Metaheuristics in Vehicle Routing

Michel Gendreau

CIRRELT and MAGI

École Polytechnique de Montréal

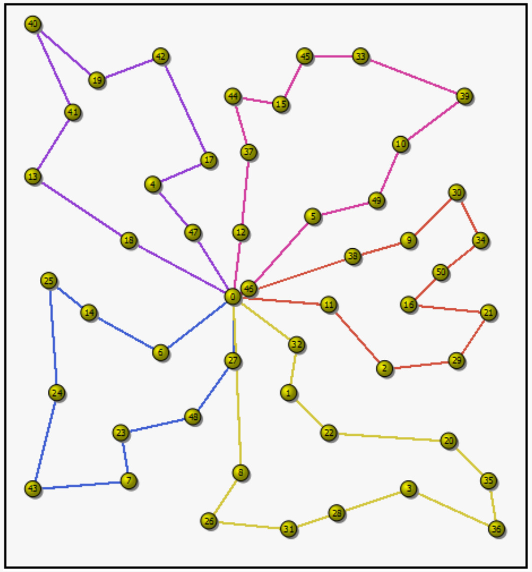
ICORES 2012

Vilamoura, Portugal, 4-6 February 2012

Presentation outline

1) Vehicle Routing Problems
2) Metaheuristics
3) Metaheuristics for VRPs
4) Unified Tabu Search (UTS)
5) The general heuristic of Pisinger and Ropke
6) A successful Hybrid Genetic Algorithm
7) Some computational results

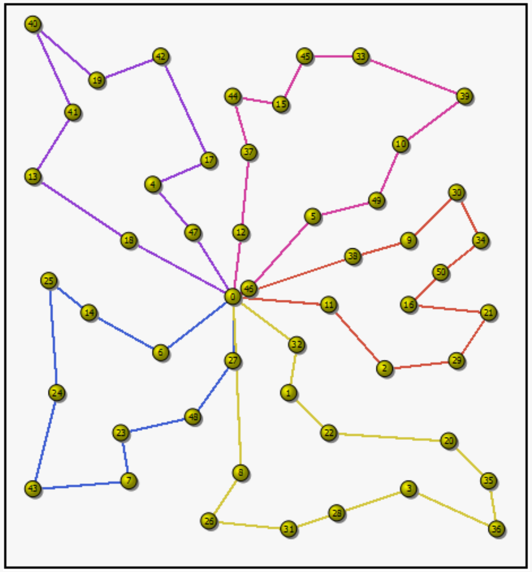
Vehicle Routing Problems (1/4)



- Vehicle Routing Problems are at the core of a huge number of practical applications in the area of the distribution of goods and services.

- One of the most studied classes of problems in applications of O.R. (about 40% of the papers published in *Transportation Science* since 2005, more than 40% of the submissions to *Odysseus 2012*).

Vehicle Routing Problems (2/4)

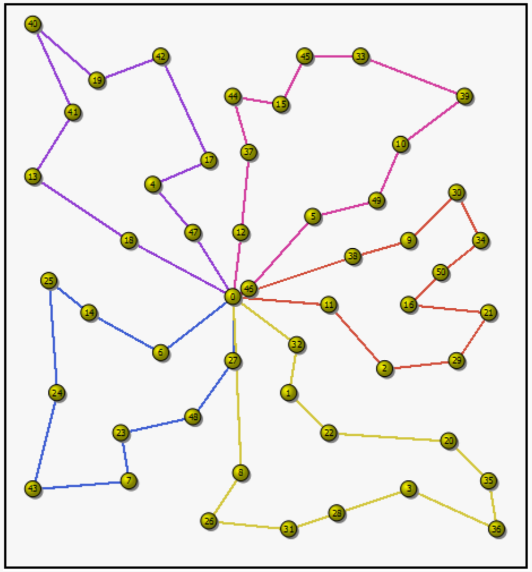


Basic problem (Classical VRP):

- Set of customers with known demands
- Single depot at which is based a fleet of identical vehicles.
- Each customer must be visited by a single vehicle.
- Capacity and route-length restrictions

Benchmark instances: set of 14 problems by Christofides *et al.* (50 to 199 customers) and some larger problems.

Vehicle Routing Problems (3/4)



Huge number of variants:

- Route length and duration
- Multi-Depot (MDVRP)
- Periodic (PVRP)
- Time-Windows
- Mixed Fleet
- Multi-Compartment
- Backhauls
- Pick-up and deliveries
- Location routing
- …

Vehicle Routing Problems (4/4)

- Most routing problems have been shown to be NP-hard and are effectively difficult to solve in practice, given the size of instances encountered in practical applications.
- While several exact methods have been developed over the last 25-30 years, heuristics still remain the method of choice for larger instances.
- Traditional heuristics
  - constructive (e.g., insertion methods) or
  - local improvement ones (e.g., r-opt)

  were the standard solution approach for routing and other tough combinatorial problems up to the early 80’s.

Traditional Local Improvement Heuristics

- **Principle:**
  - Start with a (feasible) initial solution.
  - Apply a sequence of local modifications to the current solution as long as these produce improvements in the value of the objective (monotone evolution of the objective).
- A big problem: these methods stop when they encounter a ***local optimum*** (w.r.t. to the allowed modifications).
- Solution quality (and CPU times) depends on the "richness" of the set of transformations (moves) allowed at each iteration of the heuristic.

Metaheuristics (1/5)

- 1983: Kirkpatrick, Gelatt and Vecchi publish their famous paper in *Nature* on ***simulated annealing***.
- A probabilistic local search algorithm capable of overcoming local optima and with convergence properties.
- Renewed interest for the development of new types of heuristics (***metaheuristics)***.

