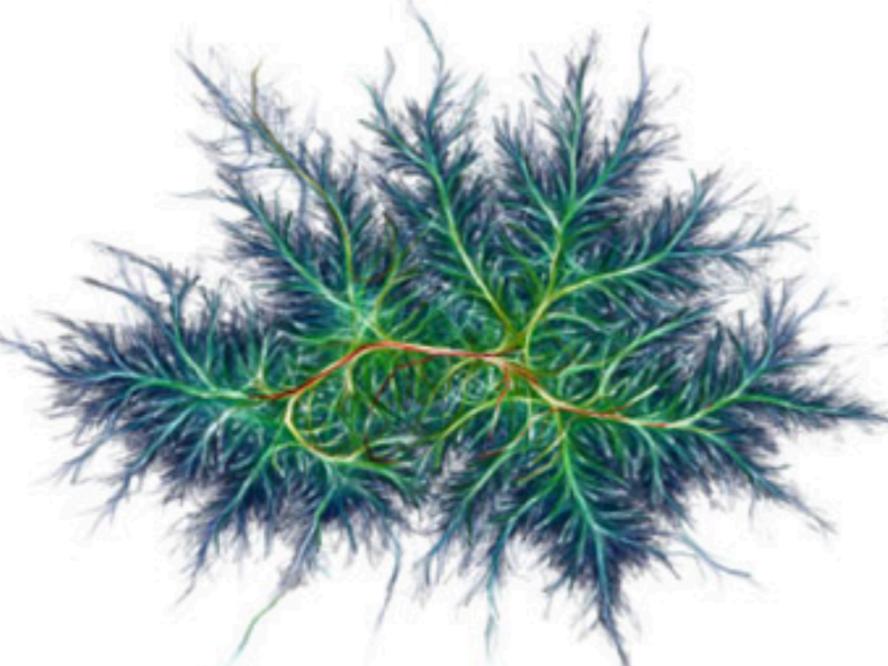
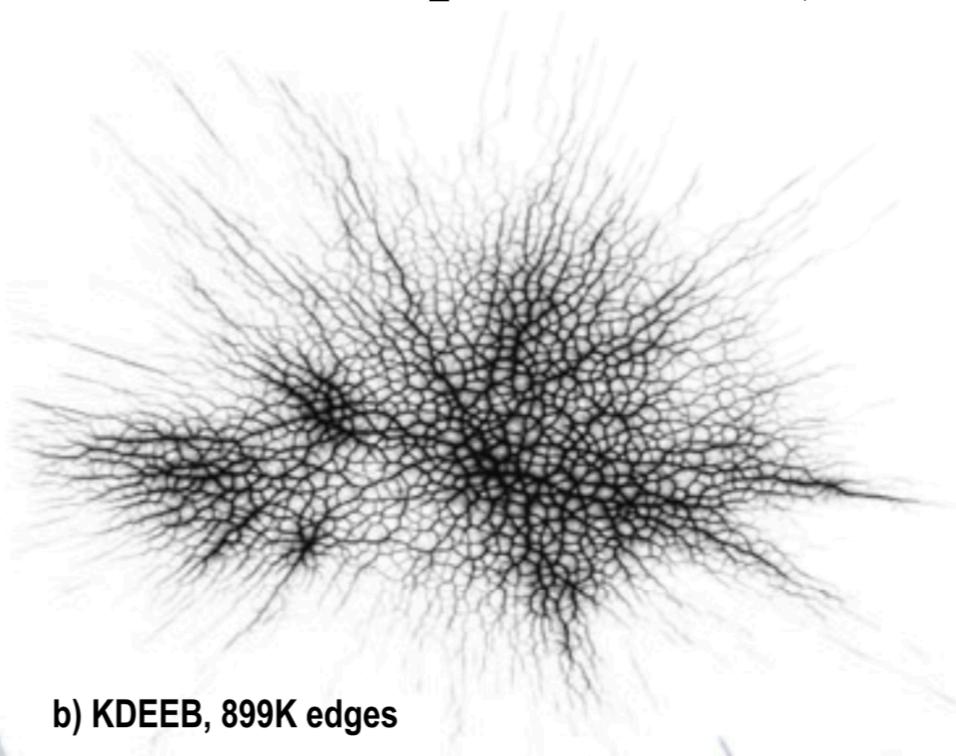
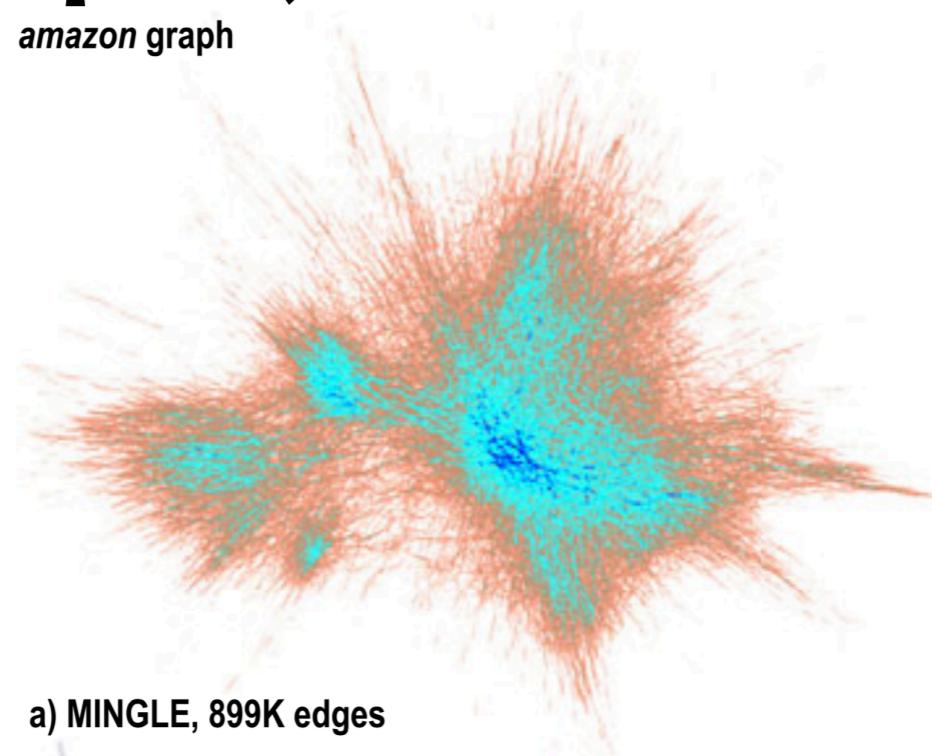


