Optimización de viajes compartidos en taxis utilizando algoritmos evolutivos

Gabriel Fagúndez de los Reyes   Renzo Massobrio

Facultad de Ingeniería,
Universidad de la República,
Montevideo, Uruguay

Contenido

Motivación

Car pooling

- Beneficios en el plano ecológico y económico, individuales y colectivos.
- Iniciativas para atender el interés del público: carriles exclusivos, campañas para compartir los viajes al trabajo y aplicaciones para encontrar compañeros de viaje.

Taxi pooling

- Los taxis son un medio de transporte rápido y confiable, especialmente en ciudades donde el transporte público es poco eficiente.
- Los taxis raramente viajan a capacidad completa, impactando en la congestión del tráfico y en la contaminación de las ciudades.
- Tarifas altas desalientan a los usuarios.
- 15 % de los accidentes fatales en Uruguay involucran a un conductor alcoholizado (UNASEV).

Motivación

Car pooling

- Beneficios en el plano ecológico y económico, individuales y colectivos.
- Iniciativas para atender el interés del público: carriles exclusivos, campañas para compartir los viajes al trabajo y aplicaciones para encontrar compañeros de viaje.

Taxi pooling

- Los taxis son un medio de transporte rápido y confiable, especialmente en ciudades donde el transporte público es poco eficiente.
- Los taxis raramente viajan a capacidad completa, impactando en la congestión del tráfico y en la contaminación de las ciudades.
- Tarifas altas desalientan a los usuarios.
- 15 % de los accidentes fatales en Uruguay involucran a un conductor alcoholizado (UNASEV).

Descripción del problema

Problema de viajes compartidos en taxis (PVCT)

Un grupo de personas ubicadas en un **mismo lugar de origen**, desean viajar hacia **diferentes destinos** utilizando taxis de forma compartida.
Se busca determinar la cantidad de taxis, la asignación de pasajeros y las rutas a seguir, de forma de minimizar el costo total del grupo de pasajeros.

Consideraciones

- Cada taxi puede trasladar a un número limitado de pasajeros.
- el número máximo de taxis para N pasajeros es N.
- Costo de un taxi = **costo inicial** ("bajada de bandera") + **costo determinado por la distancia**.
- No se consideran otros posibles costos (e.g. esperas, propinas, peajes).

Descripción del problema

Problema de viajes compartidos en taxis (PVCT)

Un grupo de personas ubicadas en un **mismo lugar de origen**, desean viajar hacia **diferentes destinos** utilizando taxis de forma compartida. Se busca determinar la cantidad de taxis, la asignación de pasajeros y las rutas a seguir, de forma de minimizar el costo total del grupo de pasajeros.

Consideraciones

- Cada taxi puede trasladar a un número limitado de pasajeros.
- el número máximo de taxis para N pasajeros es N.
- Costo de un taxi = **costo inicial** ("bajada de bandera") + **costo determinado por la distancia**.
- No se consideran otros posibles costos (e.g. esperas, propinas, peajes).

Descripción del problema

Problema de viajes compartidos en taxis (PVCT)

Un grupo de personas ubicadas en un **mismo lugar de origen**, desean viajar hacia **diferentes destinos** utilizando taxis de forma compartida. Se busca determinar la cantidad de taxis, la asignación de pasajeros y las rutas a seguir, de forma de minimizar el costo total del grupo de pasajeros.

Consideraciones

- Cada taxi puede trasladar a un número limitado de pasajeros.
- el número máximo de taxis para N pasajeros es N.
- Costo de un taxi = **costo inicial** ("bajada de bandera") + **costo determinado por la distancia**.
- No se consideran otros posibles costos (e.g. esperas, propinas, peajes).

Descripción del problema

Problema de viajes compartidos en taxis (PVCT)

Un grupo de personas ubicadas en un **mismo lugar de origen**, desean viajar hacia **diferentes destinos** utilizando taxis de forma compartida.

Se busca determinar la cantidad de taxis, la asignación de pasajeros y las rutas a seguir, de forma de minimizar el costo total del grupo de pasajeros.

Consideraciones

- Cada taxi puede trasladar a un número limitado de pasajeros.
- el número máximo de taxis para N pasajeros es N.
- Costo de un taxi = **costo inicial** ("bajada de bandera") + **costo determinado por la distancia**.
- No se consideran otros posibles costos (e.g. esperas, propinas, peajes).

Descripción del problema

Problema de viajes compartidos en taxis (PVCT)

Un grupo de personas ubicadas en un **mismo lugar de origen**, desean viajar hacia **diferentes destinos** utilizando taxis de forma compartida. Se busca determinar la cantidad de taxis, la asignación de pasajeros y las rutas a seguir, de forma de minimizar el costo total del grupo de pasajeros.

Consideraciones

- Cada taxi puede trasladar a un número limitado de pasajeros.
- el número máximo de taxis para N pasajeros es N.
- Costo de un taxi = **costo inicial** ("bajada de bandera") + **costo determinado por la distancia**.
- No se consideran otros posibles costos (e.g. esperas, propinas, peajes).

Formulación del problema

- un conjunto de pasajeros $P = \{p_1, p_2, \ldots, p_N\}$ que viajan desde un origen común O a un conjunto de destinos $D = \{d_1, d_2, \ldots, d_N\}$.
- un conjunto de taxis $T = \{t_1, t_2, \ldots, t_M\}$; con $M \leq N$; y una función $C : T \rightarrow \{0, 1, \ldots, C_{MAX}\}$ que indica la cantidad de pasajeros en un taxi. C_{MAX} es la capacidad máxima permitida en un mismo taxi.
- una constante B indica el costo inicial del taxi ("bajada de bandera").
- una función de distancia, $dist : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$.
- una función de costo asociado a la distancia recorrida por cada taxi, $cost : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

Se desea hallar la planificación $f : P \rightarrow T \times \{1, \ldots, C_{MAX}\}$ que minimice la función de costo total (CT).

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \underbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right)\right)}_{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

Formulación del problema

- un conjunto de pasajeros $P = \{p_1, p_2, \ldots, p_N\}$ que viajan desde un origen común O a un conjunto de destinos $D = \{d_1, d_2, \ldots, d_N\}$.
- un conjunto de taxis $T = \{t_1, t_2, \ldots, t_M\}$; con $M \leq N$; y una función $C : T \rightarrow \{0, 1, \ldots, C_{MAX}\}$ que indica la cantidad de pasajeros en un taxi. C_{MAX} es la capacidad máxima permitida en un mismo taxi.
- una constante B indica el costo inicial del taxi ("bajada de bandera").
- una función de distancia, $dist : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$.
- una función de costo asociado a la distancia recorrida por cada taxi, $cost : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

Se desea hallar la planificación $f : P \rightarrow T \times \{1, \ldots, C_{MAX}\}$ que minimice la función de costo total (CT).