Metaheuristics (2/5)

- Concept introduced by Glover (1986)
- Generic heuristic solution approaches designed to **control** and **guide** specific problem-oriented heuristics
- Generally inspired from analogies with natural processes
- Rapid development over the last 25 years

Metaheuristics (3/5)

- Simulated Annealing (SA)
- Tabu Search (TS)
- Genetic Algorithms (GA)
- Evolutionary Algorithms (EA)
- Adaptive Memory Procedures (AMP)
- Variable Neighborhood Search (VNS)
- Threshold Acceptance methods (TA)
- Ant Colony Optimization (ACO)
- Greedy Randomized Adaptive Search Procedure (GRASP)
- Scatter Search (SS)
- Path Relinking (PR)
- and several others…

Metaheuristics (4/5)

One should distinguish between:

- Methods based on the improvement of a single incumbent solution (neighbourhood search methods)

  SA, TS, VNS, TA, …

- Methods based on the exploitation of a population of solutions

  GA, EA, ACO, SS, PR, AMP,…

➢ The most successful implementations often blend creatively ideas and principles from both families.

Metaheuristics (5/5)

- A history of **successes**… and **failures**
  - Metaheuristics have been applied successfully to a variety of difficult combinatorial problems encountered in numerous application settings.
  - Because of that, they have become extremely popular and are often seen as a panacea.

BUT…

  - There have also been many less-than-successful applications of metaheuristics.

Metaheuristics in VRP (1/3)

- The first attempts to apply metaheuristics to routing problems in the late 80’s were not very successful.
- The situation changed with the Tabu Search heuristics of Taillard (1993) and Gendreau, Hertz, and Laporte (1994).
- All types of metaheuristics were applied to routing problems over the last 20 years.
- Some of the most successful implementations include:
  - TS + Adaptive Memory (Rochat and Taillard, 1995)
  - TS + Adaptive Memory (Taillard et al., 1997)
  - Unified Tabu Search (Cordeau et al., 1997, 2001, 2004)
  - Granular Tabu Search (Toth and Vigo, 2003)
  - Adaptive Large Neighborhood Search (Pisinger and Ropke, 2007)

Metaheuristics in VRP (2/3)

- Up until the early 2000's, the few implementations of GA's for CVRPs proved extremely disappointing.
- Things changed radically with the work of Prins (2004).
  - Solution representation without trip delimiters
  - Solution is constructed by applying a specialized splitting algorithm (shortest path on an auxiliary graph)
  - Memetic algorithm (GA + Local search)
- The same approach has been applied since to several variants of the CVRP (Fleet Mix, Multi-depot, …)
  - Vidal et al. (2011)

Metaheuristics in VRP (3/3)

- Several successful implementations of population-based methods (as well as of other approaches) for VRPTW's:
  - Homberger and Gehring (1999, 2001, 2002, 2005)
  - Mester (2002)
  - Berger et al. (2003)
  - Mester (2006)
  - Repoussis et al. (2009)
  - Nagata et al. (2010)
  - …
- For more details, see the surveys by Bräysy and Gendreau (2005) and Gendreau and Tarantilis (2011).

Unified Tabu Search (1/3)

- Originally proposed in Cordeau et al. (1997) to solve Periodic and Multi-Depot VRPs.
- Extended in Cordeau et al. (2001) to handle problems with time windows (VRPTW, PVRPTW, MDVRPTW)
- Extended in Cordeau and Laporte (2001) to handle problems with site dependencies (SDVRP and SDVRPTW)
- Improved in Cordeau et al. (2004) for better handling of route duration constraints
- Imbedded in an iterated local search scheme, using a simple parallel computing framework, to take advantage of the multiple cores available on modern computers (2011).

Unified Tabu Search (2/3)

Key features of the algorithm

- Neighborhood search method with basic moves:
  - For problems without time windows, customers are added/ removed using the GENI heuristic of Gendreau et al. (1992).
  - For problems with time windows, customers are added by simple insertions and removed by simple reconnection.
- Search space includes infeasible solutions: capacity, route duration, and time window constraints are relaxed.
- Constraint violations are penalized in the objective through dynamically self-adjusting weights:
  - Value increased (resp. decreased) at the end of each iteration in which the related set of constraints is infeasible (resp. feasible).

Unified Tabu Search (3/3)

Key features of the algorithm (con'd)

- Local optima do not stop the exploration of the search space: search proceeds to the best neighbor.
- Cycling is prevented by forbidding reversal of recent moves recorded in a short-term memory (tabu list).
- Solutions are characterized by sets of attributes (e.g., the assignment of customers to routes or depots).
- Attributes are the basis for the definition of the tabus.
- Attributes are also a key element to implement a ***continuous diversification strategy*** through special terms that are added to the objective: these are based on the addition frequency of each attribute.

Adaptive Large Neighborhood Search (1/4)

- General routing algorithm proposed by Pisinger and Ropke (2007).
- The method can solve 5 different VRP variants:
  - vehicle routing problem with time windows (VRPTW)
  - capacitated vehicle routing problem (CVRP),
  - multi-depot vehicle routing problem (MDVRP),
  - site-dependent vehicle routing problem (SDVRP),
  - open vehicle routing problem (OVRP).
- Solution approach based on the ALNS framework presented in Ropke and Pisinger (2006) for solving the pickup and delivery problem with time windows.

Adaptive Large Neighborhood Search (2/4)

- An extension of the Large Neighborhood Search procedure proposed by Shaw (1998).
- ALNS is also based on the ***Ruin and Recreate*** paradigm presented by Schrimpf et al. (2000).
- In a typical iteration, part of the solution is destroyed and then reconstructed using sets of suitably defined destruction and repair operators.
- The new solution obtained is accepted according to a simulated annealing acceptance criterion.
- Operators are selected according to dynamically adjusted selection probabilities:
  - Probabilities change as a function of the effectiveness of operators.