# 内容の変換

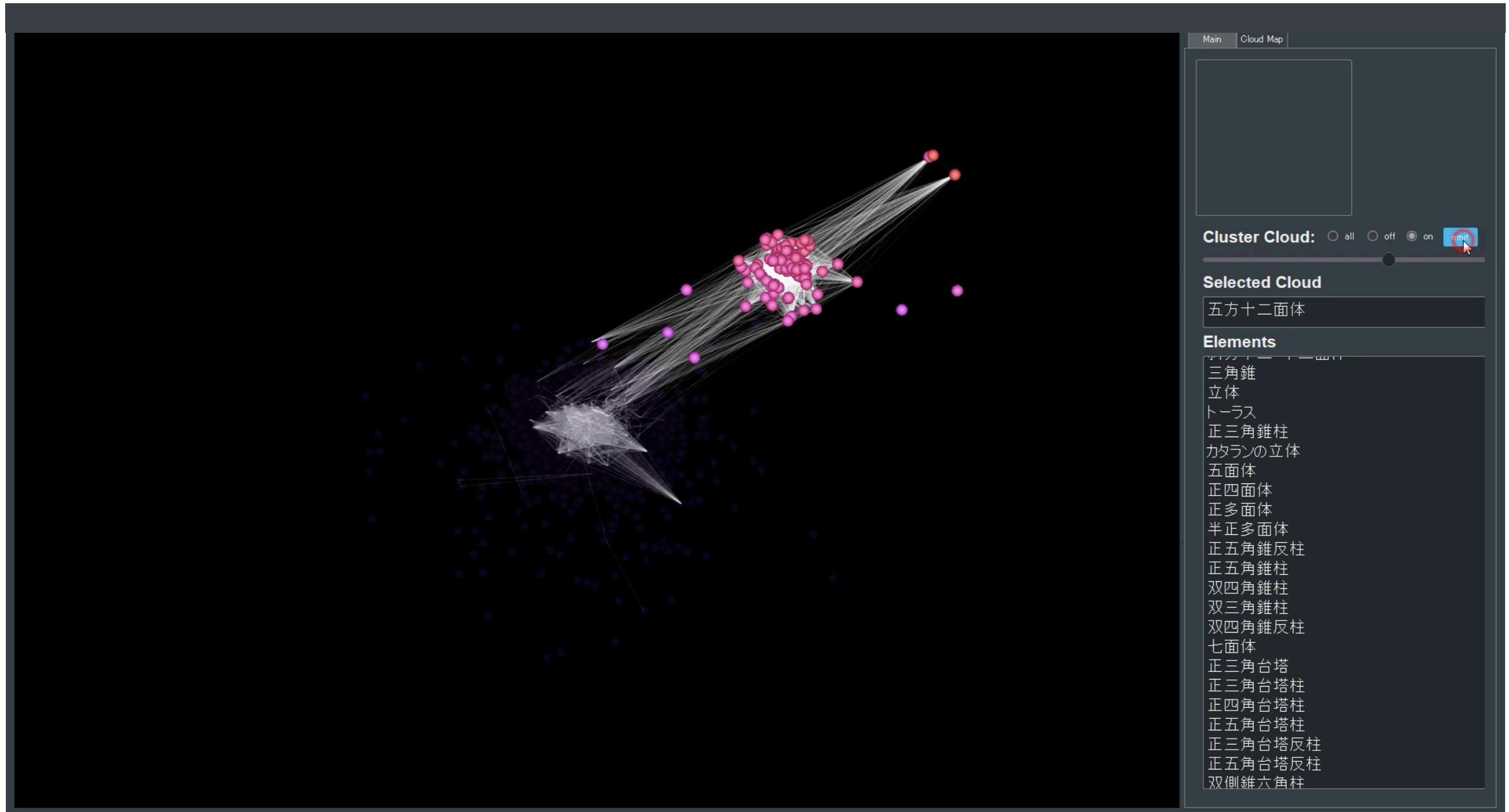
- 時間に沿った彩色
- 空間に沿った彩色
- 時刻の注釈
- 安定化（外的要因の相殺）
- 束化（bundling）
- シェーディング
- フィルター
- 凝集表現（クラスタ）

