

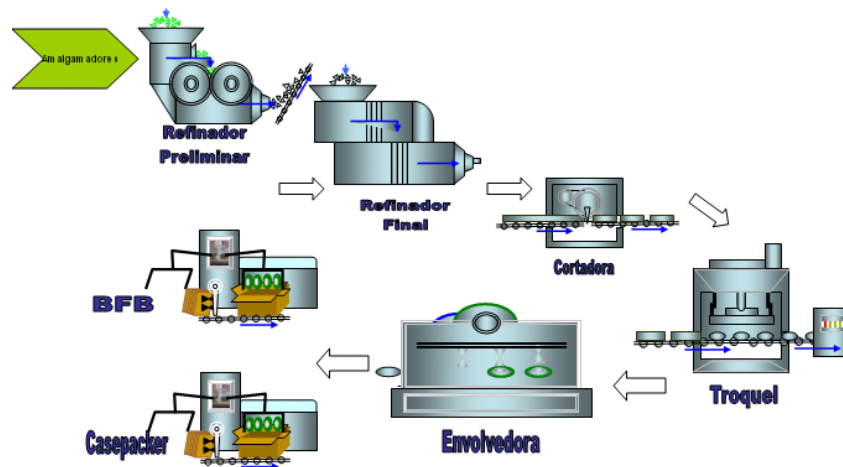
# Aplicaciones — programación producción

Tesis Andrea Rosales (P&G)

Para la producción de jabón se tienen diferentes líneas de producción, donde el jabón pasa por diferentes etapas.

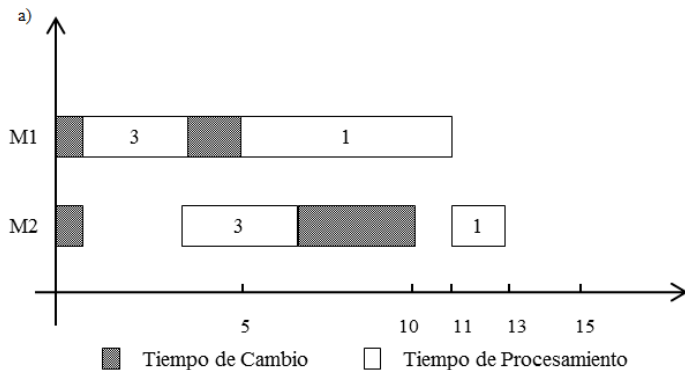
Terminación de la producción de jabón:

- La viruta de jabón se le adiciona fragancia y se amalgama para pasar a refinarse
- Posteriormente se corta y se le da forma (proceso de troquelado)
- Para pasar a la envolvedora donde se empacan los jabones



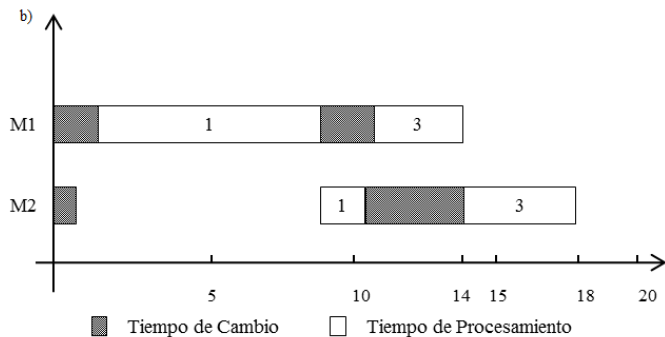
# Aplicaciones — programación producción

- En una misma semana se pueden realizar diferentes tipos de jabones (y ordenes)
- Entre cada orden se necesita:
  - limpiar las máquinas (refinadora)
  - ajustar las máquinas cambiar piezas (troquel, cortadora, etc.)
- Lo que lleva diferentes tiempo de cambio entre cada orden



