



HAL
open science

A note on “A unified solution framework for multi-attribute vehicle routing problems”

Thierry Garaix, Mohammed Skiredj

► To cite this version:

Thierry Garaix, Mohammed Skiredj. A note on “A unified solution framework for multi-attribute vehicle routing problems”. *European Journal of Operational Research*, 2024, 314 (3), pp.1215-1219. 10.1016/j.ejor.2023.11.037 . hal-04420027

HAL Id: hal-04420027

<https://hal.science/hal-04420027>

Submitted on 26 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A note on "A Unified Solution Framework for Multi-Attribute Vehicle Routing Problems"

Thierry Garaix^a, Mohammed Skiredj^b

^a*Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, Saint-Etienne, F-42023, France*

^b*Lomaco, Figeac, 41600, France*

Abstract

In this note, the authors propose correcting one erroneous formula from [Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2014). A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*, 234(3), 658-673] in charge of lunch breaks. In the original paper, the authors propose to compute several attribute values from the solution of a vehicle routing problem; like the earliest and latest starting time for sequences of customers to visit. The computed values allow us to quickly evaluate the feasibility and the marginal cost of some neighbor solutions. Several variants of the class of vehicle routing problems can be addressed using this approach.

In the case of drivers' lunch break scheduling, the proposed formula combines optimistic values for the earliest and the latest completion times of the sequence of customers to visit. Using these values to evaluate neighbor solutions, may conclude that unfeasible solutions are feasible, or underestimate the completion time of a driver's route.

In this note, we describe a counter-example to identify the error in the formula. We also adapt the formula to the correct result.

Keywords: routing, vehicle routing problem, unified solution framework

1. Introduction

Vehicle routing problems (VRP) form a class of problems widely studied by the operations research community. A large number of variants exist with time windows, capacity, pick-up, delivery, and many other constraints.

In [1], the authors propose a unified solution framework that can be applied to many VRP variants. For each specific constraint, they provide a dedicated set of formulas to use.

Unfortunately, the set of formulas proposed for the lunch break constraint is not valid. This paper aims to highlight this error and to give a correction that can be used by the numerous researchers that are using the solution framework provided by [1].

In Section 2, the unified solution framework (UFS) is described in the case of time windows and lunch break constraints. The incorrectness of the method is explained through a counterexample in Section 2.2. A new set of formulas is proposed in Section 2.3 and applied to the counterexample.

2. The unified solution framework applied to lunch breaks

For a more detailed description of UFS, the reader is referred to the original paper [1].

UFS gives a generic pattern that can be used to build efficient local search algorithms for several VRP variants. The general idea considers all the sequences of nodes (customers or depots in classical VRPs) present in a given current solution. The neighbor solutions that can be reached by UFS concatenate some sequences of the current solution. For each sequence, some attribute values are computed. The evaluation (cost and feasibility) of the neighbor solutions is obtained by some calculations based on these attributes.

In [1], the authors provide for each type of constraint the set of attributes to consider, and how to combine them to evaluate the result of the concatenation operator.

We describe this mechanism through a simple example that involves time window and lunch break constraints.

Let us consider the current solution \mathcal{S} made of two routes $\sigma_a = \{v_1, v_2, v_3, v_4\}$ and $\sigma_b = \{v_5\}$. The node at position u in the sequence σ is named $\sigma(u)$. The nodes v_i , $i = 1, \dots, 5$, have null service times ($s_i = 0$) and the following earliest (e_i) and latest (l_i) start times:

$$\{[10, 20]; [70, 80]; [100, 110]; [150, 170]; [160, 170]\}.$$

The distance between two nodes v_i and v_j is denoted d_{v_i, v_j} . The distance equals 10 in the example for any pair of nodes.

Table 1: List of sequences of \mathcal{S}

Singleton	$v_1 ; v_2 ; v_3 ; v_4 ; v_5$
Doublet	$\{v_1, v_2\} ; \{v_2, v_3\} ; \{v_3, v_4\}$
Triplet	$\{v_1, v_2, v_3\} ; \{v_2, v_3, v_4\}$
Quadruplet	$\{v_1, v_2, v_3, v_4\}$

A lunch break is a single break of duration s_{LB} that must be taken in a non-empty route between $[e_{LB}, l_{LB}]$. The location could be unconstrained or at one dedicated location v^{LB} selected in a set of potential locations V^{LB} .

