Sistema para el Despacho Dinámico de Técnicos

Sebastián Souyris, DII
Cristián Cortés, Ingeniería Civil Trasporte
Andrés Weintraub, DII

Temas

- Introducción: Ruteo de Vehículos
- Problema Real: Despacho de técnicos
- Enfoques posibles de solución
- Enfoque aplicado
- Resultados y Conclusiones

Ruteo de Vehículos en la Industria

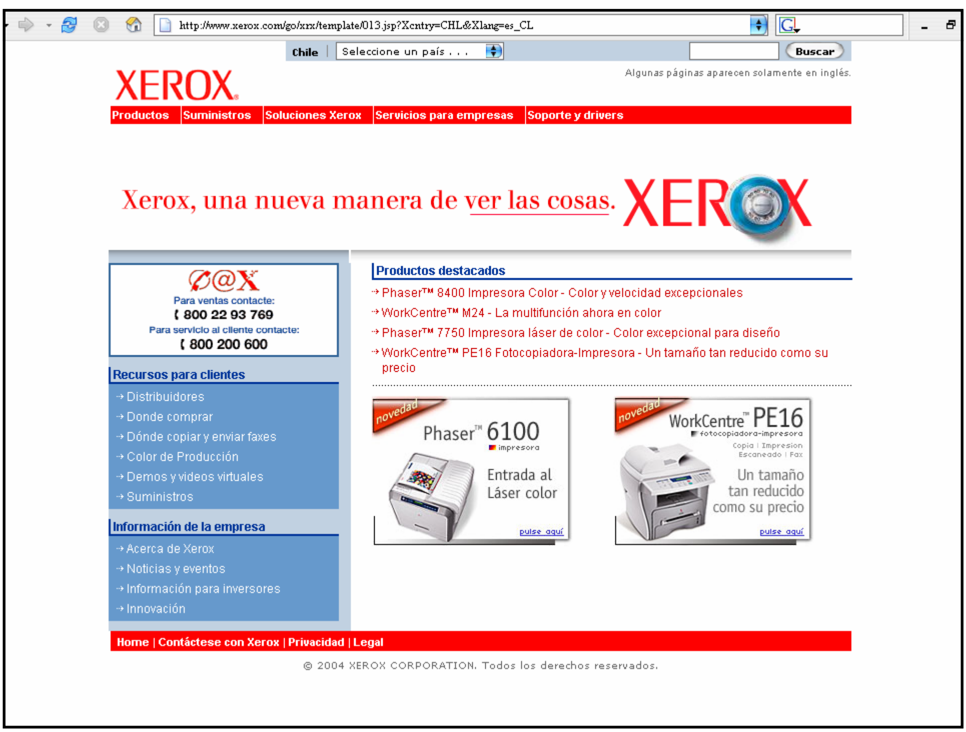
•Asignación de vehículos a tareas o clientes.

•Ruteo de vehículos.

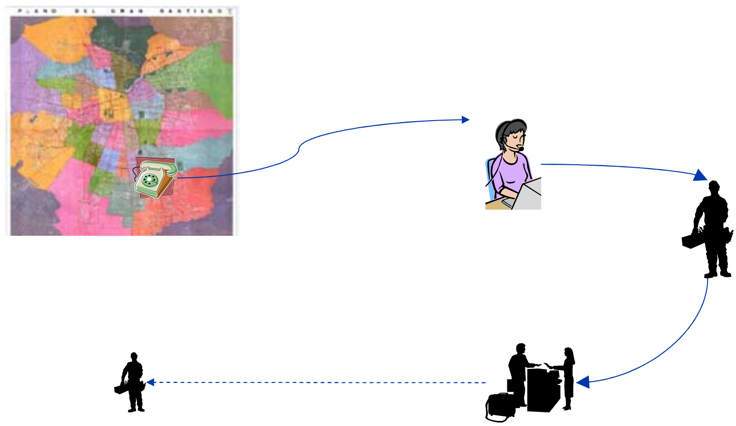
Trasporte de Pasajeros:
- •Buses
- •Taxis
- •Tren

Productos:
- •Embotelladora
- •Supermercados.
- •Minerales
- •Bosques.
- •Bencina

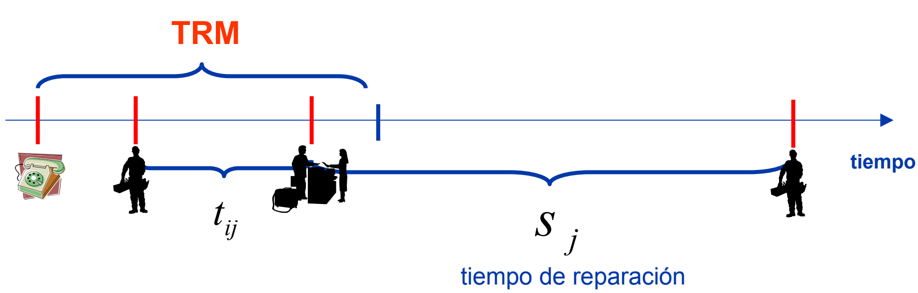
Servicios:
- •Correo
- •Basura
- •Emergencia médica
- •Servicio Técnico



