Computers & Operations Research 137 (2022) 105526

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

A multi-tiered vehicle routing problem with global cross-docking

A. Smith [a,*], P. Toth [b], L. Bam [c], J.H. van Vuuren [a]

[a] *Stellenbosch Unit for Operations Research in Engineering, Department of Industrial Engineering, Stellenbosch University, Private Bag X1, Matieland, 7602, South Africa*

[b] *DEI, University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy*

[c] *Health Systems Engineering and Innovation Hub, Department of Industrial Engineering, Stellenbosch, Private Bag X1, Matieland, 7602, South Africa*

ARTICLE INFO

Keywords:
Combinatorial optimisation
Vehicle routing problem
Integer programming model
Global cross-docking
Multi-objective ant colony optimisation
Healthcare optimisation
Supply chain management

ABSTRACT

A new "rich" variation on the multi-objective *vehicle routing problem* (VRP), called the *multi-tiered vehicle routing problem with global cross-docking* (MTVRPGC), is introduced in this paper. With respect to previously studied VRPs, the MVRPTGC includes the following novel features: (i) segregation of facilities into different tiers that distinguish them in terms of different processing and storage capabilities, (ii) cross-docking at a pre-specified subset of facilities in the network (a feature referred to as *global cross-docking*), and (iii) the possibility of spill-over into subsequent planning periods of demand for facility visitation. The problem originated from a real-life application concerning the collection and delivery of pathology specimens in the transportation network of a pathology health-care service provider. Other industrial applications may, however, benefit from this type of VRP, such as mail sorting. A *mixed integer linear programming* (MILP) model for this VRP is proposed, and tested computationally in respect of seventeen small hypothetical test instances. A *multi-objective ant colony optimisation* (MACO) algorithm for solving larger real-world instances of the MTVRPGC is also proposed. The solutions returned by the MACO algorithm are compared with those achieved by the MILP in respect to sixteen instances and also compared to actual collection and delivery routes of a real pathology healthcare service provider operating in South Africa and it is found that adopting the routes suggested by the algorithm results in substantial improvements of all the objectives pursued relative to the *status quo*.

1. Introduction

The class of *vehicle routing problems* (VRPs) has enjoyed a long and colourful history since its inception in 1959 by Dantzig and Ramser (1959), resulting in numerous variations introduced to accommodate practical considerations (see, *e.g.*, Toth and Vigo, 2014).

In this paper, we consider the *multi-tiered vehicle routing problem with global cross-docking* (MTVR-PGC), a variation on the multi-objective VRP with time-windows that arises in a real-life application related to the collection and delivery of pathology specimens (called *commodities* in the following description). With respect to previously studied VRPs, the MVTRPGC includes the following three novel features:

(i) *Segregation of facilities into different tiers*. There are different types of commodities that have to be collected from a set of facilities (*e.g.*, hospitals and clinics) and processed in potentially different ways at a set of facilities (*e.g., laboratories*) within a transportation network. The variation in commodity type may be due to the nature of the commodities themselves, such as their purpose and processing requirements, as well as maintaining standards associated with a commodity, or may even be due to the intended destinations of the commodities. We segregate the available processing facilities according to their respective processing and storage capabilities into a set of tiers. This tier allocation is nested in the sense that a facility of tier i can process any type of commodity that can be processed at a facility of tier j if $j < i$, but there exist certain commodity types which can be processed at a facility of tier i that cannot be processed at any facility of a lower tier. Facilities of the lowest tier represent facilities at which the commodities originate and have to be collected—these facilities have no commodity processing or storage capabilities—their only role is that they introduce new commodities into the system. Facilities of higher tiers may or may not introduce new commodities into the system, but their distinguishing feature is that they all offer commodity processing capabilities or intermediate commodity storage capabilities. All facilities, excluding facilities of the lowest tier, are assumed to offer the same storage capabilities.

* Corresponding author.
E-mail address: anthonysmith_uae@protonmail.com (A. Smith).

https://doi.org/10.1016/j.cor.2021.105526
Received 7 August 2019; Received in revised form 15 August 2021; Accepted 17 August 2021
Available online 27 August 2021
0305-0548/

(ii) *Global cross-docking at a pre-specified subset of facilities*. We allow for handover of commodities at facilities in the sense that a commodity requiring processing at a facility of a specific tier may be transported by one vehicle to a facility of a lower tier than the required one, and then be collected later by some other vehicle(s) which transport it to a facility of the required tier. We refer to this type of commodity handover, which may occur at a facility of any tier (save the lowest and the highest[1]), as *global cross-docking*.[2]

(iii) *Rolling demand horizon*. We allow demand for commodity collection to spill-over into a subsequent planning period. We essentially assume that the time continuum may be partitioned into planning periods of fixed length. One planning period is considered at a time, and if demand for commodity collection occurs at a facility after the last vehicle has departed from that facility, then this commodity is simply collected from the facility during the following planning period (all demand for commodity collection is assumed to be the same in each planning period and is known at the beginning of the planning period). Individual commodities are not tracked as they travel through the system, but they nevertheless all require collection at their originating facilities and transportation to facilities with adequate processing capabilities. This requirement is met by constructing a model which produces a flow route (perhaps consisting of several individual vehicle sub-routes) for commodities from any facility (except facilities of the highest tier) to a facility of a strictly higher tier, thereby facilitating delivery of the commodities to facilities of the tiers required, perhaps after repeated global cross-docking operations, performed in one or more planning periods.

The requirements of the aforementioned problem conform to suggestions in the so-called *Maputo Declaration* (WHO, 2008) to which a large number of countries are signatories. The declaration suggests that the pathology specimen processing facilities of a national health laboratory service should be segregated into different tiers indicative of their processing capabilities (in terms of both specimen processing variation and quantity). There are typically four tiers of specimen processing laboratories: A tier-one laboratory is typically referred to as a primary laboratory where only doctors, nurses, and medical assistants are stationed, whereas a tier-two laboratory additionally has laboratory specialists and senior technologists available. A tier-three laboratory has staff of the same qualifications as those at a tier-two laboratory, but additionally has equipment available which enables it to offer a complete menu of testing blood samples for HIV/AIDS, tuberculosis and malaria as well as many other diseases at a much higher throughput. Finally, a tier-four laboratory performs the tasks of the lower-tiered laboratories, and additionally acts as a reference laboratory providing linkages with research laboratories, academic institutions and other public laboratories that can provide assistance in clinical trials, the evaluation of new technology and surveillance. The clinics at which specimens originate are referred to as laboratories of tier zero as they do not offer any processing capabilities. In rural settings, the distribution of the specimen processing laboratories is such that for specimens to reach a processing laboratory of the required tier, global cross-docking and spill-over into subsequent planning periods are a necessity since it would be impossible for a single vehicle to deliver during a single planning period specimens originating in such settings over the long distances required to reach a suitable tier of processing facility in view of legal maximum driving times.

[1] Global cross-docking of commodities at facilities of the highest tier is not necessary as we assume that all commodities considered in the transportation network can be processed at facilities of the highest tier. Global cross-docking of commodities may also not occur at facilities of the lowest tier as they do not offer any processing or storage capabilities.

[2] As opposed to the traditional notion of *cross-docking* in the supply chain literature where goods are consolidated at a dedicated cross-docking facility (Liao et al., 2010), referred to here as *local cross-docking*.

The MVRPTGC considered in this paper can also model a postal service collection and consolidation network. In this case, the segregation of facilities may refer to the extent to which mail sorting takes place in each sorting centre within the system. There may, for example, be local, provincial, national, and international mail sorting centres in the system, giving rise to four tiers of mail sorting facilities. Letters destined to be sent abroad may then conceivably experience repeated global cross-docking operations—first at a local sorting centre, then at a provincial sorting centre and finally at a national sorting centre before finally being consolidated, within one or more planning periods, at an international sorting centre.

The commodity collection and processing system with global cross-docking and demand spill-over to subsequent planning periods described above is modelled in this paper as a tri-objective VRP which may form the basis of a decision support system capable of assisting tiered-facility services in respect of cost-effective planning, routing and scheduling of a fleet of homogeneous vehicles dedicated to commodity collection. An acceptable trade-off between the three objectives is pursued in the model, namely minimisation of the total time required to transport commodities, minimisation of the maximum travel time associated with the vehicles utilised and, finally, minimisation of the number of vehicles required to implement the commodity collection routing schedule.