In the example, we develop in the paper, $s_{LB} = 60$; the lunch can be taken at any location $V^{LB} = \emptyset$, at any time ($[e_{LB} = -\infty, l_{LB} = +\infty]$).

All the sequences present in \mathcal{S} are listed in Table 1, sorted by their length.

The concatenation operator of two sequences σ_1 and σ_2 is denoted \oplus ; for instance, $\{v_1, v_2\} \oplus \{v_3\} = \{v_1, v_2, v_3\}$.

2.1. Attributes for VRP with time window constraints

We first describe the case where only time window constraints are considered. This case is simple to treat and the related formulas remain the same when the lunch break constraint is added.

2.1.1. Definition

The attributes defined for each sequence of nodes are the total travel time (servicing + driving), T , the earliest finishing time; E , the latest starting time, L ; and the feasibility, F .

2.1.2. Concatenation

Equations (1)-(4) describe how to compute the values of these attributes for the sequence obtained by the concatenation of two sequences σ_1 and σ_2 .

$$T(\sigma_1 \oplus \sigma_2) = T(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + T(\sigma_2) \quad (1)$$

$$E(\sigma_1 \oplus \sigma_2) = \max\{E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + T(\sigma_2), E(\sigma_2)\} \quad (2)$$

$$L(\sigma_1 \oplus \sigma_2) = \min\{L(\sigma_1), L(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - T(\sigma_1)\} \quad (3)$$

$$F(\sigma_1 \oplus \sigma_2) \equiv F(\sigma_1) \wedge F(\sigma_2) \wedge (E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} \leq L(\sigma_2)) \quad (4)$$

In Equation (1), the total time is the sum of the travel times of each sequence plus the distance between the last node of σ_1 and the first node of σ_2 .

Table 2: Values of attributes T , E , L , and F for \mathcal{S}

σ	T	E	L	F
$\{v_1\}$	0	10	20	<i>true</i>
$\{v_2\}$	0	70	80	<i>true</i>
$\{v_3\}$	0	100	110	<i>true</i>
$\{v_4\}$	0	150	170	<i>true</i>
$\sigma_b = \{v_5\}$	0	160	170	<i>true</i>
$\{v_1, v_2\}$	10	70	20	<i>true</i>
$\{v_1, v_2, v_3\}$	20	100	20	<i>true</i>
$\sigma_a = \{v_1, v_2, v_3, v_4\}$	30	150	20	<i>true</i>

If the driver leaves the last node of σ_1 at the earliest completion time, directly travels from σ_1 to σ_2 , and reaches the end of σ_2 , the arrival time is greater or equal to the earliest completion time of σ_2 . This statement gives Equation (2).

A symmetrical reasoning gives Equation (3) to compute the latest starting time of $(\sigma_1 \oplus \sigma_2)$.

The feasibility of $\sigma_1 \oplus \sigma_2$ is ensured by the feasibility of each sub-sequence and the possibility of reaching the beginning of σ_2 starting from the end of σ_1 before the latest starting time of σ_2 . This logical constraint is defined by Equation (4).

2.1.3. Initialisation

For a sequence $\sigma_0 = (v_i)$ containing a single vertex, $T(\sigma_0) = s_i$, $E(\sigma_0) = e_i + s_i$, $L(\sigma_0) = l_i$, and $F(\sigma_0) = \textit{true}$.

By the iterative concatenations from singletons to quadruplets of Table 1, the attributes of all the sequences. They are displayed in Table 2 for the solution \mathcal{S} .

2.1.4. Neighbors evaluation

All the neighbor solutions made of concatenations of the sequences listed in Table 1 can be easily evaluated with Equations (1)-(4).

For instance, the solution $\mathcal{S}' = \{\sigma_c, \sigma_d\}$ with $\sigma_c = \{v_1, v_2, v_4\}$ and $\sigma_d = \{v_3, v_5\}$ is evaluated by the two concatenations: $\{v_1, v_2\} \oplus \{v_4\}$ and $\{v_3\} \oplus \{v_5\}$.

The values obtained for the solution $\mathcal{S}' = \{v_1, v_2, v_3, v_4, v_5\} = \sigma_a \oplus \sigma_b$ are listed in Table 3.

Table 3: Values of attributes T , E , L , and F for \mathcal{S}'

σ	T	E	L	F
$\sigma_a \oplus \sigma_b$	40	160	20	<i>true</i>

2.2. Attributes for VRP with time windows and lunch break

We describe in this section the proposition of [1] for the lunch break. We conclude by the inconsistency revealed by the counter-example.