Adaptive Large Neighborhood Search (3/4)

Adaptive Large Neighborhood Search

1 Construct a feasible solution x; set $x^* := x$

2 Repeat

3 Choose a destroy neighborhood N^- and a repair neighborhood N^+ using roulette wheel selection based on previously obtained scores $\{\pi_j\}$

4 Generate a new solution x' from x using the heuristics corresponding to the chosen destroy and repair neighborhoods

5 If x' can be accepted then set $x := x'$

6 Update scores π_j of N^- and N^+

7 If $f(x) < f(x^*)$ set $x^* := x$

8 Until stop criteria is met

9 Return x^*

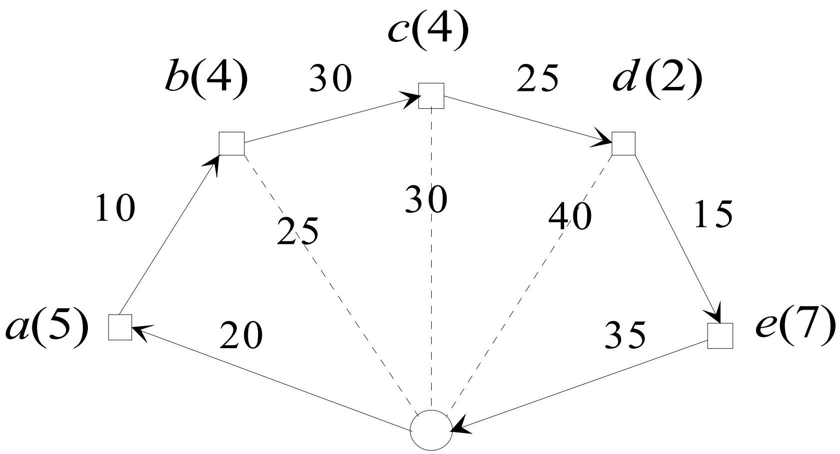
Adaptive Large Neighborhood Search (4/4)

- Important: The destruction operators should be a proper mix of operators that can intensify and diversify the search.
- Possible destruction operators:
  - Random removal
  - Worst or critical removal
  - Related removal (same route, same time, cluster, …)
  - History-based removal
- Reconstruction operators are typically based on existing well-performing heuristics for the problem at hand.
  - These heuristics can make use of variants of the greedy paradigm.

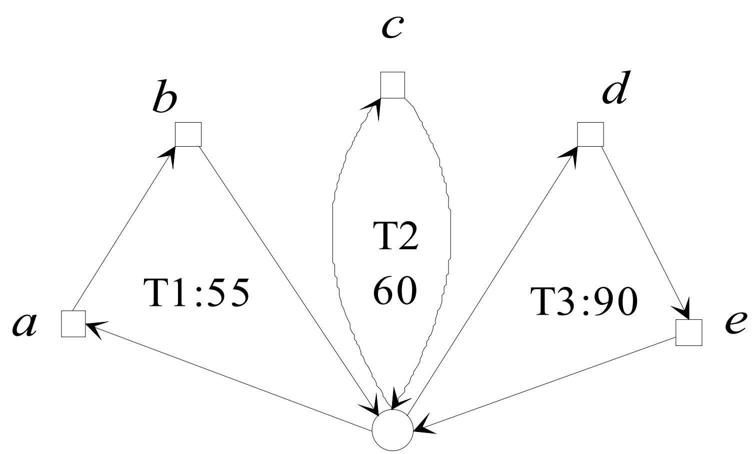
The Hybrid Genetic Algorithm of Vidal et al.

- Vidal, T., Crainic, T.G., Gendreau M., Lahrichi, N., Rei, W., “A hybrid genetic algorithm for multi-depot and periodic vehicle routing problems” (forthcoming)
- An algorithm designed to solve periodic, multi-depot VRP, as well as MDVRP and PVRP.
- Basic version without time windows.
- Based on Prins’ memetic algorithm.

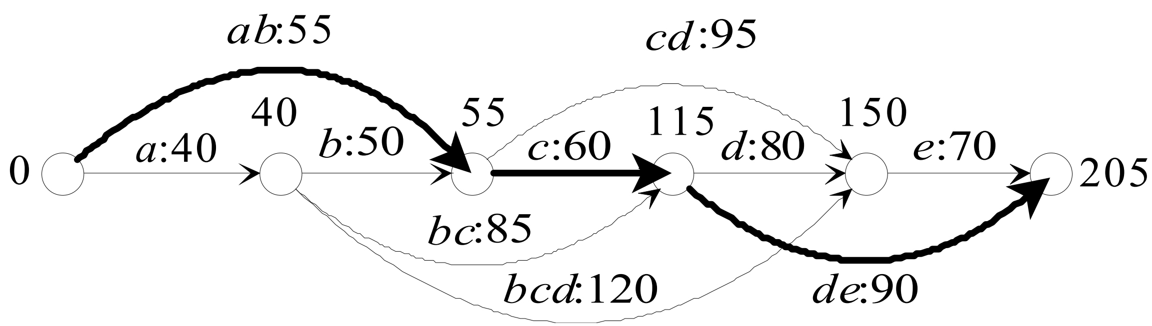
Evaluation: procedure SPLIT



Chromosome *S* = (*a*,*b*,*c*,*d*,*e*)



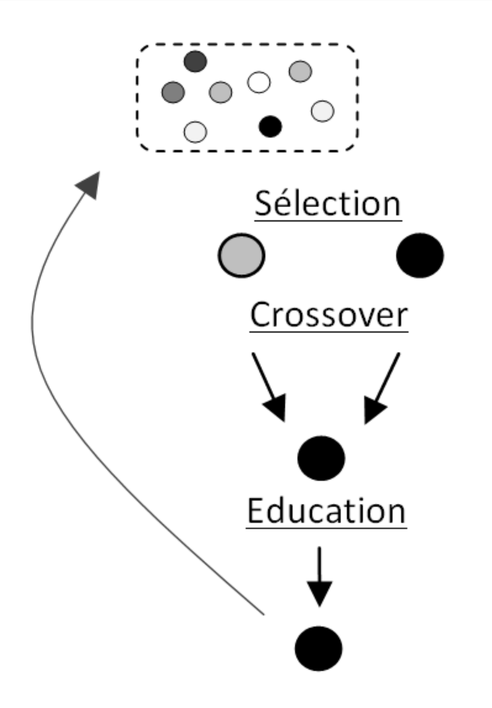
Optimal splitting, cost 205



Auxiliary graph of possible trips for *W*=10 and shortest path in boldface
(Bellman's algorithm for directed acyclic graphs)