The paper is organised as follows. Section 2 is devoted to a brief review of various VRPs from the literature that are related to the problem considered here. After carefully noting the assumptions underlying our novel VRP in Section 3, we proceed to cast the problem as a *mixed integer linear programming* (MILP) model in Section 4. The MILP model formulation builds on a combination of well-known model components proposed in the literature for various VRP variants, and on new model components introduced for accommodating the novel features outlined above. We then validate the model logic in Section 5 by implementing the model in the commercial MILP solver CPLEX and applying it to seventeen small-size, hypothetical problem instances. For the solution of larger real-world instances, we propose a novel *multi-objective ant colony optimisation* (MACO) algorithm in Section 6, customised to the unique requirements of the MTVRPGC, which incorporates several novel algorithmic components, such as a global cross-docking component, in an attempt to yield approximate solutions of a high quality to the MTVRPGC. A case study is carried out in Section 7 in order to compare the performance of a solution returned by the MACO algorithm with the *status quo* in a real tiered pathology service network. Finally, the paper closes in Section 8 with a brief summary. The main goal of the paper is to introduce a new rich variation of the VRP, having many possible real-world applications and to propose a MILP model and a MACO algorithm for solving the model, taking into account the objectives and constraints of the problem considered.

2. Literature review

The MTVRPGC considered in this paper, which is described in further detail in Section 3, belongs to the family of the so-called *rich VRPs*, since it represents a real-world VRP. In particular, the MTVRPGC is a generalisation on the classical *VRP with Time Windows, the Multi-Depot VRP* and the *Multi-objective VRP*. In addition, it represents a variation of the *Pickup and Delivery Problem, the Multi-echelon VRP* and the *VRP with Cross-docking*. A short review of the literature on the above-mentioned problems is given in this section. For a more extensive review of these problems, see also Irnich et al. (2014).

The classical *Vehicle Routing Problem with Time Windows* (VRPTW) is an extension of the VRP in which each facility is associated with a time interval (called a *time window*) and a *service time*. It is required that servicing of a facility must start within the associated time window, and that the vehicle must stop at the facility location for a time period equal to the associated service time. In addition, in case of arrival at a facility before the start of the associated time window, the vehicle is allowed to wait until servicing may commence. The VRPTW has been considered, among others, in the surveys by Kolen et al. (1987), Bräysy and Gendreau (2005a,b), Kallehauge (2008), and Desaulniers et al. (2014).

When the available vehicles are homogeneous, but must start and end their routes at different depots, the corresponding variation on the VRP is called the *Multi Depot VRP* (MDVRP). Although each available vehicle could potentially have its own specific starting and ending locations, the vehicles are generally grouped and assigned to a limited number of depots in the classical MDVRP. Recent reviews on the MDVRP may be found in Montoya-Torres et al. (2015) and Braekers et al. (2016).

Most of the existing variations on the VRP involve the optimisation of a *single objective* (*i.e.*, minimisation of the global distance travelled by the vehicles utilised), or of *hierarchical objectives* (*e.g.*, first minimising the number of vehicles utilised, and then minimising the global distance). Other variations on the VRP reside within the realm of *multi-objective* optimisation, where the aim is to find an acceptable compromise between the optimisation of several conflicting objectives (*e.g.*, global distance, completion time, or the balancing of the routes). See Jozefowiez et al. (2008) for a survey on the variations on the *Multi-objective VRP*.

In the basic version of the *Pickup and Delivery Problem* (PDP), each transportation request consists of the transportation of a commodity between two locations: One where the commodity is picked up (the origin), and a corresponding location where the commodity is delivered (the destination). It is generally required that each transportation request is served by a single vehicle, which first visits the origin and then the destination. The PDP has been considered, among many others, in surveys by Savelsbergh and Sol (1995), Desaulniers et al. (2002), Battarra et al. (2014), and Doerner and Salazar-Gonzalez (2012). The MTVRPGC considered in this paper differs from the classical PDP in that each commodity has to be transported from its origin to any destination belonging to a specific subset of facilities having the appropriate tier. In addition, each transportation request can be served, possibly over consecutive planning periods, by more than one vehicle.

There are variations on the VRP that consider more than one level of the distribution network, referred to in the literature as *Multi-echelon VRPs*, with city logistics and multi-modal transportation systems among the most cited examples of such a network. Two-echelon VRPs involve transportation networks in which the goods are available from different origins and have to be delivered to the respective destinations while necessarily moving through intermediate facilities. Models, exact algorithms and metaheuristics for the two-echelon VRP have been proposed in Baldacci et al. (2013), Cuda et al. (2015), Dondo et al. (2011) and Perboli et al. (2011). See Cuda et al. (2015) for a survey on two-echelon routing problems. The MTVRPGC presented in this paper differs from Multi-echelon VRPs in that consolidation at intermediary facilities is not compulsory as the commodities may be delivered directly to appropriate facilities.

Cross-docking has been applied in industry since the 1980s, but has only recently attracted attention from academia with more than 85% of the papers on this subject published from 2004 onwards Van Belle et al. (2012). The two key points of cross-docking are, typically, simultaneous arrival and consolidation. If all vehicles do not arrive simultaneously, some vehicles have to wait and therefore the core issue is to synchronise the arrival of vehicles at cross-docking facilities. The cross-docking facilities typically do not offer any processing or storage capabilities and are known *a priori*. Several applications of cross-docking exist in the supply chain management literature (Dondo et al., 2011; Liao et al., 2010). Models, exact algorithms and metaheuristics for the VRP with Cross-docking have been proposed in Grangier et al. (2017), Maknoon and Laporte (2017) and Rais et al. (2014). Recent reviews of VRPs with cross-docking may be found in Buijs et al. (2014) and Van Belle et al. (2012). The global cross-docking component of the MTVRPGC presented in this paper differs from typical cross-docking model components in that the cross-docking facilities offer both storage and processing capabilities, and in addition, the facilities acting as consolidation centres are not known *a priori*, but must be selected by considering the objectives and the constraints of the VRP considered in this paper.

3. Problem description

The MVRPTGC introduced in Section 1 is described in more detail in the form of an assumptions list in this section.[3] The section closes with a description of the input data required.

1. *The nature of the facilities.* The transportation network consists of facilities, consolidation points, and facilities of varying commodity processing and storage capabilities, which are collectively referred to as *facilities*. The facilities are segregated into a collection of nested tiers according to their processing capabilities, with a higher tier indicative of superior processing capabilities. The lowest-tier facilities only require collection, the highest-tier facilities only offer processing capabilities, and all the other facilities both require collection, and offer processing capabilities as well as the same storage or consolidation capabilities.
2. *The nature of the vehicles*. It is assumed that a fleet of homogeneous vehicles is available for commodity collection. The capacities of the vehicles are assumed to be sufficiently large to handle any demand requirements. This is usually a realistic assumption in the case of pathology specimen or mail transportation, because these commodities typically exhibit negligible volume and weight. A capacity constraint may nevertheless easily be included in the model formulation and in the MACO algorithm described in Section 4 and Section 5, respectively, if required. Each vehicle may perform at most one route during the planning period considered.
3. *Home depot allocation.* It is assumed that each vehicle has a fixed home depot which may be located at any of the facilities within the network. All vehicles must begin and end their routes at their respective home depots.
4. *Multiple visits and global cross-docking.* Each lowest-tier facility must be visited by exactly one vehicle during the planning period. The other facilities may each be visited by more than one vehicle during the planning period, although any specific vehicle may visit any facility at most once during the planning period. In particular, according to the global cross-docking feature, a commodity may be delivered to a facility of a tier different from the lowest and the highest tiers by a vehicle, and later be collected from this facility by a different vehicle for further transportation in the network.
5. *Service times.* The service time of a facility by a vehicle is limited to the loading and/or unloading of commodities at the facility and does not include the processing times of the commodities. The collection and delivery of commodities by vehicles must be performed within certain time windows that reflect the operational hours of each facility.
6. *Rolling demand horizon.* It is assumed that demand for commodity collection is the same during each planning period and that demand not fully satisfied during the previous planning period may be brought forward to the current planning period. This allows for a vehicle to deliver and collect commodities at the same facility without having to wait at the facility for all demand to be realised there. These assumptions allow one to consider a single planning period, representative of all the demand to be satisfied in a problem instance.

[3] These assumptions have been agreed in conjunction with a senior decision maker at a large pathology healthcare service provider in South Africa.

