Events-Vis: Integrando Espaço e Tempo para a Visualização de Eventos em um gráfico 2D

Giovani de Almeida Valdrighi

Jorge Poco

Nivan Ferreira

Introdução

1. Dados espaço-temporais

Dados espaço-temporais

- Dados com informação espacial (2D) e temporal.
- Com o **avanço tecnológico**, grande quantidade de dados vem sendo coletada:
  - dados de GPS de celulares;
  - sensores remotos para medições meteorológicas;
  - sensores para medições do trânsito.
- São **aplicados em diversas áreas** como: mobilidade humana, assistência médica, sismologia e meteorologia.

Imagem por Dariusz Sankowski (Pixabay)

Imagem por Sebastiaan Stam (Unsplash)

Classificação de dados espaço-temporais

- Diversas classificações de dados espaço-temporais existem na literatura.
- **Eventos**: **posição espacial fixa** e um **intervalo de tempo de existência**. Exemplo: atividade sísmica.
- **Séries temporais georreferenciadas: posição espacial fixa**, e uma **sequência de instantes de tempo** em que uma **medição** ocorre. Exemplo: medição de umidade a cada 4 horas realizada por um sensor fixo.
- **Trajetórias**: **posição espacial que se altera** a cada instante de tempo, e um identificador para reconhecer o percurso de um mesmo objeto. Exemplo: registro do percurso de uma corrida de táxi.

Por que “eventos”?

Eventos

- A visualização proposta irá focar na visualização de eventos.
- Apresentam dinamismo espacial e temporal e são comumente obtidos em dados urbanos.
- Eventos podem ser gerados a partir de **agrupamentos** em dados pontuais espaço-temporais, obtendo uma **região de ocorrência** e um **intervalo de tempo de existência**.
- Em particular, iremos focar **em eventos que ocorrem em uma região no espaço**, apresentam **área**, e a técnica poderá ser generalizada para dados pontuais considerando um valor constante para a área.

- Slide explicando agrupamento e envoltório convexo

Quais técnicas para visualização espaço-temporal já existem?

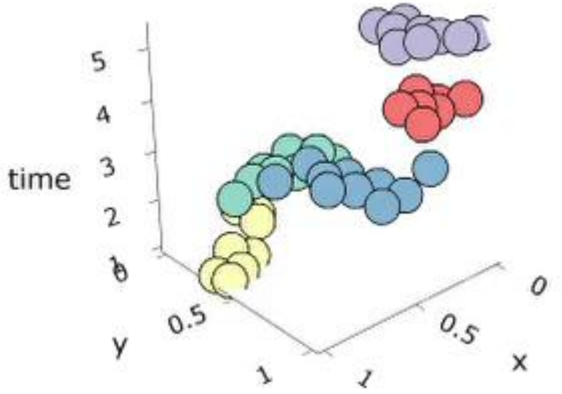
2. Visualizações de dados espaço-temporais

2. Visualização de dados espaço-temporais

- Muitas técnicas para a visualização de dados espaciais foram herdadas da **cartografia**.
- Para lidar com a dimensão adicional (tempo), técnicas conectaram os métodos cartográficos com a dinamicidade permitida ao se utilizar computadores, utilizando de **interatividade e animação**.
- Técnicas mais comuns:
  - cubo de espaço-tempo;
  - animações;
  - “small multiples”.

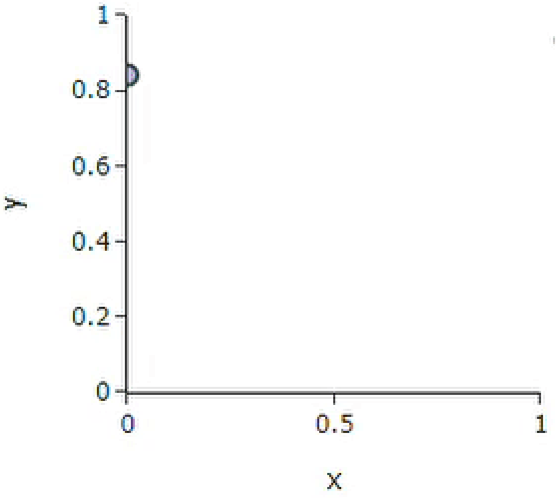
2. Visualização de dados espaço-temporais

Cubo de espaço-tempo



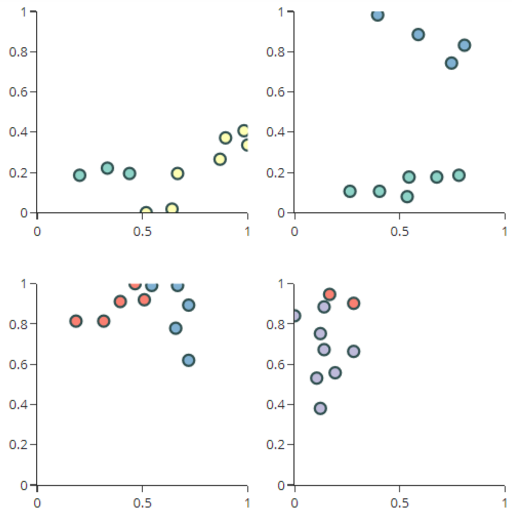
Projeções realizadas para exibir um **objeto 3D** em uma **tela 2D** pode gerar **interpretações equivocadas**.

Animações



Necessidade de **memorizar** os dados de diferentes momentos para **realizar comparações.**

"Small multiples"



Dificuldade em selecionar os intervalos temporais e **número limitados de sub-gráficos**.

Quais os objetivos de **Events-Vis**?

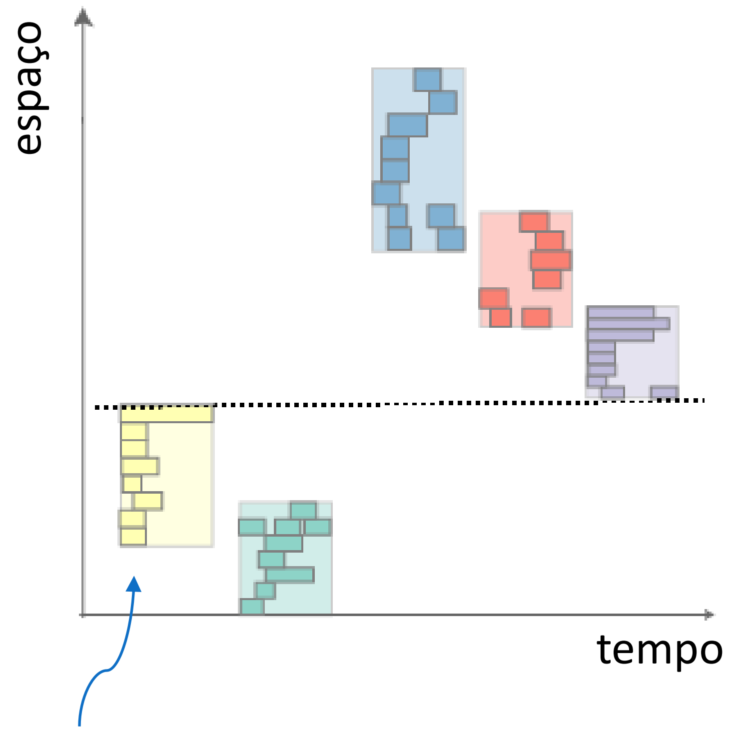
3. Objetivos

3. Objetivos

- Gerar uma visualização de dados espaço-temporais que:
- Seja capaz de gerar uma **visão geral** de todo o conjunto de dados;
- Permita a **interpretabilidade** para identificação de características temporais e espaciais dos dados originais;
- Evite os problemas apresentados pelas técnicas convencionais;