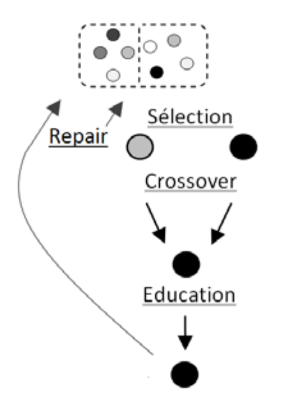
Hybrid genetic algorithm for the MDPVRP (1/4)

- Existing Hybrid GA's for VRP, VRPTW, MDVRP
  - Few work on periodic problems



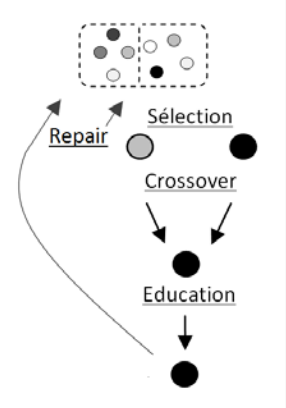
- General Methodology:
  - Evolving a population of solutions by means of genetic operators such as *selection*, *crossover* and *mutation*.
  - Survival of the fittest drives the population towards good solutions.
  - To speed up the evolution, random mutation replaced by a local search based *education* operator.

Hybrid genetic algorithm for the MDPVRP (3/4)



- Double population management:
  - A feasible individual goes in the feasible subpopulation
  - An infeasible individual → included in the infeasible subpopulation → probability P_{rep} to be repaired & added into the feasible one
- Each subpopulation → (μ+λ) strategy where any new offspring is directly included (and thus can reproduce):
  - μ individuals initially
  - Each new individual is included in the population
  - As a population reaches the size (μ+λ), selection of survivors to discard λ individuals
- Good properties :
  - Profit from new individuals, including those with bad fitness
  - Preserve an elite

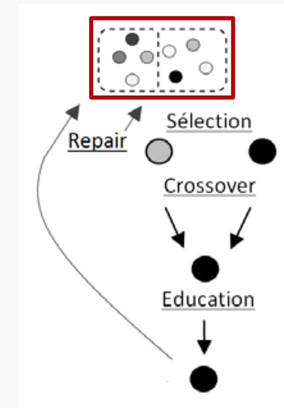
Hybrid genetic algorithm for the MDPVRP (2/4)



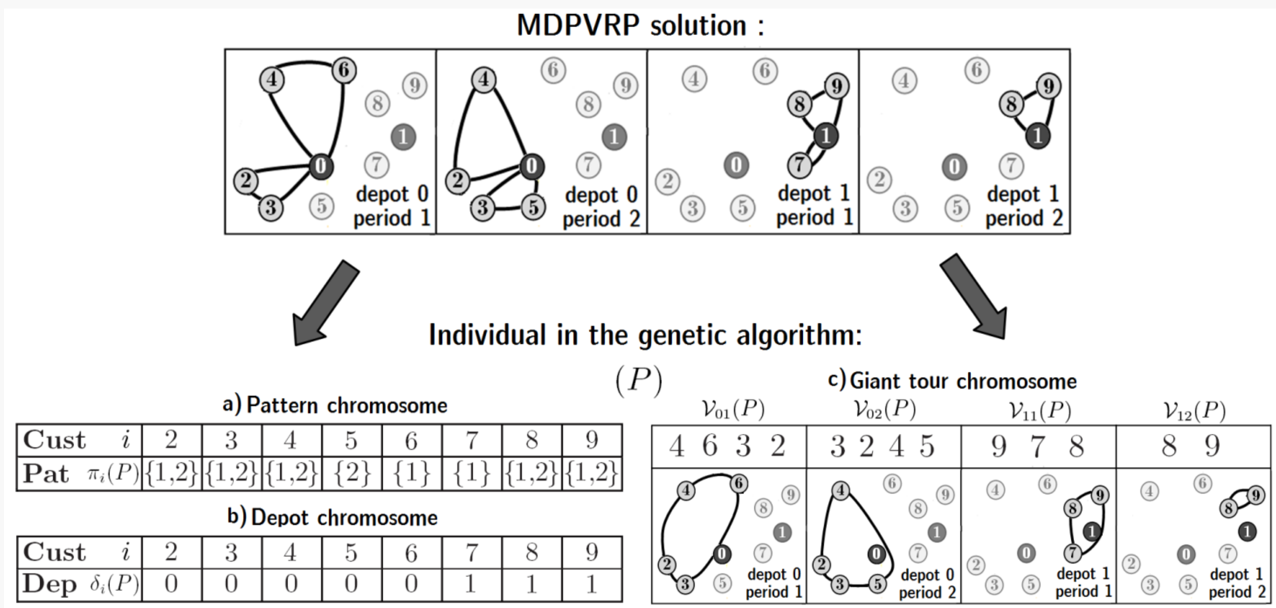
- Search Space:
  - Accepting infeasible solutions not respecting route related constraints : load or duration
  - Always respect the number of vehicles

- Adaptive penalties:
  - Amount of infeasible solutions is monitored; penalties are adjusted during run time to obtain about 20% feasible solutions following education
  - Repair operator to obtain more feasible solutions
  - Double population to manage feasible and infeasible individuals

Hybrid genetic algorithm for the MDPVRP (4/4)



- Solution representation
  - Representation as a giant TSP tour without trip delimiters (Prins 2004)
  - In MDPVRP context, a tour for each couple (day, depot)
  - Polynomial « Split » algorithm to obtain the best segmentation of each sequence into routes



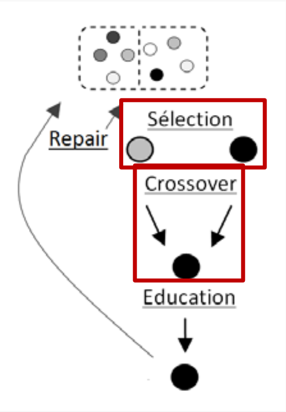
a) Pattern chromosome

Cust i	2	3	4	5	6	7	8	9
Pat $\pi_i(P)$	{1,2}	{1,2}	{1,2}	{2}	{1}	{1}	{1,2}	{1,2}

b) Depot chromosome

Cust i	2	3	4	5	6	7	8	9
Dep $\delta_i(P)$	0	0	0	0	0	1	1	1

New Crossover operator for the MDPVRP (1/3)