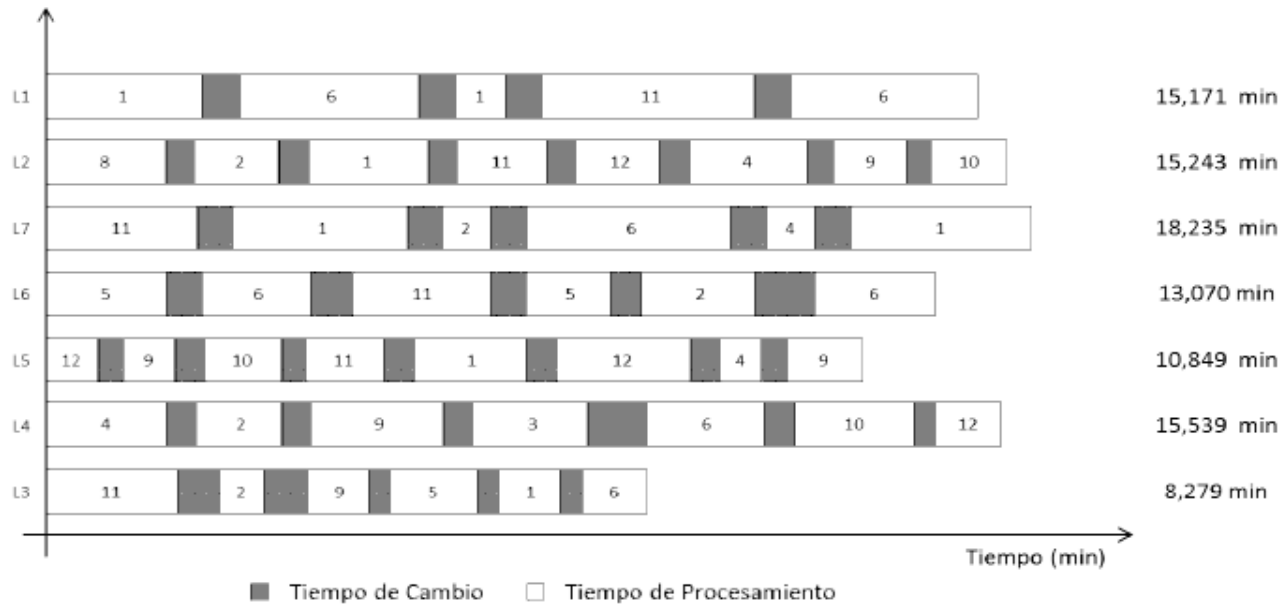
No lleva el mismo tiempo hacer la orden  $i$  antes de la  $j$  que la  $j$  antes de la  $i$

Tenemos  $n$  ordenes por procesar en un día. Desea minimizar el tiempo total de producción



Idea es transformar el problema a uno del agente viajero donde cada ciudad es una orden a procesar y la distancia entre ciudades el tiempo de cambio entre las órdenes

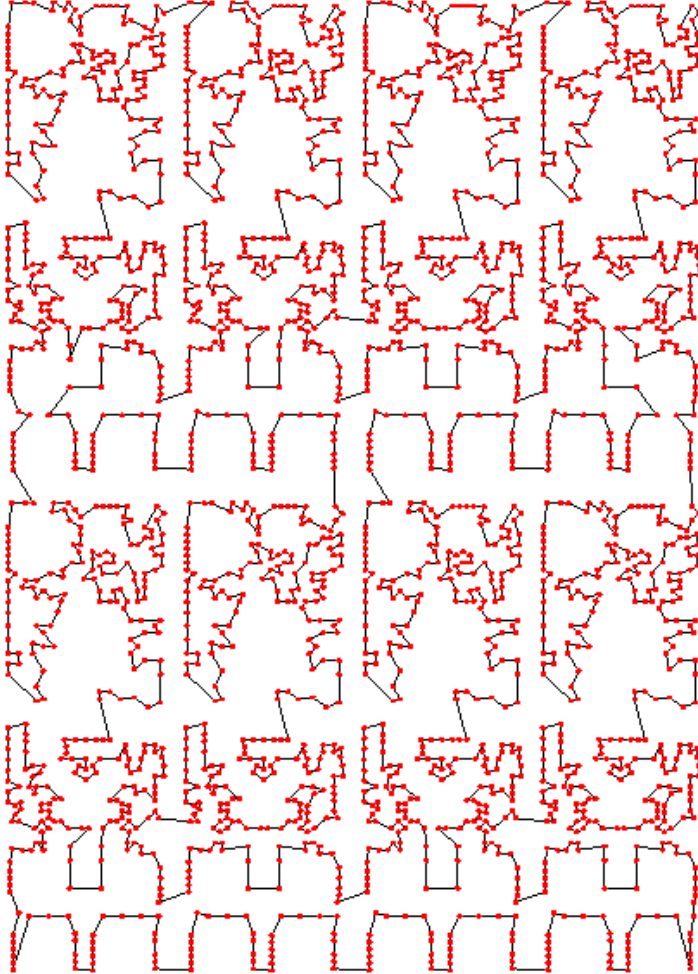
# Aplicaciones — programación producción