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \underbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right) \right)}_{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

Formulación del problema

- un conjunto de pasajeros $P = \{p_1, p_2, \ldots, p_N\}$ que viajan desde un origen común O a un conjunto de destinos $D = \{d_1, d_2, \ldots, d_N\}$.
- un conjunto de taxis $T = \{t_1, t_2, \ldots, t_M\}$; con $M \leq N$; y una función $C : T \rightarrow \{0, 1, \ldots, C_{MAX}\}$ que indica la cantidad de pasajeros en un taxi. C_{MAX} es la capacidad máxima permitida en un mismo taxi.
- una constante B indica el costo inicial del taxi ("bajada de bandera").
- una función de distancia, $dist : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$.
- una función de costo asociado a la distancia recorrida por cada taxi, $cost : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

Se desea hallar la planificación $f : P \rightarrow T \times \{1, \ldots, C_{MAX}\}$ que minimice la función de costo total (CT).

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist\, \underbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right)\right)}_{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

Formulación del problema

- un conjunto de pasajeros $P = \{p_1, p_2, \ldots, p_N\}$ que viajan desde un origen común O a un conjunto de destinos $D = \{d_1, d_2, \ldots, d_N\}$.
- un conjunto de taxis $T = \{t_1, t_2, \ldots, t_M\}$; con $M \leq N$; y una función $C : T \rightarrow \{0, 1, \ldots, C_{MAX}\}$ que indica la cantidad de pasajeros en un taxi. C_{MAX} es la capacidad máxima permitida en un mismo taxi.
- una constante B indica el costo inicial del taxi ("bajada de bandera").
- una función de distancia, $dist : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$.
- una función de costo asociado a la distancia recorrida por cada taxi, $cost : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

Se desea hallar la planificación $f : P \rightarrow T \times \{1, \ldots, C_{MAX}\}$ que minimice la función de costo total (CT).

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \underbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right)\right)}_{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

Formulación del problema

- un conjunto de pasajeros $P = \{p_1, p_2, \ldots, p_N\}$ que viajan desde un origen común O a un conjunto de destinos $D = \{d_1, d_2, \ldots, d_N\}$.
- un conjunto de taxis $T = \{t_1, t_2, \ldots, t_M\}$; con $M \leq N$; y una función $C : T \rightarrow \{0, 1, \ldots, C_{MAX}\}$ que indica la cantidad de pasajeros en un taxi. C_{MAX} es la capacidad máxima permitida en un mismo taxi.
- una constante B indica el costo inicial del taxi ("bajada de bandera").
- una función de distancia, $dist : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$.
- una función de costo asociado a la distancia recorrida por cada taxi, $cost : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

Se desea hallar la planificación $f : P \rightarrow T \times \{1, \ldots, C_{MAX}\}$ que minimice la función de costo total (CT).

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\Big(dist \underbrace{\Big(dest\big(f^{-1}(t_i, j-1)\big), dest\big(f^{-1}(t_i, j)\big)\Big)}_{\text{destinos consecutivos en el recorrido del taxi } t_i} \Big) \right]$$

Formulación del problema

- un conjunto de pasajeros $P = \{p_1, p_2, \ldots, p_N\}$ que viajan desde un origen común O a un conjunto de destinos $D = \{d_1, d_2, \ldots, d_N\}$.
- un conjunto de taxis $T = \{t_1, t_2, \ldots, t_M\}$; con $M \leq N$; y una función $C : T \rightarrow \{0, 1, \ldots, C_{MAX}\}$ que indica la cantidad de pasajeros en un taxi. C_{MAX} es la capacidad máxima permitida en un mismo taxi.
- una constante B indica el costo inicial del taxi ("bajada de bandera").
- una función de distancia, $dist : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$.
- una función de costo asociado a la distancia recorrida por cada taxi, $cost : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$.

Se desea hallar la planificación $f : P \rightarrow T \times \{1, \ldots, C_{MAX}\}$ que minimice la función de costo total (CT).

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \underbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right)\right)}_{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

Variante multiobjetivo del PVCT: formulación matemática

Se busca minimizar simultáneamente el costo total y la demora total.

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \overbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right)\right)}^{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

$$DT = \sum_{t_i} \left[\sum_{j=1}^{C(t_i)} \left[\overbrace{\sum_{h=1}^{j} time\left(dest\left(f^{-1}(t_i, h-1)\right), dest\left(f^{-1}(t_i, h)\right)\right)}^{\text{tiempo efectivo de traslado del pasajero en la posición } j \text{ del taxi } t_i} - \underbrace{tol\left(f^{-1}(t_i, j)\right) + time\left(O, dest\left(f^{-1}(t_i, j)\right)\right)}_{\text{tiempo tolerado por el pasajero en la posición } j \text{ del taxi } t_i} \right] \right]$$

- $time : \{\{O\} \cup D\} \times D \to \mathbb{R}_0^+$ indica el tiempo de recorrido.
- $tol : P \to \mathbb{R}_0^+$ indica el tiempo adicional tolerado por cada pasajero.

Variante multiobjetivo del PVCT: formulación matemática

Se busca minimizar simultáneamente el costo total y la demora total.

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \overbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right)\right)}^{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

$$DT = \sum_{t_i} \left[\sum_{j=1}^{C(t_i)} \left[\overbrace{\sum_{h=1}^{j} time\left(dest\left(f^{-1}(t_i, h-1)\right), dest\left(f^{-1}(t_i, h)\right)\right)}^{\text{tiempo efectivo de traslado del pasajero en la posición } j \text{ del taxi } t_i} \right.\right.$$
$$\left.\left. - \underbrace{tol\left(f^{-1}(t_i, j)\right) + time\left(O, dest\left(f^{-1}(t_i, j)\right)\right)}_{\text{tiempo tolerado por el pasajero en la posición } j \text{ del taxi } t_i} \right] \right]$$

- $time : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$ indica el tiempo de recorrido.
- $tol : P \rightarrow \mathbb{R}_0^+$ indica el tiempo adicional tolerado por cada pasajero.

Variante multiobjetivo del PVCT: formulación matemática

Se busca minimizar simultáneamente el costo total y la demora total.

$$CT = \sum_{t_i, C(t_i) \neq 0} \left[B + \sum_{j=1}^{C(t_i)} cost\left(dist \overbrace{\left(dest\left(f^{-1}(t_i, j-1)\right), dest\left(f^{-1}(t_i, j)\right) \right)}^{\text{destinos consecutivos en el recorrido del taxi } t_i} \right) \right]$$

$$DT = \sum_{t_i} \left[\sum_{j=1}^{C(t_i)} \left[\overbrace{\sum_{h=1}^{j} time\left(dest\left(f^{-1}(t_i, h-1)\right), dest\left(f^{-1}(t_i, h)\right) \right)}^{\text{tiempo efectivo de traslado del pasajero en la posición } j \text{ del taxi } t_i} - \underbrace{tol\left(f^{-1}(t_i, j)\right) + time\left(O, dest\left(f^{-1}(t_i, j)\right) \right)}_{\text{tiempo tolerado por el pasajero en la posición } j \text{ del taxi } t_i} \right] \right]$$

- $time : \{\{O\} \cup D\} \times D \rightarrow \mathbb{R}_0^+$ indica el tiempo de recorrido.
- $tol : P \rightarrow \mathbb{R}_0^+$ indica el tiempo adicional tolerado por cada pasajero.