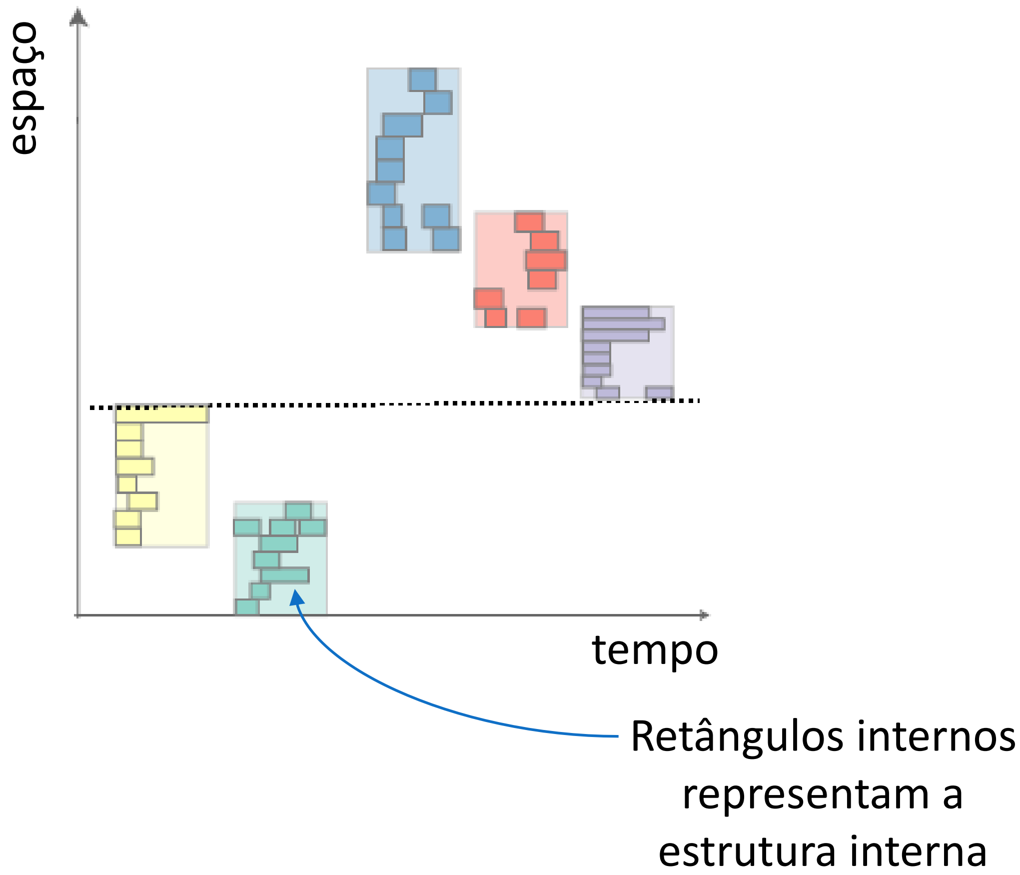
Events-Vis: Método

Events-Vis: Método



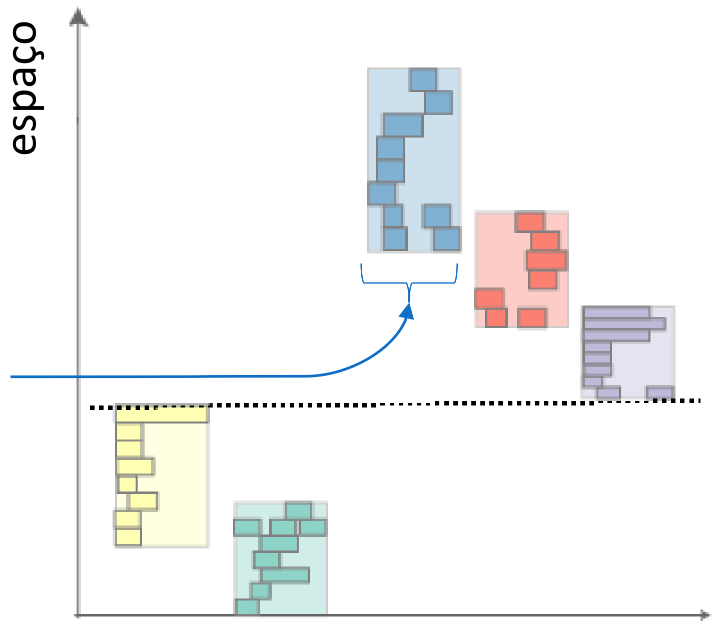
Retângulos externos
representam eventos

Events-Vis: Método

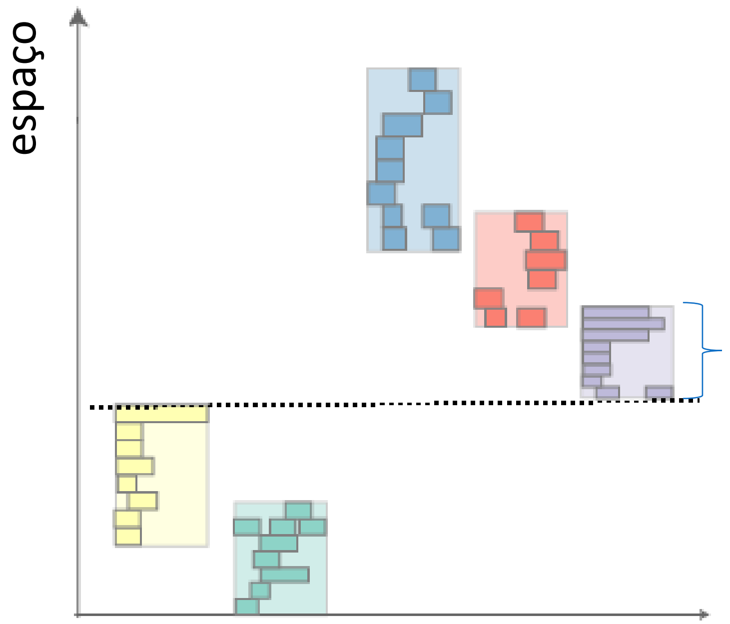


Events-Vis: Método

Largura representa a duração temporal

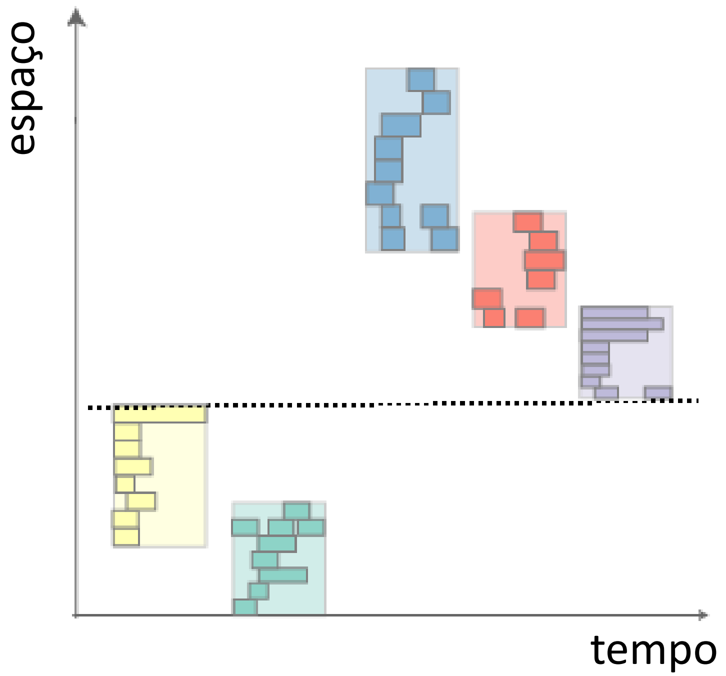


Events-Vis: Método

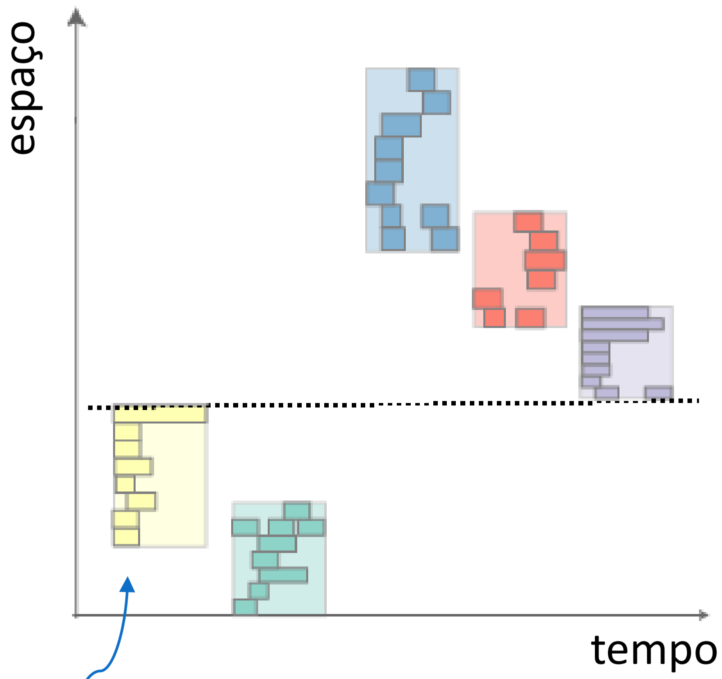


Altura representa