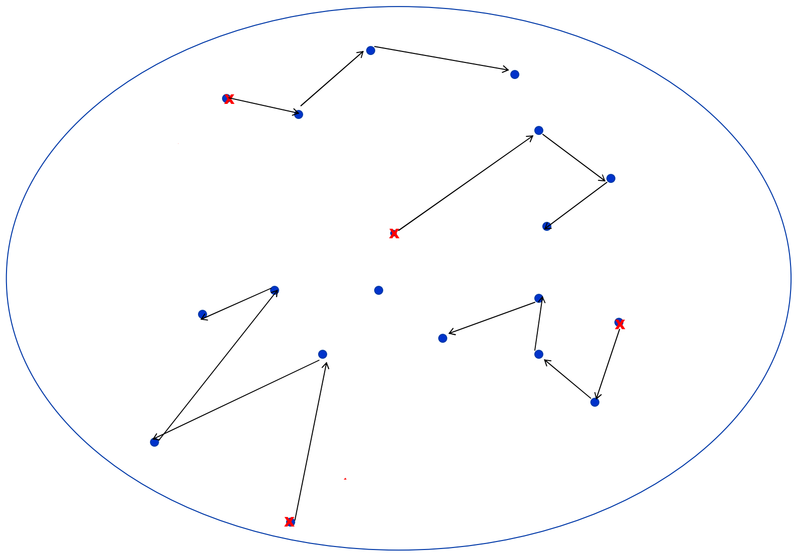
Motivación: Despacho de técnicos



Prioridad Cliente: TRM



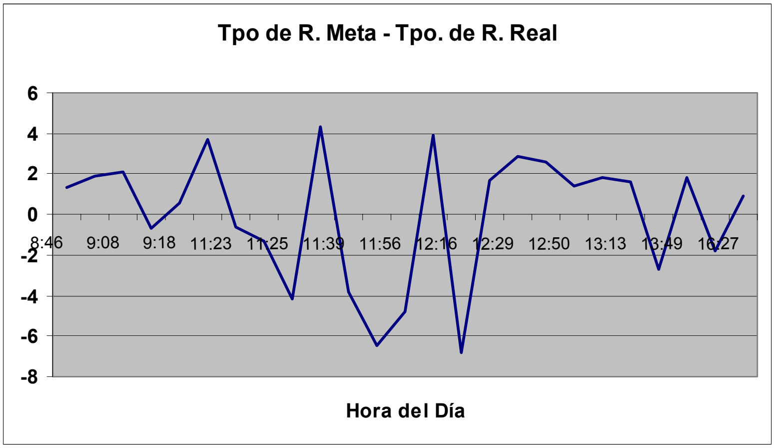
Cliente tiene prioridad establecida por contrato → Tiempo en que debe llegar técnico } **TRM**



Sistema Actual
Servicio Técnico Santiago

- Ciudad dividida territorios con técnicos asignados.
- Más de 100 técnicos en total.
- Más de 400 llamadas diarias.
- Sistema de cola de espera para cada territorio
- Asignación por staff de despachadoras.
  - Mantener cola de cada territorio ordenada,
    - Tiempo espera
    - Prioridad clientes
  - Asignar los técnicos a los clientes.

Problema



¿Qué Hacer?

Sistema de despacho dinámico de técnicos basado en Algoritmos Matemáticos

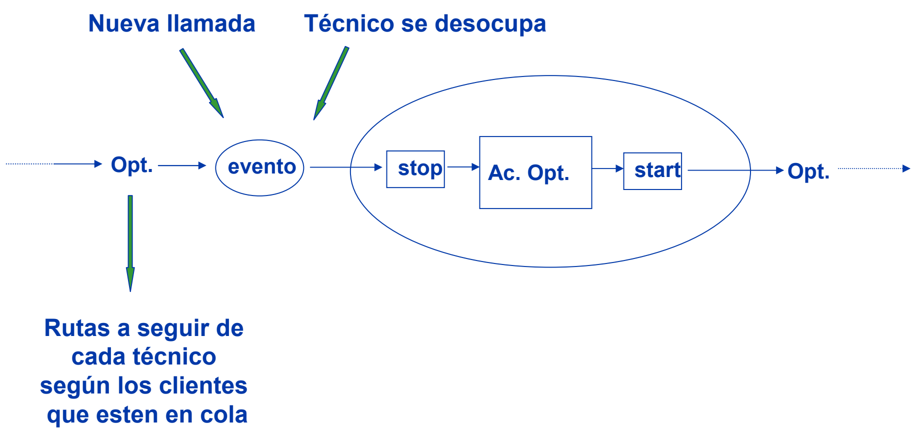
Enfoques

1. Teoría de colas.
2. Algorítmico:
   - Modelos Analíticos
   - Heurísticas y Metaheurísticas: (Tabu Search, Ant System)

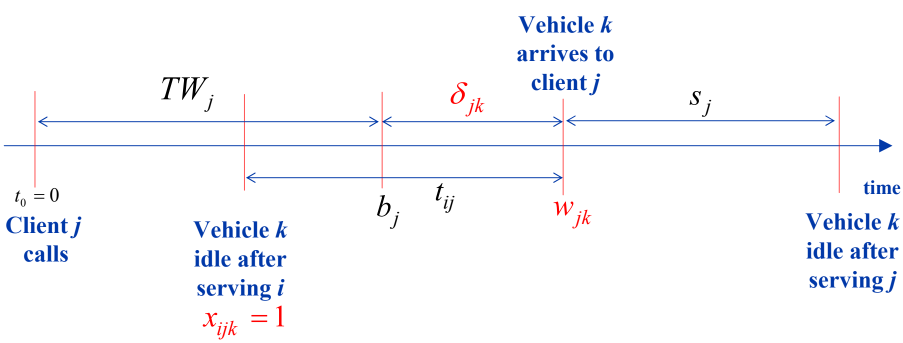
Enfoques

1. Teoría de colas.
2. Algorítmico:
   - Modelos Analíticos
   - Heurísticas y Metaheurísticas: (Tabu Search, Ant System)

Enfoque “Algorítmico”



Model: VRPTW



Variables

$$x_{ijk} : \begin{cases} 1 \text{ if vehicle } k \text{ goes from client } i \text{ to } j \\ 0 \text{ otherwise} \end{cases}$$

w_{ik} : time in which vehicle k starts service of client i

δ_{ik} : time window violation of vehicle k to service client i

FO: Minimize TW violation and travel time

$$\min_{x,\delta} \underbrace{\sum_{k\in K}\sum_{i\in I}\delta_{ki}}_{\substack{\text{Violaciones}\\ \text{de las ventanas}\\ \text{de tiempo}}} + \overbrace{\beta}^{\substack{\text{Penalización}\\ \text{tiempos}\\ \text{de viaje}}} \underbrace{\sum_{k\in K}\sum_{i,j\in I} t_{ij}x_{ijk}}_{\substack{\text{Tiempos}\\ \text{de viaje}}}$$

Todos las máquinas deben ser atendidas por un y solo un técnico.

$$\sum_{k\in K}\sum_{j\in I} x_{ijk} = 1$$
$$\forall i \in \{I \backslash m_{I+1}\}$$

De todas las máquinas, en cola de espera, entra y sale el mismo técnico

$$\sum_{i} x_{ijk} - \sum_{i} x_{jik} = 0$$
$$\forall j \in \{m_{K+1},\ldots,m_I\}, \forall k \in K$$

El inicio del servicio de cada máquina debe ser anterior al final del día

$$w_{ik} \le L\sum_{j\in I} x_{jik}$$
$$\forall i \in I, \forall k \in K$$

Concordancia entre tiempos:
Si técnico *k* visita máquina *i* y luego a máquina *j*, entonces el tiempo de inicio de servicio de *j* debe ser mayor que el tiempo de inicio de servicio de *i* más el tiempo de reparación de *i* más el tiempo de viaje entre *i* y *j*

$$w_{ik} + s_i + t_{ij} - w_{jk} \le (1 - x_{ijk}) * M$$
$$\forall i, j \in I, \forall k \in K$$

Se incurre violación de ventana de tiempo si el tiempo de inicio de servicio es mayor que la cota superior de la ventana de tiempo.

$$w_{ik} - \delta_{ik} \le b_i$$
$$\forall i \in I, \forall k \in K$$

Naturaleza de las variables

$$x_{ijk} \in \{0,1\}$$
$$w_{i,k}, \delta_{i,k} \ge 0 \quad \forall i, j \in I, \forall k \in K$$

Características del Modelo

- El VRPTW es NP-hard.
- Programación y resolución resolución con CPLEX 7.5 permitió resolver instancias con 20 clientes y 5 técnicos en 3.5 hrs. (Pentium IV 2.2 Ghz, 256 RAM).
- Instancias de tamaño real no es posible resolver con este modelo en tiempo razonable.

