# GRASP para el diseño de redes de transporte público

*Antonio Mauttone*

Depto. de Investigación Operativa - InCo

Facultad de Ingeniería - UdelaR

*Metaheurísticas y Optimización sobre Redes*

Setiembre 2019

# Agenda

- Greedy Randomized Adaptive Search Procedures.
- Planificación de sistemas de transporte público.
- Optimización multi-objetivo: conceptos y metodologías.
- Diseño de recorridos en transporte público: problema y algoritmo de solución.
- Aplicación: caso de estudio real.

# Greedy Randomized Adaptive Search Procedures

# Contexto

Problema de optimización combinatoria:

- Conjunto de base $E = \{1,\ldots,n\}$
- Conjunto de soluciones factibles $F \subseteq 2^E$
- Función objetivo: $f$: $2^E \rightarrow \Re$
- En la versión de minimización buscamos la solución óptima $S^* \in F$, tal que $f(S^*) \leq f(S)$, $\forall S \in F$

(Resende y Ribeiro, 2010)

# GRASP: estrategia general

```
procedure GRASP(Max_Iterations, Seed)
    Read_Input();
    for k = 1,...,Max_Iterations do
        Solution ← Greedy_Randomized_Construction(Seed);
        if Solution is not feasible then
            Solution ← Repair(Solution);
        end;
        Solution ← Local_Search(Solution);
        Update_Solution(Solution, Best_Solution);
    end;
    return Best_Solution;
end GRASP.
```

# GRASP: construcción

```
procedure Greedy_Randomized_Construction(Seed)
    Solution ← ∅;
    Initialize the set of candidate elements;
    Evaluate the incremental costs of the candidate elements;
    while there exists at least one candidate element do
        Build the restricted candidate list (RCL);
        Select an element s from the RCL at random;
        Solution ← Solution ∪ {s};
        Update the set of candidate elements;
        Reevaluate the incremental costs;
    end;
    return Solution;
end Greedy_Randomized_Construction.
```

# GRASP: búsqueda local

```
procedure Local_Search(Solution)
    while Solution is not locally optimal do
        Find s' ∈ N(Solution) with f(s') < f(Solution);
        Solution ← s';
    end;
    return Solution;
end Local_Search.
```

# Planificación de sistemas de transporte público

# Planificación de sistemas de transporte público

| **Etapa** | **Decisiones** | **Objetivos principales** | **Principal tomador de decisiones** | **Horizonte temporal** |
|---|---|---|---|---|
| Diseño de recorridos | Trazados de recorridos | Mín. tiempo viaje y costo de operación | Entidad reguladora | Largo plazo |
| Determinación de frecuencias | Valores de frecuencias | Mín. tiempo viaje y costo de operación | Entidad reguladora | Mediano plazo |
| Construcción de tablas de horarios | Hora de partida y llegada de viajes | Mín. tiempo viaje y costo de operación | Entidad reguladora, operadores | Mediano/corto plazo |
| Asignación de flota | Asignación de buses a viajes | Mín. costo de operación | Operadores | Mediano/corto plazo |
| Asignación de personal | Asignación de personal a buses | Mín. costo de operación | Operadores | Mediano/corto plazo |

# Planificación de sistemas de transporte público

| Etapa | Decisiones | Objetivos principales | Principal tomador de decisiones | Horizonte temporal |
|---|---|---|---|---|
| Diseño de recorridos | Trazados de recorridos | Mín. tiempo viaje y costo de operación | Entidad reguladora | Largo plazo |
| Determinación de frecuencias | Valores de frecuencias | Mín. tiempo viaje y costo de operación | Entidad reguladora | Mediano plazo |
| Construcción de tablas de horarios | Hora de partida y llegada de viajes | Mín. tiempo viaje y costo de operación | Entidad reguladora, operadores | Mediano/corto plazo |
| Asignación de flota | Asignación de buses a viajes | Mín. costo de operación | Operadores | Mediano/corto plazo |
| Asignación de personal | Asignación de personal a buses | Mín. costo de operación | Operadores | Mediano/corto plazo |

# Problemas de optimización en TP

- De complejidad combinatoria y no lineal.
- Con estructura subyacente de red.
- Gran tamaño.
- Sub-sistemas complejos: comportamiento de pasajeros.
- Metaheurísticas: alternativa de solución viable.

# Optimización multi-objetivo

# Optimización multi-objetivo

Optimización (Ehrgott y Gandibleux, 2002):

Hacer una tarea de la “mejor forma posible”, respecto a un <u>único criterio</u>, p.e. minimización de costos o maximización de beneficios.

$$\boxed{\begin{array}{l} \min f(x) \\ \text{s.a.} \\ \qquad x \in X \end{array}}$$

# Optimización multi-objetivo

- En algunos casos es imposible resumir en un único objetivo, diferentes opiniones, motivaciones y metas encontradas en un proceso de decisión donde hay <u>intereses en conflicto</u>.
- Imposible arribar a una solución “ideal”, es decir, aquella que optimiza cada objetivo en forma individual.

# Optimización multi-objetivo

Optimización multi-objetivo:

Diferentes funciones objetivo, parcialmente contradictorias y a veces expresadas en unidades diferentes.

$$\boxed{\begin{array}{l} \text{“min”}\ [f_1(x), \ldots, f_m(x)] \\ \text{s.a.} \\ \qquad x \in X \end{array}}$$

# Optimización multi-objetivo

Definiciones y Notación (Deb, 2001; Ehrgott, 2005):

- $x' \in X$ es <u>eficiente</u> (o Pareto optimal) si no existe otra $x \in X$ tal que $f(x)$ "≤" $f(x')$.
- $f(x)$ "≤" $f(x')$ si $f_i(x) \leq f_i(x')$ $\forall i \in 1..m$ y

  $\exists j \in 1..m$ tal que $f_j(x) < f_j(x')$

- Si $x'$ es eficiente entonces $f(x')$ es un punto <u>no dominado</u>.
- Si $x^1, x^2 \in X$ cumplen $f(x^1)$ "≤" $f(x^2)$ entonces se dice que $x^1$ domina a $x^2$ y $f(x^1)$ domina a $f(x^2)$.