a área

Events-Vis: Método

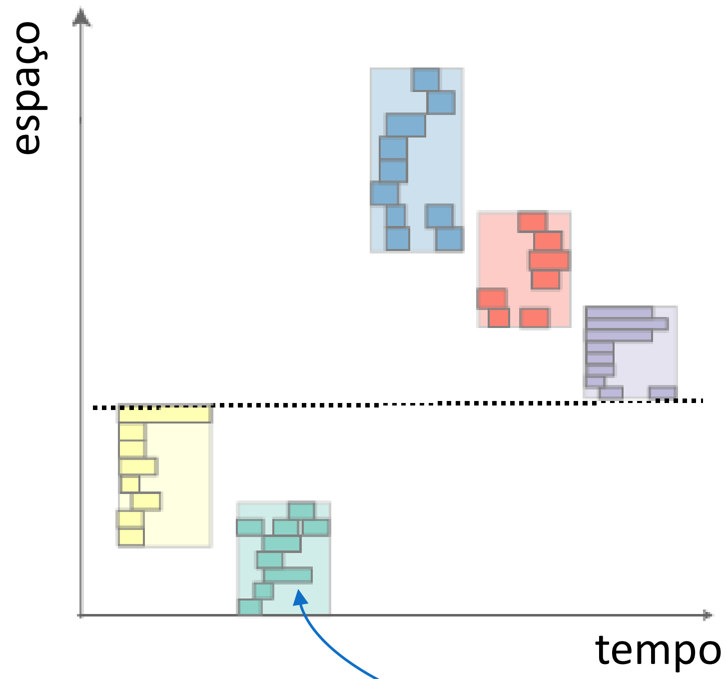


Events-Vis: Método



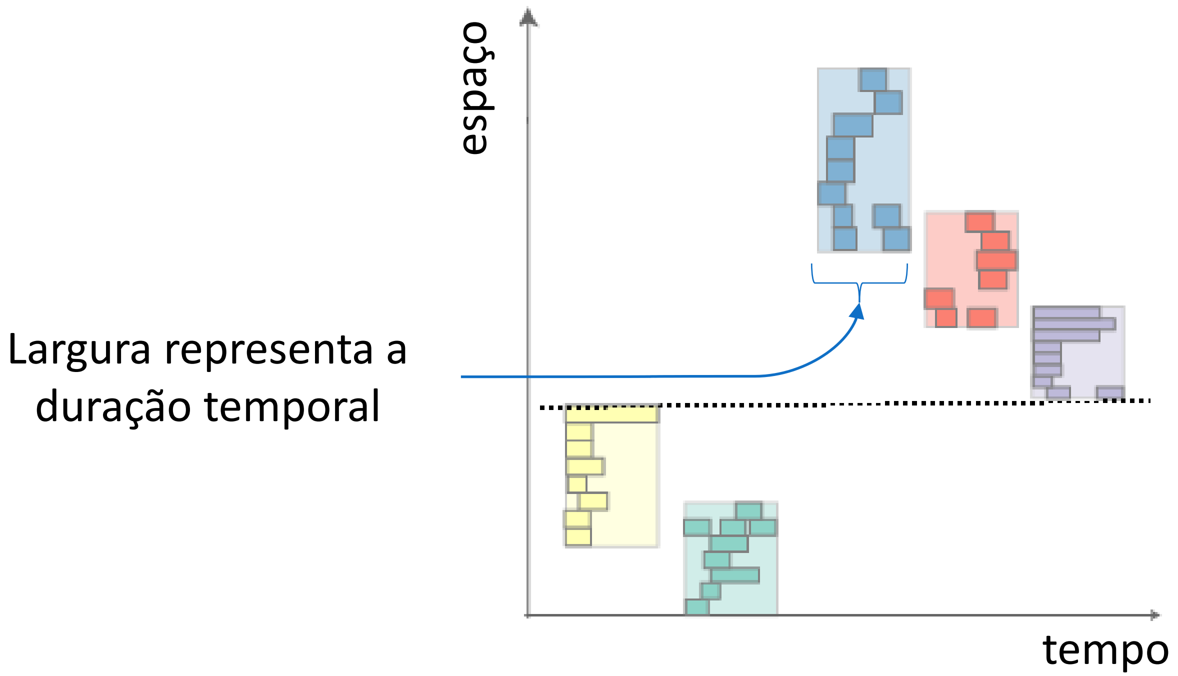
Retângulos externos
representam eventos

Events-Vis: Método

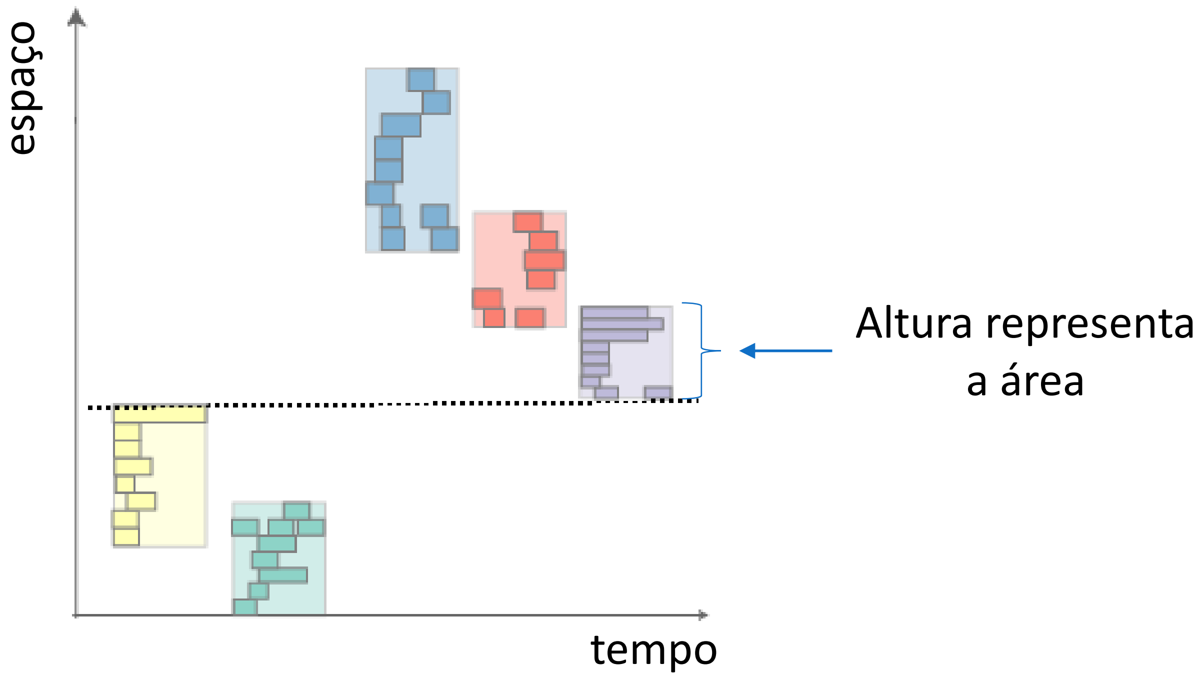


Retângulos internos
representam a
estrutura interna

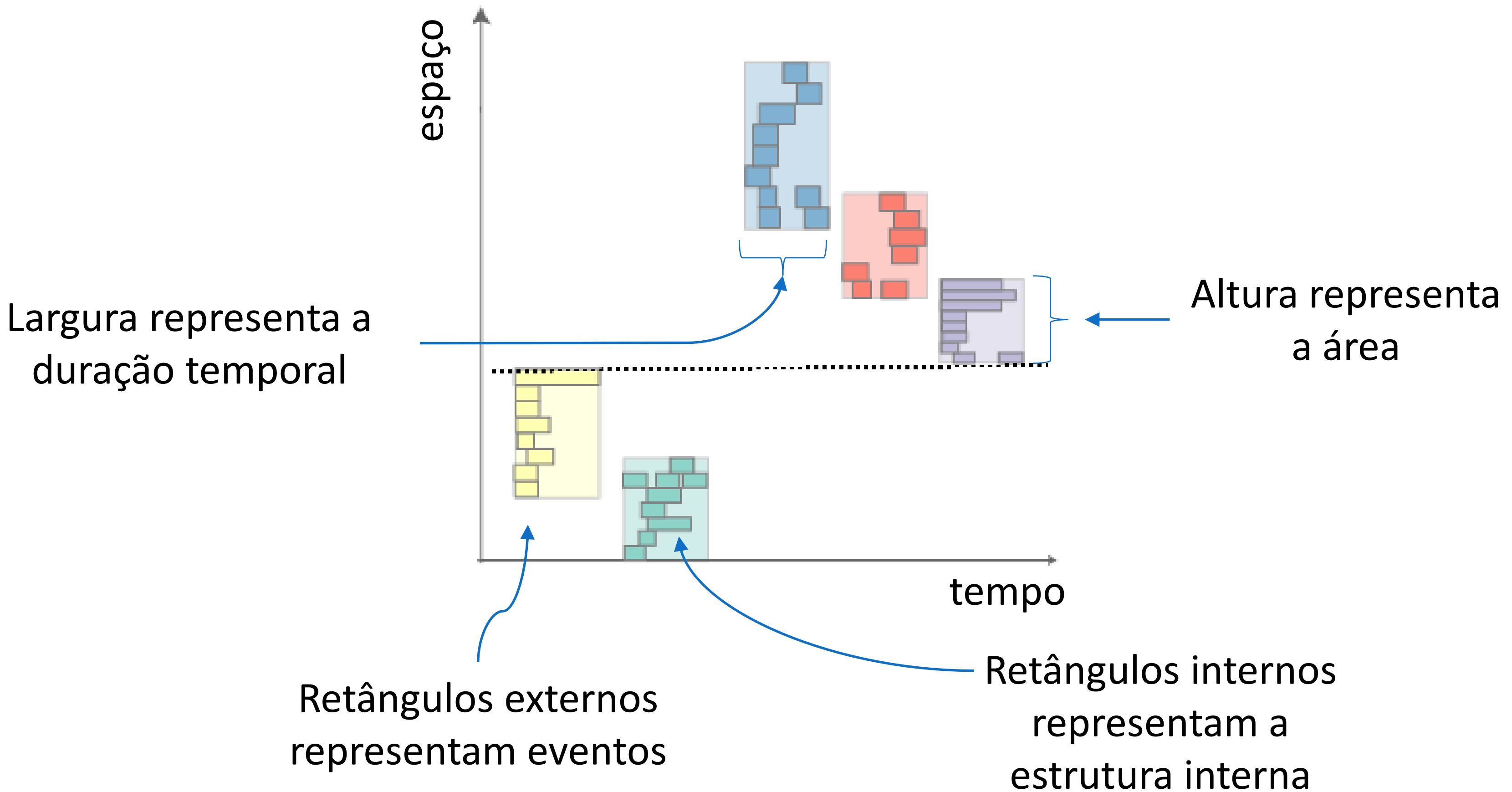
Events-Vis: Método



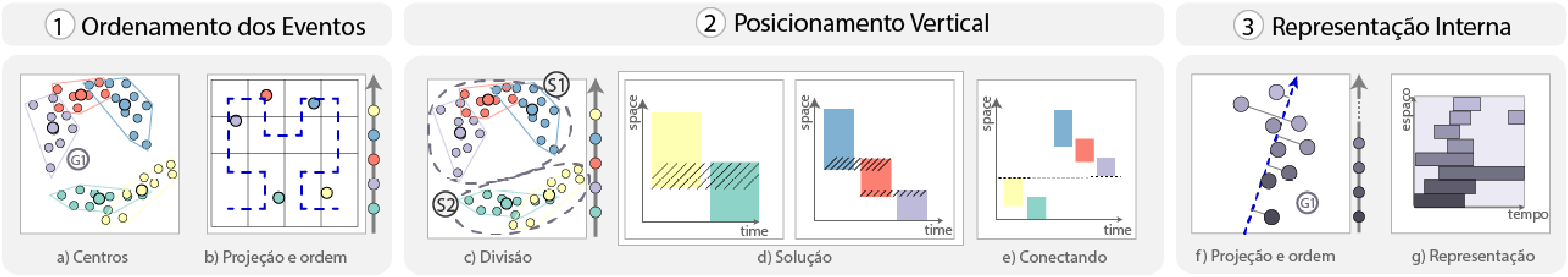
Events-Vis: Método



