FILO2^{xe}: Linking Multiple FILO2^{x} Runs

Luca Accorsi[1], Demetrio Laganà[2], Federico Michelotto[3], Roberto Musmanno[2], and Daniele Vigo[3, 4]

[1]*Google*
[2]*DIMEG, University of Calabria, Via Pietro Bucci, 87036 - Rende (CS), Italy*
[3]*Department of Electrical, Electronic and Information Engineering "G. Marconi", University of Bologna, Italy*
[4]*CIRI ICT, University of Bologna, Italy*

accorsi@google.com
daniele.vigo@unibo.it

1 Background: FILO2^{x}

FILO2^{x} is a single-trajectory parallel metaheuristic for the efficient solution of large-scale capacitated vehicle routing problem instances (see Accorsi et al. (2025)).

The core optimization procedure, based on the iterated local search paradigm (Lourenço, Martin, and Stützle, 2003), is the primary tool for improving initial solutions. These solutions are generated using an adaptation of the Savings algorithm proposed by Clarke and Wright (1964), followed by an optional quick route minimization procedure.

Each core optimization iteration generates a neighbor solution S' by applying ruin, recreate, and local search steps to a current reference solution S. The neighbor solution S' may replace S according to a standard simulated annealing (SA) acceptance criterion (Kirkpatrick, Gelatt, and Vecchi, 1983).

Because the difference between S and S' is usually small (as optimization steps are localized), x solvers are used to perform multiple core optimization iterations in parallel. This results in a low probability of generating overlapping changes that lead to infeasible neighbor solutions. Iterations that result in infeasibility are discarded.

The core optimization procedure is performed for Δ_{CO} iterations, during which the SA temperature is exponentially lowered from t_0 to t_f.

2 Extension for the BKS Challenge

The solution approach proposed for the BKS challenge, hereafter referred to as FILO2^{xe}, builds on the FILO2^{x} approach and extends it to handle long-running and potentially indefinite executions.

FILO2^{xe} performs several rounds of the core optimization procedure. During the first round, as in FILO2^{x}, Δ_{CO} iterations are performed while the SA temperature is lowered from t_0 to t_f. Subsequent rounds perform $\bar{\Delta}_{CO}$ iterations with the temperature cooling from $\bar{t}_0$ to t_1, where $\bar{\Delta}_{CO}$ and $\bar{t}_0$ are randomly selected from uniform distributions in $[0.1 \cdot \Delta_{CO}, 10 \cdot \Delta_{CO}]$ and $[t_0, t_1]$, respectively. The cooling schedule is adjusted to match the new temperature range and number of iterations.

The first round starts from the initial solution obtained by the savings and route minimization procedures. Subsequent rounds are restarted with a previously visited solution found at temperature $\bar{t}_0$. Specifically, the algorithm's search trajectory is discretized into k buckets of exponentially decreasing

size based on the initial SA temperatures t_0 and t_1. Every bucket b, representing a temperature range $[t_a, t_b]$, holds a random solution found within that range. This solution is selected using a reservoir sampling approach with a probability of $1/C_b$, where C_b is the total number of solutions visited in $[t_a, t_b]$.

2.1 Backup Strategy

If the aforementioned approach does not prove effective, we may explore alternative methodologies.

3 Computational Setting

The algorithm will be executed on several small cloud machines, each equipped with two virtual processors and 8GB of RAM.

Initially, 100 machines will be utilized for several days. Subsequently, the exact number of machines will be determined based on forecast cloud expenses and a maximum budget of $5000.

Occasionally, we may use an additional 64-bit GNU/Linux Ubuntu 22.04 workstation with an Intel Xeon Gold 6254 CPU (3.1 GHz) and 384 GB of RAM.

4 Algorithm Parameters

All parameters are inherited from FILO2^x (Accorsi et al., 2025). The initial number of core optimization iterations Δ_{CO} is set to one million, consistent with long runs of FILO2^x. As a consequence, $\bar{\Delta}_{CO}$ ranges from one hundred thousand to ten million. The simulated annealing temperatures t_0 and t_f are defined to be proportional to the average cost of an arc in an instance.

Two solvers ($x = 2$) will be used for experiments on cloud machines, whereas up to 16 solvers may be used to process selected instances on the additional workstation.

References

[1] Luca Accorsi et al. *Asynchronous Cooperative Optimization of a Capacitated Vehicle Routing Problem Solution*. 2025. arXiv: 2511.19445 [cs.DC].

[2] G. Clarke and J. W. Wright. "Scheduling of vehicles from a central depot to a number of delivery points". In: *Operations Research* 12.4 (1964), pp. 568–581.

[3] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. "Optimization by Simulated Annealing". In: *Science* 220.4598 (1983), pp. 671–680. ISSN: 0036-8075. eprint: https://science.sciencemag.org/content/220/4598/671.full.pdf.

[4] H. R. Lourenço, O. C. Martin, and T. Stützle. "Iterated Local Search". In: *Handbook of Metaheuristics*. Boston, MA: Springer US, 2003, pp. 320–353.