Alternativas de Solución:

•Heurísticas
•Metaheurísticas (Tabu, etc).
•Formulación Alternativa.

→ **Column Generation**

Características del Modelo

- El VRPTW es NP-hard.
- Programación y resolución resolución con CPLEX 7.5 permitió resolver instancias con 20 clientes y 5 técnicos en 3.5 hrs. (Pentium IV 2.2 Ghz, 256 RAM).
- Instancias de tamaño real no es posible resolver con este modelo en tiempo razonable.

Alternativas de Solución:

•Heurísticas
•Metaheurísticas (Tabu, etc).
•Formulación Alternativa.

→ **Column Generation**

Dantzig Wolfe Decomposition
Column Generation

Decompose Original Problem into:

- Master Problem – Choose routes
  OF and consistency (coverage) constraint
- Sub Problem – Generate Routes
  Modified OF, route constraints and time

$$\min_{x,\delta} \sum_{k\in K}\sum_{i\in I}\delta_{ki} + \sum_{k\in K}\sum_{i,j\in I} t_{ij}x_{ijk}$$

$$\sum_{k\in K}\sum_{j\in I} x_{ijk} = 1 \qquad \forall i \in \{I \backslash m_{I+1}\}$$

Master Problem: Set Partitioning Model

$$\sum_{\mathrm{i}} x_{ijk} - \sum_{\mathrm{i}} x_{jik} = 0$$

$$\forall j \in \{m_{K+1},\ldots,m_I\}, \forall k \in K$$

$$w_{ik} + s_i + t_{ij} - w_{jk} \le (1 - x_{ijk}) * M \qquad \forall i,j \in I, \forall k \in K$$

$$x_{ijk} \le c_{jk} \qquad \forall i,j \in I, \forall k \in K$$

$$w_{ik} \le L\sum_{j\in I} x_{ijk} \qquad \forall i \in I, \forall k \in K$$

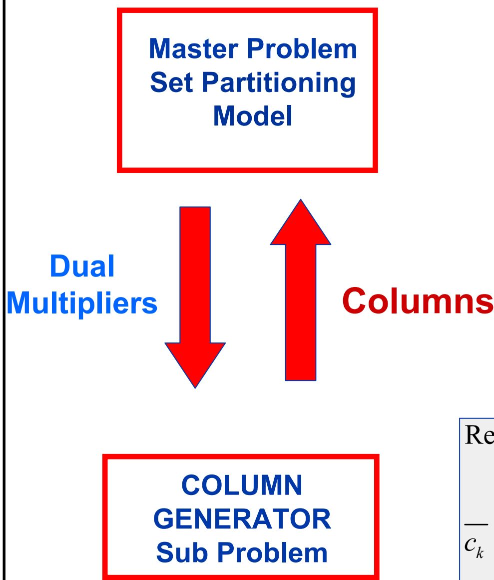
$$\delta_{ik} \ge [w_{ik} - b_i] \qquad \forall i \in I, \forall k \in K$$

$$x_{ijk} \in \{0,1\}$$

$$w_{i,k}, \delta_{i,k} \ge 0 \quad \forall i,j \in I, \forall k \in K$$

COLUMN GENERATOR
Sub Problem

c



$$P: \quad \min_{x} \sum_{r\in R} c^r x^r$$

$$\sum_{r\in R} a_i^r x^r = 1 \quad \forall i \in I \;\longrightarrow\; \alpha_i$$

$$x^r \in \{0,1\}, \quad r \in R$$

$$x^r = \begin{cases} 1 \text{ si ruta r es seguida} \\ 0 \text{ si no} \end{cases}$$

$$c^r = \text{costo de la ruta r}$$

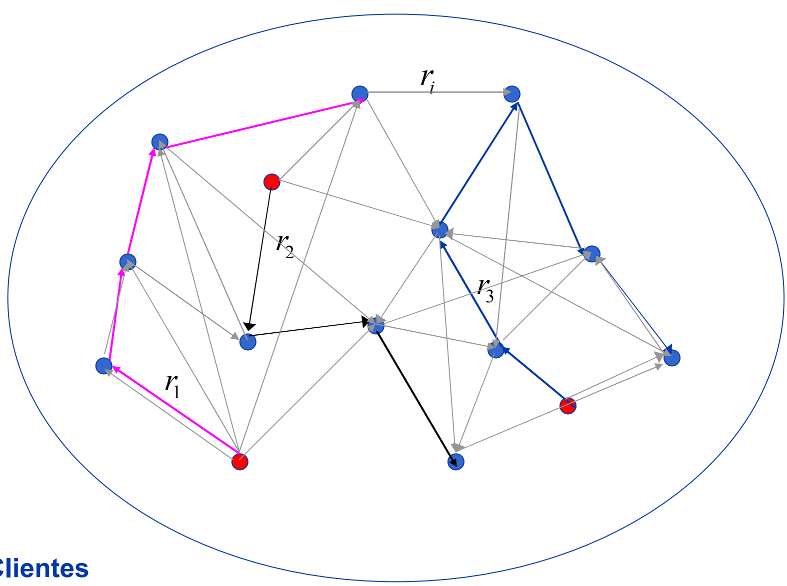
$$a_i^r = \begin{cases} 1 \text{ si máquina } i \text{ pertenece a ruta } r \\ 0 \text{ si no} \end{cases}$$