Events-Vis: Método

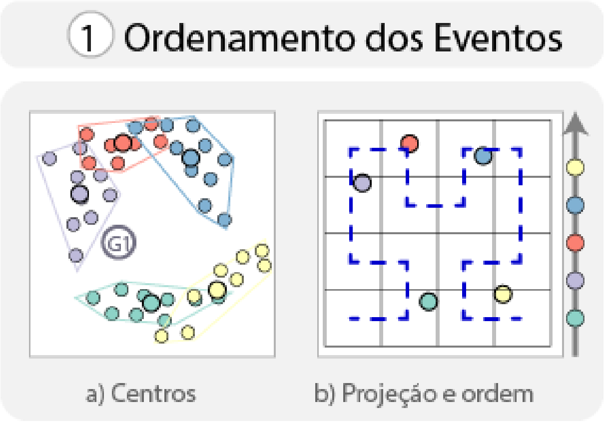


Events-Vis: Método



1. Ordenamento dos Eventos

1. Ordenamento dos Eventos

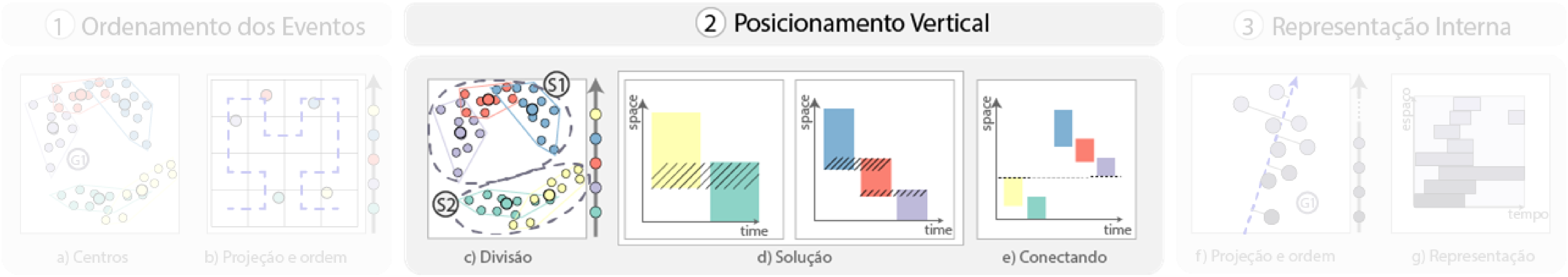


- Calculamos **pontos centrais dos eventos**, como a média entre os pontos de um evento ou o centro de massa.
- Aplicamos uma transformação de **projeção de 2D para 1D** nos pontos centrais e obtemos um ordenamento dos eventos.