2.2.1. Definition

Three additional attributes are defined. Compared to the previous attributes E , L , and F they consider the schedule of the lunch break in the sequence. They are denoted, E' , the earliest starting time, L' , the latest starting time, and F' , the feasibility. Note that there is no equivalence of the travel time T including the lunch break duration since it could be obtained by adding the lunch break duration, $T' = T + s_{LB}$.

2.2.2. Concatenation

Equations (5)-(16) compute the values for E' , L' and F' for the two sequences' concatenation.

Four cases are considered: (1) the break is in the first sequence, (2) the break is between the two sequences, (3) the break is in the second sequence, and (4) there is no break (this case refers to E, L, F).

Some intermediate parameters are evaluated for each case, and denoted $E'_{case\ i}$, $L'_{case\ i}$, and $F'_{case\ i}$, for each Case i .

For Case 1, the attributes $E'(\sigma_1)$, $L'(\sigma_1)$, and $F'(\sigma_1)$ are used instead of the attributes that do not include the lunch break in σ_1 . Thus, Equations (2), (3), and (4) are respectively translated to (8), (9), and (10).

Case 3 is symmetric to Case 1.

Three possibilities have to be computed to evaluate $E'_{case\ 2}$:

1. The first member of Equation (11) enforces the earliest completion time of $\sigma_1 \oplus \sigma_2$, $E'(\sigma_1 \oplus \sigma_2)$, to be compatible with the earliest completion time of σ_1 .
2. The second member enforces $E'(\sigma_1 \oplus \sigma_2)$ to be compatible with the earliest starting time of the lunch break. This case is active when the driver is reaching $\sigma_2(1)$ before e_{LB} after the schedule of σ_1 .

3. The third term enforces $E'(\sigma_1 \oplus \sigma_2)$ to be greater or equal to the earliest completion time of the last sequence σ_2 .

A similar approach is applied to compute L'_{case2} :

1. The third member of Equation (12) means that the schedule of σ_1 and the lunch break before σ_2 has to reach $\sigma_2(1)$ before $L(\sigma_2)$.
2. The second member enforces to reach the lunch break before its latest starting time if σ_1 is scheduled before.
3. The first member means that the latest starting time of $\sigma_1 \oplus \sigma_2$ is lower or equal to $L(\sigma_1)$.

The formula given for F'_{case2} ensures the feasibility of the following schedules:

- σ_1 ;
- σ_2 ;
- σ_1 before the lunch break;
- the lunch break before σ_2 ;
- σ_1 followed by the trip to σ_2 , the lunch break, and σ_2 .

$$E'(\sigma_1 \oplus \sigma_2) = \min(\{E'_{casei} | F'_{casei} = true, \forall i \in \{1, 2, 3\}\} \cup +\infty) \quad (5)$$

$$L'(\sigma_1 \oplus \sigma_2) = \max(\{L'_{casei} | F'_{casei} = true, \forall i \in \{1, 2, 3\}\} \cup -\infty) \quad (6)$$

$$F'(\sigma_1 \oplus \sigma_2) \equiv \{F'_{case1} \vee F'_{case2} \vee F'_{case3}\} \quad (7)$$

$$E'_{case1} = \max\{E'(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + T(\sigma_2), E(\sigma_2)\} \quad (8)$$

$$L'_{case1} = \min\{L'(\sigma_1), L(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - T(\sigma_1) - s_{LB}\} \quad (9)$$

$$F'_{case1} \equiv F'(\sigma_1) \wedge F(\sigma_2) \wedge (E'(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} \leq L(\sigma_2)) \quad (10)$$

$$E'_{case2} = \max\{E(\sigma_1) + d_{\sigma_1(|\sigma_1|),\sigma_2(1)} + s_{LB} + T(\sigma_2), e_{LB} + s_{LB} + T(\sigma_2), E(\sigma_2)\} \quad (11)$$

$$L'_{case2} = \min\{L(\sigma_1), l_{LB} - T(\sigma_1), L(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - s_{LB} - T(\sigma_1)\} \quad (12)$$

$$F'_{case2} \equiv F(\sigma_1) \wedge F(\sigma_2) \wedge (E(\sigma_1) \leq l_{LB}) \wedge (e_{LB} + s_{LB} \leq L(\sigma_2)) \wedge (E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + s_{LB} \leq L(\sigma_2)) \quad (13)$$

$$E'_{case3} = \max\{E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + T(\sigma_2) + s_{LB}, E(\sigma_2)\} \quad (14)$$

$$L'_{case3} = \min\{L(\sigma_1), L'(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - T(\sigma_1)\} \quad (15)$$

$$F'_{case3} \equiv F(\sigma_1) \wedge F'(\sigma_2) \wedge (E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} \leq L'(\sigma_2)) \quad (16)$$