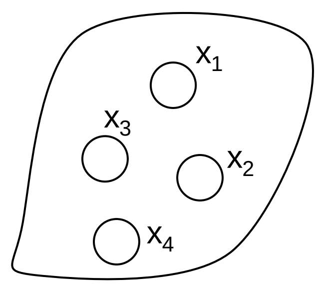
# Optimización multi-objetivo

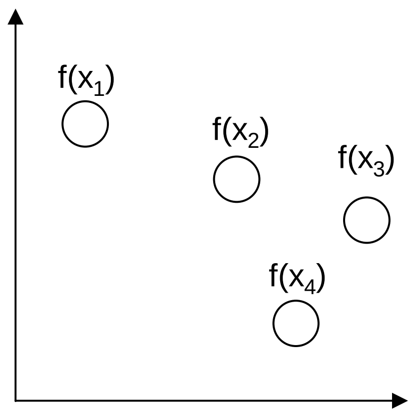
Definiciones y Notación:

- El subconjunto de todas las soluciones eficientes de $X$ se denomina <u>conjunto eficiente</u> $X_E$.
- El conjunto de todos los puntos no dominados

  $y'=f(x') \in Y$, donde $x' \in X_E$, se denomina <u>conjunto no dominado</u> $Y_N$ (o frente de Pareto).
- Muchas veces se utilizan los términos solución eficiente y no dominada indistintamente.

# Optimización multi-objetivo




Conjunto eficiente $X_E = \{x_1,x_2,x_4\}$.

Conjunto no dominado $Y_N = \{f(x_1),f(x_2),f(x_4)\}$.

$x_4$ domina a $x_3$.

$x_1,x_2,x_4$ son no dominados.

# Optimización multi-objetivo

- Para seleccionar una única solución de $X_E$ se requiere de información adicional, que se asume dada por un agente decisor (decision maker, DM).

- Información que incluye criterios subjetivos, políticos, etc.

- Modos *a priori*, *a posteriori* e *interactivo*.

# Optimización multi-objetivo

Optimización combinatoria multi-objetivo (MOCO):

- El dominio *X* es discreto.
- Cuando se adopta el modo *a posteriori*, el problema resultante es usualmente *NP*-hard (no hay chances de encontrar un algoritmo de orden polinomial para encontrar $X_E$). Incluso cuando los problemas asociados de objetivo único son fáciles de resolver. Por ejemplo

$$
\begin{array}{l}
\text{“min” } \left(\Sigma_{\text{i}=1..n} c_{i1} x_i,\ \Sigma_{\text{i}=1..n} c_{i2} x_i\right) \\
\text{s.a.} \\
\quad x \in \{0,1\}^n \\
\quad x \neq (0,\ldots,0) \\
\quad c_{i1}, c_{i2} \geq 0,\ \text{i}=1..n
\end{array}
$$

(Ehrgott y Gandibleux, 2004)

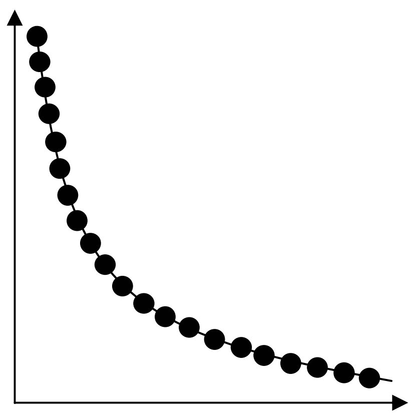
# Optimización multi-objetivo

Métodos aproximados para MOCO:

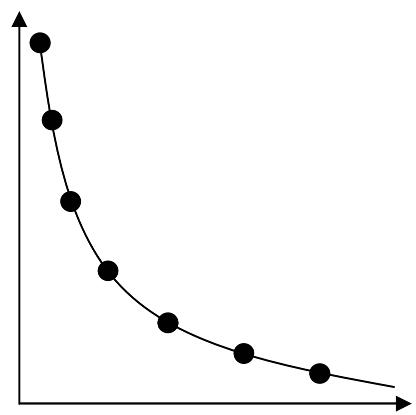
- La resolución exacta de un problema MOCO (adoptando el modo *a posteriori*) además de ser costosa (por su complejidad computacional) no es práctica (desde el punto de vista del DM).
- Significado del término “aproximado”.
- Algoritmos que encuentran un sub-conjunto de $X_E$ (Sayin y Kouvelis, 2004). Cada solución encontrada es eficiente (o Pareto optimal).
- <u>Heurísticas multi-objetivo</u>: encuentran un conjunto de soluciones no dominadas no necesariamente en $X_E$.

# Optimización multi-objetivo

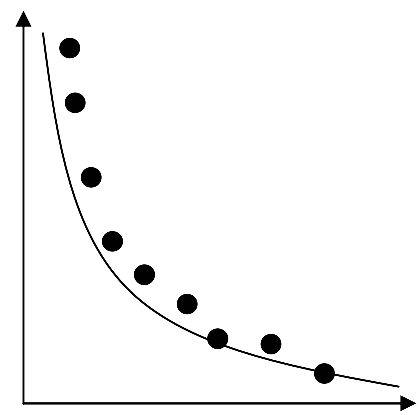
Conjunto $Y_N$
(solución exacta)



Subconjunto de $Y_N$
(soluciones Pareto optimales)



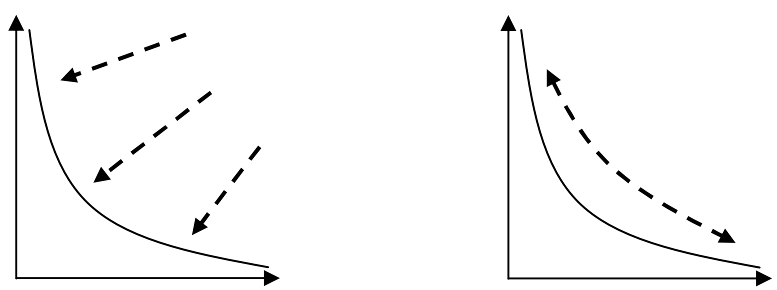
Conjunto de soluciones no dominadas (no necesariamente Pareto optimales)



# Optimización multi-objetivo

## Heurísticas multi-objetivo:

- Mecanismos específicos de exploración del espacio de soluciones. Cercanía y diversidad.