- Parent selection by *binary tournament*

- New *Periodic Crossover with insertions*:
  one offspring inherits information from both parents

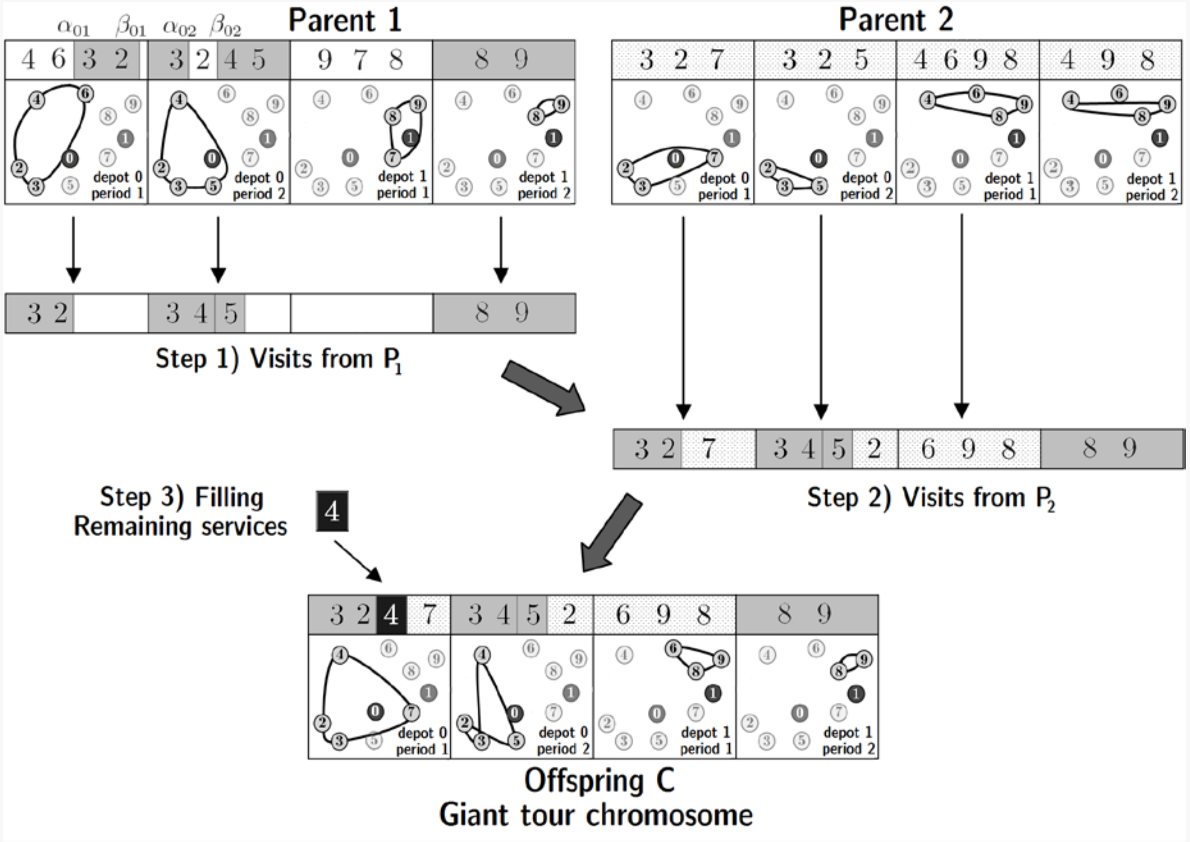
1) Choose for each day which parent (or both parents) provide the genetic material
2) Transmit the genetic information from the first parent
3) Complete with information from the second parent
4) Eventually fill the remaining required visits

New Crossover operator for the MDPVRP (2/3)

- For each couple (day, depot) choosing randomly the amount of information transmitted from parent 1 :
  - Copy the whole sequence of services for this couple,
  - **or** Do not copy any information for this couple,
  - **or** Copy a substring

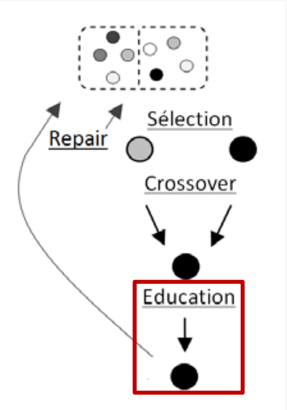
- In a random order of (day, depot), visits are added from parent 2. A visit is copied only if:
  - The entire sequence of parent 1 has not been copied for this couple
  - The insertion is compatible with at least one pattern of the customer

New Crossover operator for the MDPVRP (3/3)



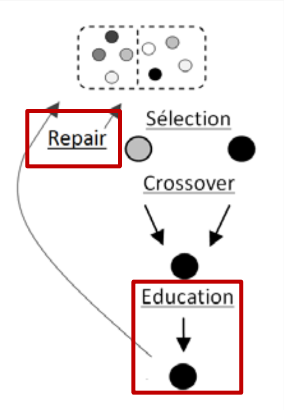
- After this process, some customers can have an “incomplete pattern”:
  - Remaining visits are added after the split algorithm, using a minimum cost insertion criteria.

Education operator (1/2)



- Two level local search:
  - *Route Improvement* (RI) dedicated to improve the routes by moving customer or depot visits (nodes).
    For each node v_1 in random order and each node v_2 in random order, we test *insertion, swap, 2-opt, 2-opt** involving v_1 and v_2 (some restrictions if v_1 is a depot).
  - *Pattern Improvement (PI)* = calculate for each route in each (day/depot) the insertion cost of a customer → evaluate the cost of a pattern change and operate if negative.
  - First improvement rule. Stops when all moves have been tested without success.
  - Called in sequence RI-PI-RI.