Complejidad del PVCT

Complejidad

Baldacci et al. (2004) estudiaron una variante del *Car Pooling Problem (CPP)* donde trabajadores desean compartir vehículos hacia y desde el lugar de trabajo.
Esta variante es un caso particular del *Vehicle Routing Problem (VRP)* con demanda unitaria, el cual es $\mathcal{NP}$–difícil [Letcheford et al. (2002)].

Estrategias de resolución

Cuando se utilizan instancias de tamaños realistas, los algoritmos exactos tradicionales no resultan útiles para una planificación eficiente.
Heurísticas y metaheurísticas permiten calcular soluciones de calidad aceptable en tiempos razonables.

Complejidad del PVCT

Complejidad

Baldacci et al. (2004) estudiaron una variante del *Car Pooling Problem (CPP)* donde trabajadores desean compartir vehículos hacia y desde el lugar de trabajo.
Esta variante es un caso particular del *Vehicle Routing Problem (VRP)* con demanda unitaria, el cual es $\mathcal{NP}$–difícil [Letcheford et al. (2002)].

Estrategias de resolución

Cuando se utilizan instancias de tamaños realistas, los algoritmos exactos tradicionales no resultan útiles para una planificación eficiente.
Heurísticas y metaheurísticas permiten calcular soluciones de calidad aceptable en tiempos razonables.

Trabajo relacionado

Car pooling problem (CPP)

Yan et al. (2011) **CPP con histórico de viajes** (relajación lagrangeana).

Dial–a–ride problem (DARP)

Cordeau et al. (2003) **DARP estático con ventanas de tiempo**.

Búsqueda tabú con tiempos de ejecución de hasta 90 minutos.

Taxi pooling problem (TPP)

Tao et al. (2007) Heurísticas ávidas para **one–to–many** y many–to–one.

Las mejoras se reportan en términos absolutos.

Ma et al. (2013) TPP dinámico con pedidos en tiempo real.

13 % de ahorro en distancia con un algoritmo ávido en **instancias realistas**.

Resumen

Pocos trabajos centrados en el usuario y con un enfoque multiobjetivo.

Trabajo relacionado

Car pooling problem (CPP)

Yan et al. (2011) **CPP con histórico de viajes** (relajación lagrangeana).

Dial–a–ride problem (DARP)

Cordeau et al. (2003) **DARP estático con ventanas de tiempo**.

Búsqueda tabú con tiempos de ejecución de hasta 90 minutos.

Taxi pooling problem (TPP)

Tao et al. (2007) Heurísticas ávidas para **one–to–many** y many–to–one.

Las mejoras se reportan en términos absolutos.

Ma et al. (2013) TPP dinámico con pedidos en tiempo real.

13 % de ahorro en distancia con un algoritmo ávido en **instancias realistas**.

Resumen

Pocos trabajos centrados en el usuario y con un enfoque multiobjetivo.

Trabajo relacionado

Car pooling problem (CPP)

Yan et al. (2011) **CPP con histórico de viajes** (relajación lagrangeana).

Dial–a–ride problem (DARP)

Cordeau et al. (2003) **DARP estático con ventanas de tiempo**.

Búsqueda tabú con tiempos de ejecución de hasta 90 minutos.

Taxi pooling problem (TPP)

Tao et al. (2007) Heurísticas ávidas para **one–to–many** y many–to–one.

Las mejoras se reportan en términos absolutos.

Ma et al. (2013) TPP dinámico con pedidos en tiempo real.

13 % de ahorro en distancia con un algoritmo ávido en **instancias realistas**.

Resumen

Pocos trabajos centrados en el usuario y con un enfoque multiobjetivo.

Trabajo relacionado

Car pooling problem (CPP)

Yan et al. (2011) **CPP con histórico de viajes** (relajación lagrangeana).

Dial–a–ride problem (DARP)

Cordeau et al. (2003) **DARP estático con ventanas de tiempo**.

Búsqueda tabú con tiempos de ejecución de hasta 90 minutos.

Taxi pooling problem (TPP)

Tao et al. (2007) Heurísticas ávidas para **one–to–many** y many–to–one.

Las mejoras se reportan en términos absolutos.

Ma et al. (2013) TPP dinámico con pedidos en tiempo real.

13 % de ahorro en distancia con un algoritmo ávido en **instancias realistas**.

Resumen

Pocos trabajos centrados en el usuario y con un enfoque multiobjetivo.

Trabajo relacionado

Car pooling problem (CPP)

Yan et al. (2011) **CPP con histórico de viajes** (relajación lagrangeana).

Dial–a–ride problem (DARP)

Cordeau et al. (2003) **DARP estático con ventanas de tiempo**.

Búsqueda tabú con tiempos de ejecución de hasta 90 minutos.

Taxi pooling problem (TPP)

Tao et al. (2007) Heurísticas ávidas para **one–to–many** y many–to–one.

Las mejoras se reportan en términos absolutos.

Ma et al. (2013) TPP dinámico con pedidos en tiempo real.

13 % de ahorro en distancia con un algoritmo ávido en **instancias realistas**.

Resumen

Pocos trabajos centrados en el usuario y con un enfoque multiobjetivo.

Algoritmos evolutivos

Definición

- Los *algoritmos evolutivos* (AE) son técnicas estocásticas que emulan el proceso de evolución natural de las especies para resolver problemas de optimización, búsqueda y aprendizaje.
- Un AE es una técnica iterativa (cada iteración se denomina **generación**) que aplica operadores estocásticos sobre un conjunto de individuos (la **población**).
- Cada individuo en la población codifica una solución tentativa al problema y tiene un valor de **fitness**, dado por una función de evaluación que determina su adecuación para resolver el problema.
- El propósito del AE es mejorar el fitness de los individuos en la población mediante la aplicación iterativa de **operadores evolutivos** a individuos seleccionados según su fitness, guiando al AE hacia soluciones tentativas de mayor calidad.

AE para el PVCT monoobjetivo

Aspectos comunes

- Tuplas de largo $2N - 1$
  $N = \#$pasajeros.
- Inicialización:
  aleatoria y ávida.
- Cruzamiento basado en posición (PBX).
- Mutación por intercambio (EM).
- Función correctiva:
  desplaza ceros para romper secuencias de dígitos inválidas.
- Implementados en Malva.

AE para el PVCT monoobjetivo

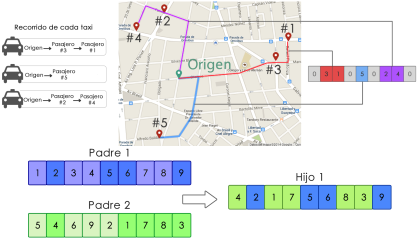
Aspectos comunes

- Tuplas de largo $2N - 1$
  $N = \#$pasajeros.
- Inicialización:
  aleatoria y ávida.
- Cruzamiento basado en posición (PBX).
- Mutación por intercambio (EM).
- Función correctiva:
  desplaza ceros para romper secuencias de dígitos inválidas.
- Implementados en Malva.