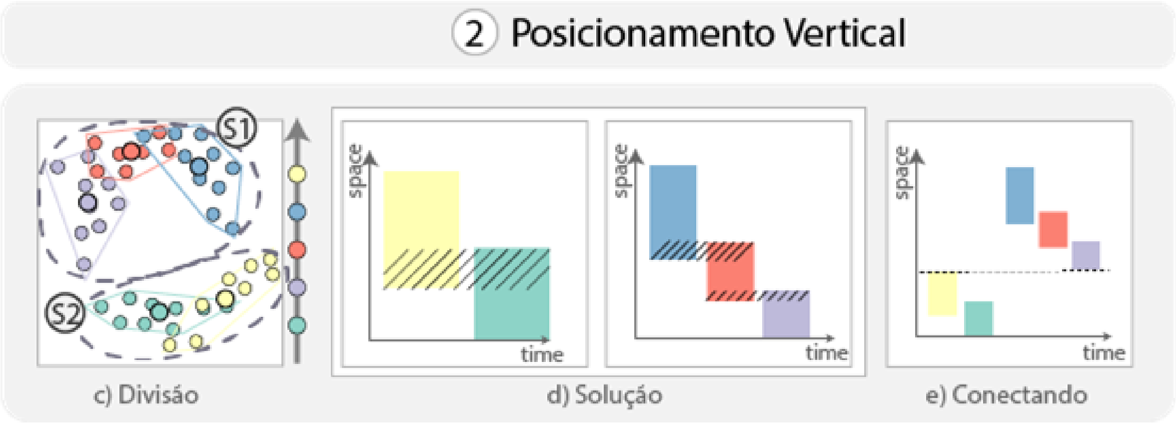
Projeções

- Utilizadas **para representar o espaço bidimensional no eixo vertical** unidimensional preservando o máximo da informação.
- Métodos de projeção considerados:
  - Redução de dimensionalidade: PCA, MDS, t-SNE, UMAP.
  - Indexação espacial: curvas de preenchimento de espaço Hilbert e Morton.

2. Posicionamento Vertical



2. Posicionamento Vertical



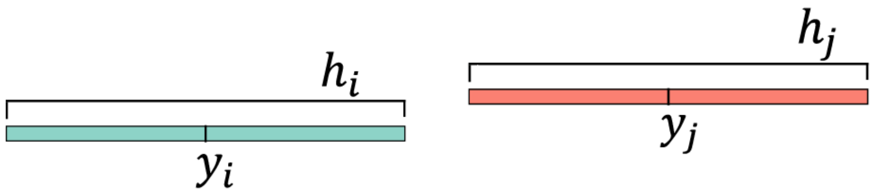
- Espaço será representado verticalmente, dessa forma, a posição **horizontal** e **largura** dos retângulos **não será considerada**.
- Posicionar os eventos verticalmente de forma em que eventos seja representada:
  - **suas áreas;**
  - **suas vizinhanças;**
  - **A área de intersecção** entre eventos.

2.I Formulação do problema de posicionamento

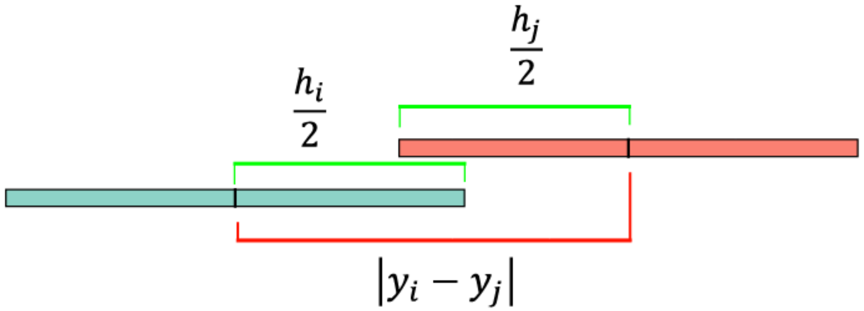
- Temos n eventos denotados por e_i e respectivos segmentos S_i, ambos numerados pela ordem obtida na etapa de projeção.
- Cada evento possui uma área a_i (área do envoltório convexo) e para cada segmento um altura h_i.
- Para cada par de eventos (e_i, e_j) temos uma medida $w_{i,j}$ da área de intersecção entre suas regiões, para cada par de segmentos (S_i, S_j) podemos também calcular a intersecção geométrica entre eles $I_{i,j}$.
- Considere os segmentos posicionados no plano cartesiano sobre a reta $x = 0$, com o ponto médio de S_i em $(0, y_i)$.
- **Problema:** calcular $(y_i)|_{i=1}^{n}$ tais que a ordem vertical dos segmentos será a mesma dos reespectivos eventos, a altura representa a área $h_i = a_i$, e que a intersecção $I_{i,j}$ seja o mais próximo possível de $w_{i,j}$**.**

Intersecção entre segmentos

Seja $I_{i,j}$ uma função que recebe dois segmentos e retorna o valor da intersecção entre eles. Observamos:



Caso os retângulos não apresentem intersecção, o valor é 0.



Caso apresentem, temos:

$$I_{i,j} = \frac{h_i + h_j}{2} - |y_i - y_j|$$

Intersecção entre segmentos

Note que quando os eventos não se intersectam:

$$\frac{h_i + h_j}{2} - |y_i - y_j| < 0$$

Logo podemos definir a forma geral:

$$I_{i,j} \ = \max\left(0, \frac{h_i + h_j}{2} - |y_i - y_j|\right)$$

Quando sabemos a ordem dos segmentos, isto é, sabemos que $y_j \leq y_i$ então:

$$I_{i,j} = \max\left(0, \frac{h_i + h_j}{2} - (y_i - y_j)\right)$$

Solução do problema de posicionamento

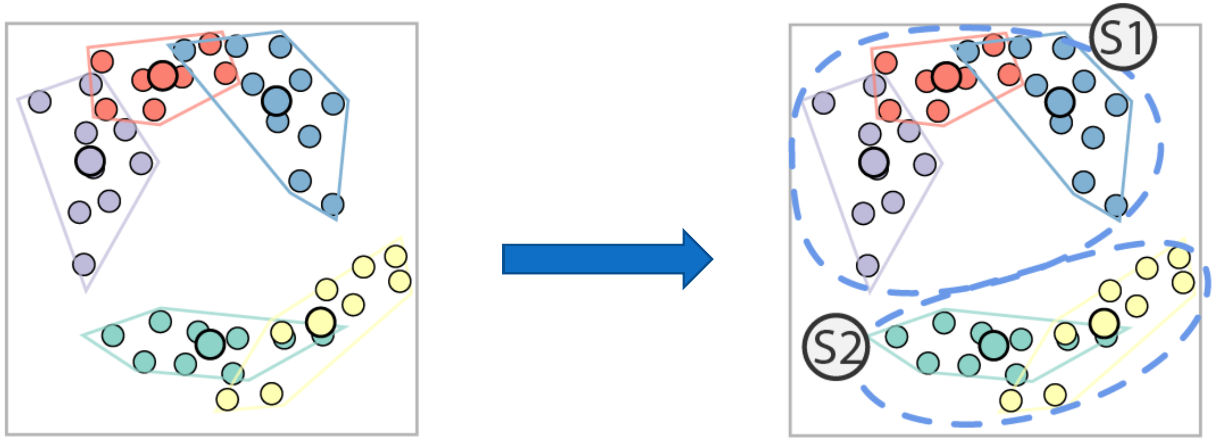
- Soluções propostas:
  - Heurística gulosa
  - Programação Quadrática Mista Inteira
- É vantajoso **separar os eventos em subconjuntos que se intersectam**.
- O problema pode ser resolvido em subproblemas menores.
- Para unir os resultados dos subconjuntos basta posiciona-los de forma sequencial.

Separação em subconjuntos

- Dois eventos (i, j) pertencem a um mesmo subconjunto se, e somente se:
  - eles se intersectam;
  - existe uma sequência $k_1, k_2, \ldots, k_n$ tal que $w_{ik_1} > 0, \ldots, w_{ik_n} > 0$ (“existe uma sequência de eventos que ligam (i, j) por meio de intersecções”).