Finally Equations (5)-(7) means that $F'(\sigma_1 \oplus \sigma_2)$ is set to true only if at least one case is feasible; and $E'(\sigma_1 \oplus \sigma_2)$ and $L'(\sigma_1 \oplus \sigma_2)$ are respectively set the minimum and maximum feasible values.

2.2.3. Initialisation

The initial values for singletons are given in the five first rows of Table 4. The three last sequences are built by consecutive additions of the last vertex. The evaluation of attributes E' , L' , F' and intermediate ones is obtained through Equations (5)-(16). Calculations are detailed in Appendix A.

Table 4: Values of attributes E' , L' , and F' for the sequence of concatenations

σ	E'_{case1}	E'_{case2}	E'_{case3}	L'_{case1}	L'_{case2}	L'_{case3}	F'_{case1}	F'_{case2}	F'_{case3}	E'	L'	F'
$\{v_1\}$	-	-	-	-	-	-	-	-	-	$+\infty$	0	<i>false</i>
$\{v_2\}$	-	-	-	-	-	-	-	-	-	$+\infty$	0	<i>false</i>
$\{v_3\}$	-	-	-	-	-	-	-	-	-	$+\infty$	0	<i>false</i>
$\{v_4\}$	-	-	-	-	-	-	-	-	-	$+\infty$	0	<i>false</i>
$\{v_5\}$	-	-	-	-	-	-	-	-	-	$+\infty$	0	<i>false</i>
$\{v_1, v_2\}$	-	80	-	-	10	-	<i>false</i>	<i>true</i>	<i>false</i>	80	10	<i>true</i>
$\{v_1, v_2, v_3\}$	100	-	-	10	-	-	<i>true</i>	<i>false</i>	<i>false</i>	100	10	<i>true</i>
$\{v_1, v_2, v_3, v_4\}$	150	170	-	10	20	-	<i>true</i>	<i>true</i>	<i>false</i>	150	20	<i>true</i>

Table 5: Values of attributes E' , L' , and F' for S'

σ	E'_{case1}	E'_{case2}	E'_{case3}	L'_{case1}	L'_{case2}	L'_{case3}	F'_{case1}	F'_{case2}	F'_{case3}	E'	L'	F'
$\sigma_a \oplus \sigma_b$	160	-	-	20	-	-	<i>true</i>	<i>false</i>	<i>false</i>	160	20	<i>true</i>

Table 6: Possible schedules for \mathcal{S}'

lunch position	v_1	v_2	v_3	v_4	v_5
v_1, v_2	[10, 10]	[80, 80]	[100, 110]	[150, 160]	[160, 170]
v_2, v_3	[10, 20]	[70, 40]	[140, 110]	[150, 160]	[160, 170]
v_3, v_4	[10, 20]	[70, 80]	[150, 90]	[220, 160]	[230, 170]

2.2.4. Neighbors evaluation

The solution \mathcal{S}' that concatenates $\sigma_a = \{v_5\}$ at the end of $\sigma_b = \{v_1, v_2, v_3, v_4\}$ is evaluated in Table 5.

We point out that starting the resulting sequence $\sigma_a \oplus \sigma_b$ at 20 is not feasible. In Table 6, the three possible schedules with the lunch break in σ_a are detailed with the impact on the starting time window for each node in \mathcal{S}' .

The lunch can be placed only between v_1 and v_2 with a latest starting time enforced to 10 instead of 20.

The feasibility of Case 1 is checked in (10) with $E'(\sigma_a)$, the earliest completion time of σ_a that could be incompatible with $L'(\sigma_a)$. In addition, the computation of L'_{case1} does not take into account waiting times, and may therefore set L'_{case1} to $L'(\sigma_a)$ and give an inconsistency.

The waiting time enforced between v_1 and v_2 is the key feature to build the counter-example \mathcal{S}' .