AE para el PVCT monoobjetivo

Aspectos comunes

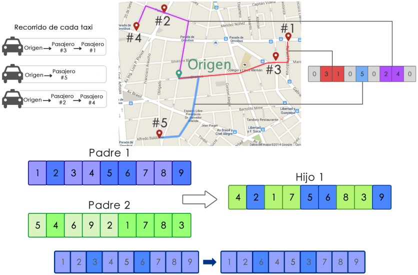
- Tuplas de largo $2N - 1$
  $N = \#$pasajeros.
- Inicialización:
  aleatoria y ávida.
- Cruzamiento basado en posición (PBX).
- Mutación por intercambio (EM).
- Función correctiva:
  desplaza ceros para romper secuencias de dígitos inválidas.
- Implementados en Malva.



AE para el PVCT monoobjetivo

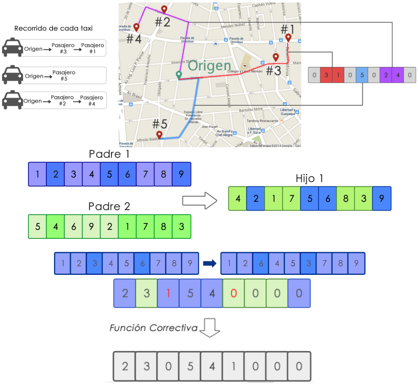
Aspectos comunes

- Tuplas de largo $2N - 1$
  $N = \#$pasajeros.
- Inicialización:
  aleatoria y ávida.
- Cruzamiento basado en posición (PBX).
- Mutación por intercambio (EM).
- Función correctiva:
  desplaza ceros para romper secuencias de dígitos inválidas.
- Implementados en Malva.



Aspectos comunes

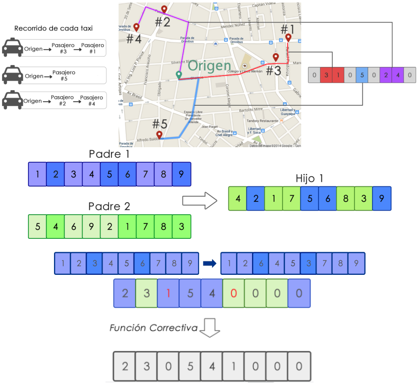
- Tuplas de largo $2N - 1$
  $N = \#$pasajeros.
- Inicialización:
  aleatoria y ávida.
- Cruzamiento basado en posición (PBX).
- Mutación por intercambio (EM).
- Función correctiva:
  desplaza ceros para romper secuencias de dígitos inválidas.
- Implementados en Malva.



AE para el PVCT monoobjetivo

Aspectos comunes

- Tuplas de largo $2N - 1$
  $N = \#$pasajeros.
- Inicialización:
  aleatoria y ávida.
- Cruzamiento basado en posición (PBX).
- Mutación por intercambio (EM).
- Función correctiva:
  desplaza ceros para romper secuencias de dígitos inválidas.
- Implementados en Malva.



AE para el PVCT monoobjetivo

seqEA

AE **secuencial**. Utiliza selección proporcional.

Modelos paralelos en AE

Buscan **mejorar el desempeño** de los AE.

Modelo de subpoblaciones distribuidas: divide la población en **islas** que intercambian individuos mediante **migración**.

AE paralelo con micro–población ($p\mu EA$)

- Poblaciones pequeñas.
- Selección por torneo (m, k).
- Migración asíncrona.
- Topología de anillo unidireccional.

AE para el PVCT monoobjetivo

seqEA

AE **secuencial**. Utiliza selección proporcional.

Modelos paralelos en AE

Buscan **mejorar el desempeño** de los AE.
Modelo de subpoblaciones distribuidas: divide la población en **islas** que intercambian individuos mediante **migración**.

AE paralelo con micro–población ($p\mu EA$)

- Poblaciones pequeñas.
- Selección por torneo (m, k).
- Migración asíncrona.
- Topología de anillo unidireccional.

AE para el PVCT monoobjetivo

seqEA

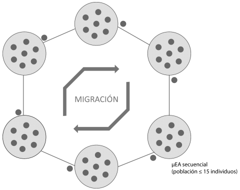
AE **secuencial**. Utiliza selección proporcional.

Modelos paralelos en AE

Buscan **mejorar el desempeño** de los AE.
Modelo de subpoblaciones distribuidas: divide la población en **islas** que intercambian individuos mediante **migración**.

AE paralelo con micro–población ($p\mu EA$)

- Poblaciones pequeñas.
- Selección por torneo (m, k).
- Migración asíncrona.
- Topología de anillo unidireccional.



AE para el PVCT multiobjetivo

Propósitos en AE multiobjetivos (MOEA)

Acercarse al frente de Pareto del problema (**convergencia**) y muestrear adecuadamente el frente de soluciones (**diversidad**).

pμMOEA/D

$F = w_C \times CT + w_D \times DT,$

$$w_C = [0 : \tfrac{1}{\#islas} : 1],\ w_D = 1 - w_C\,.$$

NSGA–II

Ordenamiento no–dominado (elitista) y *crowding* para preservar diversidad.

Aspectos comunes

Función correctiva considera vehículos de distintas capacidades.
Inicialización de la población ávida y selección por torneo.

AE para el PVCT multiobjetivo

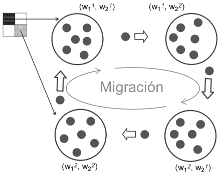
Propósitos en AE multiobjetivos (MOEA)

Acercarse al frente de Pareto del problema (**convergencia**) y muestrear adecuadamente el frente de soluciones (**diversidad**).

pµMOEA/D

$$F = w_C \times CT + w_D \times DT,$$

$$w_C = [0 : \frac{1}{\#islas} : 1],\ w_D = 1 - w_C.$$



NSGA–II

Ordenamiento no–dominado (elitista) y *crowding* para preservar diversidad.

Aspectos comunes

Función correctiva considera vehículos de distintas capacidades.

Inicialización de la población ávida y selección por torneo.

AE para el PVCT multiobjetivo

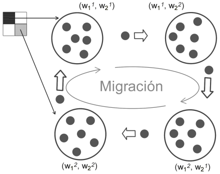
Propósitos en AE multiobjetivos (MOEA)

Acercarse al frente de Pareto del problema (**convergencia**) y muestrear adecuadamente el frente de soluciones (**diversidad**).

pμ*MOEA/D*

$$F = w_C \times CT + w_D \times DT,$$

$$w_C = [0 : \tfrac{1}{\#islas} : 1],\ w_D = 1 - w_C.$$



NSGA–II

Ordenamiento no–dominado (elitista) y *crowding* para preservar diversidad.

