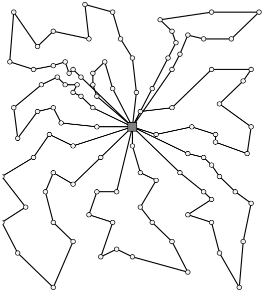
Vehicle Routing Heuristics: A Quick Tour d'Horizon

Thibaut Vidal

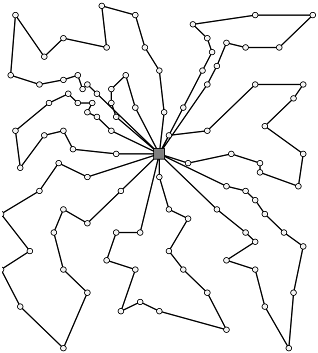
CIRRELT & SCALE-AI Chair in Data-Driven Supply Chains
Department of Mathematics and Industrial Engineering, Polytechnique Montréal



EURO Meets NeurIPS 2022
Vehicle Routing Competition
December 7^{th}, 2022

Capacitated Vehicle Routing Problem (CVRP)

- Capacitated Vehicle Routing Problem (CVRP)
  - **INPUT :** n customers, with locations and demand quantity. All-pair distances. Homogeneous fleet of m vehicles with capacity Q located at a central depot.
  - **OUTPUT :** Least-cost delivery routes (at most one route per vehicle) to service all customers.



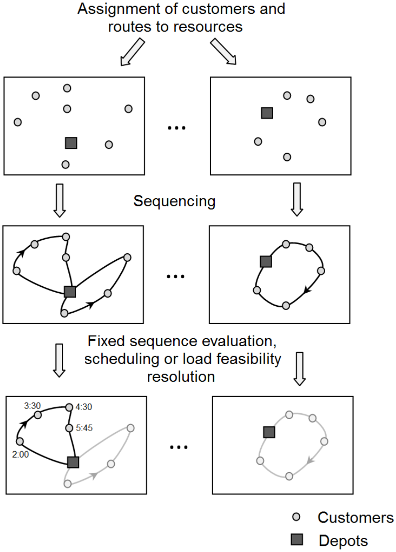
-
  - NP-Hard problem
  - Recent exact methods can reliably solve problems of moderate size with 200-300 customers.
  - Extensive research on (meta-)heuristics for several decades
  - Long stream of comparisons on common benchmark instances permitting to gauge the strengths of proposed methods: Uchoa et al. (2017) and Queiroga et al. (2021)

Multi-attribute vehicle routing problems (MAVRPs)

- **VRP "attributes":** Supplementary decisions, constraints and objectives combined with the classic VRP (Vidal et al., 2013)
  - **Practical objectives:** Profitability, equity, service Levels, persistence, compactness, robustness, externalities
  - **Integrated problems:** Multiple periods, depots, echelons, fleet mix, location routing, inventory-routing, production-routing, synchronization...
  - **Fine-grained modeling:** **<u>Time windows</u>** (can even be soft or multiple), loading constraints (2D,3D), driver skills, time-dependent travel times, charging stations, engine modes, drones etc...

Multi-attribute vehicle routing problems (MAVRPs)

- Vehicle routing problem attributes generally impact different decisions and tasks:
- **ASSIGNMENT** (assignment of customers and routes to time-periods or depots)
  - *multi-period, multi-depot, heter. fleet, location routing...*
- **SEQUENCING** (choice of the sequence of visits)
  - *P&D, Backhauls, 2-echelon...*
- **ROUTE EVALUATIONS** (route feasibility/cost & other decisions)
  - *Time windows, time-dep travel time, loading constraints, HOS regulations, lunch breaks, load-dependent costs...*



Multi-attribute vehicle routing problems (MAVRPs)

- Challenges: **VARIETY** and **COMBINATION** of attributes
- *Over 200 attributes* have been proposed to date...

  ...also combined together $\Rightarrow$ 2^{200} problems... 2^{200} methods... 2^{200} papers ?!!!
- **Combinatorial explosion**: Combinatorial optimization problem + combinatorial *family of problems*

$\Rightarrow$ Need for unified solution concepts and methods (i.e., *generalization* capabilities)

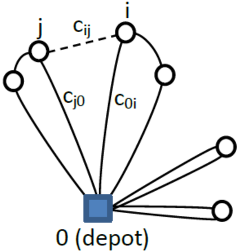
$\Rightarrow$ Solution methods that can address a wide range of problems without need for extensive adaptation or user expertise.

$\Rightarrow$ Necessary for faster application to industrial settings.

Constructive Algorithms

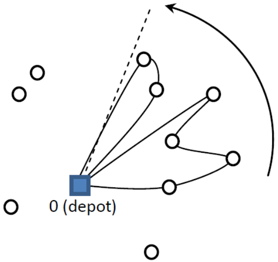
- **Constructive methods:** mostly between 1960s and 1980s.
  - Step-by-step definitive decisions which cannot be revoked afterward
  - Performance ranging typically around 8-15% error gap on common benchmark instances

- Savings method (Clarke and Wright 1964)
  - Merge routes step by step based on a savings measure s_{ij} $s_{ij} = c_{i0} + c_{0j} - c_{ij}$
  - Refinements by Gaskell (1967), Yellow (1970), and Solomon (1987).