2.3. New attributes for VRP with time windows and lunch break

Regarding the counter-example and the default of the attributes E' , L' , and F' , one can state that each possible position of the lunch break in a sequence has to bring its own set of attributes. In this section, we propose an alternative setting with a reduced set of attributes for each sequence.

2.3.1. Definition

In this section, we propose a new set of attributes to compute for each sequence. The attributes E , L , F , and T are defined exactly as previously. The concatenation operator remains the same for these attributes.

Instead of the attributes E' , L' , and F' , we define the pair of attributes (E'_x, L'_x) for any not-dominated position of the lunch break in the sequence – named configuration x for short. The set of not-dominated positions for the lunch break is denoted C . An empty set C means infeasibility for a lunch break scheduled in the sequence.

2.3.2. Concatenation

As in [1], we consider the same four cases to compute the value of these attributes after a concatenation. But for each case, we consider all the configurations that are matching with the current case.

Each configuration $x \in C(\sigma_1)$ with a break in σ_1 is concatenated with the sequence σ_2 without the lunch break. This configuration of $\sigma_1 \oplus \sigma_2$ is denoted $f(x, \cdot)$. The concatenation operator in this case is given by Equations (17)-(18), which are similar to Equations (8) and (9).

$$\begin{aligned} & \forall x \in C(\sigma_1) \\ E'_{f(x, \cdot)}(\sigma_1 \oplus \sigma_2) &= \max\{E'_x(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + T(\sigma_2), E(\sigma_2)\} \quad (17) \end{aligned}$$

$$L'_{f(x, \cdot)}(\sigma_1 \oplus \sigma_2) = \min\{L'_x(\sigma_1), L(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - T(\sigma_1) - s_{LB}\} \quad (18)$$

The second case refers to the single configuration $f(\cdot, \cdot)$, with the lunch break between σ_1 and σ_2 . Equations (19) and (20) are derived from Equations (11) and (12).

$$\begin{aligned} & E'_{f(\cdot, \cdot)}(\sigma_1 \oplus \sigma_2) = \\ \max\{E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + s_{LB} + T(\sigma_2), e_{LB} + s_{LB} + T(\sigma_2), E(\sigma_2)\} & \quad (19) \end{aligned}$$

$$\begin{aligned} & L'_{f(\cdot, \cdot)}(\sigma_1 \oplus \sigma_2) = \\ \min\{L(\sigma_1), l_{LB} - T(\sigma_1), L(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - s_{LB} - T(\sigma_1)\} & \quad (20) \end{aligned}$$

The third case adapts Equations (14) and (15) to obtain Equations (21) and (22).

$$\begin{aligned} & \forall y \in C(\sigma_2) \\ E'_{f(\cdot, y)}(\sigma_1 \oplus \sigma_2) &= \max\{E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + T(\sigma_2) + s_{LB}, E'_y(\sigma_2)\} \quad (21) \end{aligned}$$

$$L'_{f(\cdot, y)}(\sigma_1 \oplus \sigma_2) = \min\{L(\sigma_1), L'_b(\sigma_2) - d_{\sigma_1(|\sigma_1|)\sigma_2(1)} - T(\sigma_1)\} \quad (22)$$

Thus, the number of subcases considered is equal to $|C(\sigma_1)| + 1 + |C(\sigma_2)|$ configurations with a break, plus one final sequence without any break. The time complexity of this procedure is, therefore, linear in the number of nodes.

The set of acceptable configurations with a lunch break for the sequence $\sigma_1 \oplus \sigma_2$ is obtained by checking the feasibility of each resulting sequence of the

tested combinations, as in Equations (23), (24), and (25). These equations are similar to Equations (10), (13), and (16), respectively.

Finally, this set is limited to configurations that are not-dominated. The dominance between the two configurations is determined by the values of the earliest completion time and the latest starting time, like in Equation (26).

$$C^1 = \{z = f(x, \cdot) | x \in C(\sigma_1) : F(\sigma_2) \wedge (E'_x(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} \leq L(\sigma_2)) = true\} \quad (23)$$

$$C^2 = \{z = f(\cdot, \cdot) | F(\sigma_1) \wedge F(\sigma_2) \wedge (E(\sigma_1) \leq l_{LB}) \wedge (e_{LB} + s_{LB} \leq L(\sigma_2)) \wedge (E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} + s_{LB} \leq L(\sigma_2)) = true\} \quad (24)$$

$$C^3 = \{z = f(\cdot, y) | y \in C(\sigma_2) : F(\sigma_1) \wedge (E(\sigma_1) + d_{\sigma_1(|\sigma_1|)\sigma_2(1)} \leq L'_y(\sigma_2)) = true\} \quad (25)$$

$$C(\sigma_1 \oplus \sigma_2) = \{z \in \{C^1 \cup C^2 \cup C^3\} | \nexists w \in \{C^1 \cup C^2 \cup C^3\} : (E'_w \leq E'_z) \wedge (L'_w \geq L'_z)\} \quad (26)$$