$$R = \text{Cjto. de Rutas factibles}$$

Reduced Cost Column

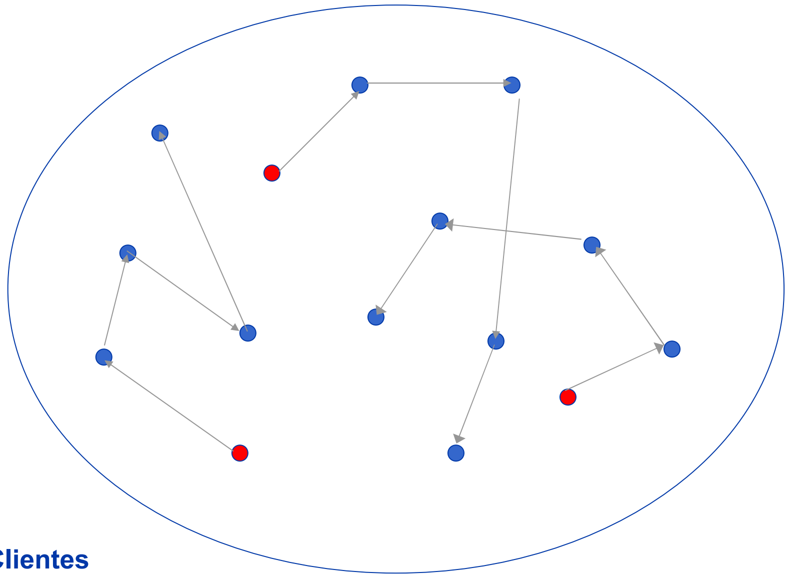
$$\overline{c_k} = \sum_{i\in P^k} \delta_{ki} + \beta \sum_{i,j\in P^k} t_{ij} - \overbrace{\sum_{i\in P^k} \alpha_i}^{\text{Suma de los multiplicadores duales de las máquinas que pertenecen a esta columna}}$$

2. Column Generation cial para formar solución factible



- Clientes
- Técnicos

3. Selección de las mejores columnas



Clientes

Técnicos

Generate feasible routes

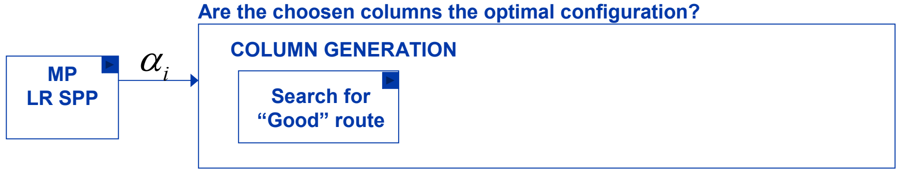
	r_1	r_2	r_3	$\cdots$	r_{R_1}
c_1	1	1	1	$\cdots$	0
c_2	1	0	1	$\cdots$	1
c_3	0	1	0	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
c_i	1	1	1	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
c_m	0	0	1	$\cdots$	1

MP
LR SPP

Generate feasible paths

	r_1	r_2	r_3	$\cdots$	r_{R_1}	Σ
c_1	1	1	1	$\cdots$	0	1
c_2	1	0	1	$\cdots$	1	1
c_3	0	1	0	$\cdots$	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
c_i	1	1	1	$\cdots$	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
c_m	0	0	1	$\cdots$	1	1

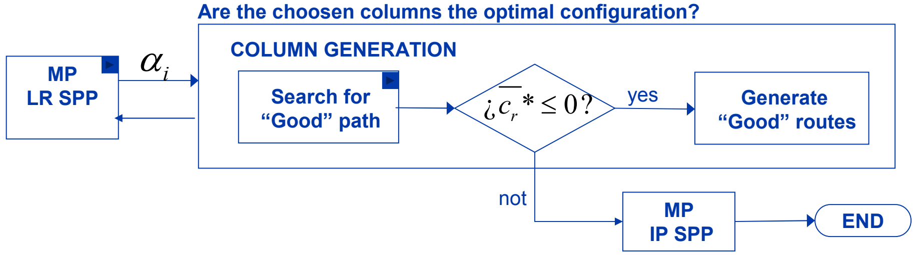
Are the choosen columns the optimal configuration?



Generate feasible paths

	r_1	r_2	r_3	$\cdots$	r_{R_1}	Σ	r_{R_1+1}	r_{R_1+2}	$\cdots$	r_{R_2}	Σ	r_{R_2+1}	r_{R_2+2}	$\cdots$	r_{R_3}	Σ
c_1	1	1	1	$\cdots$	0	1	1	1	$\cdots$	0	1	0	0	$\cdots$	1	1
c_2	1	0	1	$\cdots$	1	1	0	1	$\cdots$	1	1	0	0	$\cdots$	0	1
c_3	0	1	0	$\cdots$	1	1	0	0	$\cdots$	0	1	1	0	$\cdots$	0	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
c_i	1	1	1	$\cdots$	1	1	1	0	$\cdots$	1	1	1	1	$\cdots$	0	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
c_m	0	0	1	$\cdots$	1	1	0	0	$\cdots$	0	1	0	1	$\cdots$	1	1

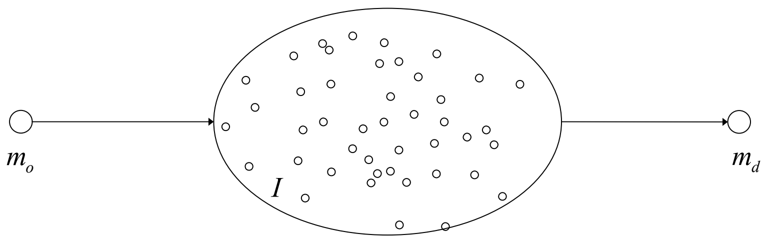
Are the choosen columns the optimal configuration?



Sub Problem: Shortest Path Problem on G=(N,A)

Encontrar el camino más corto entre *mo* y *md*

Costo de arco$_{ij}$ = tiempo de viaje + violación ventana de tiempo nodo j – variable dual de nodo j



$$\text{Costo Path } m_o \text{ a } m_d = F_{m_o - m_d} = \sum_{i \in Path} \delta_i + \beta \sum_{(i,j) \in Path} t_{ij} - \sum_{i \in Path} \alpha_i$$

Sub Problem: Shortest Path Problem on G=(N,A)

Problema NP-Hard Métodos de Solución

- **Programación Dinámica**
- **Constraint Programming**

F(S, i,w): costo mínimo de la ruta desde m_o hasta $i \in M \setminus m_o$, visitando todos los nodos del conjunto $S \subseteq M \setminus m_o$ solamente una vez y sirviendo al nodo i en w o después.

$$F(\phi, m_o, 0) = 0$$

$$F(S, j, w) = \min_i \Big\{ F(S - \{j\}, i, w') + (t_{ij} + \delta j - \alpha_j) \mid i \in S - \{j\},$$
$$w' \leq w - (t_{ij} + s_i), w' \leq L, \delta_j = \max\{0, w - b_j\} \Big\},$$
$$\forall S \subseteq M, j \in S \quad \text{y} \quad w \leq L.$$

Sub Problem: Shortest Path Problem on G=(N,A)

•Problema NP-Hard.
•Largo pequeño de las rutas permite resolverlo en tiempo razonable

Metodología Usada: Constraint Programming
Rápida resolución dada estructura del problema (pocas máquinas por técnico).