Constructive Algorithms

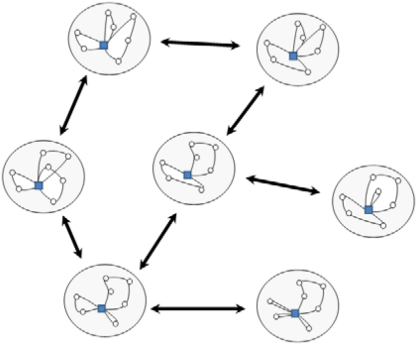
- Sweep algorithm (Gillett and Miller, 1974)
  - Sweep the deliveries in circular order to create routes.
  - A new route is initiated each time the capacity is exceeded.



- Route first cluster second (see, e.g., Beasley, 1983)
  - constructs a giant circuit (TSP tour) that visits all customers.
  - Segment this tour into several routes. An optimal segmentation (Split) can be found by solving a shortest path problem in an auxiliary directed acyclic graph
  - Possible to model and solve the Split problem in $\mathcal{O}(n)$ for the CVRP (Vidal, 2016)

Classical Local Searches

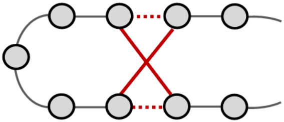
- **Local-improvement procedures :**
  - From an *incumbent solution* s define a *neighborhood* $N(s)$ of solutions obtained by applying some changes.
  - The set of solutions, linked by neighborhood relationships = search space.
  - LS-improvement method progress from one solution to another in this search space as long at the cost improves.



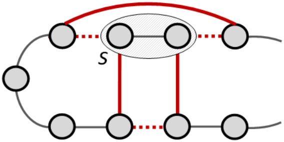
- Performance of multi-start LS ranging typically around 4-8% error gap on common benchmark instances

Classical Local Searches

- For optimizing a single route (TSP tour);
  - in the terminology of Lin (1965), λ-OPT neighborhood = subset of moves obtained by deleting and reinserting λ arcs.
  - 2-OPT and 3-OPT are commonly used,
  - OR-OPT: relocates a sequence of visits, is a subset of 3-OPT.



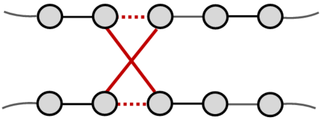
2-opt



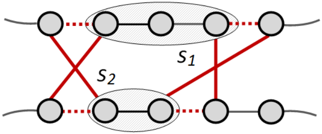
Or-exchange

Classical Local Searches

- For optimizing multiple routes together,
  - RELOCATE neighborhood (relocate a delivery)
  - SWAP neighborhoods (swap two deliveries from different routes)
  - CROSS (exchanges two sequences of visits)
  - I-CROSS (exchange and reverse two sequences)
  - 2-OPT* exchange two route tails (special case of CROSS)



2-opt*



CROSS

Classical Local Searches

- These neighborhoods contain a polynomial number of moves.
  - For all moves except CROSS and I-CROSS, the number of neighbors is $O(n^2)$
  - CROSS and I-CROSS are often limited to sequences of a few customers (e.g., up to k) and explored in $O(k^2n^2)$.
- Non-enumerative large-scale neighborhoods:
  - Heuristic of Lin and Kernighan (1973) – efficient implementation from Helsgaun (2000) generalized into LKH3 for constrained vehicle routing problems (Helsgaun, 2017);
  - Ruin-and-recreate (Shaw, 1998; Schrimpf et al., 2000);
  - Ejection chains (Glover, 1996)
- Pattern mining can also be used to identify useful higher-order moves (Arnold et al., 2021)

Classical Local Searches

- Efficient move evaluations and pruning procedures are critical to efficiently solve large-scale problem instances
  - Neighborhood restrictions, granular search (Johnson and Mcgeoch, 1997; Toth and Vigo, 2003): restrain the subset of moves to spatially related customer (but so far, limited success in our experience when trying to use a learned sparsified graph in this context – Santana et al. 2022)
  - Sequential search (Christofides and Eilon, 1972; Irnich and Villeneuve, 2003): any profitable move can be broken down into a list of arc exchanges $(a_1, \dots, a_\lambda)$ with gains $(g_1, \dots, g_\lambda)$ such that for any $k \in \{1, \dots, \lambda\}$, $g_1 + \cdots + g_k \geq 0$.
    $\Rightarrow$ This condition allows the pruning of many non-promising moves.

Metaheuristics

- Local-improvement methods lead to local optima $\Rightarrow$ "Metaheuristics" are various general principles that permit to escape and guide the search towards better solutions

- We can separate two groups of methods:
  - **Individual-trajectory** search: Iterative improvements on one incumbent solution – Tabu search, Simulated annealing, ILS, VNS...
  - **Population-based** search: Improving a population of solutions – Hybrid GA, evolutionary algorithms, ACO, path relinking...

- Hybrid methods mixing different search principles are also very common.