7. *Facility visitation sequence.* For feasibility of a solution, it is required that every facility (except the highest-tier facilities) should be visited by at least one vehicle that also visits a higher-tier facility at a later stage within the planning period or should participate with another vehicle in cross-docking at a consolidation facility such that the commodities of the facility reach a strictly higher-tier facility (this allows a facility of a tier different from the lowest and the highest tiers to be visited by a vehicle that later visits a facility of the same tier, if this facility is visited by another vehicle visiting a higher-tier facility at a later stage). In other words, any facility i having a tier different from the lowest and the highest tiers must be visited, as a "collection facility", by a vehicle which later visits at least one facility having an equal or higher tier. In particular, if facility i is visited as a "consolidation facility" by vehicle h which has earlier visited facilities having the same tier, then facility i must be visited, as a "collection facility", by a vehicle k (with $k \neq h$) which later visits at least one facility having a higher tier. These vehicle visitation sequence feasibility requirements are elucidated in Fig. 1, with Fig. 1(a) representing a feasible solution and Fig. 1(b) representing an infeasible solution.
8. *Commodity destinations.* Individual commodity collection and transportation is not tracked explicitly in the model formulation as numerous types of commodities may be collected and an even larger number of possible types of commodity processing may be required by these commodities. The only constraint is that a commodity should eventually be delivered to a facility capable of processing it (perhaps over the course of several successive planning periods).

Suppose there are $f+1$ different tiers of facilities in the system, and that each facility tier (save the lowest) is associated with specific commodity processing capabilities. Suppose, furthermore, that indices are assigned to these facility tiers in such a manner that a facility of tier $d > 1$ possesses a superset of the processing capabilities of a facility of tier e for any $e \in \{1, \ldots, d-1\}$, but that all facilities of the same tier have identical processing capabilities. An indexing convention is followed where all facilities exhibiting no processing capabilities are referred to as *facilities of tier zero*, while all processing facilities of tier $d \in \{1, \ldots, f\}$ are referred to as *facilities of tier* d. Let $\mathcal{F}^d$ denote the set of all facilities of tier $d \in \{0, 1, \ldots, f\}$, and define $\mathcal{F} = \cup_{d=0}^{f} \mathcal{F}^d$ as the set of all the facilities. Any facility in $\mathcal{F}^0$ therefore has no commodity processing capability, but only exhibits demand for commodities to be collected there. Any facility in $\mathcal{F}^f$, on the other hand, only processes commodities, and exhibits no demand for the collection of such commodities. Finally, any facility in $\mathcal{F} \setminus (\mathcal{F}^0 \cup \mathcal{F}^f)$ exhibits demand for commodity collection as a result of cross-docking operations there and also offers certain processing capabilities. Facility $i \in \mathcal{F}$ furthermore has an associated vehicle arrival capacity γ_i (*i.e.* a limit on the number of vehicle arrivals the facility can accommodate during the planning period), a required service time of s_i time units and a service time window $[a_i, g_i]$ during which vehicles have access to the facility. It is assumed that if there a vehicle arrives at facility $i \in \mathcal{F}$ before the beginning a_i of the associated time window, it must wait until time a_i to start its service at facility i.

Let $\mathcal{V}$ represent the set of homogeneous vehicles that constitute the commodity collection fleet. As mentioned previously, it is assumed that this set of vehicles is sufficiently large to facilitate feasible commodity collection routing and scheduling at a 100% service level. The homogeneity of the fleet implies that all vehicles have the same autonomy level μ (the maximum allowable route duration of a vehicle, measured in units of expected travel time) and that any two vehicles are expected to traverse a given road link in the same amount of time. Denote the subset of facilities acting as home depots for vehicles by $\mathcal{D}$ and denote the home depot of vehicle $k \in \mathcal{V}$ within this set by b_k. As is customary in the VRP literature, each home depot b_k is associated with a virtual, identical copy of the depot, denoted by b_k^+, in order to be able to distinguish between the departure time of a vehicle from its home depot and the later arrival time of the vehicle when returning to its home depot. In particular, b_k represents the home depot of vehicle $k \in \mathcal{V}$ when it departs from the depot, while b_k^+ represents the same home depot when the vehicle returns to the depot upon completion of its route. The departure time $T'_{b_k k}$ of vehicle $k \in \mathcal{V}$ from the depot b_k is known *a priori*.

The set of all commodities that have to be collected is partitioned into f distinct types, indexed by the set $\mathcal{S} = \{1, \ldots, f\}$, according to the convention that a commodity of type $c \in \mathcal{S}$ can be processed at any facility in $\cup_{d=c}^{f} \mathcal{F}^d$.

Let $\mathcal{G} = (\mathcal{F}, \mathcal{E})$ be a complete directed weighted graph with vertex set $\mathcal{F}$ and arc set $\mathcal{E}$ representing all possible road network connections between facilities in $\mathcal{F}$, where the weight of an arc $(i, j) \in \mathcal{E}$ is the expected travel time t_{ij} of a vehicle traversing the arc from facility $i \in \mathcal{F}$ to facility $j \in \mathcal{F}$. It is assumed that the triangle inequality is upheld and that the travel times are positive.

The planning period is limited to a schedule of fixed length, implemented (possibly in slightly altered form as a result of demand non-realisation at certain facilities) along a rolling horizon.

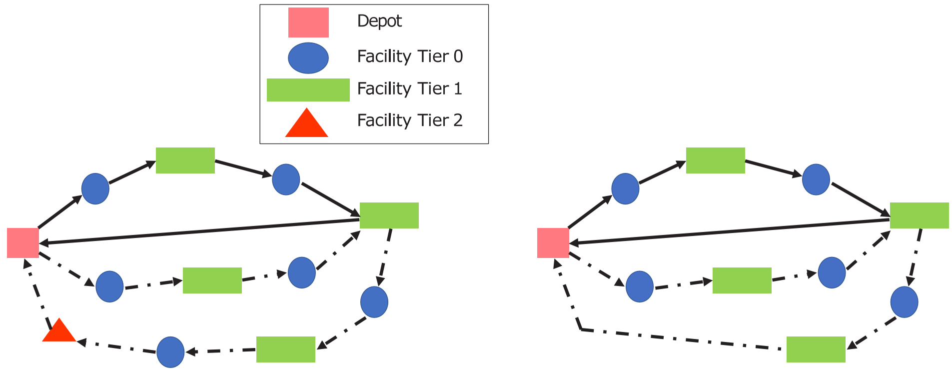
4. Mathematical model formulation

This section contains a detailed description of the sets of constraints and planning objectives of the MTVRPGC in a formal MILP model of the problem. The objectives and constraints of the model proposed were developed in consultation with the manager of a transportation network within South Africa. In order to determine the global ordering of vehicle arrivals, a set of global event numbers is introduced in Section 4.1. After defining the model variables in Section 4.2, the model objectives are formulated mathematically in Section 4.3. The focus then shifts in Section 4.4 to the formulation of the model constraints.

4.1. Global ordering of vehicle arrivals

Let $\mathcal{N}$ denote a set of global event numbers associated with the vehicle routing schedule over the planning period. The elements of this set induce a global ordering of vehicle service starting times at the various facilities in the spirit of Dondo et al. (2011) (who considered the special case of local cross-docking in supply chain management). In their application, the service starting times of each vehicle at a pre-specified local cross-docking facility was associated with a unique integer value in such a manner that a later service starting time of any vehicle at the facility was associated with a larger integer value. These values were employed in a two-echelon VRP so as to reflect real-life distribution problems in which several vehicles may stop at the same manufacturing site or warehouse to accomplish pickup or delivery operations. In this kind of application, a vehicle may be visiting a source node several times during the same tour, and product requirements at some destination may be satisfied through various partial shipments using more than one vehicle. Therefore, a sequence of operations may be performed at every location and a vehicle stop is no longer characterised by just the visited node. Dondo et al. (2011) overcame this obstacle by including an ordered set of event numbers in their model. In our application, we also adopt the practice of assigning the service starting time of each vehicle a unique integer value. Our application, however, differs from that of Dondo et al. (2011) in that we consider the service starting times of all vehicles at *all* the facilities in the network as opposed to at a specific cross-docking facility only. The integer values included in the set $\mathcal{N}$ are representative of the global service starting time sequence of vehicles at all destination facilities of the network. This sequence facilitates monitoring of the global cross-docking and tier-visitation of vehicles.

A summary of all the sets and parameters, described in detail here and in Section 3, that are required to model the MTVRPGC mathematically may be found in Table 1.



(a) Feasible solution

(b) Infeasible solution

Fig. 1. Vehicle visitation sequence feasibility requirements.

Table 1
Table of notations.