Separação em subconjuntos

- Dois eventos (i, j) pertencem a um mesmo subconjunto se, e somente se:
    - eles se intersectam;
    - existe uma sequência $k_1, k_2, \dots, k_n$ tal que $w_{ik_1} > 0, \dots, w_{ik_n} > 0$ ("existe uma sequência de eventos que ligam (i, j) por meio de intersecções").



Definição da função objetivo

- Para cada par de de segmentos (S_i, S_j), desejamos aproximar $I_{i,j}$ do valor da área intersecção dos eventos $w_{i,j}$.
- Uma opção é com a distância quadrática:

$$\left(w_{i,j} - I_{i,j}\right)^2$$

- Considerando todos os pares possíveis, o problema é:

$$\min \sum_{i=1}^{n} \sum_{j=i+1}^{n} \left(w_{i,j} - I_{i,j}\right)^2$$

2. II Heurística Gulosa

- **Heurística** é um método **simples, rápido** e que encontre soluções próximas da solução ótima.
- **Sem garantia de encontrar a solução ótima.**
- Uma heurística **gulosa** em **cada etapa** de construção da solução, **decide pela opção que dará o melhor resultado imediato.**
- Iterando baseado na ordem dos eventos iremos **posicionar os segmentos** verticalmente **verificando a intersecção os segmentos consecutivos**.

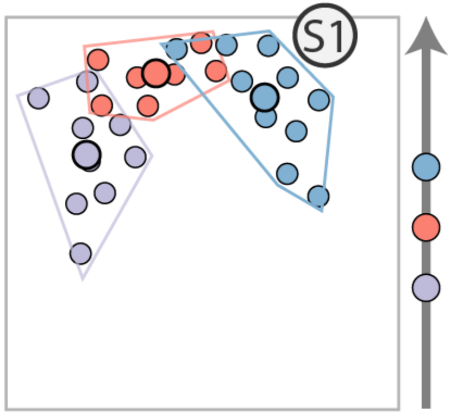
Heurística Gulosa

- A posição do primeiro: $y_1 = \frac{h_1}{2}$ (isto é, o início do segmento está em y = 0).
- Iterando pelos segmentos ordenados, definimos as posições da seguinte forma:

$$y_i = y_{i-1} + \frac{(h_i + h_{i-1})}{2} - w_{i,j}$$

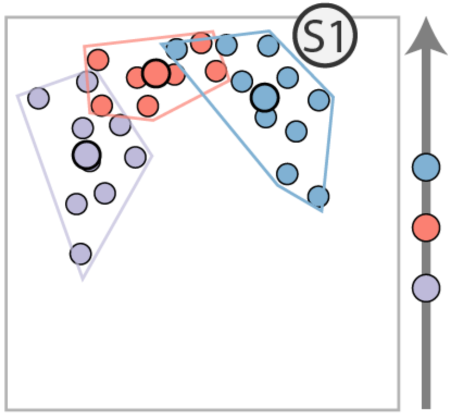
- A cada iteração $\mathrm{I}_{\mathrm{i,i-1}} = \mathrm{w}_{i,i-1}$.
- No entanto, **a intersecção de segmentos não consecutivos não é garantida.**

Heurística Gulosa - exemplo

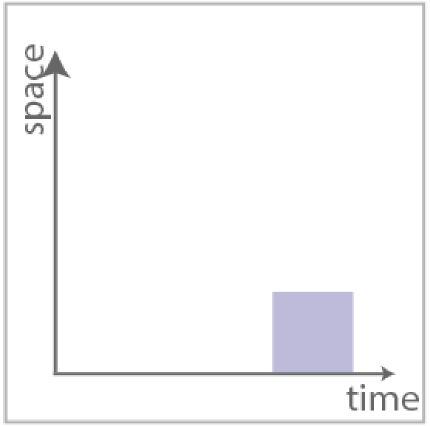


Situação espacial a ser representada e a ordem dos eventos

Heurística Gulosa - exemplo

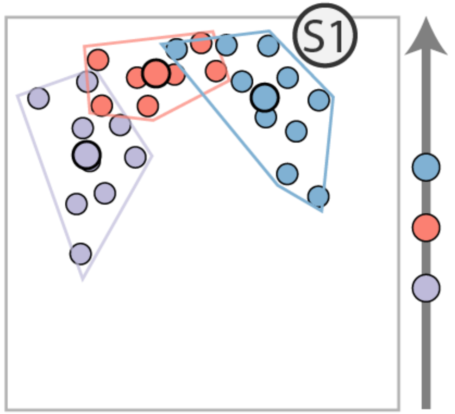


Situação espacial a ser representada e a ordem dos eventos

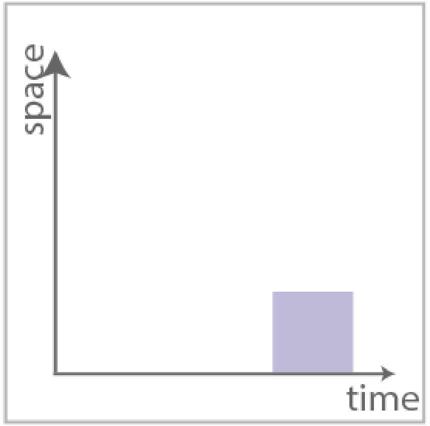


Posiciona o primeiro evento no eixo y = 0

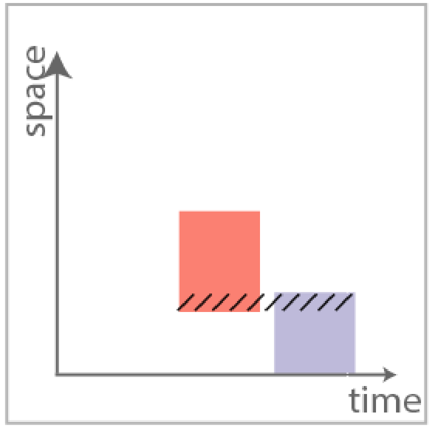
Heurística Gulosa - exemplo



Situação espacial a ser representada e a ordem dos eventos

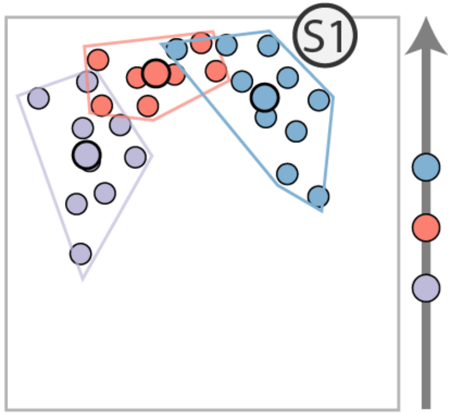


Posiciona o primeiro evento no eixo y = 0

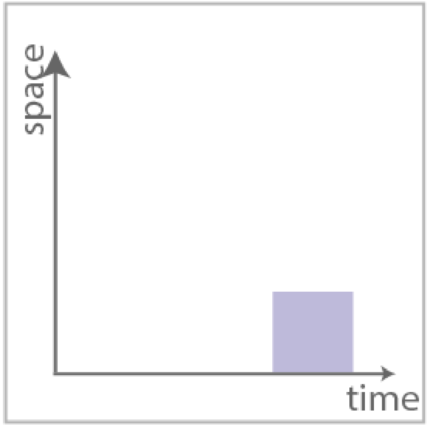


Adiciona o segundo representando a intersecção espacial

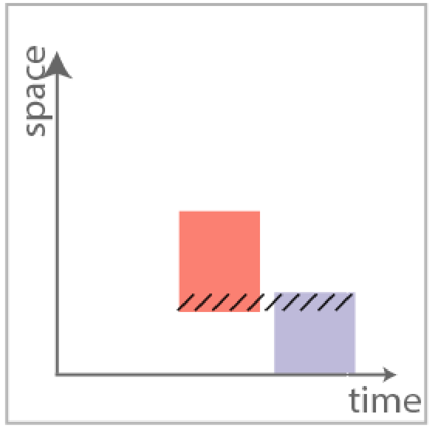
Heurística Gulosa - exemplo



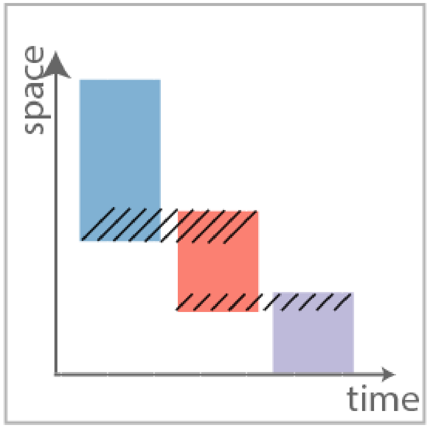
Situação espacial a ser representada e a ordem dos eventos



Posiciona o primeiro evento no eixo y = 0



Adiciona o segundo representando a intersecção espacial



Continuando com as iterações...