- Se implementaron diferentes heurísticas para mejorar el plan de producción y resolver el TSP correspondiente se obtuvieron reducciones en promedio de entre 15 y 25 horas al mes (ahorros considerables para la compañía).
- Se le proporcionó un paquete (EXCEL) que les ayuda en la planeación en una implementación sencilla.

# Aplicaciones- fabricación de circuitos

---

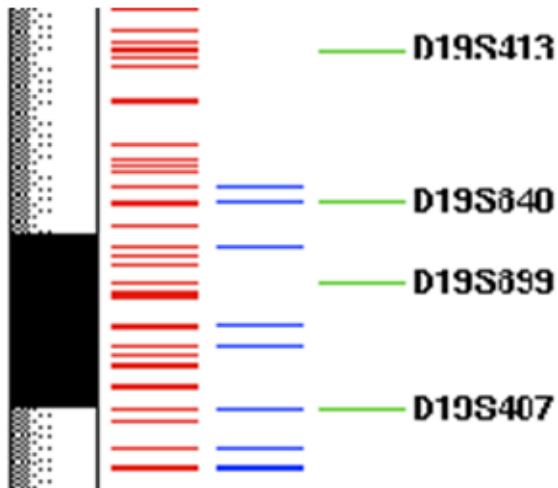


Donde cada ciudad representa un puto de soldado y lo que se quiere es minimizar la distancia recorrida por la máquina soldadora para manufacturar el circuito.

# Aplicaciones- secuenciación genoma

---

R. Agarwala et al., (2000) “A Fast and Scalable Radiation Hybrid Map Construction and Integration Strategy” *Genome Research* 10, pp350-364 .



Aquí se desea construir mapas híbridos de radiación como parte de la investigación en secuenciación de genoma.

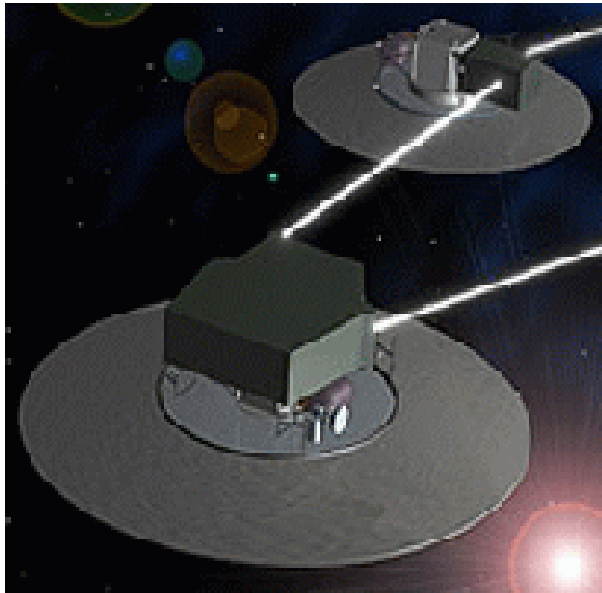
El TSP permite la integración de mapas locales para integrar mapas completos. Las ciudades son los mapas locales y la distancia entre ellos es una medida de la similitud de que un mapa siga a otro .