Education operator (2/2)



- Speeding-up the local search:
  - *Granular search*: Testing only moves in RI involving correlated nodes (X% closer in terms of distance)
  - *Memories*: Remembering the insertion costs in PI. During RI: remembering for each couple (node, route) if the route has changed since last cycle of moves involving the node.

- Repair = increasing temporarily the penalty values and use Education.

Promotion of diversity (1/2)

- Diversity management is crucial to evade premature convergence and obtain high quality solutions.

- Previous methods to maintain diversity:
  - Prins (2004): dispersal rule based on fitness during insertion in the population
  - Sörensen et Sevaux (2006) « *Memetic Algorithm with Population Management (MA|PM)*»: dispersal rule based on a distance measure

- We go a step further, and introduce a *promotion of diversity* during the very evaluation of individuals
  - *Hybrid Genetic Search with Adaptive Diversity Management* (HGSADC)

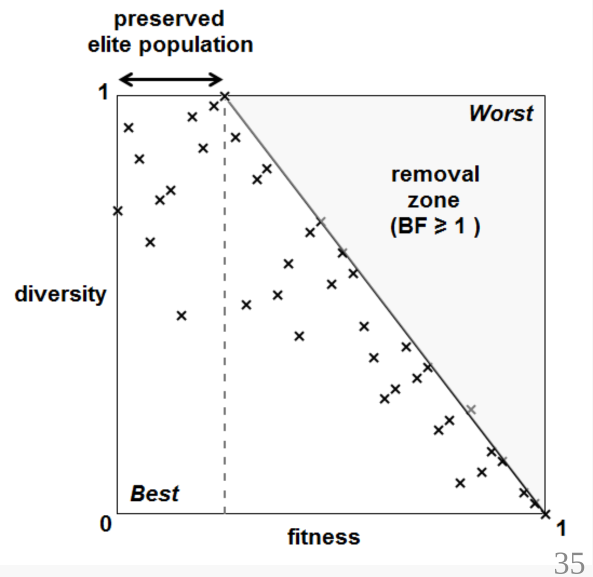
Promotion of diversity (2/2)

- Our approach for individual evaluation:

  Biased Fitness is a tradeoff between ranks in terms of **solution cost** *cost(I),* and **contribution to the diversity** *dc(I),* measured as a distance to others individuals in the population.

- During selection of the parents:
  - Balance strength with innovation during reproduction, and thus favors exploration of the search space. -> Increased level of diversity in the population.

- During selection of the survivors:
  - Removing the individual *I* with worst *BF(I)* also guarantees some elitism in terms of solution value.



Experimental setup

- Problem benchmarks:
  - Cordeau, Gendreau, Laporte (1998) instances for PVRP and MDVRP
  - New instances for MDPVRP derived from the previous benchmarks
  - CVRP instances of Christofides et al. (1979) and Golden et al. (1998)
  - Instances ranging from 48 to 483 customers, up to a planning horizon of 10 days, and 6 depots. Up to about 900 total services for some periodic problems.
- Experiments conducted on a 2.4 Ghz AMD Opteron 250 CPU
- Conversion of run-times using Dongarra factors, to compare with other authors
- Meta-calibration of parameters
  - Done using the *Evolutionary Strategy with Covariance Matrix Adaptation* (CMA-ES) of Hansen and Ostermeier (2001)

Results on PVRP instances (1/2)

➢ State of the art algorithms then and now. We compare deviations to Best Known Solutions (BKS) :

- Cordeau, Gendreau, Laporte (CGL-97): Tabu Search
- Hemmelmayr, Doerner, Hartl (HDH-09): Variable Neighborhood Search
- Gulczynski, Golden,Wasil (GGW-11): Integer programming + record-to-record travel

Benchmark	Best approach in 1997	Best approach in 2011	HGSADC
PVRP "old" set	Cordeau et al. (1997) Dev. to BKS : +1.62%	Gulczynski et al. (2011) +0.94%	**+0.14%**
PVRP "new" set	Cordeau et al. (1997) +2.48%	Hemmelmayr et al. (2009) +1.53%	**+0.38%**
Nb. customers > 150	Cordeau et al. (1997) +3.23%	Hemmelmayr et al. (2009) +2.16%	**+0.35%**

Results on PVRP instances (2/2)

- Behavior as the termination criterion increases:

	CGL-97 15.10^3 it	HDH-09 10^7 it	HDH-09 10^8 it	HDH-09 10^9 it	HGSADC 10^4 it	HGSADC 2.10^4 it	HGSADC 5.10^4 it
T	3.96 min	3.09 min	*30 min*	*300 min*	5.56 min	13.74 min	28.21 min
%	+1.82%	+1.45%	+0.76%	+0.39%	+0.20%	+0.12%	+0.07%

- All best known solutions have been retrieved, including 15 optimal results from Baldacci et al. (2010)
- Many have been improved → 19 new BKS
- Small standard deviation : ≈ 0.13% for the previous results

Results on MDVRP instances (1/2)

- State of the art algorithms then and now:
  - Cordeau, Gendreau, Laporte (CGL-97) : Tabu Search
  - Pisinger and Ropke (PR-07) : Adaptive Large Neighborhood Search

Benchmark	Best approach in 1997	Best approach in 2011	HGSADC
MDVRP "old" set	Cordeau et al. (1997) +0.58%	Pisinger and Ropke (2007) +0.35%	**+0.00%**
MDVRP "new" set	Cordeau et al. (1997) +1.85%	Pisinger and Ropke (2007) +0.34%	**-0.04%**
Nb. customers > 150	Cordeau et al. (1997) +1.40%	Pisinger and Ropke (2007) +0.45%	**-0.03%**

Results on MDVRP instances (2/2)

- Results with different running times:

	CGL 15.10^3 it	RP 25.10^3 it	RP 50.10^3 it	HGSADC 10^4 it	HGSADC 2.10^4 it	HGSADC 5.10^4 it
T	---	1.97 min	3.54 min	2.24 min	8.99 min	19.11 min
%	+0.96%	+0.52%	+0.34%	-0.01%	-0.04%	-0.06%