- Metaheurísticas multi-objetivo: Adaptaciones de sus contrapartes de objetivo único. Basadas en poblaciones (Algoritmos Genéticos, Coello 2000, Deb 2001) y en trayectorias (Tabu Search, Simulated Annealing, Ehrgott y Gandibleux, 2004).

# Optimización multi-objetivo

## GRASP (original):

Greedy Randomized Adaptive Search Procedures (Feo y Resende, 1995).

```
procedure GRASP
  bestSolution = {};
  for i=1 to Iterations
    solution = Construction(i);
    solution = LocalSearch(solution);
    Update(bestSolution, solution);
  end for;
  return bestSolution;
end GRASP;
```

# Optimización multi-objetivo

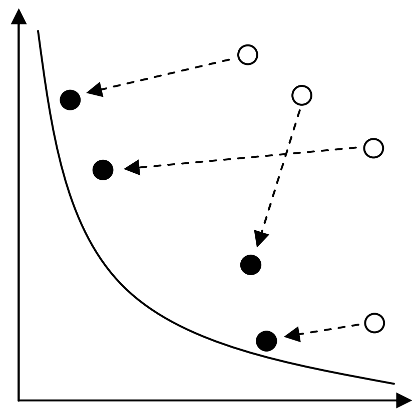
## GRASP (multi-objetivo):

Variantes para problemas MOCO (Baldoquín, 2002; Soares y Arroyo, 2004).

```
procedure MO_GRASP
  paretoFront = {};
  for i=1 to Iterations
    solution = Construction(i, tradeOff_i);
    solution = LocalSearch(solution, tradeOff_i);
    Update(paretoFront, solution);
  end for;
  return paretoFront;
end GRASP;
```

# Optimización multi-objetivo

## MO_GRASP: GRASP Multi objetivo



○ Soluciones resultantes de la construcción

● Soluciones resultantes de la búsqueda local

- - -▶ Trayectoria de la búsqueda local

—— Frente de Pareto

# Diseño de recorridos en transporte público

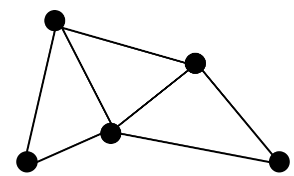
# Transit Network Design Problem

- Definición de Baaj y Mahmassani (1991).

- Diseño de <u>recorridos</u> y <u>frecuencias</u> para un sistema de transporte público,
- optimizando simultaneamente los objetivos de <u>usuarios</u> y <u>operadores</u>,
- en base a información relativa a la <u>red de calles</u> y a la <u>demanda inter-zonal de viajes</u>.

# TNDP: Modelo

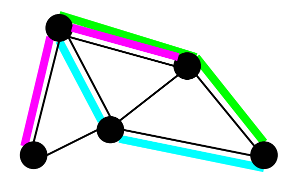
- Datos del problema:
  - Un grafo no dirigido *G* que representa la red de calles (o sistema de zonas); las aristas son ponderadas con el tiempo de viaje entre sus vértices extremos.



  - Una matriz origen-destino *D* que representa la demanda entre vértices en un horizonte de tiempo dado.

$$\begin{array}{|ccc|} \hline d_{11} & \dots & d_{1n} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ d_{n1} & \dots & d_{nn} \\ \hline \end{array}$$

- Variables de decisión:
  - Conjunto de recorridos *R*,
  - con sus correspondientes frecuencias *F*.



# TNDP: Restricciones

- Cubrimiento de demanda**:**
  - Cada elemento de *D* (par de vértices) con demanda no nula debe estar conectado por al menos un recorrido de *R*; un transbordo (como máximo) es permitido.

- Frecuencias:
  - Deben pertenecer a un conjunto de valores predeterminedos.

# TNDP: Funciones objetivo

- Usuarios:

$$\min Z_1 = \Sigma_{\mathrm{i}=1..n}\Sigma_{\mathrm{j}=1..n}\, d_{ij}(tv_{ij} + tw_{ij})$$

Minimizar tiempo de viaje y de espera para cada par de vértices (*i*,*j*), ponderado por la demanda $d_{ij}$. Valores de *tv* y *tw* son determinados por el <u>sub-modelo de asignación</u>.

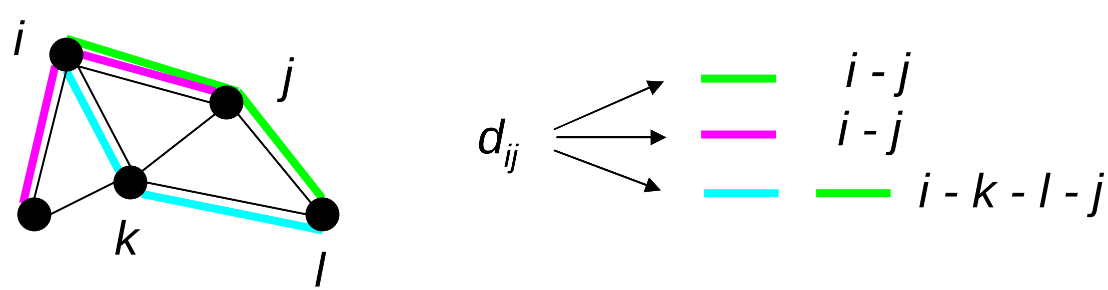
- Operadores:

$$\min Z_2 = \Sigma_{r_k \in R}\, f_k t_k$$

Minimizar tamaño de la flota, $t_k$ es la duración del recorrido $r_k$.

# TNDP: Sub-modelo de asignación

- Necesario para calcular $Z_1$.
- Aplica las hipótesis sobre el comportamiento (selección de recorridos) de la demanda *D* con respecto a los recorridos de *R*.
- Distribuye la matriz de demanda *D* sobre los recorridos de *R*.
- Usamos el modelo de Baaj y Mahmassani (1990).



# TNDP y optimización multi-objetivo

- En sistemas reales de transporte público los objetivos de usuarios y operadores son contrapuestos.
- En un sentido general, muchos recorridos con altas frecuencias redunda en bajos valores para $Z_1$ y altos para $Z_2$ y viceversa.