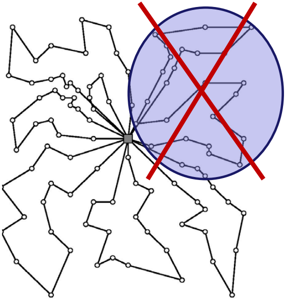
Unified Tabu Search – Cordeau et al. (1997, 2001)

- Tabu search – usually chooses the **best move** at each step (possibly non-improving).
- Neighborhood: RELOCATE
- Short-term tabu memories to avoid cycling:
  - Moving Client i from route R_1 to $R_2 \Rightarrow$ Not allowed to insert i back into route R_1 for X iterations.
- Longer term **diversification strategies:**
  - Penalizing recurrent solution attributes in the objective function
  - Penalized infeasible solutions (excess load or duration)

TS with adaptive memory – Rochat and Taillard (1995)

- Early study from 1995, already contained many key strategies:
- **Diversification**
  - Tabu search based on SWAP and RELOCATE moves
  - *Probabilistic* selection of moves driven by measures of attractiveness
- **and Intensification**:
  - Detection of good fragments of solutions that consistently appear in elite solutions and creation of new solutions from these fragments to obtain new starting points
  - Decomposition phases based on spatial proximity
  - Exact solution of the TSPs at regular intervals

- **Large neighborhoods** based on the *ruin-and-recreate* principle (Shaw, 1998; Schrimpf et al., 2000).
- Variety of operators to **partially destroy** the solutions
  - Based on randomness, cost metrics, relatedness, history...
  - Adaptive probabilities for operator selection
- Variety of operators to **reconstruct** the solutions
- Deteriorating solutions are accepted with some probability, as in a simulated annealing



HILS – Subramanian et al. (2013)

- **Iterated local search:** local search until a local optimum is encountered, **perturbation** and local search again etc...

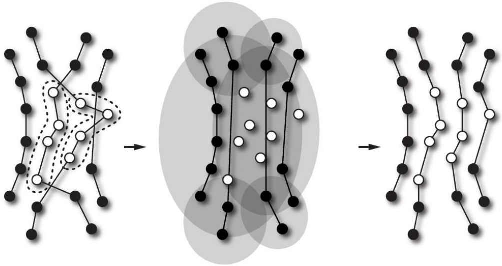
- Several neighborhoods are used
  - RELOCATE and SWAP of one to three customers in different routes, 2-OPT, 2-OPT*, empty-route, swap depot...
  - Several perturbation operations: multi-swap, multi-shift, double-bridge ...
  - Set covering model to create new solutions out of a set of high-quality routes.

Knowledge-Guided LS – Arnold and Sörensen (2019)

- Based on a "guided local search"
  - **Detect and temporarily penalize *bad* edges**
  - Characterization of bad edges result from a prior study on features of good (and bad) solutions
- Three types of neighborhoods encompassing most of the classic moves
  - CROSS-exchanges (includes RELOCATE and SWAP)
  - Ejection chains
  - Heuristic of Lin and Kernighan (1973)

SISRs – Christiaens and Vanden Berghe (2020)

- Ruin-and-recreate principle revisited
- Single ruin operator (adjacent string removals) aiming to introduce "capacity and spatial slack"
- Single recreate method (a randomized greedy insertion)

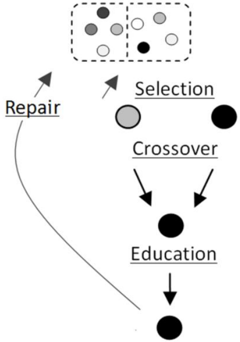


- Good performance on classical CVRP instances (Uchoa et al., 2017) as well as for other problem variants: VRPTW, PDPTW...

Hybrid GA – Prins (2004)

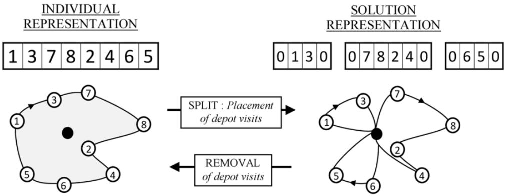
- **First Genetic Algorithm (GA)** to achieve competitive results on some VRP variants.

- **Genetic algorithms** mimic natural evolution
  - Population of solutions
  - Selection
  - Crossover
  - Mutation
    (replaced by a local search)



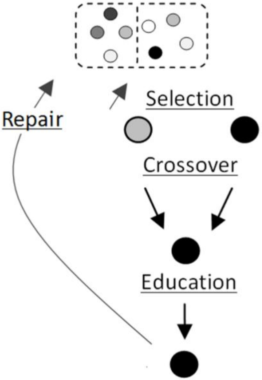
Hybrid GA – Prins (2004)

- The algorithm of Prins (2004) includes key design choices that made GA a practical approach at that time:
- **Giant-tour** solution representation
  - As there is a polynomial dynamic-programming *Split* algorithm to obtain a complete solution from it
  - Permits the use of a simple Ordered Crossover (OX)
- **Local search** on the offspring
- **Population management** (spacing constraint)



UHGS – Vidal et al. (2012, 2014)

- Hybrid genetic search (HGS) is still based on a giant-tour solution representation, and:
  - Efficient local search using neighborhood restrictions (granular search)
  - Adaptive management of penalized infeasible solutions
  - Active promotion of diversity in the population through a biased fitness measure