2. III Programação Quadrática Mista Inteira (MIQP)

- Programação quadrática é a classe de problemas que minimizam funções quadráticas.
- Desejamos minimizar a diferença quadrática entre $I_{i,j}$ e $w_{i,j}$:

$$\min \sum_{i=1}^{n} \sum_{j=i+1}^{n} \left(w_{i,j} - I_{i,j}\right)^2$$

- Utilizamos programação mista inteira pois com variáveis binárias é possível utilizar de um truque para representar a função max presente na definição da intersecção.

$$I_{i,j} = \max\left(0, \frac{h_i + h_j}{2} - (y_i - y_j)\right)$$

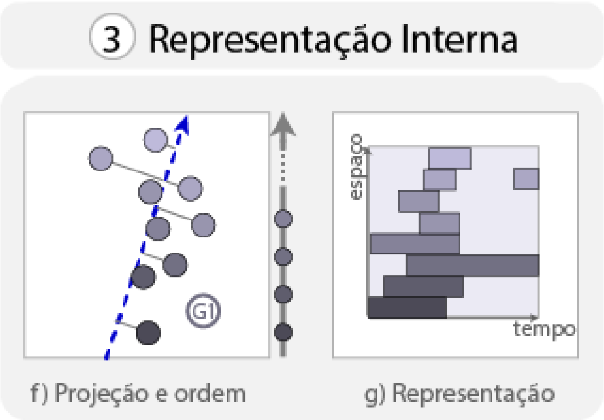
Programação Quadrática Mista Inteira (MIQP)

- O problema final MIQP com função objetivo e restrições será:

$$
\begin{cases}
\min \sum_{i=1}^{n} \sum_{j=i+1}^{n} \left(w_{i,j} - I_{i,j}\right)^2 \\
sub. \forall \text{ i} < \text{j}: \\
y_i \leq y_j \\
0 \leq I_{i,j} \\
\dfrac{h_i + h_j}{2} - \left(y_i - y_j\right) \leq I_{i,j} \\
I_{i,j} \leq b_{i,j} M \\
I_{i,j} \leq \dfrac{h_i + h_j}{2} - \left(y_i - y_j\right) + \left(1 - b_{i,j}\right) M \\
b_{i,j} \in \{0, 1\}
\end{cases}
$$

3. Representação Interna

Events-Vis: Método



- Representação dos **pontos internos de um evento** (opcional caso essa estrutura interna exista).
- Para cada evento, pontos são projetados e ordenados.
- Representados dentro do retângulo do respectivo evento, posicionados horizontalmente de acordo com o tempo e verticalmente de acordo com a ordenação.

Avaliação do método

Avaliação quantitativa do método

- Desejamos avaliar quão boa é a **representação espacial dos dados** obtida no algoritmo de posicionamento vertical.
- **Preservação de distâncias:** a **distância euclidiana** entre os centros dos **eventos** (e_i, e_j) é próxima a **distância vertical** entre os **retângulos** (R_i, R_j)?
- **Preservação de vizinhanças:** os k **eventos mais próximos** do centro do evento e_i e os k **retângulos mais próximos** do retângulo R_i são os mesmos?
- **Preservação de intersecções:** as intersecções são bem representadas?

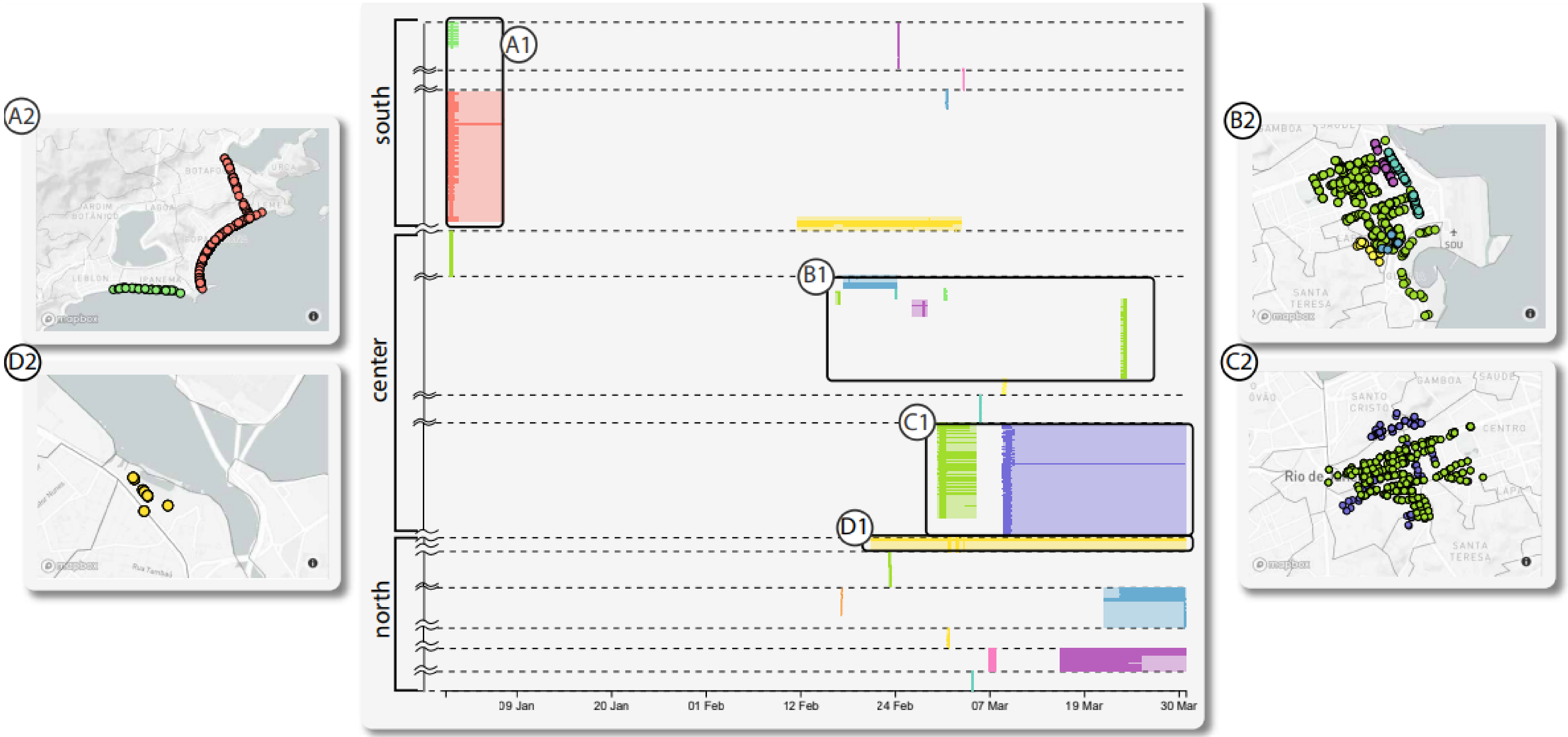
Resultados parciais

- Ambos métodos (guloso e MIQP) apresentam bons resultados.
- Seria esperado que o método MIQP superasse o guloso, porém isso não ocorre, em poucas situações o guloso apresenta resultados piores.
- Devido a demora para solução do método MIQP, guloso se torna a melhor opção.
- Entre as projeções, as que preservaram melhor o espaço foram a PCA e de Hilbert.