- All best known solutions have been retrieved, including 5 optimal results from Baldacci and Mingozzi (2009)
- Many have been improved → 9 new BKS
- Very small standard deviation : ≈ 0.03%

Results on MDPVRP instances

➢ New instances → Compare to our BKS from multiple long runs

Inst	n	d	t	Average	Gap %	T (min)	BKS
p01	48	4	4	**2019.07**	0%	0.35	2019.07
p02	96	4	4	**3547.45**	0%	1.49	3547.45
p03	144	4	4	4491.08	0.12%	7.72	4480.87
p04	192	4	4	5151.73	0.23%	22.10	5141.17
p05	240	4	4	5605.60	0.49%	30	5570.45
p06	288	4	4	6570.28	0.36%	30	6524.42
p07	72	6	6	4502.06	0.04%	2.18	4502.02
p08	144	6	6	6029.58	0.43%	7.96	6023.98
p09	216	6	6	8310.19	0.90%	27.79	8257.80
p10	288	6	6	9972.35	1.86%	30	9818.42
					+0.42%	15.96 min	

- Good overall gap for a hard problem, a relatively small standard deviation of ≈ 0.30%
- One could investigate cooperation schemes to increase performance

Results on CVRP instances

- Excellent results on Christofides et al. (1979), and Golden et al. (1998) CVRP instances.
  - Average gap of 0.11% comparable to 0.10% for Nagata and Bräysy (2010), which is the best actual state-of-the-art method, specially tailored for CVRP.
  - All BKS have been retrieved, 12 BKS improved

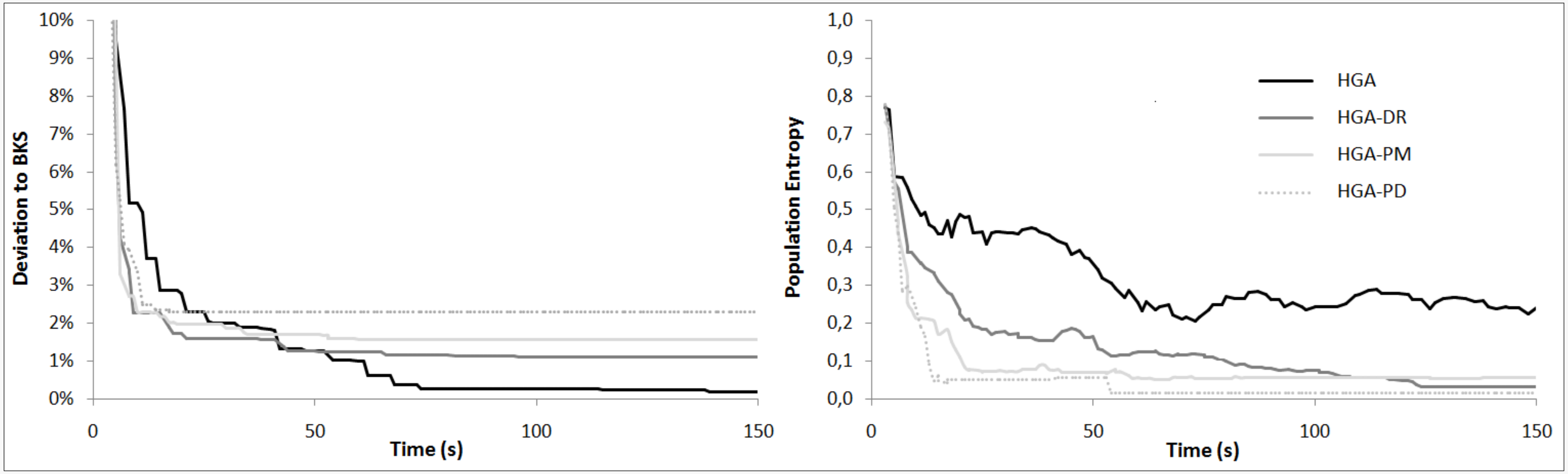
Empirical studies on diversity management methods (1/2)

- Several diversity management methods, average results:
  - **HGA** : No diversity management method
  - **HGA-DR** : Dispersal rule on objective space
  - **HGA-PM** : Dispersal rule on solution space
  - **HGSADC** : The proposed approach

Benchmark		**HGA**	**HGA-DR**	**HGA-PM**	**HGSADC**
PVRP	T	6.86 min	7.01 min	7.66 min	8.17 min
	%	+0.64%	+0.49%	+0.39%	+0.13%
MDVRP	T	7.93 min	7.58 min	9.03 min	8.56 min
	%	+1.04%	+0.87%	+0.25%	-0.04%
MDPVRP	T	25.32 min	26.68 min	28.33 min	40.15 min
	%	+4.80%	+4.07%	+3.60%	+0.44%

Empirical studies on diversity management methods (2/2)

- Behavior of HGSADC during a random run:
  - Higher entropy (average distance between two individuals)
  - Better final solution
  - Diversity can increase during run time



Results on VRPTW (Bräysy and Gendreau, 2005)