Notation	Description
f	The cardinality of the set used to denote the different tiers associated with the respective facilities based on their commodity processing capabilities.
$\mathcal{F}^d \quad (d \in \{0, \ldots, f\})$	The subset of all facilities that are capable of processing a commodity of type d or lower.
$\mathcal{F}$	The set of all facilities within the transportation network.
$\gamma_i \quad (i \in \mathcal{F})$	The maximum number of vehicles facility $i \in \mathcal{F}$ can process within a single planning horizon.
$s_i \quad (i \in \mathcal{F})$	The service time associated with facility $i \in \mathcal{F}$.
$[a_i, g_i] \quad (i \in \mathcal{F})$	The time window during which facility $i \in \mathcal{F}$ is able to process arriving commodity vehicles.
$\mathcal{V}$	The set of all available commodity collection vehicles within the transportation network.
μ	The maximum allowable time associated with a single vehicle's route.
$\mathcal{D}$	The subset of facilities that act as the home depot for vehicles within the transportation network.
$b_k \quad (k \in \mathcal{V})$	The home depot of vehicle $k \in \mathcal{V}$, with $b_k \in \mathcal{D}$.
$b_k^+ \quad (k \in \mathcal{V})$	An identical copy of the home depot of vehicle $k \in \mathcal{V}$ with $b_k^+ \in \mathcal{D}$ used to denote the depot at the subsequent return time of the vehicle at its depot.
$T'_{b_k k} \quad (k \in \mathcal{V})$	The departure time of vehicle $k \in \mathcal{V}$ from its respective home depot $b_k \in \mathcal{D}$.
$\mathcal{S}$	The set of all the possible different commodity types.
$\mathcal{G} = (\mathcal{F}, \mathcal{E})$	A complete directed weighted graph with vertex set $\mathcal{F}$ and arc set $\mathcal{E}$ representing all possible road connections between facilities in $\mathcal{F}$.
t_{ij}	The travel time associated with traversing arc $(i,j) \in \mathcal{E}$.
$\mathcal{N}$	A global set of event numbers associated with the vehicle routing schedule over the planning period.

4.2. Model variables

In our model formulation, decision and auxiliary variables are required to keep track of the movement of vehicles and their service allocation to facilities. In order to facilitate the orchestration of global cross-docking operations, a global ordering is assigned to the service starting times of all vehicles in the routing schedule, as described above. The auxiliary variables

$$y_{nik} = \begin{cases} 1, & \text{if the service starting time of vehicle } k \in \mathcal{V} \text{ at facility} \\ & i \in \mathcal{F} \text{ is global event } n \in \mathcal{N} \\ & \text{during the current planning period,} \\ 0, & \text{otherwise} \end{cases}$$

achieve this purpose in conjunction with the auxiliary variables

$$z_{ijkn} = \begin{cases} 1, & \text{if the service starting time of vehicle } k \in \mathcal{V} \\ & \text{at facility } i \in \mathcal{F} \setminus (\mathcal{F}^0 \cup \mathcal{F}^f) \text{, visited as a} \\ & \text{collection facility, is global event } n \in \mathcal{N} \text{, following} \\ & \text{which vehicle } k \text{ also visits} \\ & \text{facility } j \in \mathcal{F}^\ell \text{ (with } j \neq i \text{) as a consolidation} \\ & \text{facility at some later stage, where} \\ & \text{facilities } i \text{ and } j \text{ are of the same tier } \ell, \\ 0, & \text{otherwise.} \end{cases}$$

It follows that $|\mathcal{N}| \leq |\mathcal{F}^0| + |\mathcal{V}||\mathcal{F} \setminus \mathcal{F}^0|$. The assignment decision variables

$$r_{ikn} = \begin{cases} 1, & \text{if global event } n \in \mathcal{N} \text{ involves the assignment} \\ & \text{of vehicle } k \in \mathcal{V} \text{ to visit facility} \\ & i \in \mathcal{F} \setminus \mathcal{F}^f \text{ as a collection facility and this vehicle} \\ & \text{later visits a facility of a higher} \\ & \text{tier than that of facility } i, \\ 0, & \text{otherwise} \end{cases}$$

are used in a disjunctive fashion to enforce appropriate facility visitation sequences. Finally, the flow decision variables

$$x_{ijk} = \begin{cases} 1, & \text{if vehicle } k \in \mathcal{V} \text{ travels directly from facility } i \in \mathcal{F} \\ & \text{to facility } j \in \mathcal{F}, \\ 0, & \text{otherwise} \end{cases}$$

monitor the movement of vehicle $k \in \mathcal{V}$, while the non-negative, real auxiliary variables T_{ik} denote the time at which vehicle $k \in \mathcal{V}$ starts its service at facility $i \in \mathcal{F}$, with T_{ik} assuming the value zero if vehicle k does not visit facility i.

4.3. Model objectives

Following the discussion in Section 1, the aim of the model proposed in this paper is to pursue an acceptable trade-off between the realisation of three objectives. The first of these objectives is to minimise the expected global travel time[4] associated with the transportation of all commodities from the various original commodity collection facilities

[4] The decision not to minimise the distance travelled by vehicles stems from possibly very rural locations of some of the facilities. The potentially poor quality of roads leading to these remote facilities in a developing context could bring about considerable deviations in the expected travel time per unit distance.

to appropriate facilities where they are to be processed or stored (during a single planning period). This objective may be formulated mathematically as

$$\text{minimise} \quad \sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{F}}\sum_{k\in\mathcal{V}} t_{ij}x_{ijk}. \tag{1}$$

The second objective is to minimise the longest travel time of the delivery vehicles in terms of their total service travel times, that is to

$$\text{minimise} \max_{k\in\mathcal{V}} \quad (T_{b_k^+k} - T'_{b_kk}). \tag{2}$$

The final objective is to

$$\text{minimise} \quad \sum_{k\in\mathcal{V}}\sum_{j\in\mathcal{F}} x_{b_kjk}, \tag{3}$$

which is equivalent to minimising the number of vehicles required for commodity collection at a service level of 100% by reducing the number of vehicles departing from their home depots.

4.4. Model constraints

The model includes numerous constraints reflecting the various commodity transportation requirements outlined in Section 3. The first set of such constraints states that every vehicle utilised must initially depart from and eventually return to its home depot at the end of its route, as required by Assumption 3 of Section 3. This constraint is enforced by requiring that

$$\sum_{j\in\mathcal{F}} x_{b_kjk} \leq 1, \qquad k\in\mathcal{V}$$

and that

$$\sum_{j\in\mathcal{F}} x_{jb_k^+k} = \sum_{j\in\mathcal{F}} x_{b_kjk}, \qquad k\in\mathcal{V}.$$

The constraint set

$$\sum_{i\in\mathcal{F}} x_{ijk} \leq \sum_{\ell\in\mathcal{F}} x_{b_k\ell k}, \qquad j\in\mathcal{F}, \quad k\in\mathcal{V}$$

ensures that any vehicle $k\in\mathcal{V}$ visits a facility $j\in\mathcal{F}$ at most once during the planning period according to Assumption 4. The flow conservation constraint set

$$\sum_{i\in\mathcal{F}} x_{ijk} - \sum_{\ell\in\mathcal{F}} x_{j\ell k} = 0, \qquad j\in\mathcal{F}\setminus\mathcal{D}, \quad k\in\mathcal{V}$$

states that if any vehicle $k\in\mathcal{V}$ arrives at facility j, then the same vehicle must traverse an arc departing from facility j, for all $j\in\mathcal{F}\setminus\mathcal{D}$. Since all facilities $i\in\mathcal{F}\setminus(\mathcal{F}^f\cup\mathcal{D})$ necessarily exhibit demand for commodity collection during the planning period, the constraint set

$$\sum_{j\in\mathcal{F}}\sum_{k\in\mathcal{V}} x_{ijk} \geq 1, \qquad i\in\mathcal{F}\setminus(\mathcal{F}^f\cup\mathcal{D})$$

ensures that at least one vehicle $k\in\mathcal{V}$ should visit facility $i\in\mathcal{F}\setminus(\mathcal{F}^f\cup\mathcal{D})$. In addition, the constraint set

$$\sum_{j\in\mathcal{F}}\sum_{k\in\mathcal{V}} x_{ijk} = 1, \qquad i\in\mathcal{F}^0$$

ensures that each facility $i\in\mathcal{F}^0$ is visited by exactly one vehicle (see Assumption 4). The constraint set

$$T_{ik} + s_i + t_{ij} - T_{jk} \leq (1-x_{ijk})M, \qquad i\in\mathcal{F}, \quad j\in\mathcal{F}, \quad k\in\mathcal{V}$$

is included to monitor the service starting time of vehicle $k\in\mathcal{V}$ at each vertex along its route. This constraint set ensures, if vehicle $k\in\mathcal{V}$ travels from facility $i\in\mathcal{F}$ to facility $j\in\mathcal{F}$, that the time instant at which it starts to service facility j is bounded from below by the time instant at which it started servicing facility i together with the combined service time duration at facility i and the time required to travel from facility i to facility j. Here M is a large positive number. The services provided by the processing facilities should furthermore be rendered within acceptable time windows associated with each facility according to Assumption 5. Since there is a possibility that not all vehicles $k\in\mathcal{V}$ are utilised, the constraint set

$$T'_{b_kk} + t_{b_kj} - M(1-x_{b_kjk}) \leq T_{jk}, \qquad j\in\mathcal{F}, \quad k\in\mathcal{V}$$

defines the service starting time of vehicle $k\in\mathcal{V}$ at the first facility $j\in\mathcal{F}$ visited by vehicle k, where M is again a large positive number. The constraint set

$$a_i\sum_{j\in\mathcal{F}} x_{jik} \leq T_{ik} \leq g_i\sum_{j\in\mathcal{F}} x_{jik}, \qquad i\in\mathcal{F}, \quad k\in\mathcal{V}$$

states that vehicle k may not start its service at a facility $i\in\mathcal{F}$ outside of its associated time window and enforces the requirement mentioned above that if vehicle $k\in\mathcal{V}$ does not visit facility $i\in\mathcal{F}$, the value of T_{ik} is equal to zero. If vehicle k is not utilised, the values of T_{ik} should be equal to zero for all $i\in\mathcal{F}$. The constraint set

$$T_{b_k^+k} - T'_{b_kk} \leq \mu, \qquad k\in\mathcal{V}$$

ensures that vehicle $k\in\mathcal{V}$ does not undertake a route that is expected to take longer to complete than the allowable time autonomy level assigned to the vehicle.