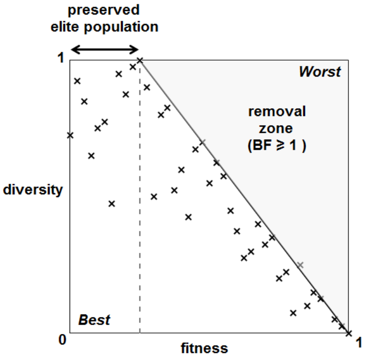


UHGS – Vidal et al. (2012, 2014)

Fitness considering ranks in terms of solution cost $C(I)$ and contribution to the population diversity $D(I)$, measured as a distance to other individuals :

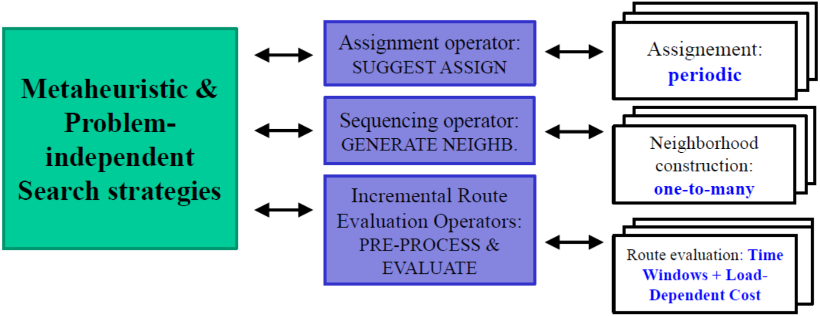
$$BF(I) = C(I) + \left(1 - \frac{nbElite}{popSize - 1}\right) D(I)$$

- Used for parents selection
  - ⇒ Balancing quality with innovation to promote a more thorough exploration of the search space.
- Used during selection of survivors
  - ⇒ Removing individuals with worst $BF(I)$ still guarantees elitism

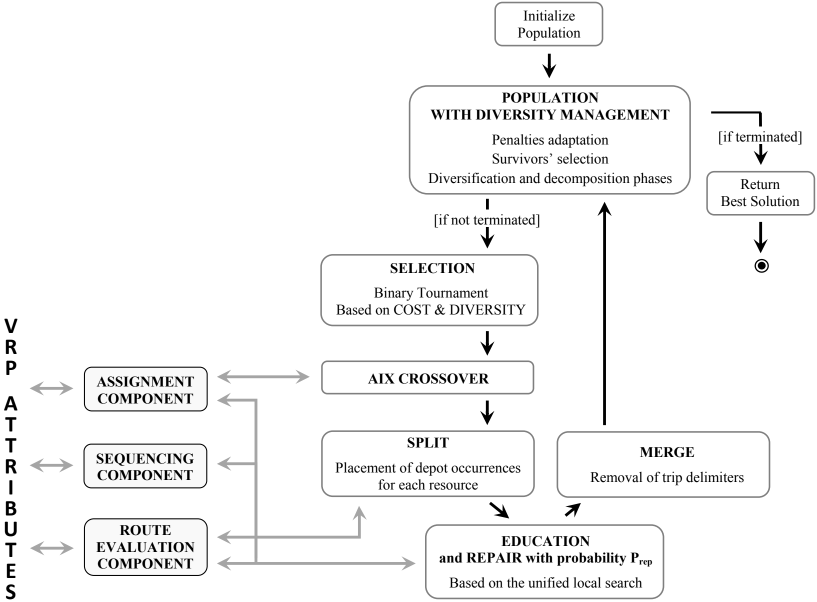


UHGS – Vidal et al. (2012, 2014)

- The HGS approach was "unified" in (Vidal et al., 2014) to solve a wide family of routing problem variants
- Exploiting a generic design with assignment, sequencing & route evaluation operators that are selected and combined by the method based on the problem structure



UHGS – Vidal et al. (2012, 2014)



UHGS – Vidal et al. (2012, 2014)

- UHGS was tested on more than 2000 benchmark instances, and 50 different problems from the vehicle routing literature
- The method has been compared to over 240 previous algorithms
  - State-of-the-art (SOTA) results in the literature on all considered problems: VRP with capacity constraints, duration, backhauls, asymmetry, cumulative costs, simultaneous and mix pickup and deliveries, fleet mix, load dependency, multiple periods, depots, generalized deliveries, open routes, time windows, time-dependent travel time and costs, soft and multiple TW, truck driver scheduling regulations, many other problems and their combinations...
  - First method that efficiently addressed so many routing problems and their combinations, equaling or outperforming SOTA in each case.