Aspectos comunes

Función correctiva considera vehículos de distintas capacidades.
Inicialización de la población ávida y selección por torneo.

AE para el PVCT multiobjetivo

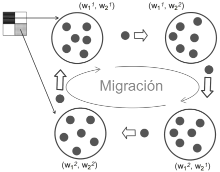
Propósitos en AE multiobjetivos (MOEA)

Acercarse al frente de Pareto del problema (**convergencia**) y muestrear adecuadamente el frente de soluciones (**diversidad**).

pμMOEA/D

$$F = w_C \times CT + w_D \times DT,$$

$$w_C = [0 : \frac{1}{\#islas} : 1],\ w_D = 1 - w_C.$$



NSGA–II

Ordenamiento no–dominado (elitista) y *crowding* para preservar diversidad.

Aspectos comunes

Función correctiva considera vehículos de distintas capacidades.
Inicialización de la población ávida y selección por torneo.

AE para el PVCT multiobjetivo

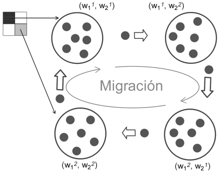
Propósitos en AE multiobjetivos (MOEA)

Acercarse al frente de Pareto del problema (**convergencia**) y muestrear adecuadamente el frente de soluciones (**diversidad**).

pμMOEA/D

$$F = w_C \times CT + w_D \times DT,$$

$$w_C = [0 : \frac{1}{\#islas} : 1],\ w_D = 1 - w_C.$$



NSGA–II

Ordenamiento no–dominado (elitista) y *crowding* para preservar diversidad.

Aspectos comunes

Función correctiva considera vehículos de distintas capacidades.
Inicialización de la población ávida y selección por torneo.

Generación de instancias

Generación de puntos realistas en el mapa

- Generador de Pedidos de Taxis (TQG) con datos de GPS de taxis de Beijing (Ma et al., 2013).
- Script para obtener instancias de un origen a muchos destinos.
- Instancias en Montevideo generadas manualmente.
- API para obtener tarifas TaxiFareFinder (TFF).

Instancias generadas

- **6 chicas:** 10 y 15 pasajeros (Beijing).
- **6 medianas:** 15 y 25 pasajeros (Beijing).
- **6 grandes:** 25 y 45 pasajeros (Beijing).
- **4 en Montevideo:** 8 y 17 pasajeros (Montevideo).

22 instancias para el PVCT monoobjetivo, y un total de 88 para el multiobjetivo, considerando distintas capacidades y tolerancias.

Generación de instancias

Generación de puntos realistas en el mapa

- Generador de Pedidos de Taxis (TQG) con datos de GPS de taxis de Beijing (Ma et al., 2013).
- Script para obtener instancias de un origen a muchos destinos.
- Instancias en Montevideo generadas manualmente.
- API para obtener tarifas TaxiFareFinder (TFF).

Instancias generadas

- **6 chicas:** 10 y 15 pasajeros (Beijing).
- **6 medianas:** 15 y 25 pasajeros (Beijing).
- **6 grandes:** 25 y 45 pasajeros (Beijing).
- **4 en Montevideo:** 8 y 17 pasajeros (Montevideo).

22 instancias para el PVCT monoobjetivo, y un total de 88 para el multiobjetivo, considerando distintas capacidades y tolerancias.

PVCT monoobjetivo

Entorno de ejecución

- La evaluación experimental fue realizada en el Cluster FING.
- **seqEA**: Dell Power Edge 2950, **1 núcleo** de Intel Xeon E5430 2.66GHz, 8GB RAM.
- **pμEA**: HP Proliant DL585, **24 núcleos** de AMD Opteron 2.09GHz, 48GB RAM.

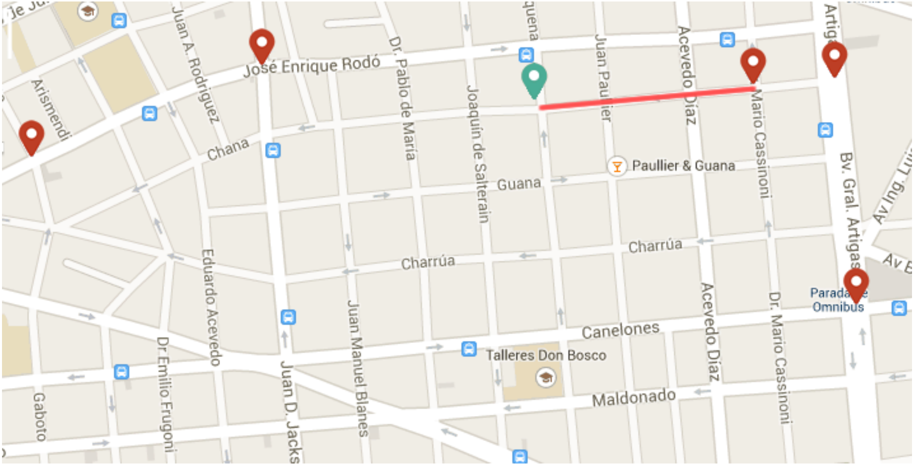
Configuración paramétrica

- **seqEA**: 20 ejecuciones de 2000 generaciones sobre 3 instancias.
  $\#P \in \{150, 200, 250\}$; $p_C \in \{0{,}6, 0{,}75, 0{,}95\}$; $p_M \in \{0{,}001, 0{,}01, 0{,}1\}$.
- **pμEA**: micro–población de 15 individuos, torneo $(m = 2,\ k = 1)$, migración cada 500 generaciones.
  20 ejecuciones de 100.000 generaciones sobre 5 instancias.
  $p_C \in \{0{,}6, 0{,}75, 0{,}95\}$; $p_M \in \{0{,}001, 0{,}01, 0{,}1\}$.

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.

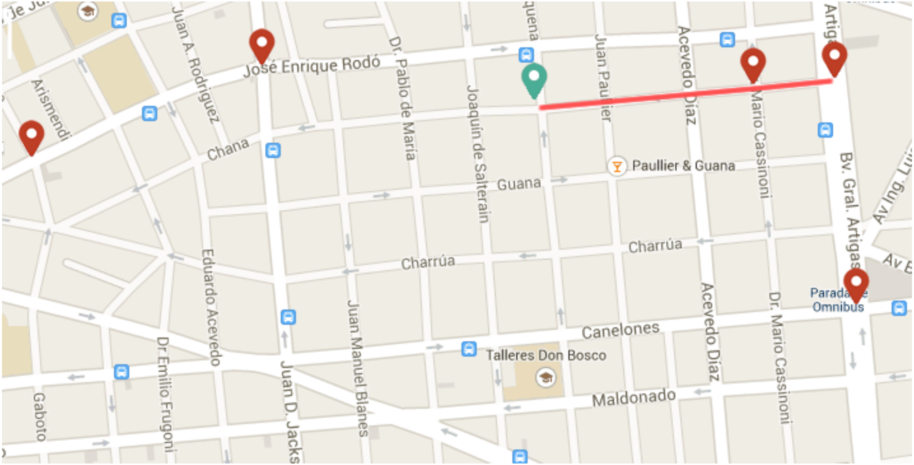
Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