Every facility tier has an associated processing capability in respect of commodities. As the model does not, however, track individual commodity processing requirements, the more practical approach described in Assumption 8 is adopted whereby the number of vehicles arriving at a facility is limited in order to prevent processing bottlenecks. The constraint set

$$\sum_{k\in\mathcal{V}}\sum_{i\in\mathcal{F}} x_{ijk} \leq \gamma_j, \qquad j\in\mathcal{F}\setminus\mathcal{F}^0$$

requires that the number of vehicles arriving at facility $j\in\mathcal{F}\setminus\mathcal{F}^0$ should not exceed the arrival capacity of the facility over the scheduling window. The novelty of the MTVRPGC is further showcased by the remaining constraint sets, which all contribute to controlling the sequencing of vehicle service starting times at facilities so as to facilitate global cross-docking. The constraint set

$$\sum_{j\in\mathcal{F}} x_{jik} = \sum_{n\in\mathcal{N}} y_{nik}, \qquad i\in\mathcal{F}, \quad k\in\mathcal{V}$$

ensures that an event $n\in\mathcal{N}$ cannot be assigned to the service starting time of a vehicle $k\in\mathcal{V}$ at a facility $i\in\mathcal{F}$, unless vehicle k actually visits facility i, and each service starting time is assigned a unique global event index. It is required that the global event indices assigned to vehicle service starting times should reflect the order of their service starting time sequence in global time. The constraint set

$$T_{j\ell} - T_{ik} \geq M(y_{nik} + y_{mj\ell} - 2), \qquad i,j\in\mathcal{F}, \quad k,\ell\in\mathcal{V}, \quad m,n\in\mathcal{N} : m > n$$

achieves this requirement by ensuring that $T_{j\ell} \geq T_{ik}$ if $y_{nik} = 1$ and $y_{mj\ell} = 1$. Here M is again a sufficiently large positive number. For every facility $i\in\mathcal{F}\setminus\mathcal{F}^f$ there must be some vehicle $k\in\mathcal{V}$ visiting a higher-tier facility at some time after having visited facility i, as explained in Assumptions 7 and 8. The disjunctive constraint sets

$$\sum_{k\in\mathcal{V}}\sum_{n\in\mathcal{N}} r_{ikn} = 1, \qquad i\in\mathcal{F}^0$$

and

$$\sum_{k\in\mathcal{V}}\sum_{n\in\mathcal{N}}\left(r_{ikn} + \sum_{j\in\mathcal{F}^\ell\setminus\{i\}} z_{ijkn}\right) = 1, \qquad i\in\mathcal{F}^\ell, \quad \ell\in\{1,\ldots,f-1\}$$

enforce this requirement. These constraint sets ensure that for each facility i of tier $\ell < f$ there exists a vehicle $k\in\mathcal{V}$ visiting the facility with a corresponding event number $n\in\mathcal{N}$ such that $r_{ikn} = 1$ (indicating that vehicle k later visits some facility of a tier higher than ℓ) or (if $i\in\mathcal{F}^\ell$ with $\ell\in\{1,\ldots,f-1\}$) $z_{ijkn} = 1$ for some facility j of tier ℓ, provided that $j \neq i$ (indicating that vehicle k later visits facility j), in accordance with Assumption 7. The second disjunctive constraint set may, however, allow for the situation where a commodity is transported by several vehicles to facilities of the same tier without eventually reaching a facility of a higher tier (as shown in Fig. 1(b)). In order to avoid this kind of situation, the constraint set

$$\sum_{k\in\mathcal{V}}\sum_{n\in\mathcal{N}} r_{ikn} \geq \sum_{j\in\mathcal{F}^{\ell}\setminus\{i\}}\sum_{h\in\mathcal{V}}\sum_{n\in\mathcal{N}} z_{jihn}/|\mathcal{F}^{\ell}|, \qquad i\in\mathcal{F}^{\ell}, \quad \ell\in\{1,\dots,f-1\}$$

is introduced, which implies that if facility i acts as a "consolidation facility" for one or more facilities of tier ℓ, visited as collection facilities by a vehicle h which later visits facility i, *i.e.*

$$\sum_{j\in\mathcal{F}^{\ell}\setminus\{i\}}\sum_{h\in\mathcal{V}}\sum_{n\in\mathcal{N}} z_{jihn} \geq 1,$$

the facility i is visited as a collection facility by a vehicle k which later visits a facility of a higher tier, *i.e.*

$$\sum_{k\in\mathcal{V}}\sum_{n\in\mathcal{N}} r_{ikn} = 1.$$

The above constraint set still allows for multiple vehicles to visit the same facility (as shown in Fig. 1(a)), but z_{ijkn} may only assume the value 1 for one of the routes along which the facility is visited by a vehicle that visits a facility of the same tier at a later stage. The linking constraint set

$$|\mathcal{F}|y_{nik} + \sum_{\substack{m\in\mathcal{N}\\ m>n}}\sum_{j\in\cup_{\ell=c+1}^{f}\mathcal{F}^{\ell}} y_{mjk} \geq (|\mathcal{F}|+1)r_{ikn}, \quad i\in\mathcal{F}^{c}, \; c\in\{0,\dots,f-1\}, \quad k\in\mathcal{V}, \; n\in\mathcal{N}$$

furthermore ensures that the variable r_{ikn} may only assume a value of 1 if vehicle $k\in\mathcal{V}$ actually visits facility $i\in\mathcal{F}^{c}$ and at some later stage also visits facility j of a tier higher than c. The powerful disjunctive constraint sets above depend on the values of the auxiliary variables r_{ikn}. The linking constraint set

$$\sum_{n\in\mathcal{N}} r_{ikn} \leq \sum_{n\in\mathcal{N}} y_{nik}, \qquad i\in\mathcal{F}\setminus\mathcal{F}^{f}, \quad k\in\mathcal{V}$$

enforces the correct assignment of values to these binary variables. The global cross-docking component of the model allows for facilities of the same tier to have their commodities consolidated at any facility of that tier within the transportation network. The constraint set

$$y_{nik} + \sum_{\substack{m\in\mathcal{N}\\ m>n}} y_{mjk} \geq 2z_{ijkn}, \qquad i,j\in\mathcal{F}^{\ell}, \quad \ell\in\{1,\dots,f-1\}, \quad n\in\mathcal{N}, \quad k\in\mathcal{V}$$

ensures that the auxiliary variable z_{ijkn} only assumes the value 1 if vehicle k visits facility $i\in\mathcal{F}^{\ell}$ (with $\ell\neq 0,f$) and then at a later time also visits facility $j\in\mathcal{F}^{\ell}$, allowing for consolidation of commodities of facility i at facility j, to be collected by a different vehicle $k\in\mathcal{V}$ for transportation to a higher-tier facility. Finally, the computational burden associated with satisfying the aforementioned constraints may be lowered by introducing the symmetry-breaking constraint set

$$\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{F}} x_{ijk} \geq \sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{F}} x_{ijk+1}, \qquad k\in\{1,\dots,|\mathcal{V}|-1\}.$$

This constraint set ensures that the number of facilities visited by vehicle $k\in\mathcal{V}$ is not smaller than the number of facilities visited by vehicle $k+1$.