UHGS – Vidal et al. (2012, 2014)

Variant	Bench.	n	Obj.	State-of-the-art methods				
				Author	Avg.%	Best%	T(min)	CPU
CVRP	CMT79	[50,199]	C	GG11:	—	+0.03%	8×2.38	8×Xe 2.3G
				MB07:	+0.03%	—	2.80	P-IV 2.8G
				UHGS*:	**+0.02%**	**+0.00%**	11.90	Opt 2.4G
CVRP	GWKC98	[200,483]	C	GG11:	—	+0.29%	8×5	8×Xe 2.3G
				NB09:	+0.27%	+0.16%	21.51	Opt 2.4G
				UHGS*:	**+0.15%**	**+0.02%**	71.41	Opt 2.4G
VRPB	GJ89	[25,200]	C	ZK12:	+0.38%	+0.00%	1.09	T5500 1.67G
				GA09:	+0.09%	+0.00%	1.13	Xe 2.4G
				UHGS:	**+0.01%**	**+0.00%**	0.99	Opt 2.4G
CCVRP	CMT79	[50,199]	C	NPW10:	+0.74%	+0.28%	5.20	Core2 2G
				RL12:	+0.37%	+0.07%	2.69	Core2 2G
				UHGS:	**+0.01%**	**-0.01%**	1.42	Opt 2.2G
CCVRP	GWKC98	[200,483]	C	NPW10:	+2.03%	+1.38%	94.13	Core2 2G
				RL12:	+0.34%	+0.07%	21.11	Core2 2G
				UHGS:	**-0.14%**	**-0.23%**	17.16	Opt 2.2G
VRPSDP	SN99	[50,199]	C	SDBOF10:	+0.16%	+0.00%	256×0.37	256×Xe 2.67G
				ZTK10:	—	+0.11%	—	T5500 1.66G
				UHGS:	**+0.01%**	**+0.00%**	2.79	Opt 2.4G
VRPSDP	MG06	[100,400]	C	SDBOF10:	+0.30%	+0.17%	256×3.11	256×Xe 2.67G
				UHGS:	+0.20%	+0.07%	12.00	Opt 2.4G
				S12 :	**+0.08%**	**+0.00%**	7.23	I7 2.93G

UHGS – Vidal et al. (2012, 2014)

Variant	Bench.	n	Obj.	State-of-the-art methods				
				Author	Avg.%	Best%	T(min)	CPU
VFMP-F	G84	[20,100]	C	ISW09:	—	+0.07%	8.34	P-M 1.7G
				SPUO12:	+0.12%	+0.01%	0.15	I7 2.93G
				UHGS:	**+0.04%**	**+0.01%**	1.13	Opt 2.4G
VFMP-V	G84	[20,100]	C	ISW09:	—	+0.02%	8.85	P-M 1.7G
				SPUO12:	+0.17%	+0.00%	0.06	I7 2.93G
				UHGS:	**+0.03%**	**+0.00%**	0.85	Opt 2.4G
VFMP-FV	G84	[20,100]	C	P09:	—	+0.02%	0.39	P4M 1.8G
				UHGS:	+0.01%	+0.00%	0.99	Opt 2.4G
				SPUO12:	**+0.01%**	**+0.00%**	0.13	I7 2.93G
LDVRP	CMT79	[50,199]	C	XZKX12:	+0.48%	+0.00%	1.3	NC 1.6G
				UHGS:	**-0.28%**	**-0.33%**	2.34	Opt 2.2G
LDVRP	GWKC98	[200,483]	C	XZKX12:	+0.66%	+0.00%	3.3	NC 1.6G
				UHGS:	**-1.38%**	**-1.52%**	23.81	Opt 2.2G
PVRP	CGL97	[50,417]	C	HDH09:	+1.69%	+0.28%	3.09	P-IV 3.2G
				UHGS*:	+0.43%	+0.02%	6.78	Opt 2.4G
				CM12:	**+0.24%**	**+0.06%**	64×3.55	64×Xe 3G
MDVRP	CGL97	[50,288]	C	CM12:	+0.09%	+0.03%	64×3.28	64×Xe 3G
				S12:	+0.07%	+0.02%	11.81	I7 2.93G
				UHGS*:	**+0.08%**	**+0.00%**	5.17	Opt 2.4G
GVRP	B11	[16,262]	C	BER11:	+0.06%	—	0.01	Opt 2.4G
				MCR12:	+0.11%	—	0.34	Duo 1.83G
				UHGS:	**+0.00%**	**-0.01%**	1.53	Opt 2.4G

UHGS – Vidal et al. (2012, 2014)