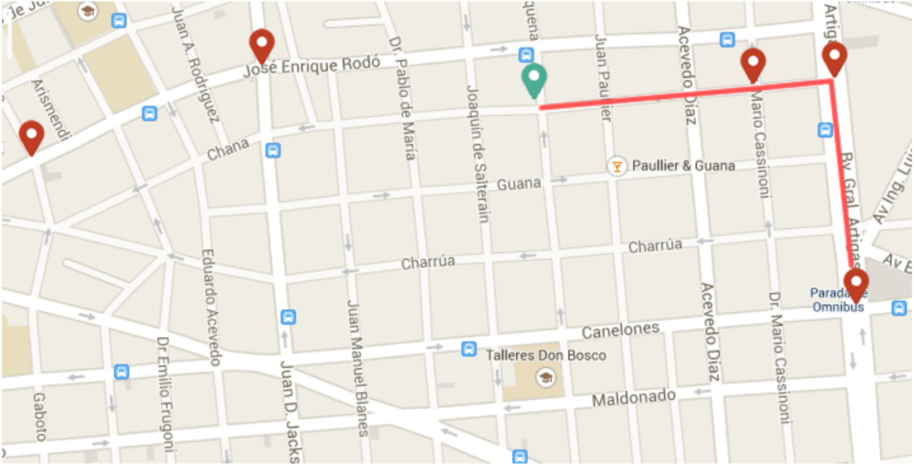
Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



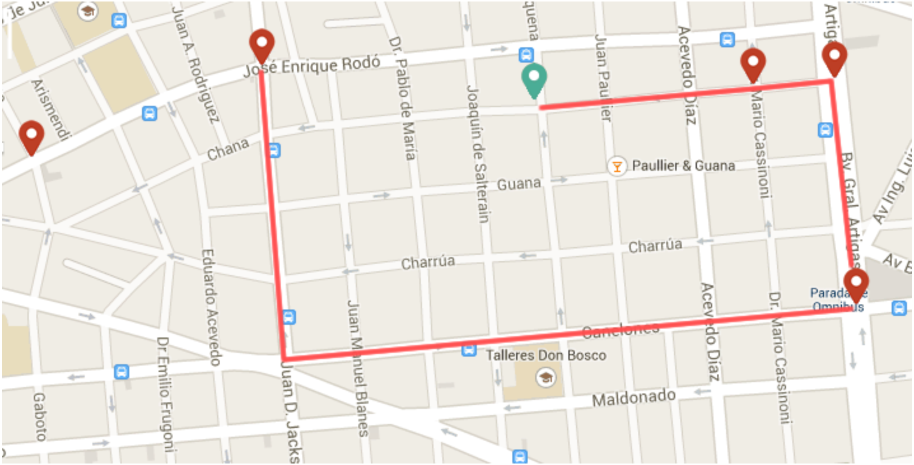
Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



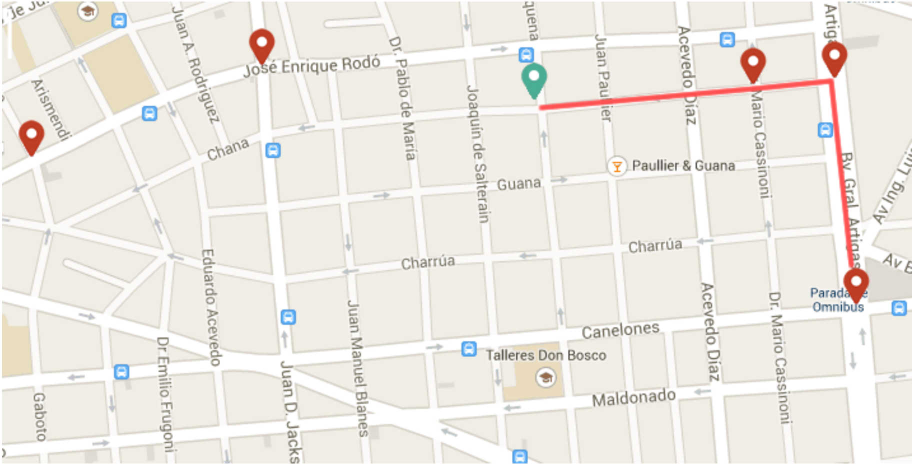
Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



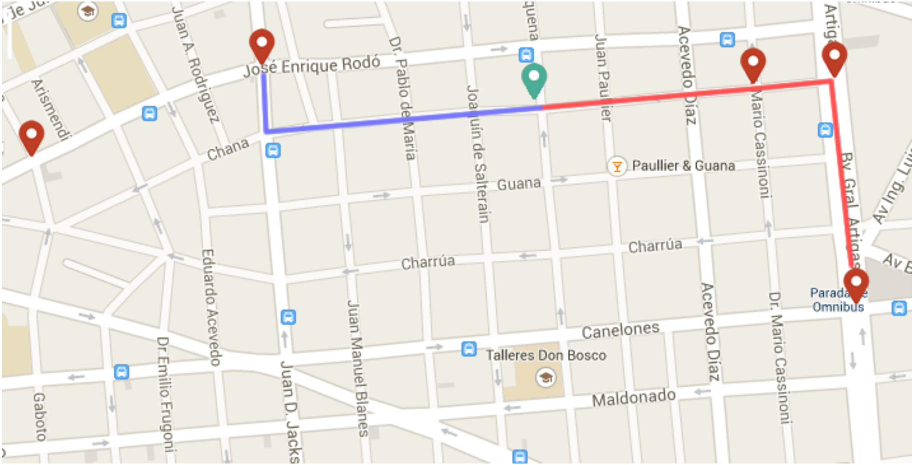
Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



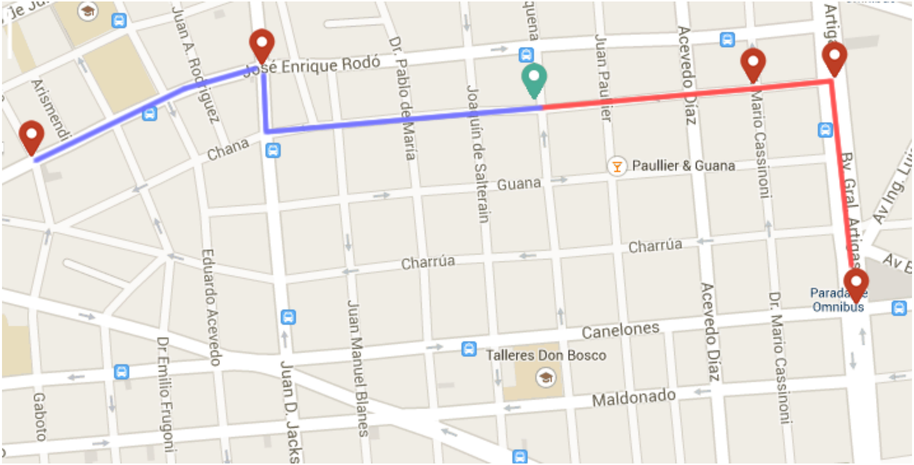
Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



Algoritmo ávido

Utiliza ideas de los trabajos relacionados. Toma **decisiones localmente óptimas** y emula el comportamiento de un grupo de usuarios humanos.