5. A worked example

The logic of the model in Section 4 is verified in this section by implementing it in a commercially available MILP solver within the context of small, hypothetical problem instances. A worked example, based on a hypothetical instance with seven facilities, is first described in detail. Computational results on an additional sixteen hypothetical instances (with up to ten facilities) are reported later. In these hypothetical instances, the number of facilities within each respective tier and the number of vehicles vary. The aim of the worked example is not to evaluate experimentally the computational performance of the proposed MILP model (which could be substantially improved upon by applying effective preprocessing procedures to decrease the number of variables and constraints while also incorporating more efficient branch-and-bound protocols), but to demonstrate its capability and robustness to deal with the novel global cross-docking properties and the peculiar constraints of the model proposed in Section 4.

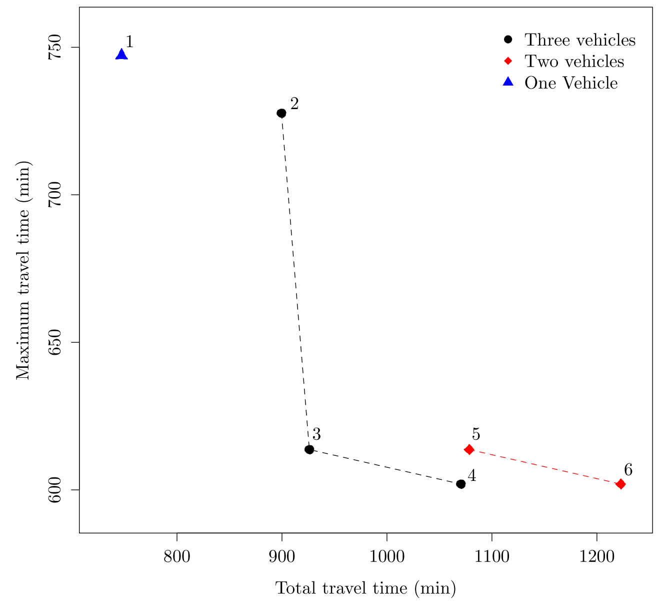


Fig. 2. True Pareto fronts for the first hypothetical test problem instance consisting of the seven facilities of Tables A.6 and A.7 in the cases of using one, two and three vehicles, respectively.

5.1. First hypothetical instance

In the first hypothetical instance there are seven facilities (*i.e.* $|\mathcal{F}|=7$) of three different tiers, and so $f=2$ in this case. The first of these facilities is depot (acting as home depot for all vehicles). Facilities 2, 5 and 6 are hospitals or clinics at which pathology samples originate. These collection stations have no commodity processing capabilities, and so they are classified as facilities of tier zero. Facilities 3 and 4 are hospitals at which commodity processing laboratories of tier one are located, while Facility 7 is a tier-two laboratory.

The coordinates of the seven facilities and the travel times (expressed in minutes) between these facilities are shown in Tables A.5 and A.6 of Appendix A, respectively, and were calculated as the corresponding Euclidean distances (rounded up) between the facilities.

This instance was constructed in a manner to highlight the concept of global cross-docking and hence some of the problem input data described in Section 3 not affecting cross-docking, such as the imposition of time windows and the adherence to arrival capacities of facilities, were set to generally unconstraining values so as to reduce the complexity of finding an initial feasible solution.

A complete enumeration of all feasible routes was performed, implemented in Wolfram's Mathematica (Wolfram, 2016), in order to generate the true Pareto front for the hypothetical problem instance, with a view to validate the logic of the model in the cases where either $|\mathcal{V}|=2$ or $|\mathcal{V}|=3$ delivery vehicles are employed (with $b_k=1$ and $b_k^{+}=8$). The enumeration process is described in Appendix A, yielding only three (for $|\mathcal{V}|=2$) and two (for $|\mathcal{V}|=3$) Pareto-optimal vehicle routing combinations, as depicted in objective function space in Fig. 2.

Although it violates the maximum travel time bound of 740 mins per vehicle, the objective function values of the optimal solution single-vehicle *travelling salesman problem* (TSP) are also included for reference purposes in Fig. 2.

Table 2
Computational times (expressed in seconds) required by CPLEX 12.9 to generate the solutions in Figs. 3(b) and 3(f) on an i7-8550 processor running at 1.80 GHz with a working memory limit of 6 GB within the Windows 10 operating system.

Solution	2	6
Time to find initial feasible solution	418.65 s	370.10 s
Time to find an optimal solution	1 741.68 s	1 980.07 s
Time to prove optimality	8 995.54 s	14 518.33 s

The six numbered solutions in Fig. 2 are depicted in solution space in Fig. 3. Among these solutions, the concept of global cross-docking is clearly illustrated in Solutions 3 and 5.

The mathematical model of Section 4 was also implemented in CPLEX 12.9 in respect of the problem instance described above in an attempt to validate the logic of the mathematical model formulation. In order to accommodate the pursuit of trade-offs between minimising the total travel time and minimising the longest travel time of the vehicles in a solution, the number of vehicles utilised was fixed first as $|\mathcal{V}| = 2$ and then as $|\mathcal{V}| = 3$. Since CPLEX 12.9 can only handle multi-objective MILPs in a hierarchical way, we decided to focus our CPLEX search on replicating Solutions 2 and 6, respectively. This allows for single-objective consideration, since the number of vehicles may be fixed, as described above, after which the non-relevant model objective may simply be disregarded.

Accordingly, the number of vehicles was fixed at two and objective (2) (see Section 4.3) was removed from consideration in order to replicate Solution 2. The values of the non-zero decision variables returned by CPLEX in this case are shown in Table A.7 of Appendix A. The facility index 8 in the tables refers to the virtual copy of the depot (Facility 1).

The total travel time of the two vehicles in Solution 2 is 899.53 min, while the times spent travelling by vehicles 1 and 2 are, respectively, 171.87 and 727.66 min, giving a maximum travel time of the vehicles of 727.66 min.

Similarly, the number of vehicles was fixed at three and objective (1) (see Section 4.3) was removed from consideration in order to replicate Solution 6. The non-zero decision variables returned by CPLEX in this case are shown in Table A.8 of Appendix A. The total travel time of the three vehicles in solution 6 is 1222.79 min, while the times spent by vehicles 1, 2 and 3 are, respectively, 468.50, 601.97 and 152.32 min, giving a maximum travel time of the vehicles of 601.97 min.

The solutions presented in Tables A.7 and A.8 of Appendix A are exactly those depicted in Figs. 3(b) and 3(f), respectively. The computation times required by CPLEX to reach these solutions are listed in Table 2.

5.2. Additional hypothetical instance

The numerical experiment described above was repeated for eight, nine, and ten facilities, respectively. The data and parameter values for these sixteen instances are available online (Smith, 2019a). Objective (2) was removed from consideration for all instances. The combinatorial explosion associated with solving these problem instances is elucidated in Appendix B. The CPLEX implementation was allocated a computational budget of 200 000 s and was not able to prove optimality for the instances with ten facilities within the allotted time.

It is evident from Tables B.9–B.12 that the computational time required to solve the mathematical model exactly is extremely high, even for small instances. This is to be expected in view of the model versatility and complexity. In order to find good approximate solutions to large, real-world instances of the MTVRPGC, a MACO algorithm is proposed in the following section.

6. Description of the MACO algorithm

It was decided to employ a MACO algorithm to solve the MTVRPGC approximately based on the results published by García-Martínez et al. (2007) and the underlying constructive behaviour of MACO algorithms. The basic notion is for the algorithm to emulate the foraging behaviour of ants and through the use of residual pheromone trails, guide the algorithm towards high-quality solutions. The MACO algorithm presented in this paper differs from existing algorithms within the literature due to the mechanisms developed to handle the unique constraints of the MTVRPGC, such as the facility visitation sequence. The principles and mechanisms of the MACO algorithm are described in detail in the remainder of this section.

Our underlying algorithmic approach towards solving instances of the MTVRPGC approximately is to employ three single-objective *ant colony systems* (ACSs) for the construction of initial vehicle routes based on a colony heuristic and a collection of pheromone matrices. Local update mechanisms are subsequently employed in respect of each colony's pheromone. Routing sequences that are infeasible with respect to higher-tier visitations are then identified and corrected by means of several procedures tailored specifically to the MTVRPGC, adopting a top-down correction paradigm. The algorithm employs three colonies, each focusing on a single objective (*i.e.* minimisation of the total travel time, minimisation of the maximum travel time of the vehicles, and compromise solutions between total travel time and maximum travel time for vehicles). The minimisation of the number of vehicles is not explicitly modelled in the MACO algorithm. The routes are then collectively examined and penalty weights are applied with respect to the remaining constraints, such as time windows and maximum travel time for vehicles violations, after which a global ranking of the solution population is determined according to the standard NSGA-II ranking mechanism (Deb et al., 2002), complemented by a crowding distance function. Finally, a global pheromone update mechanism is applied to the respective pheromone matrices.