Variant	Bench.	n	Obj.	State-of-the-art methods Author	Avg.%	Best%	T(min)	CPU
OVRP	CMT79 &F94	[50,199]	F/C	RTBI10:	0%/+0.32%	—	9.54	P-IV 2.8G
				S12:	—/+0.16%	0%/+0.00%	2.39	I7 2.93G
				UHGS:	**0%/+0.11%**	**0%/+0.00%**	1.97	Opt 2.4G
OVRP	GWKC98	[200,480]	F/C	ZK10:	0%/+0.39%	0%/+0.21%	14.79	T5500 1.66G
				S12:	0%/+0.13%	0%/+0.00%	64.07	I7 2.93G
				UHGS:	**0%/-0.11%**	**0%/-0.19%**	16.82	Opt 2.4G
VRPTW	SD88	100	F/C	RTI09:	0%/+0.11%	0%/+0.04%	17.9	Opt 2.3G
				UHGS*:	0%/+0.04%	0%/+0.01%	2.68	Xe 2.93G
				NBD10:	**0%/+0.02%**	**0%/+0.00%**	5.0	Opt 2.4G
VRPTW	HG99	[200,1000]	F/C	RTI09b:	—	+0.16%/+3.36%	270	Opt 2.3G
				NBD10:	+0.20%/+0.42%	+0.10%/+0.27%	21,7	Opt 2.4G
				UHGS*:	**+0.18%/+0.11%**	**+0.08%/-0.10%**	141	Xe 2.93G
OVRPTW	SD88	100	F/C	RTI09a:	+0.89%/+0.42%	0%/+0.24%	10.0	P-IV 3.0G
				KTDHS12:	0%/+0.79%	0%/+0.18%	10.0	Xe 2.67G
				UHGS:	**+0.09%/-0.10%**	**0%/-0.10%**	5.27	Opt 2.2G
TDVRPTW	SD88	100	F/C	KTDHS12:	+2.25%	0%	10.0	Xe 2.67G
				UHGS:	**-3.31%**	**-3.68%**	21.94	Opt 2.2G
VFMPTW	LS99	100	D	BDHMG08:	—	+0.59%	10.15	Ath 2.6G
				RT10:	+0.22%	—	16.67	P-IV 3.4G
				UHGS:	**-0.15%**	**-0.24%**	4.58	Opt 2.2G
VFMPTW	LS99	100	C	BDHMG08:	—	+0.25%	3.55	Ath 2.6G
				BPDRT09:	—	+0.17%	0.06	Duo 2.4G
				UHGS:	-0.38%	-0.49%	4.82	Opt 2.2G

UHGS – Vidal et al. (2012, 2014)

Variant	Bench.	n	Obj.	State-of-the-art methods				
				Author	Avg.%	Best%	T(min)	CPU
CVRP	CMT79	[50,199]	C	GG11:	—	+0.03%	8×2.38	8×Xe 2.3G
				MB07:	+0.03%	—	2.80	P-IV 2.8G
				UHGS*:	**+0.02%**	**+0.00%**	11.90	Opt 2.4G
CVRP	GWKC98	[200,483]	C	GG11:	—	+0.29%	8×5	8×Xe 2.3G
				NB09:	+0.27%	+0.16%	21.51	Opt 2.4G
				UHGS*:	**+0.15%**	**+0.02%**	71.41	Opt 2.4G
VRPB	GJ89	[25,200]	C	ZK12:	+0.38%	+0.00%	1.09	T5500 1.67G
				GA09:	+0.09%	+0.00%	1.13	Xe 2.4G
				UHGS:	**+0.01%**	**+0.00%**	0.99	Opt 2.4G
CCVRP	CMT79	[50,199]	C	NPW10:	+0.74%	+0.28%	5.20	Core2 2G
				RL12:	+0.37%	+0.07%	2.69	Core2 2G
				UHGS:	**+0.01%**	**-0.01%**	1.42	Opt 2.2G
CCVRP	GWKC98	[200,483]	C	NPW10:	+2.03%	+1.38%	94.13	Core2 2G
				RL12:	+0.34%	+0.07%	21.11	Core2 2G
				UHGS:	**-0.14%**	**-0.23%**	17.16	Opt 2.2G
VRPSDP	SN99	[50,199]	C	SDBOF10:	+0.16%	+0.00%	256×0.37	256×Xe 2.67G
				ZTK10:	—	+0.11%	—	T5500 1.66G
				UHGS:	**+0.01%**	**+0.00%**	2.79	Opt 2.4G
VRPSDP	MG06	[100,400]	C	SDBOF10:	+0.30%	+0.17%	256×3.11	256×Xe 2.67G
				UHGS:	+0.20%	+0.07%	12.00	Opt 2.4G
				S12 :	**+0.08%**	**+0.00%**	7.23	I7 2.93G

HGS-CVRP – Vidal (2022)

- An open-source implementation of UHGS dedicated to the CVRP: `https://github.com/vidalt/HGS-CVRP`
- Goal: find the best possible **trade-off between conceptual simplicity and performance**
- Simple structures due to the code specialization to CVRP
- Additional LS operator (SWAP*)
- Using the $\mathcal{O}(n)$-time SPLIT algorithm of Vidal (2016)

HGS-CVRP – Vidal (2022)

- No need for any external library ⇒ easy to set up
- The complete C++ code fits on ≈ 20 pages ⇒ easier learning curve permitting extensions to different applications
  - Python/Julia/C APIs contributed by the community: thanks Changhyun Kwon (*chkwon*), and more recently Niels Wouda (*N-Wouda*) in the scope of the challenge
- State-of-the-art results on the CVRP
  - ⇒ The starter code of the EURO-NeurIPS challenge is an optimized variant of this algorithm adapted to the static VRPTW (Kool et al., 2022)

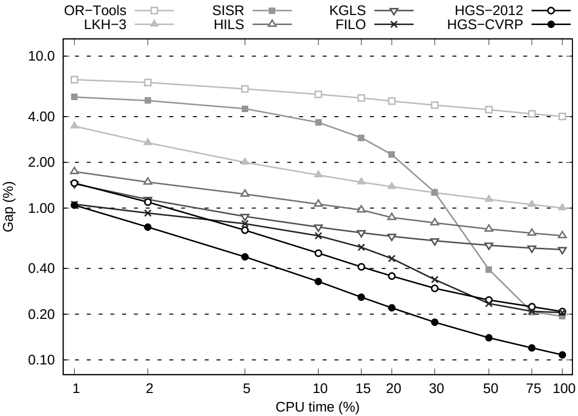
Current State of Research

What to expect regarding heuristics on vehicle routing problems?