2.3.3. Initialisation

For any singleton $\{v_1\}$, there is no feasible configuration with a lunch break ($C(\{v_1\}) = \emptyset$), since the lunch break cannot start or end a sequence. Attributes E , L , T , and F are defined as in the original approach, illustrated in Table 2.

For any doublet, only the configuration $f(\cdot, \cdot)$ with the break between the two vertices is accepted; Equation (24) given for C^2 is applied. The result obtained for $\{v_1, v_2\}$ is displayed in the first row of Table 7. Attributes E , L , T , and F are computed through Equations (1)-(4); the values for $\{v_1, v_2\}$ form the sixth row of Table 2.

In Table 7, the symbol $*$ locates the lunch break in the sequence. In the notation E'_x and L'_x the index x represents the current configuration.

2.3.4. Neighbors evaluation

The proposed approach is applied to the example presented in the previous section where $\{v_1, v_2, v_3, v_4, v_5\}$ is built by iterative concatenations of the last vertex.

The blocks of equations (1)-(4) and (17)-(22) compute the values of all the attributes, while Equations (23)-(26) check the feasibility of the configurations that include a lunch break. The results are found in Tables 2, 3 and 7. Detailed calculations are given in Appendix B.

Table 7: Computation details with the set of configurations for lunch break

σ	C	Case	Feasible	E'_x	L'_x
$\{v_1\} \oplus \{v_2\}$	$\{v_1, *, v_2\}$	C^2	True	80	10
$\{v_1, v_2\} \oplus \{v_3\}$	$\{v_1, *, v_2, v_3\}$,	C^1	True	100	10
	$\{v_1, v_2, *, v_3\}$	C^2	False		
$\{v_1, v_2, v_3\} \oplus \{v_4\}$	$\{v_1, *, v_2, v_3, v_4\}$	C^1	True	150	10
	$\{v_1, v_2, v_3, *, v_4\}$	C^2	True	170	20
$\{v_1, v_2, v_3, v_4\} \oplus \{v_5\}$	$\{v_1, *, v_2, v_3, v_4, v_5\}$	C^1	True	160	10
	$\{v_1, v_2, v_3, *, v_4, v_5\}$,	C^1	False		
	$\{v_1, v_2, v_3, v_4, *, v_5\}$	C^2	False		

In this example, dominance never applies.

The configuration $\{v_1, v_2, *, v_3\}$ imposes to visit v_3 after 140 that is greater than $L(\{v_3\}) = 110$. The parameter E'_x is evaluated at 170 with the configuration $x = \{v_1, v_2, v_3, *, v_4, v_5\}$ that violates $L(\{v_5\}) = 170$ (the travel time is added to E'_x). The schedule of the break after v_4 in $\{v_1, v_2, v_3, v_4, *, v_5\}$ makes this configuration infeasible since v_5 could not be visited before time 220.

The unique feasible configuration $x = \{v_1, *, v_2, v_3, v_4, v_5\}$ has the lunch break between v_1 and v_2 with its latest starting time L'_x set to 10 instead of 20 with the original formulas.

3. Conclusion

Through a counter-example, we show that some erroneous evaluations can be processed while exploring neighbor solutions of a VRP with time windows and lunch breaks when the approach proposed in [1].

We propose an alternative evaluation process that may drastically increase the computational complexity of several neighborhood operators.

However, the proposed approach should be efficient when a few positions of the lunch break are feasible and not dominated in a sequence. We guess that several real-world applications satisfy this hypothesis.

For future works, we propose to extend this approach to several types of breaks to schedule in the routes.

Appendix A. Detailed calculations with the original attributes

In this section, we describe how the values of the attributes are computed after each concatenation. Because of the infeasibility of the related case, the values of some attributes are not calculated.