Table 2 Comparison of Tabu Search Algorithms

Authors	R1	R2	C1	C2	RC1	RC2	CNV/CTD
Garcia et al. (1994)	12.92	3.09	10.00	3.00	12.88	3.75	436
	1,317.7	1,222.6	877.1	602.3	1,473.5	1,527.0	65,977
Rochat and Taillard (1995)	12.25	2.91	10.00	3.00	11.88	3.38	415
	1,208.50	961.72	828.38	589.86	1,377.39	1,119.59	57,231
Potvin and Bengio (1996)	12.50	3.09	10.00	3.00	12.63	3.38	426
	1,294.5	1,154.4	850.2	594.6	1,456.3	1,404.8	63,530
Taillard et al. (1997)	12.17	2.82	10.00	3.00	11.50	3.38	410
	1,209.35	980.27	828.38	589.86	1389.22	1,117.44	57,523
Chiang and Russell (1997)	12.17	2.73	10.00	3.00	11.88	3.25	411
	1,204.19	986.32	828.38	591.42	1,397.44	1,229.54	58,502
De Backer and Furnon (1997)	14.17	5.27	10.00	3.25	14.25	6.25	508
	1,214.86	930.18	829.77	604.84	1,385.12	1,099.96	56,998
Brandão (1999)	12.58	3.18	10.00	3.00	12.13	3.50	425
	1,205	995	829	591	1,371	1,250	58,562
Schulze and Fahle (1999)	12.25	2.82	10.00	3.00	11.75	3.38	414
	1,239.15	1,066.68	828.94	589.93	1,409.26	1,286.05	60,346
Tan et al. (2000)	13.83	3.82	10.00	3.25	13.63	4.25	467
	1,266.37	1,080.24	870.87	634.85	1,458.16	1,293.38	62,008
Lau et al. (2001)	14.00	3.55	10.00	3.00	13.63	4.25	464
	1,211.54	960.43	832.13	612.25	1,385.05	1,232.65	58,432
Cordeau et al. (2001)	12.08	2.73	10.00	3.00	11.50	3.25	407
	1,210.14	969.57	828.38	589.86	1,389.78	1,134.52	57,556
Lau et al. (2003)	12.17	3.00	10.00	3.00	12.25	3.38	418
	1,211.55	1,001.12	832.13	589.86	1,418.77	1,170.93	58,477

Note. For each algorithm, the average results with respect to Solomon's benchmarks are depicted. The notations CNV and CTD in the rightmost column indicate the cumulative number of vehicles and cumulative total distance over all 56 test problems.

Results on VRPTW (Bräysy and Gendreau, 2005)

Table 4 Comparison of Evolutionary and Genetic Algorithms

Authors	R1	R2	C1	C2	RC1	RC2	CNV/CTD
(1) Thangiah (1995a)	12.75	3.18	10.00	3.00	12.50	3.38	429
	1,300.25	1,124.28	892.11	749.13	1,474.13	1,411.13	65,074
(2) Potvin and Bengio (1996)	12.58	3.00	10.00	3.00	12.13	3.38	422
	1,296.83	1,117.64	838.11	590.00	1,446.25	1,368.13	62,634
(3) Berger et al. (1998)	12.58	3.09	10.00	3.00	12.13	3.50	424
	1,261.58	1,030.01	834.61	594.25	1,441.35	1,284.25	60,539
(4) Homberger and Gehring (1999)	11.92	2.73	10.00	3.00	11.63	3.25	406
	1,228.06	969.95	828.38	589.86	1,392.57	1,144.43	57,876
(5) Gehring and Homberger (1999)	12.42	2.82	10.00	3.00	11.88	3.25	415
	1,198	947	829	590	1,356	1,140	56,942
(6) Gehring and Homberger (2001)	12.00	2.73	10.00	3.00	11.50	3.25	406
	1,217.57	961.29	828.63	590.33	1,395.13	1,139.37	57,641
(7) Berger et al. (2003)	11.92	2.73	10.00	3.00	11.50	3.25	405
	1,221.10	975.43	828.48	589.93	1,389.89	1,159.37	57,952
(8) Tan et al. (2001a)	13.17	5.00	10.11	3.25	13.50	5.00	478
	1,227	980	861	619	1,427	1,123	58,605
(9) Tan et al. (2001b)	12.91	5.00	10.00	3.00	12.60	5.80	471
	1,205.0	929.6	841.96	611.2	1,392.3	1,080.1	56,931
(10) Wee Kit et al. (2001)	12.58	3.18	10.00	3.00	12.75	3.75	432
	1,203.32	951.17	833.32	593.00	1,382.06	1,132.79	57,265
(11) Mester (2002)	12.00	2.73	10.00	3.00	11.50	3.25	406
	1,208	954	829	590	1,387	1,119	57,219
(12) Jung and Moon (2002)	13.25	5.36	10.00	3.00	13.00	6.25	486
	1,179.95	878.41	828.38	589.86	1,343.64	1,004.21	54,779
(13) Le Bouthillier and Crainic (2005)	12.17	2.82	10.00	3.00	11.50	3.25	409
	1,209.27	965.91	828.38	589.86	1,389.22	1,143.70	57,574
(14) Homberger and Gehring (2005)	11.92	2.73	10.00	3.00	11.50	3.25	405
	1,212.73	955.03	828.38	589.86	1,386.44	1,123.17	57,309

Note. For each algorithm, the average results with respect to Solomon's benchmarks are reported. Notations CNV and CTD in the rightmost column indicate the cumulative number of vehicles and cumulative total distance over all 56 test problems.

Results on large VRPTW (Gendreau and Tarantilis, 2012)

Table 8: Assessment of advanced heuristics for large-scale VRPTWs

Reference	Effectiveness	Efficiency	Simplicity	Flexibility
HY	Very High	High	High	Very High
GH99	Medium	Very High	Medium-High	Medium
GH02	Medium-High	Very High	Medium-High	Medium
HG	Medium-Low	Very High	Medium-High	Medium
BHD	Medium-Low	Very High	Very High	High
MB	Medium-High	High	High	High
MBD	Medium-High	Medium	High	High
LC	Medium	Very High	Medium-Low	Medium
LCK	Medium-High	Very High	Medium-Low	Medium-High
BVH	Medium	-	Medium-High	High
I	High	High	Medium-High	High
LZ	High	Low	Medium-Low	Medium-Low
RTI	Very High	Medium	Medium-High	Medium-High
HYI	High	Medium	Medium	High
PR	High	High	Very High	Very High
B	Medium-High	High	High	High
HKI	Very Low	Medium	High	Medium-Low
NBD	Very High	High	Medium-High	High
DPR	High	Low	Medium-Low	Medium-High

Conclusions

- Over the last 25 years, metaheuristics have proven themselves to be the most effective methods for tackling vehicle routing problems, especially large instances.
- The most recent methods are capable of tackling complex problems with a large number of attributes.
- The most effective methods are now often hybrids combining the basic features of several pure methods.
- Population management seems critical in population-based methods.
- Given the recent developments in MIP solvers, methods combining metaheuristic concepts with exact solution techniques (matheuristics) may prove the most promising to explore.