# TNDP y optimización multi-objetivo

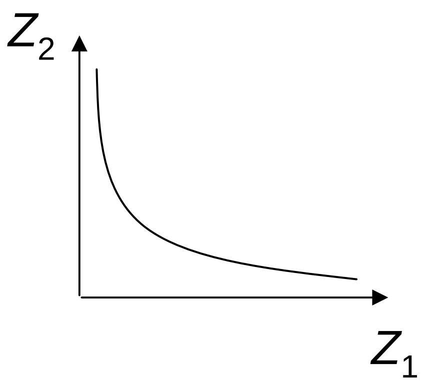
- Modo *a posteriori* adoptado por primera vez por Israeli y Ceder (1993).

- Modelo planteado por Mauttone y Urquhart (2006).

- Problema de optimización combinatoria multi-objetivo (Ehrgott y Gandibleux, 2004).

- Solución: un conjunto de soluciones no-dominadas.

$$\boxed{\begin{array}{l} \min Z_1(x) \\ \min Z_2(x) \\ \text{s.a.} \\ \quad x \in X \end{array}}$$



# TNDP: Metaheurística multi-objetivo

- Formulaciones con función objetivo única: alta complejidad combinatoria (Israeli y Ceder, 1993); resultado teórico de complejidad disponible para ciertas variantes (Börndorfer et al, 2007; Schöbel and Scholl, 2006). Enfoque multi-objetivo agrega dificultad.

- Aproximación al conjunto eficiente $X_E$.

- Metaheurística multi-objetivo GRASP TNDP propuesta en (Mauttone y Urquhart, 2006).

# TNDP: Metaheurística multi-objetivo

- GRASP TNDP:
  - Construcción: Produce un conjunto de recorridos *R* que cubre la matriz de demandas.
  - Búsqueda local: Asigna frecuencias *F* factibles y óptimas (aproximación) en los recorridos.
  - Parámetro trade off :

    Duración máxima de recorridos (construcción) y coeficientes $\alpha$ y $\beta$ en la función objetivo compuesta $Z = \alpha Z_1 + \beta Z_2$ (búsqueda local).

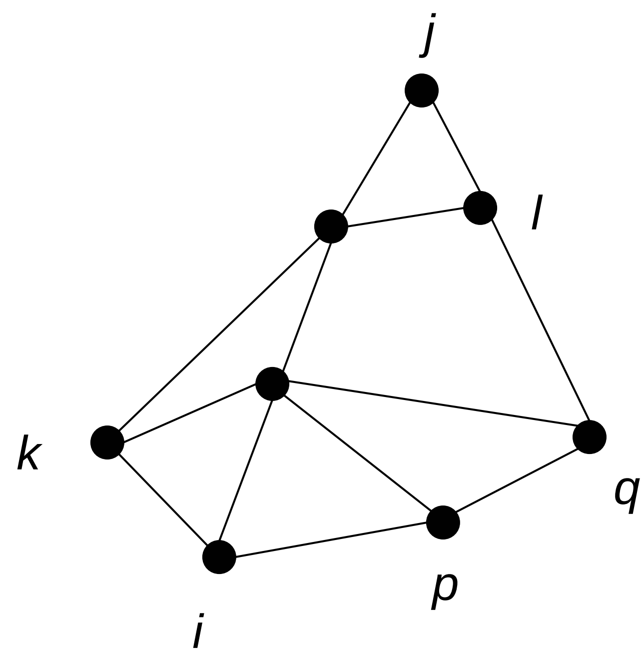
```
procedure MO_GRASP
  paretoFront = {};
  for i=1 to Iterations
    solution = Construction(i, tradeOff_i);
    solution = LocalSearch(solution, tradeOff_i);
    Update(paretoFront, solution);
  end for;
  return paretoFront;
end GRASP;
```

# TNDP: Metaheurística multi-objetivo

- Construcción:
  - Pair Insertion Algorithm (Mauttone y Urquhart, 2009).
  - Genera recorridos usando el camino más corto entre pares de vértices con alta demanda.
  - Inserta pares de vértices en recorridos existentes en la solución en construcción.

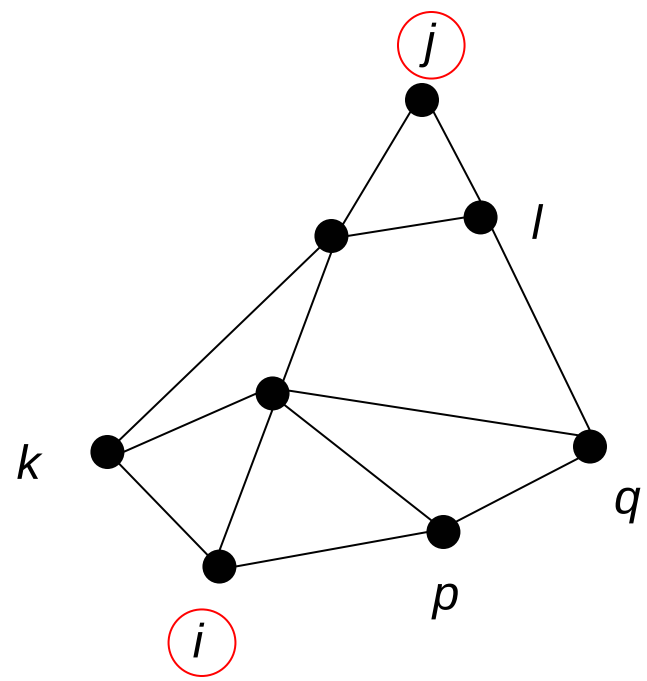
# TNDP: Metaheurística multi-objetivo

- Ejemplo:



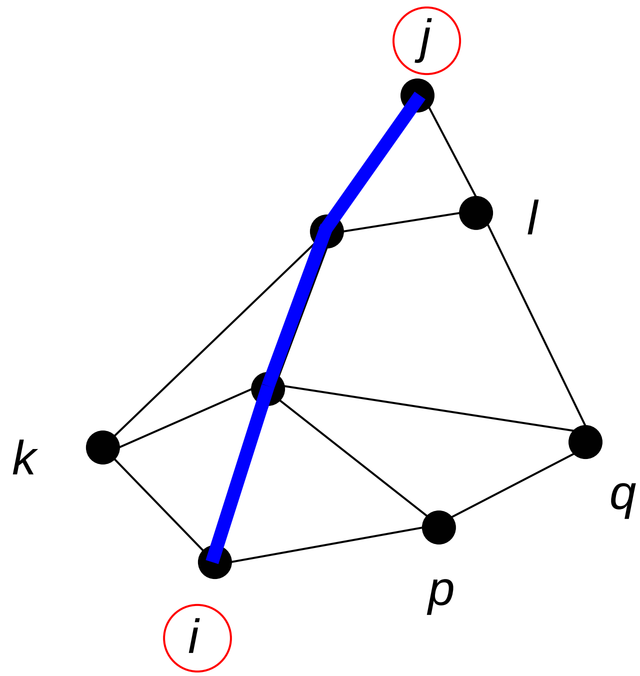
# TNDP: Metaheurística multi-objetivo

- Ejemplo:



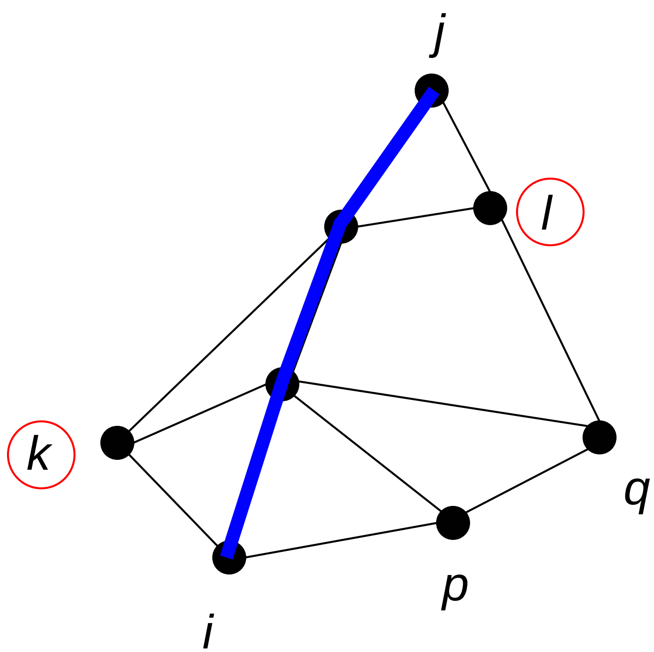
# TNDP: Metaheurística multi-objetivo

- Ejemplo:



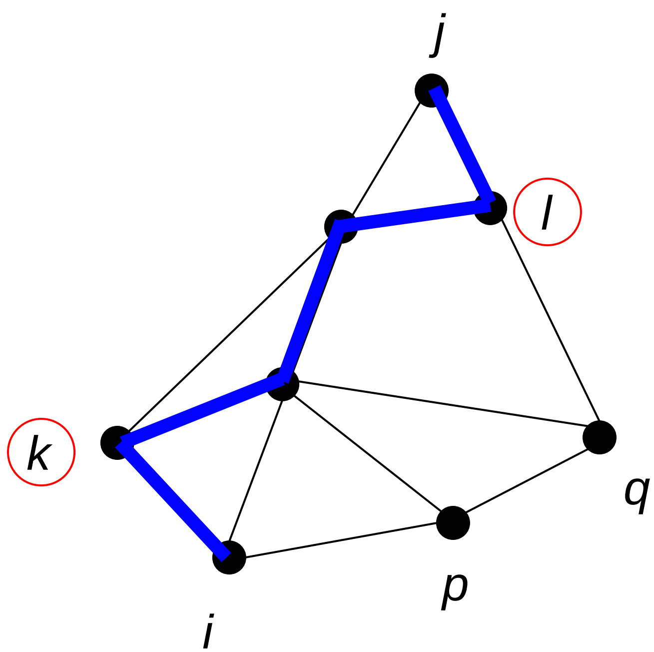
# TNDP: Metaheurística multi-objetivo

- Ejemplo:



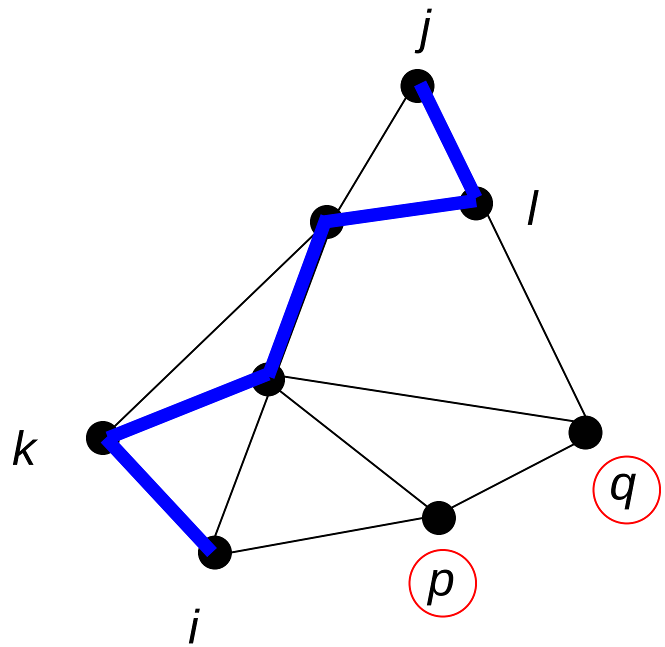
# TNDP: Metaheurística multi-objetivo

- Ejemplo:



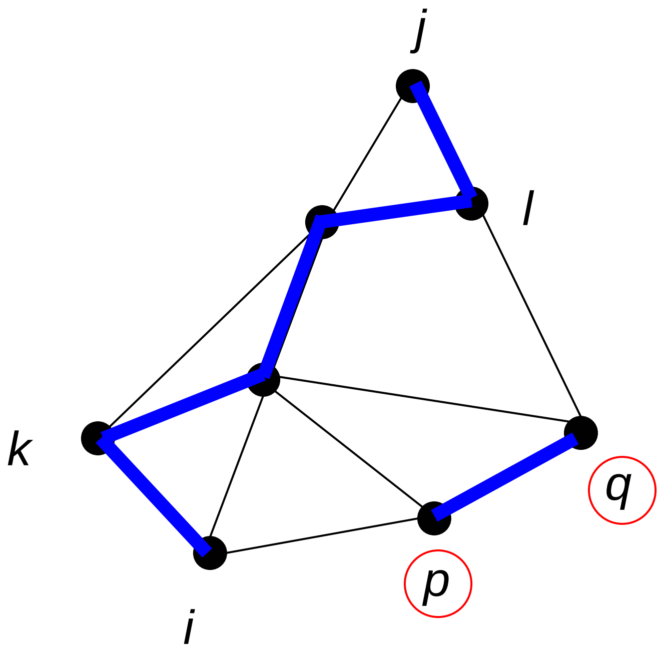
# TNDP: Metaheurística multi-objetivo

- Ejemplo:



# TNDP: Metaheurística multi-objetivo

- Ejemplo:



# TNDP: Metaheurística multi-objetivo

**procedure PIA**
  $l$ = lista de pares de vértices ($i$,$j$) con $d_{ij}$<>0;
  **while** (demanda $D$ no cubierta) **do**
    ($u$,$v$) = seleccionar elemento ($i$,$j$) con máxima demanda $d_{ij}$ en $l$;
    $r$ = Crear un recorrido con el camino más corto entre $u$ y $v$ en $G$;
    $r'$ = Crear un recorrido insertando $u$ y $v$ en las posiciones más adecuadas en el recorrido
        más conveniente $r''$ en $R$;
    **if** costo($r$) < costo($r'$) - costo($r''$) **then**
      $R = R \cup \{r\}$;
      Eliminar de $l$ los pares de vértices cuya demanda es cubierta directamente por $r$;
    **else**
      $R = R \cup \{r'\} - \{r''\}$;
      Eliminar de $l$ los pares de vértices cuya demanda es cubierta directamente por $r'$;
    **end if;**
    Actualizar demanda cubierta por recorridos de $R$;
  **end while;**
  Filtrar recorridos en $R$;
  return $R$;
**end PIA;**

# TNDP: Metaheurística multi-objetivo

- Construcción:
  - Complejidad computacional: O($n^2$) posibilidades de inserción para un par de vértices en un recorrido formado por $n$ vértices.

# TNDP: Metaheurística multi-objetivo

- Búsqueda local:
  - Varía el valor de frecuencia de cada recorrido en la solución.
  - Vecindad: Valores contiguos en el dominio de frecuencias $\Theta$. Tamaño $2|F|$.
  - Avanza en la dirección dada por la función objetivo compuesta $Z = \alpha Z_1 + \beta Z_2$.

# TNDP: Metaheurística multi-objetivo

- Ejemplo:

$$\Theta = \{1/60, 1/40, 1/30, 1/20, 1/10, 1/5\}$$

$$F^i = \{1/30, 1/40, 1/5\}$$

# TNDP: Metaheurística multi-objetivo

- Ejemplo:

$$\Theta = \{1/60,\ 1/40,\ 1/30,\ 1/20,\ 1/10,\ 1/5\}$$

$$F^i = \{1/30,\ 1/40,\ 1/5\}$$

# TNDP: Metaheurística multi-objetivo

- Ejemplo:

$$\Theta = \{1/60,\ 1/40,\ 1/30,\ 1/20,\ 1/10,\ 1/5\}$$

$$F^i = \{1/30,\ 1/40,\ 1/5\}$$

$$\Rightarrow$$

$$F^{i+1} = \{1/20,\ 1/40,\ 1/5\}$$

# TNDP: Metaheurística multi-objetivo

- Evaluación de las funciones objetivo:
  - Sub-modelo de asignación de Baaj y Mahmassani (1990).
  - Necesario para evaluar cada cada solución en la búsqueda local.
  - Asigna la matriz de demandas *D* a los recorridos *R*, teniendo en cuenta las frecuencias *F*, asumiendo que los pasajeros tratan de:
    1. Minimizar transbordos.
    2. Minimizar tiempo de viaje entre recorridos posibles.
    3. Toman el primer bus de cualquier línea que les sirve, según 1 y 2.

# TNDP: Metaheurística multi-objetivo

- Sub-modelo de asignación:

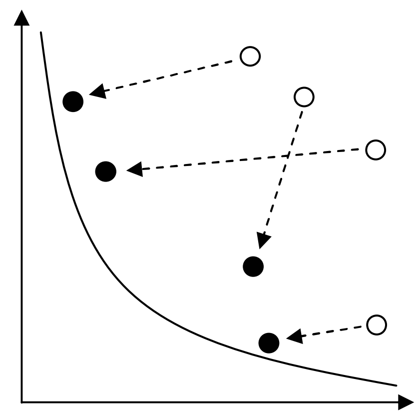
Calcula tiempo de viaje y espera para $d_{ij}$ como:

$$tv_{ij}=\Sigma_{k\in R_{ij}} w_k t_{ij}^k \qquad\qquad tw_{ij}=1/(2\Sigma_{k\in R_{ij}} f_k)$$

donde $w_k=f_k/(2\Sigma_{l\in R_{ij}} f_l)$, $R_{ij}$ es el conjunto de recorridos que conectan la demanda $d_{ij}$ y $t_{ij}^k$ es el tiempo de $i$ a $j$ usando el recorrido $r_k$.