# Aplicaciones- investigación espacial

---

## NASA *Starlight* space interferometer program

Bailey, A.C., McLain T.W. and Beard R.W. (2000) “Fuel saving strategies for separated spacecraft interferometry” *Proceedings of the AIAA Guidance, Navigation and Control Conference* . CiteSeerX 10.1.1.40.1906



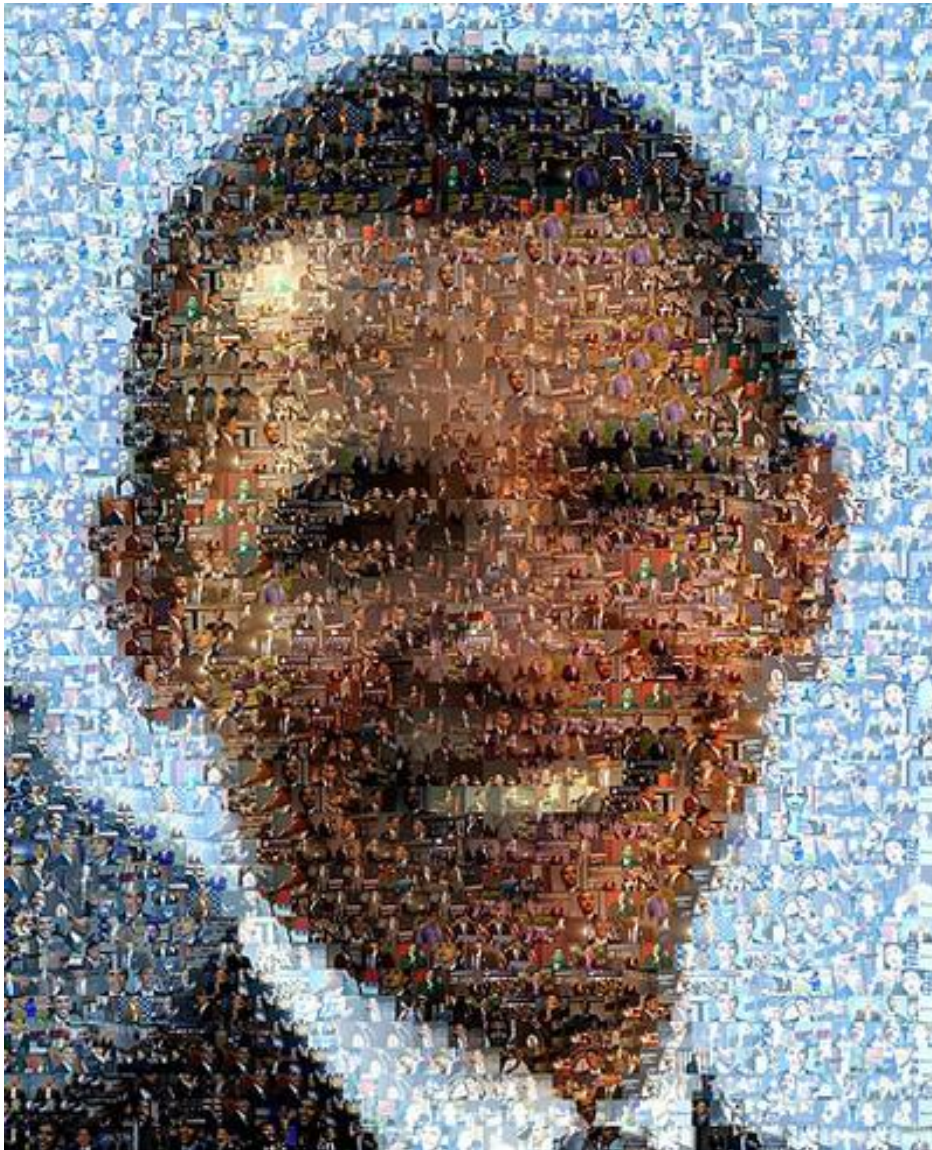
Un grupo de investigadores en Hernandez Engineering en Houston y la Universidad de Brigham Young, ha estudiado la posibilidad de usar heurísticas que resuelven el TSP para optimizar la secuencia en que se mapean los objetos celestes y minimizar el uso de energía para mover el par de satélites involucrados en la misión.