Comparativa de métodos de inicialización

Metodología

- Shapiro–Wilk afirma que los resultados no siguen una distribución normal.
- Se utilizó Kruskal–Wallis para comparar ambas inicializaciones.
- Kruskal–Wallis permite afirmar con un nivel de confianza del 95 % que una de las inicializaciones obtuvo mejores resultados que la otra.

seqEA con inicialización aleatoria vs. incialización ávida

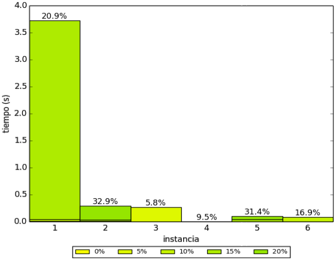
La inicialización ávida supera a la inicialización aleatoria en 10 instancias de prueba, mientras que la inicialización aleatoria lo hace en tan solo 2.

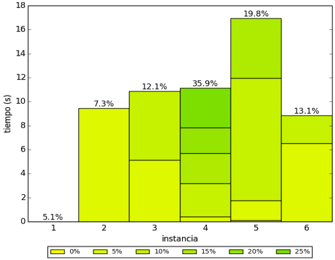
pμEA con inicialización aleatoria vs. incialización ávida

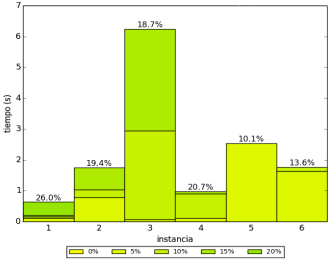
La inicialización ávida super a la inicialización aleatoria en 11 instancias de prueba, mientras que no hubo instancias en las que se pueda afirmar que la inicialización aleatoria haya alcanzado mejores soluciones.

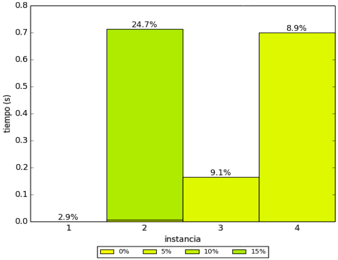
Mejora *seqEA* sobre algoritmo ávido

Se alcanzaron mejores valores en **todas** las instancias. En el mejor caso se superó el costo del algoritmo ávido en un 35.9 %.



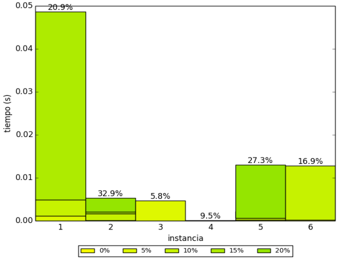


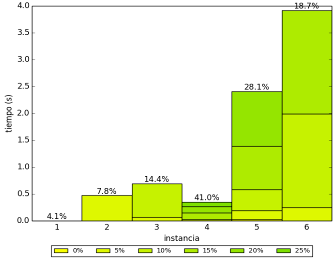


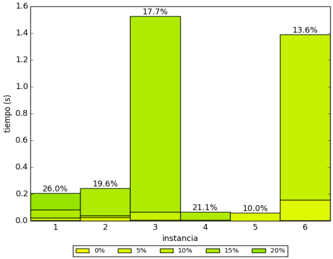


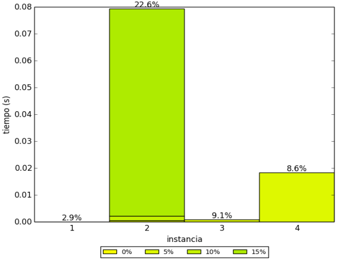
Mejora $p\mu EA$ sobre algoritmo ávido

Se alcanzaron mejores valores en **todas** las instancias. En el mejor caso se superó el costo del algoritmo ávido en un 41.0 %.









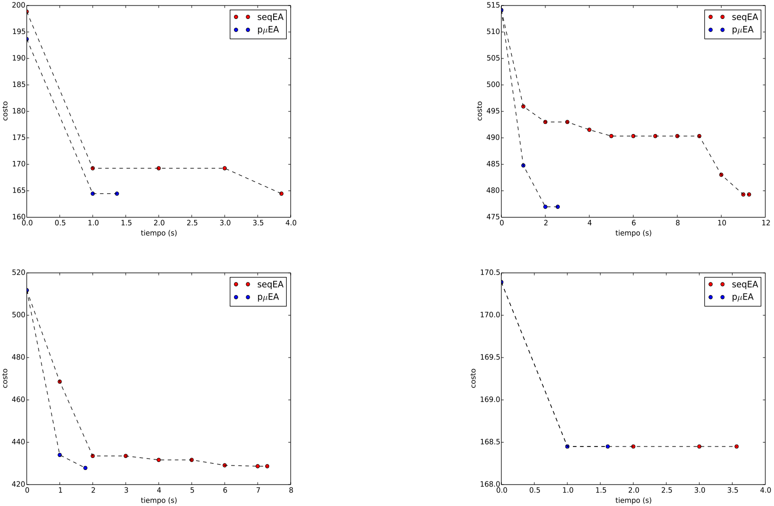
Comparativa *seqEA* vs. *pμEA*

Sobre un total de 22 instancias, *pμEA* es capaz de encontrar mejores resultados que *seqEA* en 17. Únicamente en una instancia de prueba *seqEA* alcanzó mejores soluciones que *pμEA*.