# TNDP: Metaheurística multi-objetivo

```
procedure GRASP TNDP
  Calcular caminos más cortos entre todos los
          pares de vértices en G;
  P = {};
  for i = 1 to NumIters do
    t_max = Valor aleatorio uniforme en [t_max^ini, t_max^end];
    R = Construcción(i, α, t_max);
    F = Frecuencias iniciales;
    S = (R,F);
    λ = Vector aleatorio de pesos;
    BúsquedaLocal(λ, S, P);
  end for;
  return P;
end GRASP TNDP;
```



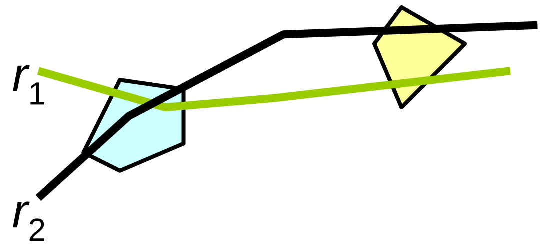
# Aplicación

# Caso de estudio

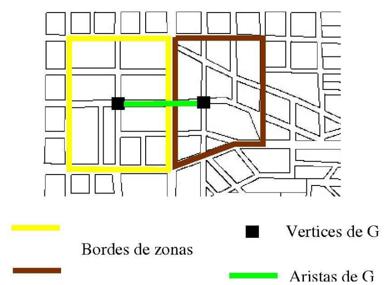
- Ciudad de Rivera, Uruguay, 65.000 habitantes aprox.
- 13 líneas de ómnibus.
- Tiempos entre pasadas: 20, 30, 40 y 60 minutos, depende de la línea.
- Fuerte presencia del transporte público en la ciudad, debido a: tradición, geografía, crecimiento.
- Publicado en Mauttone y Urquhart (2007).

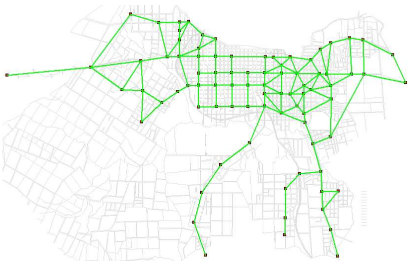
# Construcción del caso

- Zonificación: 4 x 4 cuadras

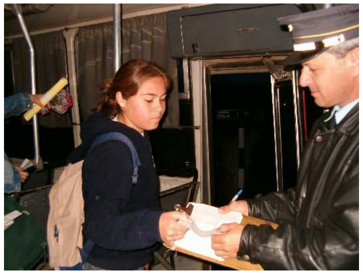



- Grafo: 84 vértices y 143 aristas



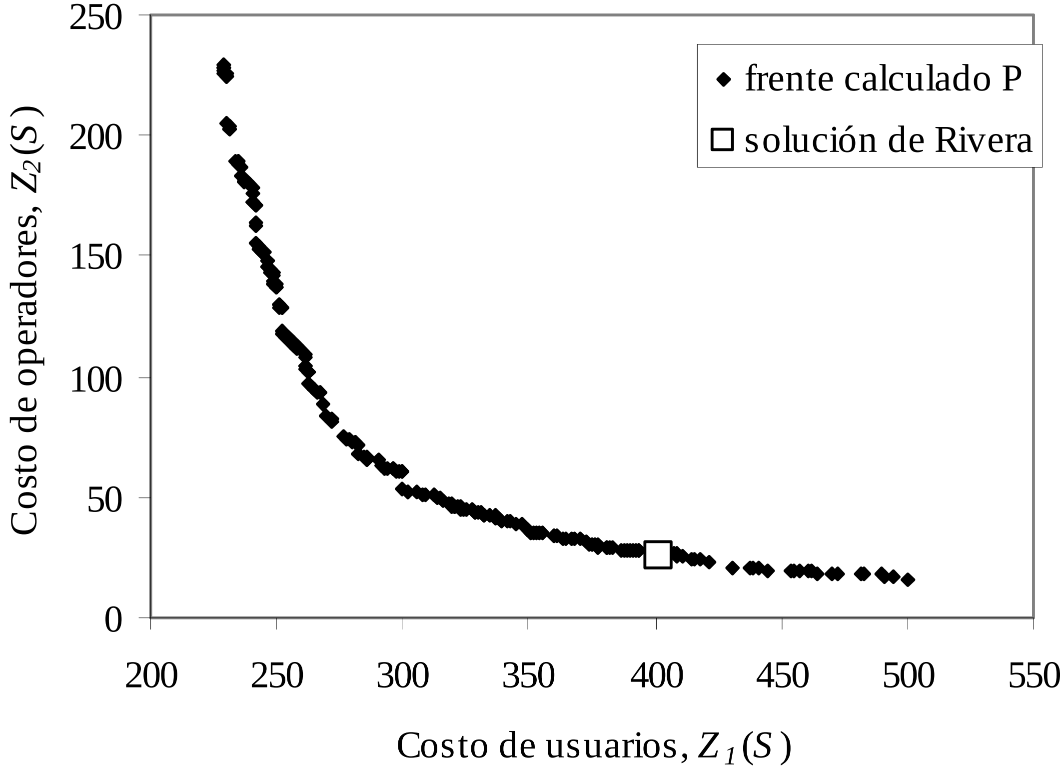



- Demanda: conteos origen-destino



# Resultados numéricos

238 soluciones no dominadas en 3 horas y 16 minutos (Pentium 4).



# Resultados numéricos

| *Solución* | $Z_1$ | $Z_2$ | $\|R\|$ | $1/f$ | $t$ |
|---|---|---|---|---|---|
| 1 | 229,34 | 226,22 | 44 | 13 | 45 |
| 3 | 239,69 | 179,53 | 46 | 16 | 43 |
| 5 | 247,09 | 144,99 | 43 | 17 | 45 |
| 7 | 252,43 | 117,44 | 26 | 16 | 53 |
| 9 | 262,88 | 96,69 | 18 | 16 | 62 |
| 11 | 292,46 | 62,39 | 28 | 30 | 52 |
| 13 | 321,60 | 45,98 | 16 | 28 | 69 |
| 15 | 349,58 | 36,64 | 11 | 33 | 90 |
| 17 | 381,58 | 28,94 | 16 | 41 | 65 |
| 19 | 437,81 | 20,54 | 10 | 47 | 90 |
| **Rivera** | **401,56** | **25,65** | **13** | **37** | **63** |