The initial route construction mechanism consists of Q ants concurrently building routes from starting vertices chosen randomly within a set $\mathcal{F}$ of facility vertices. At each construction step, each ant applies a probabilistic proportionality rule, as suggested by Wang et al. (2016), to determine which vertex to visit next. The vertex selection mechanism involves the following three parameters:

1. A heuristic value η_{ij} denotes the attractiveness of a move along the arc joining vertex i to vertex j.
2. A parameter τ_{ij} denotes the pheromone level along the arc joining vertex i to vertex j, which is an indication of the past usefulness of the arc in previous route constructions.
3. A parameter s_{ij} denotes the savings value[5] associated with the inclusion of vertices i and j in a single vehicle route.

The route construction process also involves three parameters, α, β, and φ. During route construction, ant q, located at vertex i, moves to an adjacent vertex according to the following pseudo-random proportionality rule: A real number λ is randomly generated within the interval $[0, 1]$ according to a uniform distribution, and if λ is below a pre-determined threshold λ_o, the index of the vertex visited next is

$$n = \mathrm{argmax}_{j \in C_i^q} \{(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}(s_{ij})^{\varphi}\}, \tag{4}$$

where C_i^q denotes the set of feasible neighbours of vertex i for ant q. The maximum travel time μ for vehicles in the MTVRPGC is utilised as a stopping criterion during route construction, as opposed to the typical capacity constraint in the capacitated VRP. In an attempt to allow for more flexibility during the correction phase of the algorithm,

[5] The savings values are calculated according to the method proposed by Clarke and Wright (1964).

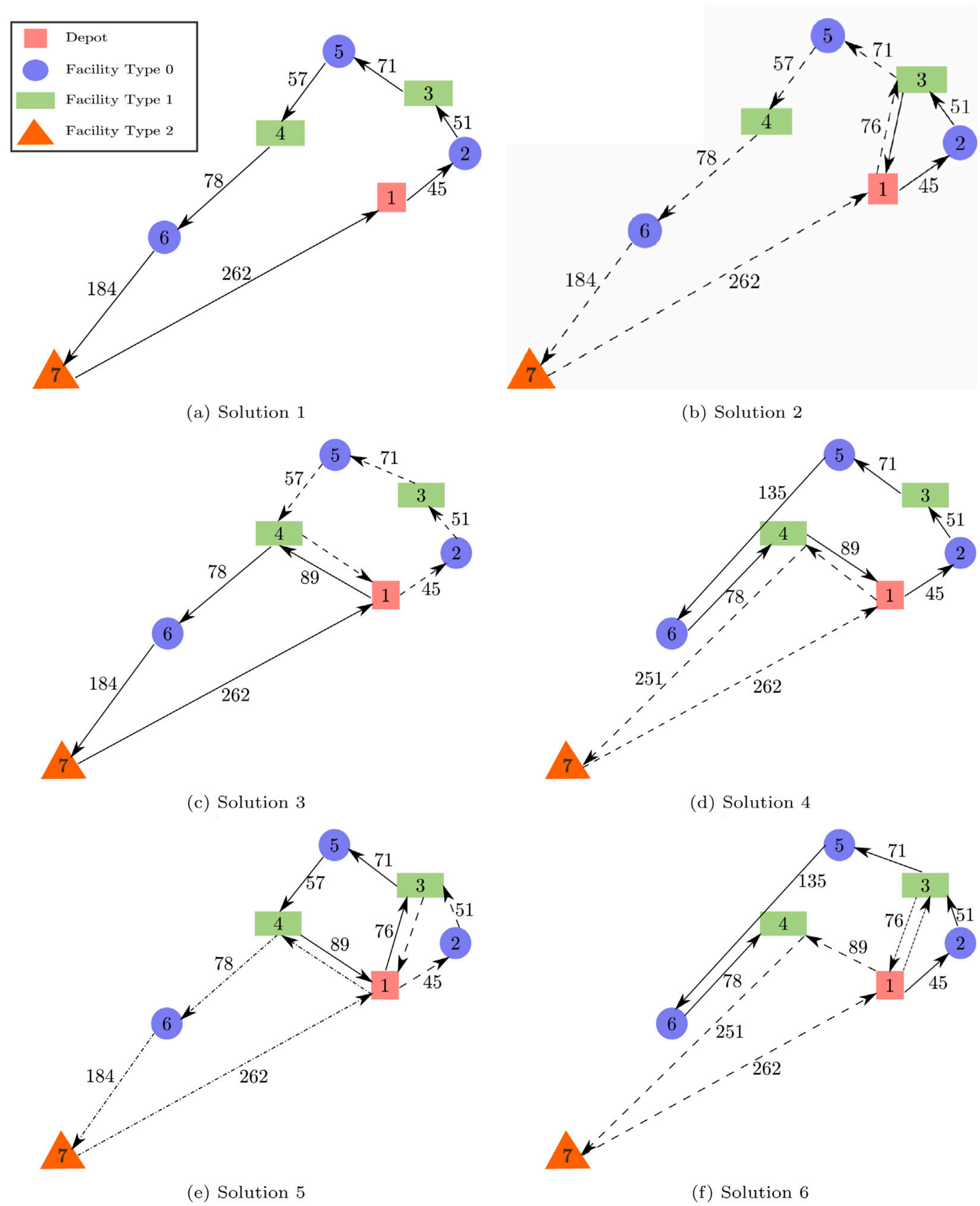


Fig. 3. The numbered solutions reported in objective function space in Fig. 2 are depicted here in solution space. The travel times reported in the figures are rounded to the nearest integer.

a random number r_a is generated within the interval $[\ell_b, 1]$, where $\ell_b \in (0, 1)$ is a predefined parameter. The new maximum travel time μ' for vehicles is determined as $\mu' = \mu r_a$. The adjusted maximum travel time for vehicles value μ' is uniquely determined for each vehicle utilised for commodity transportation and incorporated in the identification of feasible neighbours. Otherwise, if λ is not below the threshold λ_o, the probability of visiting vertex $j \in C_i^q$ next is given by

$$\mathcal{P}_{ij} = \frac{(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}(s_{ij})^{\varphi}}{\sum_{\ell \in C_i^q}(\tau_{i\ell})^{\alpha}(\eta_{i\ell})^{\beta}(s_{i\ell})^{\varphi}}. \tag{5}$$

This probability is employed in conjunction with a roulette wheel mechanism to determine which vertex ant q should visit next, allowing for a biased exploration of the arcs. There are, however, multiple vehicle depots under consideration. The route construction begins by probabilistically selecting a depot based on the distance between the depots and their respective nearest facilities.

The MACO algorithm allows for two phases of pheromone updates, a global updating step and a local updating step. The local pheromone update mechanism is only applied to the specific pheromone matrix under consideration. It is adapted from Tan et al. (2012), and is performed every time an arc is traversed. This local pheromone level along the arc (i, j) is updated as

$$\tau_{ij} \leftarrow \left(\rho + \frac{\delta}{L_k}\right)\tau_{ij}, \tag{6}$$

where ρ and δ are both user-defined parameters. The parameter ρ is referred to as *trail persistence* in the literature and is typically a real value in the interval $[0, 1]$, while δ is an elitist-related parameter which typically takes an integer value in the interval $[0, L^*]$, where L^* is the total travel time of the incumbent solution. The variable L_k refers to the total travel time of the route traversed by ant q.

A global updating step is, however, only applied to those arcs that appear in the best solutions uncovered by the entire colony of ants, by applying the substitution

$$\tau_{ij} \leftarrow \tau_{ij} \sum_{r=1}^{\sigma} \Delta\tau_{ij}^{r} + \Delta\tau_{ij}^{*} \tag{7}$$

to the relevant arcs, where

$$\Delta\tau_{ij}^{r} = \begin{cases} \frac{(\sigma-r)}{L_k}, & \text{if the } r\text{th best ant traverses arc } (i,j) \\ 0, & \text{otherwise} \end{cases}$$

and

$$\Delta\tau_{ij}^{*} = \begin{cases} \frac{\sigma}{L^*}, & \text{if arc } (i,j) \text{ is contained within the incumbent solution} \\ 0, & \text{otherwise.} \end{cases}$$

Accordingly, only the σ most elitist ants will deposit a pheromone trail in which the solution quality returned by the ant determines the quantity of pheromone deposited by the ant. The value of σ is known *a priori*. This approach was suggested by Bullnheimer et al. (1997) in an attempt to provide strong additional reinforcement of the edges belonging to the best solutions found so far. The incorporation of the ranking mechanism is aimed at avoiding the danger of over-emphasised pheromone trails caused by many ants following suboptimal routes.

The σ most elitist ants are determined based on the objective under consideration. The global pheromone update for the colony focusing on minimising the total travel time is performed only in respect of the best solutions in terms of the shortest total travel time. The global pheromone update for the colony focusing on minimising maximum travel times of vehicles occurs in a similar manner, although only the best solutions are considered with respect to minimisation of maximum travel time of vehicles.