instancia		*seqEA*		*pμEA*		*pvK–W*
		min(c)	$\bar{c} \pm std$	*min*(c)	$\bar{c} \pm std$	
chicas	#1	164.4	165.6±2.0	164.4	164.4±0.0	0.2×10^{-3}
	#2	220.7	225.7±5.0	220.7	220.7±0.0	9.7×10^{-6}
	#3	160.4	160.4±0.0	160.4	160.4±0.0	1.0
	#4	181.3	181.3±0.1	181.3	182.4±1.9	0.5×10^{-1}
	#5	152.1	155.6±4.5	152.1	152.1±0.0	5.1×10^{-6}
	#6	118.4	119.6±2.5	118.4	118.4±0.0	0.1×10^{-1}
medianas	#1	211.9	216.0±4.2	211.9	211.9±0.0	5.2×10^{-11}
	#2	428.6	444.1±11.7	427.9	429.4±1.6	7.0×10^{-10}
	#3	361.7	378.7±6.5	364.5	370.4±4.5	1.6×10^{-6}
	#4	267.5	279.8±5.5	266.8	266.8±0.0	7.6×10^{-12}
	#5	479.3	487.1±6.5	479.6	479.8±0.2	5.1×10^{-7}
	#6	306.0	321.2±7.7	306.0	307.7±3.4	2.0×10^{-9}
grandes	#1	421.9	435.1±5.0	425.9	437.7±3.2	0.1×10^{-1}
	#2	479.3	489.9±4.3	477.0	481.1±2.3	1.9×10^{-9}
	#3	332.8	349.7±7.7	326.3	331.7±4.0	2.6×10^{-10}
	#4	351.1	390.7±26.3	338.4	344.8±6.1	5.1×10^{-11}
	#5	395.9	429.6±16.2	370.2	380.0±4.4	2.7×10^{-11}
	#6	360.8	382.4±8.1	343.8	350.6±3.8	2.6×10^{-11}
Montevideo	#1	168.4	168.4±0.0	168.4	168.4±0.0	1.0
	#2	319.3	331.2±3.8	324.9	328.6±3.2	5.6×10^{-6}
	#3	266.7	269.1±2.3	266.7	266.7±0.0	3.1×10^{-7}
	#4	303.2	304.7±0.5	304.1	304.5±0.4	0.1

Evolución del costo a lo largo de la ejecución

pμEA alcanza mejores soluciones que *seqEA* en menos tiempo. En el mejor caso la aceleración es de 7,5x (4,6x en promedio).



PVCT multiobjetivo

Entorno de ejecución

- La evaluación experimental fue realizada en el Cluster FING.
- **pμMOEA/D**: HP Proliant DL585, **24 núcleos** de AMD Opteron 2.09GHz, 48GB RAM.
- **NSGA–II**: HP Proliant DL385 G7, **1 núcleo** de AMD Opteron 6172 2.10GHz, 72GB RAM.

Configuración paramétrica

- **pμMOEA/D**: 30 ejecuciones de 20000 generaciones sobre 4 instancias.
  $P = 15$; $p_C \in \{0{,}6, 0{,}75, 0{,}95\}$; $p_M \in \{0{,}001, 0{,}01, 0{,}1\}$; operador de migración cada 1000 generaciones.
- **NSGA–II**: 30 ejecuciones de 5000 generaciones sobre 4 instancias.
  $P = 80$; $p_C \in \{0{,}6, 0{,}75, 0{,}95\}$; $p_M \in \{0{,}001, 0{,}01, 0{,}1\}$.

Algoritmo ávido para minimizar demora

FALTA

Métricas multiobjetivo

$p\mu$MOEA/D

Se logra una buena convergencia y diversidad. Sin embargo, la cantidad de soluciones no dominadas es baja, indicando que se puede mejorar.

	#ND	DG	spacing	spread	RHV
chicas	8.5±2.1 (16.0)	3.1±2.5 (0.0)	740.2±746.3 (58.1)	0.6±0.2 (0.1)	0.9±0.1 (1.0)
medianas	9.1±2.2 (19.0)	5.7±2.5 (0.0)	1448.5±1064.1 (141.6)	0.6±0.1 (0.1)	0.9±0.1 (1.0)
grandes	8.5±2.2 (17.0)	7.9±3.4 (2.0)	2917.2±2041.5 (175.3)	0.6±0.1 (0.0)	0.8±0.1 (1.0)
Montevideo	8.0±2.1 (14.0)	3.0±2.0 (0.0)	663.5±542.4 (61.5)	0.6±0.2 (0.0)	0.9±0.0 (1.0)

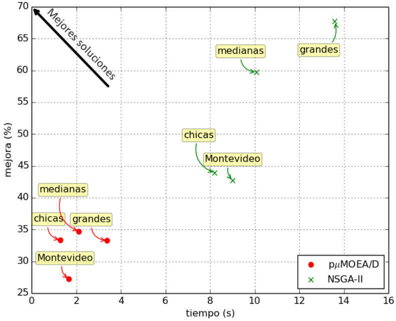
NSGA–II

La cantidad de puntos no dominados es mayor. La distancia generacional indica una buena convergencia y se logra una mayor dispersión.

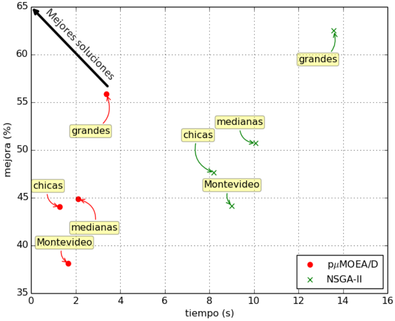
	#ND	DG	spacing	spread	RHV
chicas	32.6±9.5 (55.0)	0.3±0.6 (0.0)	236.2±222.7 (43.2)	0.9±0.1 (0.7)	1.0±0.0 (1.0)
medianas	54.5±4.2 (67.0)	1.0±0.7 (0.0)	193.6±202.4 (26.2)	0.7±0.2 (0.4)	1.0±0.0 (1.0)
grandes	55.2±3.5 (67.0)	1.8±1.1 (0.4)	243.6±229.8 (26.4)	0.7±0.2 (0.4)	1.0±0.0 (1.0)
Montevideo	43.9±16.4 (61.0)	0.4±0.5 (0.0)	142.3±143.2 (20.8)	0.8±0.1 (0.5)	1.0±0.0 (1.0)

Mejora frente a algoritmos ávidos vs. tiempo de ejecución

El enfoque multiobjetivo explícito de *NSGA–II* permite alcanzar mejores resultados que el enfoque por descomposición de dominio aplicado en *pμMOEA/D*. Sin embargo, los tiempos de ejecución son significativamente mayores.



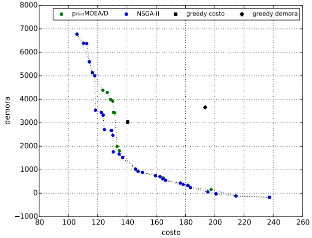
costo

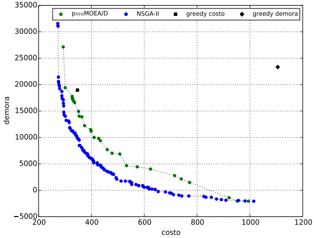


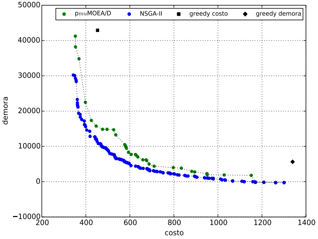
demora

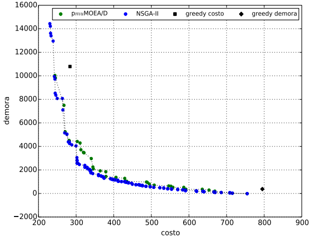
Frentes de Pareto: pμMOEA/D vs. NSGA–II

NSGA–II alcanza mejores soluciones con una mayor cantidad de puntos no dominados distribuidos homogéneamente a lo largo del frente.









Planificador de viajes compartidos en línea

Conclusiones y trabajo futuro