Aquí las ciudades son los objetos celestes a ser mapeados y el costo de viajar de una ciudad a otra es la cantidad de combustible necesario para reposicionar los satelites para pasar de la toma de una imagen a la otra.



# Foto-mosaicos

---



# Foto-mosaicos de dominó

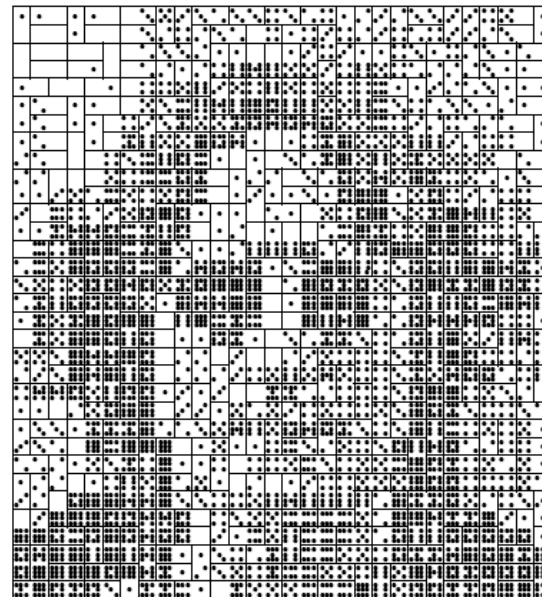
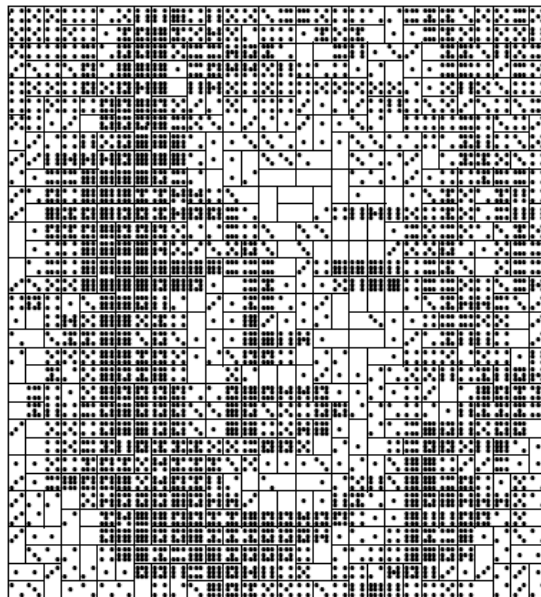
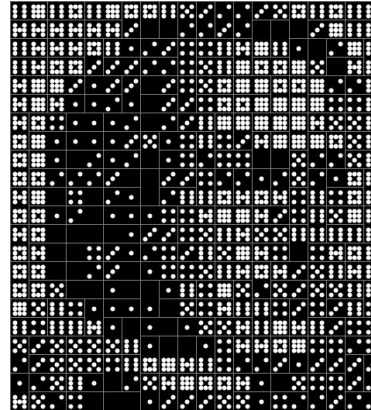
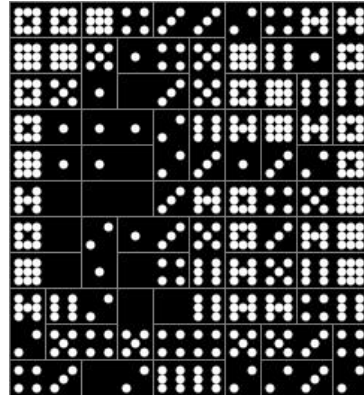
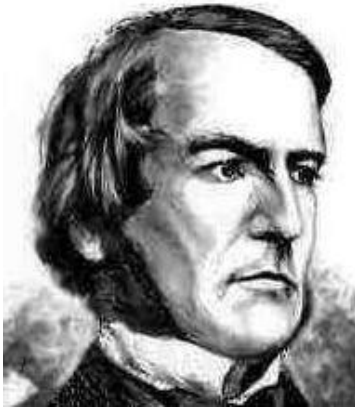