Constraint Programming

- A computer programming methodology
- Solves
  - Constraint satisfaction problems
  - Combinatorial optimization problems
- Methodology
  - Represent a model of a problem in a computer programming language.
  - Describe a search strategy for solving the problem

Constraint satisfaction problems:	Combinatorial Optimization Problems:
•Find a Feasible Solution. •Subject to Constraints. •Over a set of values of decision variables	•Minimize (or maximize) an Obj Function. •Subject to Constraints. •Over a set of values of decision variables

Constraint Programing Model

SETS

M :	clients
$MT \subseteq M$:	clients with technician
$MQ \subseteq M$:	clients in queue

PARAMETERS

s_i :	reparation time of client i
b_i :	due time of client i
F :	period end of evaluation
tv_{ij}:	travel time between clients i and j
L :	lenght of the path to generate
c_i :	dual variable of client i given by the LR of SSP
$tvmax$:	maximum time of trip allowed in the path to generate

VARIABLES

$MaqSeq[l] \in M$:	secuence of clients in path to generate, $l = 0..L$
$w[l] \in [0..F]$:	time to begin service of client l in position l in the path, $l = 0..L$
$d[l] \in [0..F]$:	time windows violation, $l = 1..L$
$t[l] \in [0..tvmax]$:	travel time to clients in position l - 1 to l

OBJECTIVE FUNCTION

$$Min \ \ ReduceCost = \left[\left(B * \sum_l d[l] + \sum_l t[l] \right) - \sum_l c_{MaqSeq[l]} \right]$$

SUBJECT TO

1. Time to begin the service

$$w[l] = w[l-1] + s_{MaqSeq[l-1]} + t[l] \qquad , \forall l = 1..L$$

2. Violation of time windows

$$d[l] \geq w[l] - b_{MaqSeq[l-1]} \qquad , \forall l = 0..L$$

3. Travel time between clients in the path

$$t[l] = tv_{MaqSeq[l-1], MaqSeq[l]} \qquad , \forall l = 1..L$$

4. Path begin time (depends of the state of service of the first client)

$$w[0] = b_{MaqSeq[0]}$$

5. All the clients in the path must be different

$$MaqSeq[m] <> MaqSeq[n] \qquad , \forall m < n = 0..L$$

$$alldiferent(MaqSeq)$$

6. First client in the path must have technician

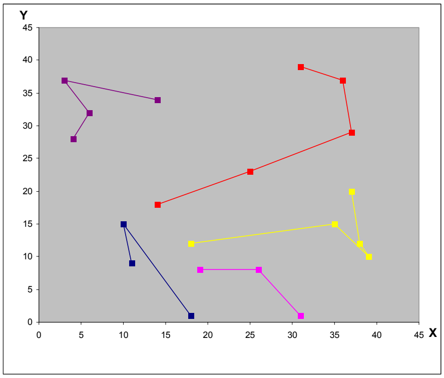
$$MaqSeq[0] \in MT$$

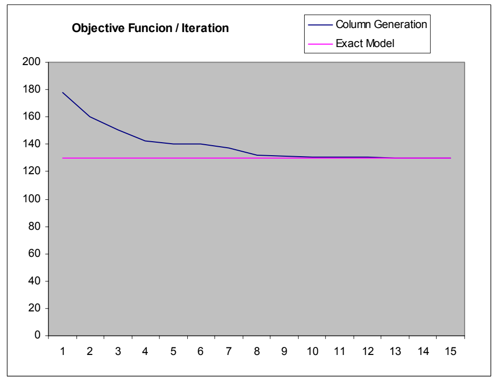
7. Other clients in the path be not have technician

$$MaqSeq[l] \in MQ \qquad , \forall l = 1..L$$

Exact Model v/s Column Generation

Instance: 5 technicians, 20 clients
Exact Solution in 3.5 hrs.
Same solution with Column Generation in 15 iterations, 3.7 sec.





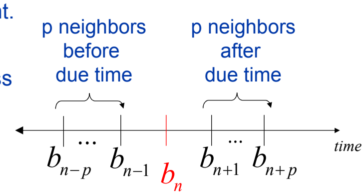
Real Instance

- Model coded in OPL Studio 3.5, and solved by using CPLEX 7.5 and SOLVER 5.2
- Real Operation: Normal working day, 36 clients (machines), 9 technicians.
- Optimization total time 500 seg
- The following maps compare observed routing versus optimized routing.
- Illogical patterns are due to TW and priority constraints

	REAL	CG
TOTAL TRAVEL TIME (min)	918	656
TOTAL Violation TW (min)	3965	1530
Z*	**8848**	**3716**

Dynamic Insertion

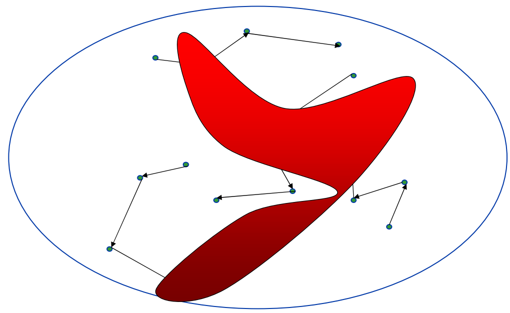
1. Generate a Neighborhood around new client.
2. Insert it in the path with minimum cost.
3. After N new clients run the complete process