1. $\{v_1\} \oplus \{v_2\}$
 - (4) $F = True \wedge True \wedge (10 + 10 \leq 80) = True$
 - (1) $T = 0 + 10 + 0 = 10$
 - (2) $E = \max\{10 + 10 + 0; 70\} = 70$
 - (3) $L = \min\{20; 80 - 10 - 0\} = 20$
 - (10) $F'_{case1} = False \wedge True \wedge (\infty + 10 \leq 80) = False$
 - (13) $F'_{case2} = True \wedge True \wedge (10 \leq \infty) \wedge (-\infty + 60 \leq 80) \wedge (10 + 10 + 60 \leq 80) = True$
 - (11) $E'_{case2} = \max\{10 + 10 + 60 + 0; -\infty + 60 + 0; 80\} = 80$
 - (12) $L'_{case2} = \min\{20; \infty - 0; 80 - 10 - 60 - 0\} = 10$
 - (16) $F'_{case3} = True \wedge False \wedge (10 + 10 \leq 0) = False$
 - (7) $F' = False \vee True \vee False = True$
 - (5) $E' = \min\{\{80\} \cup \infty\} = 80$
 - (6) $L' = \max\{\{10\} \cup -\infty\} = 10$
2. $\{v_1, v_2\} \oplus \{v_3\}$
 - (4) $F = True \wedge True \wedge (70 + 10 \leq 110) = True$
 - (1) $T = 10 + 10 + 0 = 20$
 - (2) $E = \max\{70 + 10 + 0; 100\} = 100$
 - (3) $L = \min\{20; 110 - 10 - 10\} = 20$
 - (10) $F'_{case1} = True \wedge True \wedge (80 + 10 \leq 110) = True$
 - (8) $E'_{case1} = \max\{80 + 10 + 0; 100\} = 100$
 - (9) $L'_{case1} = \min\{10; 110 - 10 - 10 - 60\} = 10$
 - (13) $F'_{case2} = True \wedge True \wedge (70 \leq \infty) \wedge (-\infty + 60 \leq 110) \wedge (70 + 10 + 60 \leq 110) = False$
 - (16) $F'_{case3} = True \wedge False \wedge (70 + 10 \leq 0) = False$

$$(7) F' = True \vee False \vee False = True$$

$$(5) E' = \min\{\{100\} \cup \infty\} = 100$$

$$(6) L' = \max\{\{10\} \cup -\infty\} = 10$$

$$3. \{v_1, v_2, v_3\} \oplus \{v_4\}$$

$$(4) F = True \wedge True \wedge (100 + 10 \leq 170) = True$$

$$(1) T = 20 + 10 + 0 = 30$$

$$(2) E = \max\{100 + 10 + 0; 150\} = 150$$

$$(3) L = \min\{20; 170 - 10 - 20\} = 20$$

$$(10) F'_{case1} = True \wedge True \wedge (100 + 10 \leq 170) = True$$

$$(8) E'_{case1} = \max\{100 + 10 + 0; 150\} = 150$$

$$(9) L'_{case1} = \min\{10; 170 - 10 - 20 - 60\} = 10$$

$$(13) F'_{case2} = True \wedge True \wedge (100 \leq \infty) \wedge (-\infty + 60 \leq 170) \wedge (100 + 10 + 60 \leq 170) = True$$

$$(11) E'_{case2} = \max\{100 + 10 + 60 + 0; -\infty + 60 + 0; 150\} = 170$$

$$(12) L'_{case2} = \min\{20; \infty - 20; 170 - 10 - 60 - 20\} = 20$$

$$(16) F'_{case3} = True \wedge False \wedge (100 + 10 \leq 0) = False$$

$$(7) F' = True \vee True \vee False = True$$

$$(5) E' = \min\{\{150; 170\} \cup \infty\} = 150$$

$$(6) L' = \max\{\{10; 20\} \cup -\infty\} = 20$$

$$4. \{v_1, v_2, v_3, v_4\} \oplus \{v_5\}$$

$$(4) F = True \wedge True \wedge (150 + 10 \leq 170) = True$$

$$(1) T = 30 + 10 + 0 = 40$$

$$(2) E = \max\{150 + 10 + 0; 160\} = 160$$

$$(3) L = \min\{20; 170 - 10 - 30\} = 20$$

$$(10) F'_{case1} = True \wedge True \wedge (150 + 10 \leq 170) = True$$

$$(8) E'_{case1} = \max\{150 + 10 + 0; 160\} = 160$$

$$(9) L'_{case1} = \min\{20; 170 - 10 - 30 - 60\} = 20$$

$$(13) F'_{case2} = True \wedge True \wedge (150 \leq \infty) \wedge (-\infty + 60 \leq 170) \wedge (150 + 10 + 60 \leq 170) = False$$

$$(16) F'_{case3} = True \wedge False \wedge (150 + 10 \leq 0) = False$$

$$(7) F' = True \vee False \vee False = True$$

$$(5) E' = \min\{\{160\} \cup \infty\} = 160$$

$$(6) L' = \max\{\{20\} \cup -\infty\} = 20$$

Appendix B. Detailed calculations with the new attributes

In this section, we describe how the values of Table 7 are computed.