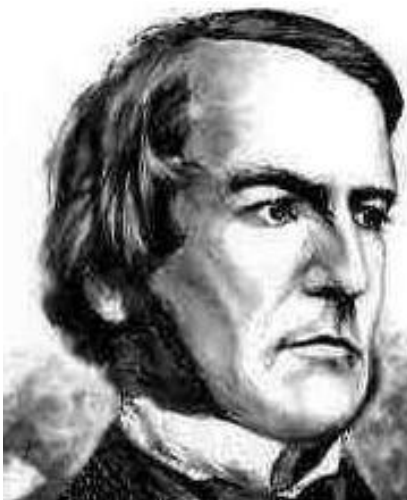
Fig. 1. Marilyn (9 sets) and John (9 sets)



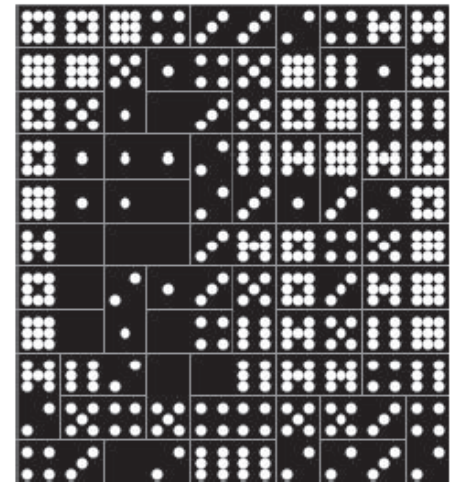
# Fotomosaicos- formulación de programación lineal

Dividimos la foto sobre la cual se desea hacer el foto mosaico en  $m \times n$  sub-áreas.

Si tenemos  $u_f$  fichas de tipo  $f \in F$  (colección de fotos, fichas de dominó), calculamos la intensidad lumínica  $b_f$  de la ficha  $f$  y  $\beta_{ij}$  la intensidad de la sub-área.



9	9	9	7	5	4	3	6	9	9
9	9	7	3	4	7	8	5	3	9
9	7	2	2	4	7	9	9	7	7
9	3	1	2	3	7	9	9	8	9
9	2	2	1	3	4	3	5	4	8
9	3	2	1	4	8	8	6	6	9
9	1	1	2	4	7	8	4	8	9
9	2	2	1	4	7	8	5	7	9
8	7	2	1	3	7	8	7	5	7
5	6	6	5	3	4	7	7	5	4
4	5	1	4	7	8	4	3	5	3

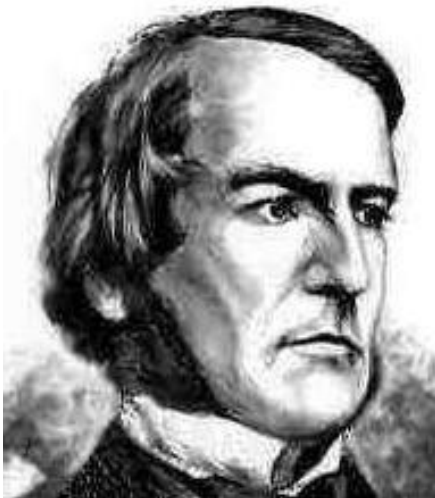


# Fotomosaicos- formulación de programación lineal

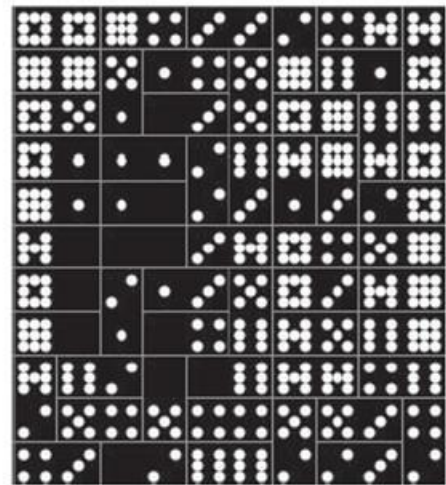
Dividimos la foto sobre la cual se desea hacer el foto mosaico en  $m \times n$  sub-áreas.