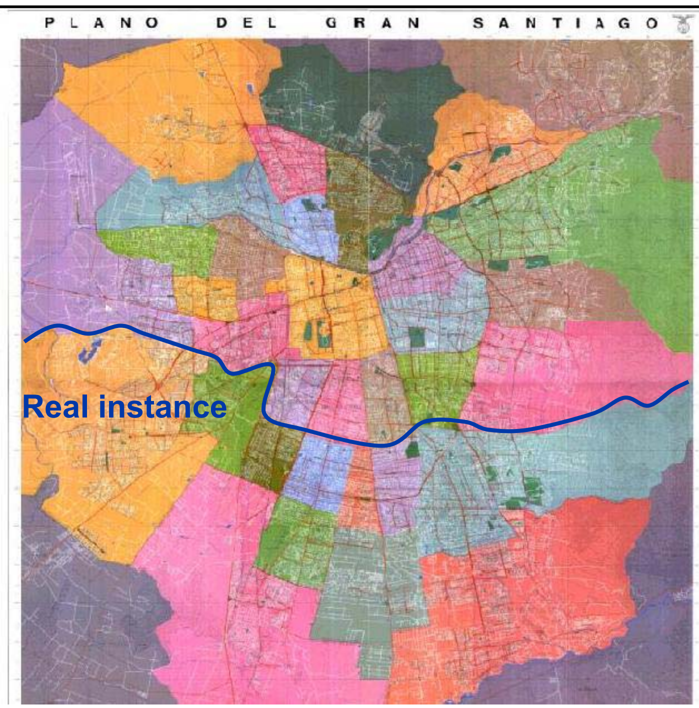


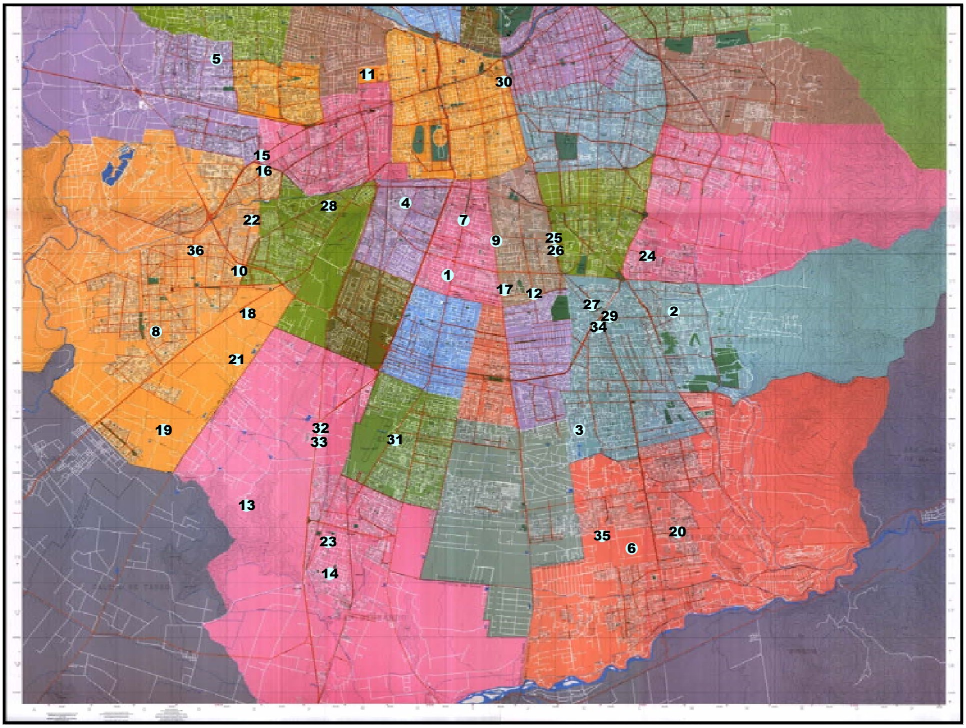
Neighborhood

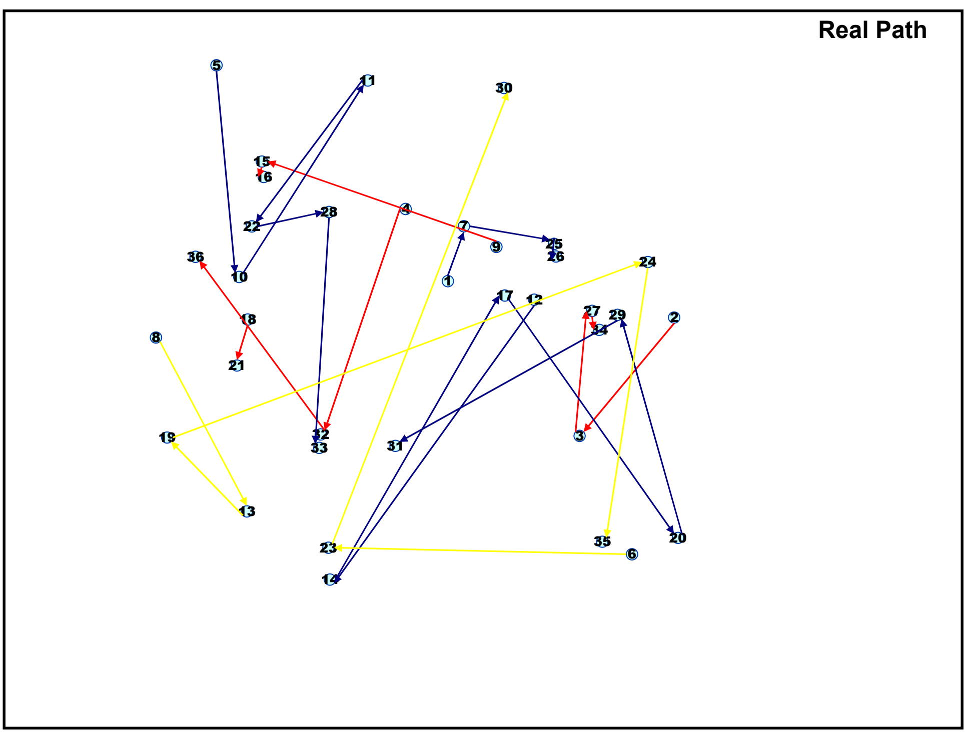
•Due Time(p): p neighbors before and after due time