# Discusión de resultados

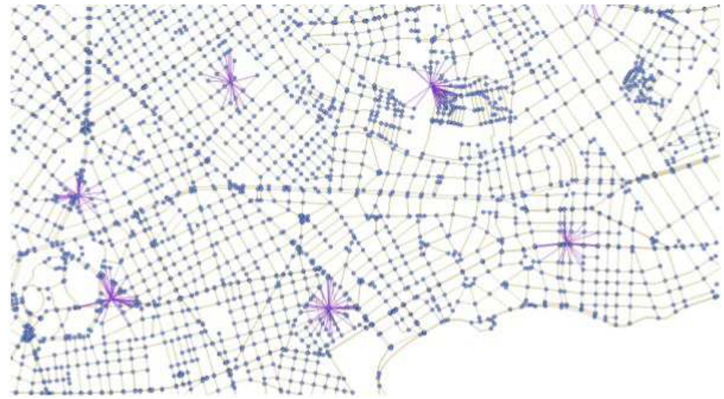
Cercanía de la solución de Rivera al frente:

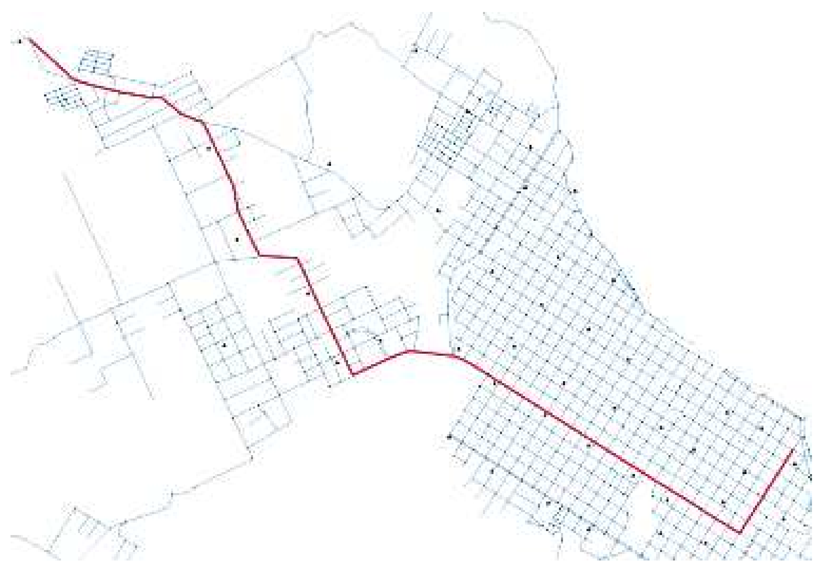
- Algoritmo aproximado, pueden existir mejores soluciones, que no fueron encontradas.
- Si el algoritmo encontró soluciones muy buenas:
    - Solución histórica de Rivera es “buena”, ajustes permanentes ralizados con conocimiento local.
    - Matriz OD utilizada se estimó en base a las líneas de la solución de Rivera. La demanda relevada está fuertemente adaptada a la oferta de la solución, por lo que es esperable que su evaluación en el modelo sea buena.

# Desarrollos recientes

Consideración de giros y casos más grandes

(Mauttone y Riganti, 2018).





# Referencias bibliográficas

Baaj, M.; Mahmassani, H. TRUST: A LISP program for the analysis of transit route configurations. Transportation Research Record 1283:125–135, 1990.

Baaj, M; Mahmassani, H. An AI-based approach for transit route system planning and design. Journal of Advanced Transportation 25(2):187–210, 1991.

Baldoquín, G. Approximate solution of an extended 0/1 knapsack problem using GRASP. En: XI Congreso Latino-Iberoamericano de Investigación de Operaciones, Concepción, Chile, 2002.

Borndörfer, R.; Grötschel, M.; Pfetsch, M. A column-generation approach to line planning in public transport. Transportation Science, 41(1):123-132, 2007.

Coello, C. An updated survey of GA-based multiobjective optimization techniques. ACM Computing Surveys 32(2):109-143, 2000.

Deb, K. Multi-Objective Optimization using Evolutionary Algorithms. Wiley, 2001.

Ehrgott, M.; Gandibleux, X. Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys (International Series in Operations Research & Management Science). Springer, 2002.

Ehrgott, M.; Gandibleux, X. Approximative solution methods for multiobjective combinatorial optimization. Top: Revista de la Asociación Española de Estadística e Investigación Operativa 12(1):1–89, 2004.

Ehrgott, M. Multicriteria Optimization. Springer, 2005.

Feo, T.; Resende, M. Greedy Randomized Adaptive Search Procedures. Journal of Global Optimization 6:109-133, 1995.

Israeli, Y.; Ceder, A. Transit route design using scheduling and multiobjective programming techniques. En: Daduna, J.; Branco, I.; Pinto, J. (eds). Proceedings of the Sixth International Workshop on Computer Aided Scheduling of Public Transport, Springer, 1993.

# Referencias bibliográficas


Mauttone, A.; Riganti, P. Modelling Turns in Transit Network Design. En: Conference on Advanced Systems in Public Transport, Brisbane, Australia, 2018.

Mauttone, A.; Urquhart, M. A Multi-Objective Metaheuristic approach for the Transit Network Design Problem. En: 10th International Conference on Computer Aided Scheduling of Public Transport, Leeds, United Kingdom, 2006.

Mauttone, A.; Urquhart, M. Optimización multi-objetivo de recorridos y frecuencias en transporte público aplicado a un caso de estudio real. En: XIII Congreso Chileno de Ingeniería de Transporte, Santiago, Chile, 2007.

Mauttone, A.; Urquhart, M. A route set construction algorithm for the Transit Network Design Problem. Computers and Operations Research, 36(8):2440–2449, 2009.

Resende, M.; Ribeiro, C. Greedy Randomized Adaptive Search Procedures. In M. Gendreau and J.M. Potvin, editors, Handbook of Metaheuristics, International Series in Operations Research & Management Science, pp 283-319, 2010.

Sayin, S.; Kouvelis, P. The Multiobjective Discrete Optimization Problem: A Weighted Min-Max Two-Stage Optimization Approach and a Bicriteria Algorithm. Management Science 51(10):1572–1581, 2005.

Schöbel, A.; Scholl, S. Line planning with minimal traveling time. In L.G. Kroon and R.H. Möhring, editors, 5th Workshop on Algorithmic Methods and Models for Optimization of Railways, 2005.

Soares, D.; Arroyo, J. A GRASP algorithm for the multi-objective knapsack problem. En: XXIV International Conference of the Chilean Computer Science Society, Arica, Chile, 2004.