Si tenemos  $u_f$  fichas de tipo  $f \in F$  (colección de fotos, fichas de dominó), calculamos la intensidad lumínica  $b_f$  de la ficha  $f$  y  $\beta_{ij}$  la intensidad de la sub-área.

Sea  $x_{fij} = 1$  si acomodamos la ficha  $f$  en la subárea  $ij$ , la variable de decisión, entonces el problema es:



9	9	9	7	5	4	3	6	9	9
9	9	7	3	4	7	8	5	3	9
9	7	2	2	4	7	9	9	7	7
9	3	1	2	3	7	9	9	8	9
9	2	2	1	3	4	3	5	4	8
9	3	2	1	4	8	8	6	6	9
9	1	1	2	4	7	8	4	8	9
9	2	2	1	4	7	8	5	7	9
8	7	2	1	3	7	8	7	5	7
5	6	6	5	3	4	7	7	5	4
4	5	1	4	7	8	4	3	5	3



# Fotomosaicos- formulación de programación lineal

---

Dividimos la foto sobre la cual se desea hacer el foto mosaico en  $m \times n$  sub-áreas.

Si tenemos  $u_f$  fichas de tipo  $f \in F$  (colección de fotos, fichas de dominó), calculamos la intensidad lumínica  $b_f$  de la ficha  $f$  y  $\beta_{ij}$  la intensidad de la sub-área.

Sea  $x_{fij} = 1$  si acomodamos la ficha  $f$  en la subárea  $ij$ , la variable de decisión, entonces el problema es:

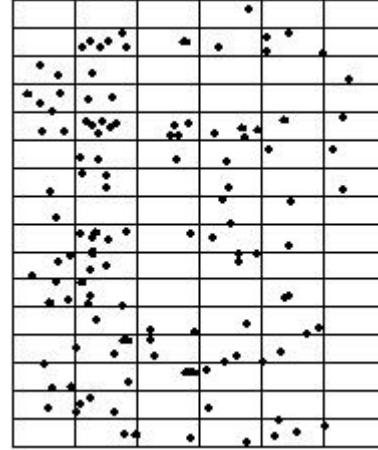
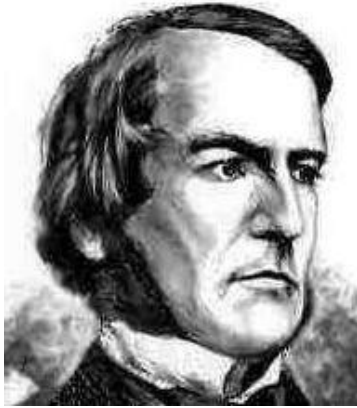
$$\text{Minimizar} \quad \sum_{f \in F} \sum_{i=1}^m \sum_{j=1}^n (b_f - \beta_{ij})^2 x_{fij}$$

Sujeto a

$$\begin{aligned} \sum_{i=1}^m \sum_{j=1}^n x_{fij} &\leq u_f && \forall f \in F \\ \sum_{f \in F} x_{fij} &= 1 && \forall 1 \leq i \leq m, 1 \leq j \leq n \end{aligned}$$

# Foto mosaicos y el agente viajero

---



Si ahora en cada sub-área ubico de manera aleatoria un número de ciudades en función de que tan obscura sea el área.

Podemos plantear un problema del agente viajero sobre esas ciudades y así obtener un foto-mosaico generado por una sola línea continua que una todas las ciudades con mínima distancia.



# Foto mosaicos y el agente viajero



Problema Fotomosaico de la Mona Lisa con 100,000 ciudades (Robert Bosch 2009) Premio \$1,000