# 束化、シェーディング

amazon graph



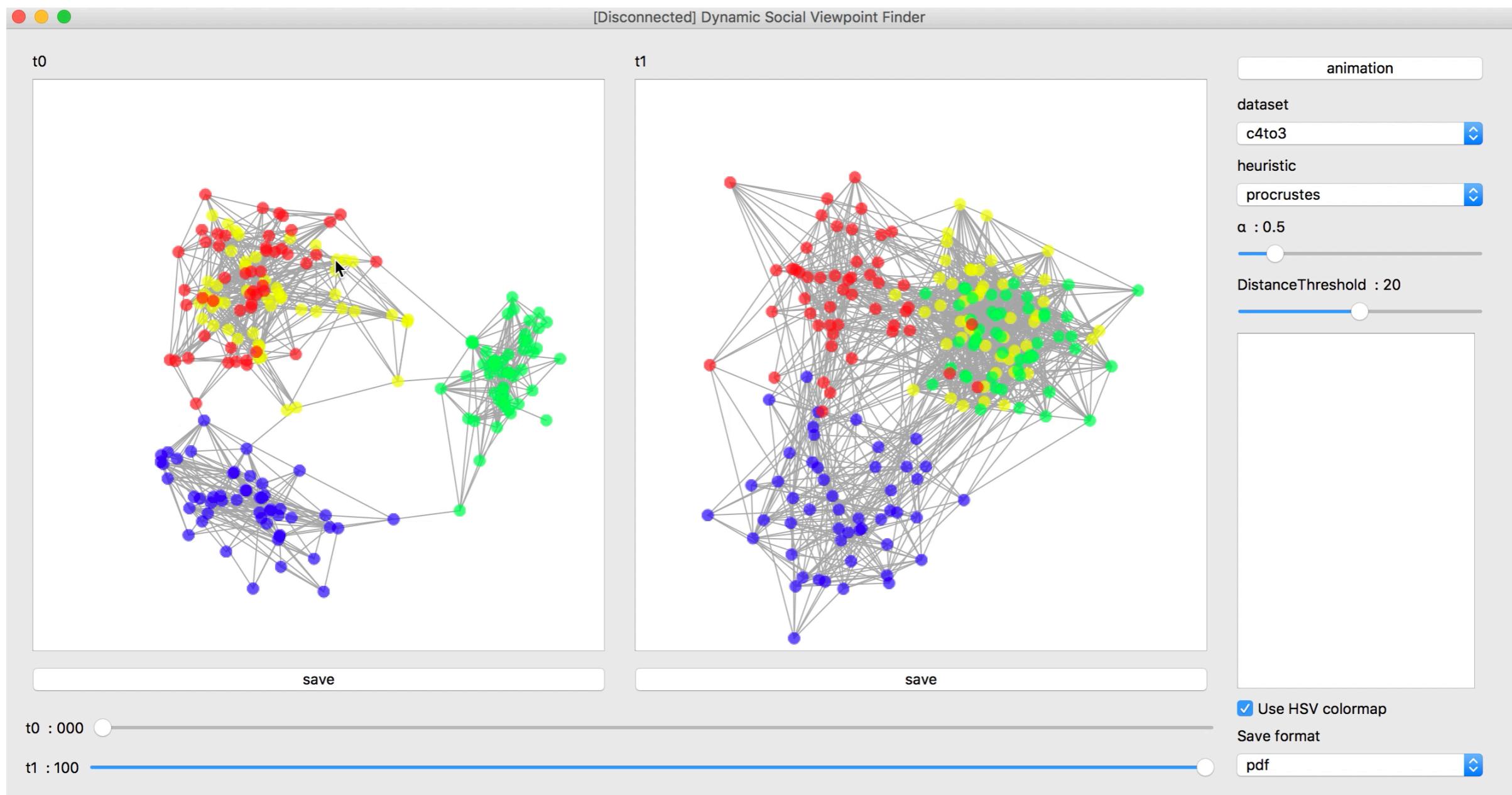
## クラスタへのインタラクション



# 注釈彩色、ビュー間の連携、

いくつか研究室の活  
動の例より

# インタラクション、幾何変形、アニメーション



# まとめ

wakita@is.titech.ac.jp

- 時間変化のあるデータの可視化にはとてもリッチな表現方法、インターラクションがあって面白いです。
  - 平面的に時間を埋め込んだ表現
  - アニメーションを用いる表現
  - データ空間を集約する表現
  - インタラクション（抽出、集約、補間、変換）