- Some research continues on better solution methods (despite smaller and smaller error gaps).
  - ⇒ Thanks to a long tradition of systematic benchmarking on common instances, vehicle routing problem variants remain a good testing ground to experiment with new search strategies.

Current State of Research

- Several metaheuristics (Christiaens and Vanden Berghe, 2020; Accorsi and Vigo, 2021; Máximo and Nascimento, 2021; Vidal, 2022) achieve $< 0.2\%$ average error gap on difficult instances



Error gap of various solution methods on the instances of Uchoa et al. (2017).

Termination criterion set to $T_{\text{MAX}} = n \times 240/100$

Current State of Research

- Some earlier ML papers did not benchmark against the current state-of-the-art regarding heuristics (e.g., OR-Tools, despite its flexibility and ease of use, is far from SOTA).
- To evaluate the benefits of new (e.g., machine learning) search methodologies, it is essential to:

  1) Compare with / build upon the best-known methods (otherwise, we run the risk of running in loops)

  2) Adopt the same conventions and benchmark instances as other studies:

  - `http://vrp.galgos.inf.puc-rio.br/index.php/en/`
  - Uchoa et al. (2017) instances are popular (and still challenging) for heuristics and exact algorithms
  - Queiroga et al. (2021) (XML set) can help establish a standardized CVRP benchmark (with 10,000 known optimums and a generator) for learning-based algorithms

Current State of Research

- **Many search components can contribute to increase performance**
  - ⇒ One can **always** improve a method by “adding more”...
  - ⇒ Success comes from a good tradeoff between performance and simplicity.
  - ⇒ To **gain methodological insights**, need to trim off all unnecessary components and avoid complex methodologies with only marginal contributions to performance.
  - ⇒ Computational experiments to assess the impact of each separate component

Current State of Research

- Some recent studies have been oriented towards heuristics for very-large instances with dozens of thousands of customers (see, e.g., Arnold et al., 2019; Accorsi and Vigo, 2021).

- Many vehicle problem variants of importance still pose great challenges (Vidal et al., 2020, see, e.g.), notably those:

  ...involving **complex interactions** between routes (e.g., synchronization between vehicles)
  ...involving **strategic decisions** (e.g., inventory, localization, or districting)
  ...considering **competitive behavior** or user choices (Gansterer and Hartl, 2018)
  ...considering **partially revealed or uncertain information** (dynamic and stochastic problems – see, e.g., Soeffker et al. 2022 for a review)

Design of the EURO-NeurIPS challenge

Given all of this, we aimed for a problem variant that would be

1) Canonical and relevant for practice
   $\Rightarrow$ VRP with time windows
2) Challenging and a promising ground for learning-based algorithms
   $\Rightarrow$ Requests revealed dynamically (dynamic problem)
3) Simple to define and evaluate
   $\Rightarrow$ Routes are final upon dispatch
   $\Rightarrow$ Fleet is unlimited but everyone must be serviced

Design of the EURO-NeurIPS challenge

- Given the significant score differences between finalist teams on the dynamic variant, it seems the problem of the challenge has lived to its expectations.
- Looking forward to learning more about the winning strategies directly from the finalist teams
- Thanks for all the energy you have dedicated to this challenge (especially Wouter Kool), and congratulations!

Thanks !

THANK YOU FOR YOUR ATTENTION !

Contact me:
`thibaut.vidal@polymtl.ca`

Open-source codes:
`https://github.com/vidalthi/`

Regular updates and announcements:
`https://twitter.com/vidalthi`
`https://mas.to/@vidalthi`

Further reading I

Accorsi, L., D. Vigo. 2021. A fast and scalable heuristic for the solution of large-scale capacitated vehicle routing problems. *Transportation Science* **55**(4) 832–856.

Arnold, F., M. Gendreau, K. Sörensen. 2019. Efficiently solving very large scale routing problems. *Computers & Operations Research* **107**(1) 32–42.

Arnold, F., Í.G. Santana, K. Sörensen, T. Vidal. 2021. PILS: Exploring high-order neighborhoods by pattern mining and injection. *Pattern Recognition* **116** 107957.

Arnold, F., K. Sörensen. 2019. Knowledge-guided local search for the vehicle routing problem. *Computers & Operations Research* **105** 32–46.

Beasley, J.E. 1983. Route first-cluster second methods for vehicle routing. *Omega* **11**(4) 403–408.

Christiaens, J., G. Vanden Berghe. 2020. Slack induction by string removals for vehicle routing problems. *Transportation Science* **54**(2) 299–564.

Christofides, N., S. Eilon. 1972. Algorithms for Large-Scale Travelling Salesman Problems. *Operational Research Quarterly* **23**(4) 511.

Cordeau, J.-F., M. Gendreau, G. Laporte. 1997. A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks* **30**(2) 105–119.

Cordeau, J.-F., G. Laporte, A. Mercier. 2001. A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society* **52**(8) 928–936.