1. $\{v_1\} \oplus \{v_2\}$, $\{v_1, *, v_2\} \in C^2$
 - (24) Feasibility. $True \wedge True \wedge (10 \leq \infty) \wedge (-\infty + 60 \leq 80) \wedge (10 + 10 + 60 \leq 80) = True$
 - (19) $E'_{\{v_1, *, v_2\}} = \max\{10 + 10 + 60 + 0; -\infty + 60 + 0; 70\} = 80$
 - (20) $L'_{\{v_1, *, v_2\}} = \min\{20; \infty - 0; 80 - 10 - 60 - 0\} = 10$
2. $\{v_1, v_2\} \oplus \{v_3\}$, $\{v_1, *, v_2, v_3\} \in C^1$
 - (23) Feasibility. $True \wedge (80 + 10 \leq 110) = True$
 - (17) $E'_{\{v_1, *, v_2, v_3\}} = \max\{80 + 10 + 0; 100\} = 100$
 - (18) $L'_{\{v_1, *, v_2, v_3\}} = \min\{10; 110 - 10 - 10 - 60\} = 10$
3. $\{v_1, v_2\} \oplus \{v_3\}$, $\{v_1, v_2, *, v_3\} \in C^2$
 - (24) Feasibility. $True \wedge True \wedge (70 \leq \infty) \wedge (-\infty + 60 \leq 110) \wedge (70 + 10 + 60 \leq 110) = False$
4. $\{v_1, v_2, v_3\} \oplus \{v_4\}$, $\{v_1, *, v_2, v_3, v_4\} \in C^1$
 - (23) Feasibility. $True \wedge (100 + 10 \leq 170) = True$
 - (17) $E'_{\{v_1, *, v_2, v_3, v_4\}} = \max\{100 + 10 + 0; 150\} = 150$
 - (18) $L'_{\{v_1, *, v_2, v_3, v_4\}} = \min\{10; 170 - 10 - 20 - 60\} = 10$
5. $\{v_1, v_2, v_3\} \oplus \{v_4\}$, $\{v_1, v_2, v_3, *, v_4\} \in C^2$
 - (24) Feasibility. $True \wedge True \wedge (100 \leq \infty) \wedge (-\infty + 60 \leq 170) \wedge (100 + 10 + 60 \leq 170) = True$
 - (19) $E'_{\{v_1, v_2, v_3, *, v_4\}} = \max\{100 + 10 + 60 + 0; -\infty + 60 + 0; 150\} = 170$
 - (20) $L'_{\{v_1, v_2, v_3, *, v_4\}} = \min\{20; \infty - 20; 170 - 10 - 60 - 20\} = 20$
6. $\{v_1, v_2, v_3, v_4\} \oplus \{v_5\}$, $\{v_1, *, v_2, v_3, v_4, v_5\} \in C^1$
 - (23) Feasibility. $True \wedge (150 + 10 \leq 170) = True$

$$(17) E'_{\{v_1, *, v_2, v_3, v_4, v_5\}} = \max\{150 + 10 + 0; 160\} = 160$$

$$(18) L'_{\{v_1, *, v_2, v_3, v_4, v_5\}} = \min\{10; 170 - 10 - 30 - 60\} = 10$$

$$7. \{v_1, v_2, v_3, v_4\} \oplus \{v_5\}, \{v_1, v_2, v_3, *, v_4, v_5\} \in C^1$$

$$(23) \text{ Feasibility. } True \wedge (170 + 10 \leq 170) = False$$

$$8. \{v_1, v_2, v_3, v_4\} \oplus \{v_5\}, \{v_1, v_2, v_3, v_4, *, v_5\} \in C^1$$

$$(24) \text{ Feasibility. } True \wedge True \wedge (150 \leq \infty) \wedge (-\infty + 60 \leq 170) \wedge (150 + 10 + 60 \leq 170) = False$$

References

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