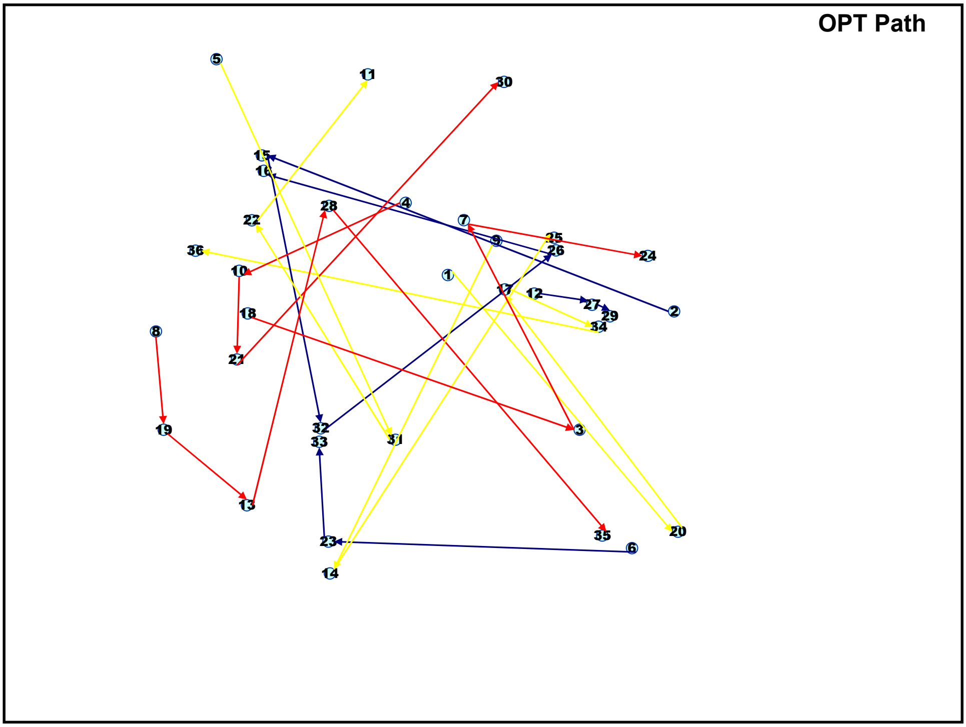
•Distance(q): q neighbors by distance proximity






5
11
30
15
16
4
28
22
7
9
25
26
36
24
10
1
17
12
27
29
2
34
18
8
21
32
33
31
3
19
13
35
20
6
23
14


Real Path
5
11
30
15
16
4
28
22
7
9
25
26
36
24
10
1
17
12
27
29
2
34
18
8
21
32
33
31
3
19
13
35
20
6
23
14



OPT PATHS

Path	Maq	w	d	s	trm	tv
1	m1	755	0	60	755	0
	m20	838	73	30	765	23
	m17	892	0	80	929	24
	m34	993	0	90	1059	21
	m36	1109	0	145	1890	26
2	m2	1002	0	120	1002	0
	m15	1152	0	120	1716	30
	m32	1290	0	70	2032	18
	m26	1371	0	180	1652	11
	m16	1581	0	120	1718	30
3	m4	1022	0	100	1022	0
	m10	1152	307	85	845	30
	m21	1262	315	60	947	25
	m30	1354	0	150	1893	32
4	m5	1024	0	45	1024	0
	m31	1083	0	90	1780	14
	m22	1190	0	120	1832	17
	m11	1325	0	40	1779	15
5	m6	1980	0	60	1980	0
	m23	2068	0	20	2250	28
	m33	2113	0	60	2214	25
6	m8	943	0	135	943	0
	m19	1101	116	120	985	23
	m13	1244	245	60	999	23
	m28	1352	0	30	1701	48
	m35	1425	0	90	2010	43
7	m9	1068	0	60	1068	0
	m14	1143	0	90	1834	15
	m25	1244	0	90	1651	11
8	m12	1662	0	75	1662	0
	m27	1761	0	30	2040	24
	m29	1812	0	60	1955	21
9	m18	970	0	50	970	0
	m3	1045	30	180	1015	25
	m7	1249	444	195	805	24
	m24	1474	0	105	1758	30
	TOTAL		**1530**			**656**
		Z*	**3716**			

REAL PATHS

Path	Maq	w	d	s	trm	tv
1	m2	660	0	120	1002	19
	m3	810	0	180	1015	30
	m27	1680	0	30	2040	30
	m34	1800	741	90	1059	0
2	m4	1670	648	100	1022	20
	m36	2085	53	70	2032	22
	m32	1820	0	145	1890	26
3	m18	870	0	50	970	0
	m21	930	0	60	947	0
4	m9	1020	0	60	1068	30
	m15	1710	0	120	1716	60
	m16	1860	142	120	1718	17
5	m20	1665	3	75	1662	15
	m17	1770	0	90	1834	20
	m29	910	0	80	929	20
	m12	765	0	30	765	0
	m14	1020	0	60	1955	20
	m31	1980	200	90	1780	40
6	m10	1665	641	45	1024	15
	m5	720	0	85	845	30
	m28	1980	201	40	1779	60
	m22	1800	0	120	1832	15
	m11	1740	39	30	1701	30
	m33	2130	0	60	2214	42
7	m7	900	145	60	755	41
	m1	600	0	195	805	14
	m25	990	0	90	1651	18
	m26	1650	0	180	1652	0
8	m6	990	0	60	1980	20
	m23	990	0	20	2250	30
	m30	1680	0	150	1893	30
9	m8	705	0	135	943	10
	m13	930	0	60	999	20
	m19	1800	815	120	985	150
	m24	1995	237	105	1758	15
	m35	2110	100	90	2010	10
	TOTAL		**3965**			**918**
		Z*	**8848**			

Ruteo de Vehículos

IN74O – MODELOS INDUSTRIALES
Agosto 2004

$$\min z = cx$$
$$Ax = b$$
$$x \geq 0$$

n variables, m restricciones
n>>m,
No se conocen todas las columnas de A

sea B base factible

$x = (x_B, x_n)$

$x_N = 0, x_B = B^{-1}b \geq 0$

Buscar "mejor" columna

$\min \overline{c_j} = \min(c_j - c_B B^{-1} a_j) = \overline{c_j}^*$

sea $\alpha = c_B B^{-1}$ = vector de multiplicadores duales

¿Óptimo? costo reducido de todas las columnas debe ser ≥ 0

$\overline{c_j} = c_j - c_B B^{-1} a_j$

¿ $\overline{c_r}^* \leq 0$?

Encontrar nueva Ruta con menor $\overline{c_r}*$

$$\left[\min_r \left(c_r - \sum_{i \in Maq} \pi_i y_{ir} \right)\right] = \overline{c_r}*$$

con

c_r = costo de la ruta r (sobreventanas y tiempo de viaje)

y_{ir} = 1 si maquina i pertenece a ruta r, 0 si no

Enfoque Estratégico
(Bertsimas y Van Ryzin, 1990 - 1993)

Basado en teoría de colas: M/G/m

Región de servicio A

$Dda \rightarrow P(\lambda)$; $X \rightarrow U$

$TSer \rightarrow G(\overline{s})$; $\rho = \lambda \overline{s}$

T_i = tpo. total de dda. i en el sistema

$= W_i^d + W_i^s + s_i$

$\underline{P|}$ $Min(\lim_{x \to \infty} E[T_i]) = T^*$

…Enfoque Estratégico

- Cota inferior para carga baja $\rho \to 0$

$$T^* \geq \underline{T}_{CB} = \frac{1}{v} E[\min_{x_0 \in D^*} \|X - x_0\|] + \bar{s}$$