Further reading II

Gansterer, M., R.F. Hartl. 2018. Collaborative vehicle routing: A survey. *European Journal of Operational Research* **268**(1) 1–12.

Gaskell, T.J. 1967. Bases for vehicle fleet scheduling. *Operational Research Quarterly* **18**(3) 281–295.

Gillett, B.E., L.R. Miller. 1974. A heuristic algorithm for the vehicle-dispatch problem. *Operations Research* **22**(2) 340–349.

Glover, Fred. 1996. Ejection chains, reference structures and alternating path methods for traveling salesman problems. *Discrete Applied Mathematics* **65**(1-3) 223–253.

Helsgaun, K. 2000. An effective implementation of the Lin-Kernighan traveling salesman heuristic. *European Journal of Operational Research* **126**(1) 106–130.

Helsgaun, K. 2017. An extension of the Lin-Kernighan-Helsgaun TSP solver for constrained traveling salesman and vehicle routing problems. Tech. rep., Roskilde University.

Irnich, Stefan, Daniel Villeneuve. 2003. *The Shortest-Path Problem with Resource Constraints and k -Cycle Elimination for k >= 3*, vol. 18. Groupe d'études et de recherche en analyse des décisions, HEC Montréal.

Johnson, D., L. Mcgeoch. 1997. The traveling salesman problem: A case study in local optimization. E. Aarts, J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*. Princeton University Press, 215–310.

Further reading III

Kool, W., J.O. Juninck, E. Roos, K. Cornelissen, P. Agterberg, J. Van Hoorn, T. Visser. 2022. Hybrid genetic Search for the vehicle routing problem with time windows: A high-performance implementation. Tech. rep.

Lin, S., B.W. Kernighan. 1973. An effective heuristic algorithm for the traveling-salesman problem. *Operations Research* **21**(2) 498–516.

Máximo, V., M.C.V. Nascimento. 2021. A hybrid adaptive iterated local search with diversification control to the capacitated vehicle routing problem. *European Journal of Operational Research* **294**(3) 1108–1119.

Pisinger, D., S. Ropke. 2007. A general heuristic for vehicle routing problems. *Computers & Operations Research* **34**(8) 2403–2435.

Prins, C. 2004. A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research* **31**(12) 1985–2002.

Queiroga, E., R. Sadykov, E. Uchoa, T. Vidal. 2021. 10,000 optimal CVRP solutions for testing machine learning based heuristics. Tech. rep.

Rochat, Y., E.D. Taillard. 1995. Probabilistic diversification and intensification in local search for vehicle routing. *Journal of Heuristics* **1**(1) 147–167.

Santana, Í., A. Lodi, T. Vidal. 2022. Neural networks for local search and crossover in vehicle routing: A possible overkill? Tech. rep., arXiv:2210.12075.

Further reading IV

Schrimpf, G., J. Schneider, H. Stamm-Wilbrandt, G. Dueck. 2000. Record Breaking Optimization Results Using the Ruin and Recreate Principle. *Journal of Computational Physics* **159**(2) 139–171.

Shaw, P. 1998. Using constraint programming and local search methods to solve vehicle routing problems. M. Maher, J.-F. Puget, eds., *Principles and Practice of Constraint Programming - CP98*, *LNCS*, vol. 1520. Springer Berlin Heidelberg, 417–431.

Soeffker, Ninja, Marlin W. Ulmer, Dirk C. Mattfeld. 2022. Stochastic dynamic vehicle routing in the light of prescriptive analytics: A review. *European Journal of Operational Research* **298**(3) 801–820.

Solomon, M.M. 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research* **35**(2) 254–265.

Subramanian, A., E. Uchoa, L.S. Ochi. 2013. A hybrid algorithm for a class of vehicle routing problems. *Computers & Operations Research* **40**(10) 2519–2531.

Toth, P., D. Vigo. 2003. The granular tabu search and its application to the vehicle-routing problem. *INFORMS Journal on Computing* **15**(4) 333–346.

Uchoa, E., D. Pecin, A. Pessoa, M. Poggi, A. Subramanian, T. Vidal. 2017. New benchmark instances for the capacitated vehicle routing problem. *European Journal of Operational Research* **257**(3) 845–858.

Vidal, T. 2016. Technical note: Split algorithm in O(n) for the capacitated vehicle routing problem. *Computers & Operations Research* **69** 40–47.

Further reading V

Vidal, T. 2022. Hybrid genetic search for the CVRP: Open-source implementation and SWAP* neighborhood. *Computers and Operations Research* **140** 105643.

Vidal, T., T.G. Crainic, M. Gendreau, N. Lahrichi, W. Rei. 2012. A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research* **60**(3) 611–624.

Vidal, T., T.G. Crainic, M. Gendreau, C. Prins. 2013. Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research* **231**(1) 1–21.

Vidal, T., T.G. Crainic, M. Gendreau, C. Prins. 2014. A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research* **234**(3) 658–673.

Vidal, T., G. Laporte, P. Matl. 2020. A concise guide to existing and emerging vehicle routing problem variants. *European Journal of Operational Research* **286** 401–416.

Yellow, P.C. 1970. A Computational Modification to the Savings Method of Vehicle Scheduling. *Operational Research Quarterly* **21**(2) 281–283.