Estudo de caso

Conjunto de dados

- Dados de tráfego urbano:
  - Alertas criado por usuários do aplicativo Waze para relatar mudanças no tráfego, trânsito, acidentes na cidade do Rio de Janeiro.
  - Cada ponto apresenta posição espacial (latitude, longitude), tempo (com informação até minutos), uma categoria e um comentário descrevendo.



A1
A2
B1
B2
C1
C2
D1
D2
south
center
north
09 Jan
20 Jan
01 Feb
12 Feb
24 Feb
07 Mar
19 Mar
30 Mar
BOTAFOGO
URCA
JARDIM BOTÂNICO
LAGOA
LEBLON
IPANEMA
GAMBOA
SAÚDE
SANTA TERESA
SANTO CRISTO
CENTRO
Rio de
LAPA
mapbox

Trabalhos futuros

- Desenvolvimento de método de otimização que obtenha solução mais rapidamente e supere os resultados do método guloso.
- Produção de estudos de casos com outros conjuntos de dados.
- Adaptação do método para outros tipos de dados espaço-temporais.

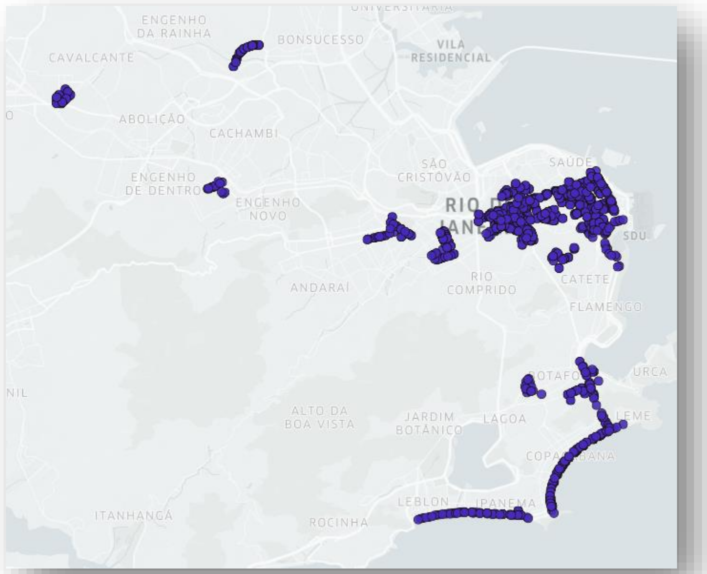
Muito obrigado!
Dúvidas? Sugestões?

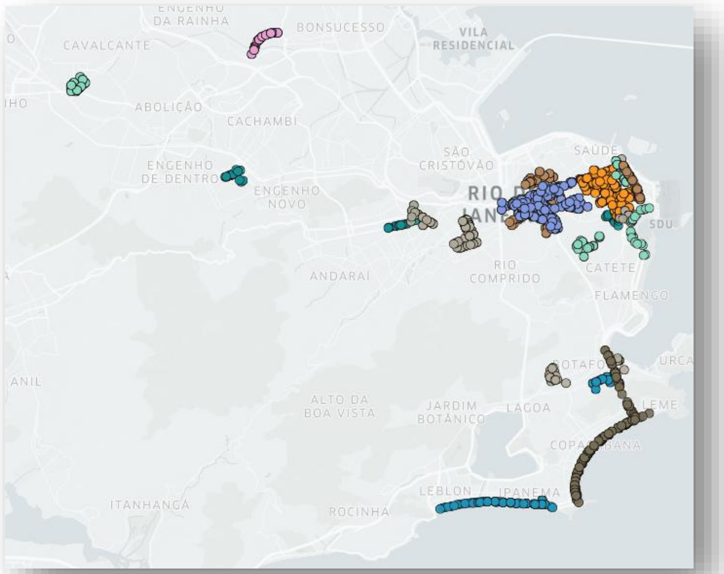
Métodos de agrupamento

- Agrupamento são métodos comuns para a **mineração de dados**, isto é, extrair informações e características importantes de grande quantidade de dados.
- Diversos **métodos adaptados** para dados espaço-temporais já foram propostos, como: ST-GRID, ST-DBSCAN, SNN.
- O método ST-DBSCAN será utilizado nos estudos de caso.

Métodos de agrupamento

- Um método adaptado deve considerar a **proximidade espacial e temporal** para determinar grupos, e além disso, considerar a diferença existente entre distâncias espaciais e distâncias temporais.





Curva de Hilbert

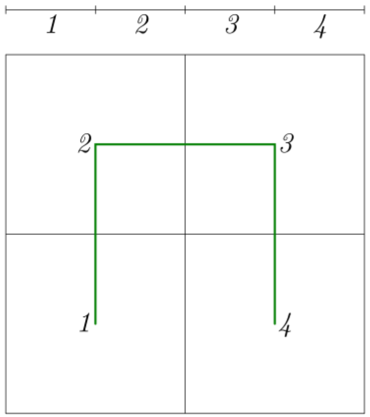


Fig. 1.

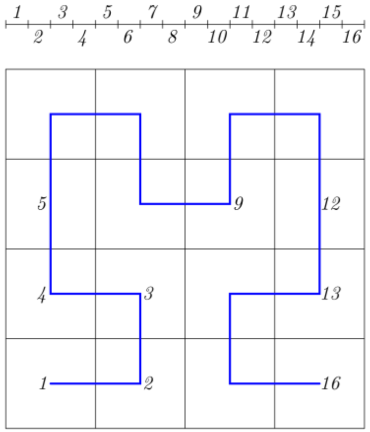


Fig. 2.

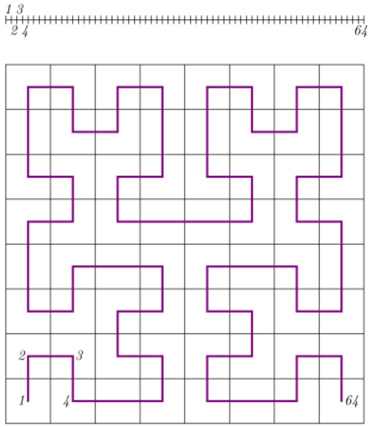
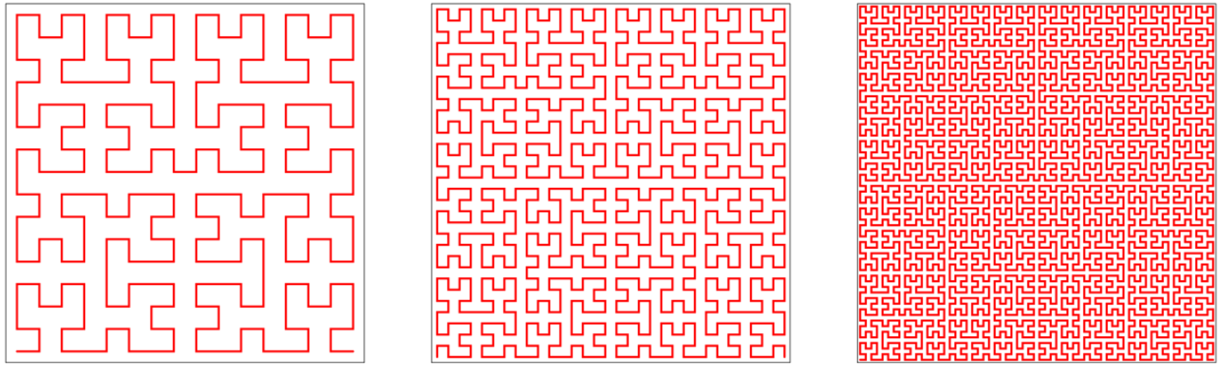


Fig. 3.



Programação Quadrática Mista Inteira (MIQP)

Utilizamos um truque que em um cenário de otimização com restrições, desejamos resolver:

$$\max f(c)$$
$$c = \max(a_1, a_2)$$

Com (a_1, a_2) variáveis de decisão, podemos definir uma variável binária b e as restrições:

$$\begin{cases} c \geq a_1 \\ c \geq a_2 \\ c \leq a_1 + bM \\ c \leq a_2 + (1 - b)M \end{cases}$$

Com M um valor alto o suficiente para não ser ultrapassado por a_1, a_2, tem-se que $c = \max(a_1, a_2)$.