- Cota inferior para carga alta $\rho \to 1$

$$T^* \geq \underline{T}_{CA} = \gamma^2 \frac{\lambda A}{m^2 v^2 (1-\rho)^2} - \frac{\bar{s}(1-2\rho)}{2\rho}$$

$donde\ \gamma \geq 2/(3\sqrt{\pi}) \approx 0.376$

Politicas Optimas: carga baja

m Stochastic Queue Median (mSQM)

- Dividir A en m subregiones y asignar un vehiculo a cada una.
- Colocar cada vehículo en el centro de la región asignada.
- En cada subregión atender a los clientes en orden FCFS, pero despues de terminar cada servicio volver al centro de la subregión.

$$\frac{T_{mSQM}}{T^*} \to 1 \quad ; \lambda \to 0$$

Politicas Optimas: carga alta

G/G/M versión de TSP

- Formar conjuntos de demandas en espera (según aparecen) ($|N_k|$=n).
- Agrupar estos conjuntos en una cola única.
- Asignar conjuntos a vehículos disponibles según FCFS.
- Atender cada conjunto según un TSP.
- Optimizar sobre n.

$$\frac{T_{GGM}}{T*} \leq \frac{m+1}{2}\frac{\beta}{\gamma^2} \approx m+1(1.8) \quad ; \quad \rho \to 1 \qquad ; \beta = 0.72$$

Politicas Optimas: carga alta

G/G/M versión modificada de TSP

- Dividir A en k subregiones de igual area, tipo “torta”
- En cada subregión formar conjuntos de clientes de tamaño n/k
- Agrupar estos conjuntos en una cola única.
- Asignar conjuntos a vehículos disponibles según FCFS.
- Atender cada conjunto según un TSP.
- Optimizar sobre n.

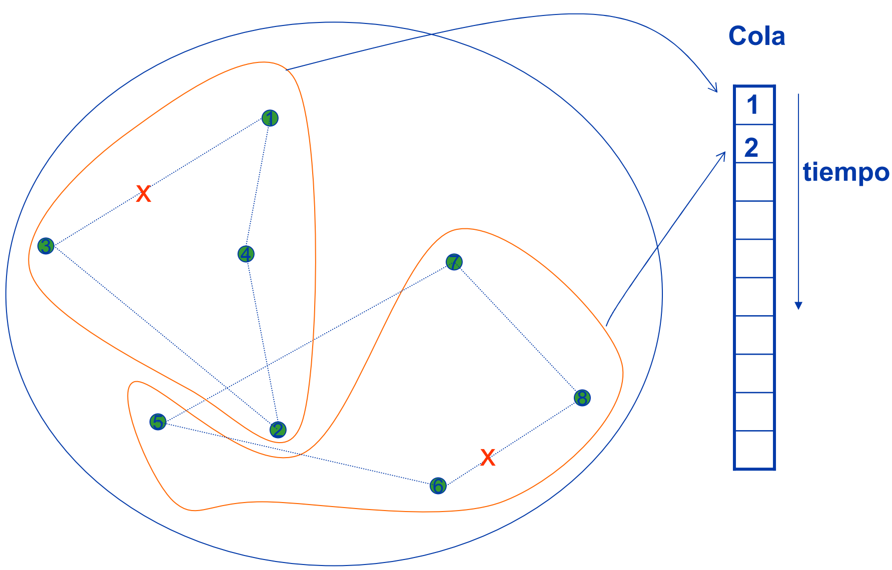
$$\frac{T_{MODGGM}}{T*} \leq \frac{\beta}{2\gamma^2} \approx 1.8 \quad ; \quad \rho \to 1$$

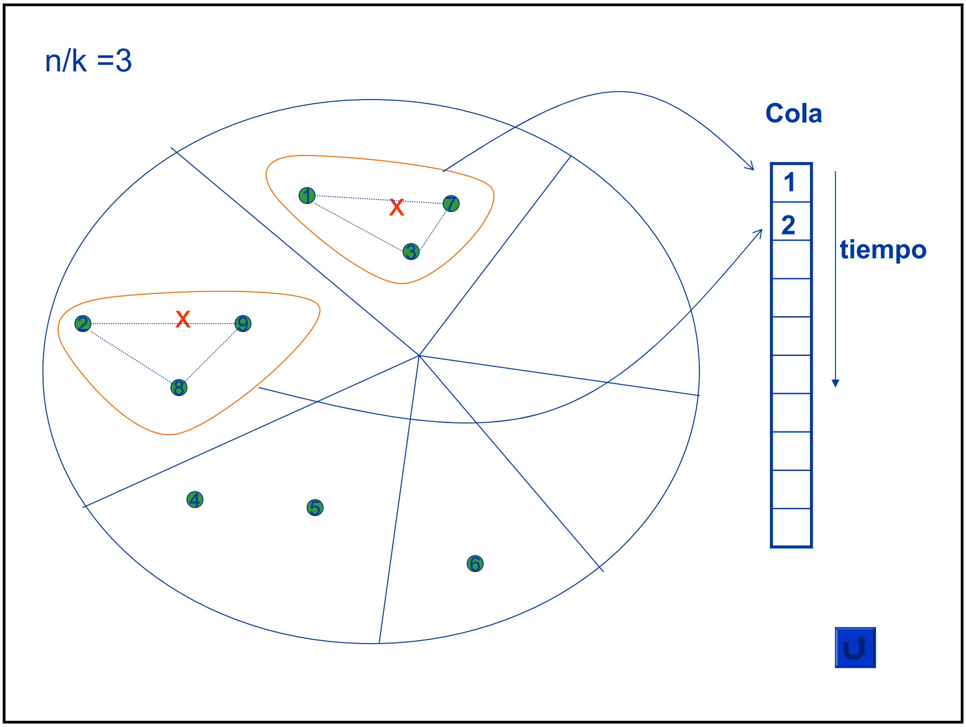
Enfoque “Algorítmico”

1. Parallel Tabu Search for real-time Vehicle Routing and Dispatching (Gengreau, Guertin, Potvin y Taillard, 1999).
   - Basado en Tabu Search diseñado para VRPTW
     - Memoria Adaptativa
     - Descomposición / Reconstrucción
     - Vecindario basado en intercambio CROSS
     - Implementación en paralelo.

2. Ant Colony System adaptado para DVRP (Montemanni, 2002).
   - Basado en comportamiento de colonia de hormigas

n = 4




n/k =3
Cola
1
2
tiempo
1
7
3
X
2
9
8
X
4
5
6