The global pheromone update for the colony aiming to discover compromise solutions is designed to improve solution diversity with a view to encourage exploration and discovery of multiple compromise solutions. The first mechanism employed to this effect is a dynamic archive of best solutions found. The dynamism of the archive refers not only to the storage of non-dominated solutions, but also to the storage of solutions that are dominated by e other solutions, with the value of e tending towards zero as the algorithm execution progresses. The calculation of a crowding distance c_i (Deb et al., 2002) is required for each solution i in the archive, excluding the extremal solutions (*i.e.* solutions on either ends of the respective approximate Pareto front). The crowding factor for solution i is determined as

$$F_i = \frac{c_i - \bar{c}}{\bar{c}}, \tag{8}$$

where $\bar{c}$ is the mean crowding distance of all the solutions in the archive (excluding the extremal solutions). Each solution s in the archive is ranked twice—once according to the overall travel time and once according to the maximum travel time of the vehicles—with a crowding factor incorporated into each ranking. The two mixture pheromone matrices, an overall travel time pheromone matrix $\boldsymbol{\tau}_a$ and a maximum travel time of the vehicles pheromone matrix $\boldsymbol{\tau}_b$, are updated by the global pheromone update mechanism based on their respective weightings. The overall pheromone matrix for colonies aiming to discover compromise solutions is calculated by applying the substitution

$$\boldsymbol{\tau} = \psi_r \boldsymbol{\tau}_a + (1-\psi_r)\boldsymbol{\tau}_b, \tag{9}$$

where ψ_r is a randomly generated number in the interval $[0.25, 0.75]$ in an attempt at biasing the compromise solution search into unexplored areas of the solution space.

Minimisation of the number of vehicles utilised for commodity transportation is not explicitly modelled in the MACO algorithm. Instead, a separate Pareto front is traced out for each distinct number of delivery vehicles employed.

The (i,j)th entry of the heuristic matrix, $\boldsymbol{\eta} = [\eta_{ij}]_{i,j\in C}$, of the colony dedicated to minimising the total travel time is simply calculated as the inverse of the travel time for the arc (i,j) under consideration.

Determination of the heuristic matrix for the colony that aims to minimise the maximum travel time of all vehicles is slightly more complicated. The constructive nature of ACSs does not allow for the fitness evaluation of the maximum travel time of the vehicles as it is not able to predict the outcome of adding a vertex to a route in respect of the vehicles' travel times. This problem is remedied by generating an initial population of routes that are feasible in terms of maximum travel time for vehicles, but not necessarily in terms of the increasing tier visitation requirement of the MTVRPGC. This initial population is then used to determine the heuristic values for the individual arcs, as described in pseudocode form in Algorithm 1. Each of the routes generated in the initial population is ranked according to its travel time and the arcs constituting the various routes are assigned weighted values according to the ranking of the route. The travel times between the respective facilities are sorted and the corresponding arcs in the heuristic matrix are assigned travel time values based on their weighted values previously assigned. The heuristic value is then taken as the inverse of the travel time value assigned to it so that the heuristic matrices employed by the different colonies are of the same order of magnitude.

Algorithm 1: Heuristic determination of maximum travel time of the vehicles

```
Input : Initial population generated randomly, travel time matrix of
        arcs between facilities
Output: Heuristic matrix
  for i ← 1 to length(population) do
      for j ← 1 to num.routes(population[i]) do
          route.travel.times[length(route.travel.times) + 1] =
          traveltime(population[i, j]);
      end
  end
  route.travel.times = sort(route.travel.times, ascending);
  for i ← 1 to length(route.travel.times) do
      route = which(traveltime(population) == route.travel.times[i]);
      for j ← 1 to length(route) − 1 do
          heursitic.arc[route[j], route[j + 1]] =
          length(route.travel.times) − i;
      end
  end
  unique.entries = unique(heuristic.arc);
  perc = percentiles(unique(route.travel.time), unique.entries);
  for i ← 1 to length(unique.entries) do
      heuristic.arc[heuristic.arc == unique.entries[i]] = 1/perc[i];
  end
  Return heuristic.arc
```

The heuristic matrix of the colony searching for compromise solutions is simply taken as a mixture of the two heuristics previously described, weighting each matrix equally so as to create a single matrix. The route construction process employs the aforementioned probabilistic rule with respect to vertex selection.

The algorithmic implementation incorporates a function aimed at fixing the sequence in which vehicles visit the facilities (*i.e.* ensuring that a facility is visited by a vehicle that later visits a facility of a strictly higher tier or another facility of the same tier visited by a vehicle that later visits a facility of a strictly higher tier). As previously mentioned, the sequence fix function adopts a top-down approach towards rectifying the visitation sequence of facilities. The first stage is aimed at ensuring that at least one route ends at a facility of the highest tier. If the routes generated do not contain such a sequence, a facility of the highest tier is inserted at the end of one of the routes, according to the roulette wheel mechanism with the probabilities calculated based on the insertion costs with respect to travel time.

A middle-tier sequence fix algorithm then considers all infeasible facilities of tiers other than the lowest and highest with respect to facility sequence visitation. These middle-tier sequence infeasibilities

Algorithm 2: Middle-tier sequence fix

```
Input  : Candidate routes, locations of the facilities together with
         their respective tiers
Output: Candidate routes with the tier visitation sequence fixed
  infeasibilities =
  determine.sequence.infeasibilities(routes, locations, tier);
  for i ← 1 to length(infeasibilities) do
      new.route[1] = insert.before(routes, infeasibilities[i]);
      new.route[2] = add.higher.after(routes, infeasibilities[i]);
      new.route[3] = cross.docking(routes, infeasibilities[i]);
      new.route[4] = higher.tier.end(routes, infeasibilities[i]);
      total.travels = travels(new.route, locations);
      random = runif(1,0,1);
      if random ≤ best.select then
          routes = new.route[which.min(total.travels)];
      end
      else
          insert = roulette(total.travel);
          routes = new.route[insert];
      end
  end
  Return routes
```

are rectified according to four paradigms incorporated within a probabilistic sequence fix function. A pseudocode description of this function is given in Algorithm 2.

The first paradigm, *insert.before*, removes those facilities that are not visited by a vehicle that later visits a facility of a higher tier and places them in a different route in which this requirement is indeed satisfied. The route and position is determined from a pool of candidates by means of a weighted probability function. The second paradigm, *add.higher.after*, adds a facility of a higher tier to the route at a later stage with respect to the infeasible facility, with the position of this facility insertion determined probabilistically, based on insertion cost with respect to travel time. The third paradigm, *cross.docking*, encourages the facilitation of global cross-docking. According to this paradigm, all facilities are determined which are visited by a vehicle that later visits a facility of the appropriate tiers. Facilities are selected from this set based on a weighted probability function biased towards lower insertion costs, and they are inserted at a later stage with respect to the infeasible facility. The final paradigm, *higher.tier.end*, simply assesses each infeasible route and adds an appropriately tiered facility at the end of the route. The overall algorithm functions based on a uniformly generated random number. If the random number is smaller than a pre-defined threshold, the repair paradigm associated with the lowest travel time is selected. Otherwise, a paradigm is selected by means of the roulette wheel mechanism.

The final phase of the sequence fix function is to rectify all sequence infeasibilities of facilities of the lowest tier. The lowest-tier fix function similarly incorporates two paradigms, aimed at moving an infeasible facility to a different route, resulting in feasibility (based on insertion cost), or adding a higher tiered facility at a later stage within the route. The algorithm is biased towards adding a higher tiered facility at a later stage of a route if there are numerous infeasible facilities within the route; otherwise, a paradigm is selected according to the roulette wheel mechanism.

The aforementioned sequence fix function is rather disruptive in respect of the quality of vehicle routes, resulting in sub-optimal facility sequence visitation within the routes. Accordingly, a probabilistic heuristic is applied to the routes once all the sequence infeasibilities have been rectified. Three simple paradigms are employed in an attempt to improve the solution quality. According to the first paradigm, it is determined whether cross-docking is present within the route. If cross-docking indeed occurs, then the facility at which consolidation occurs is swapped with another facility of the same tier within the route. The selected facility serves as the new consolidation point and a 2-opt mechanism is performed on the route, keeping the selected facility as the consolidation point. This also involves switching the original consolidation facility with the new facility in all routes that contain the original consolidation facility. A 2-opt mechanism that respects sequence visitation feasibility is applied and if this results in an improvement in solution quality, the proposed swap is adopted. It may, however, happen that only one objective function value is improved, while the other objective function value deteriorates. In such a case, both solutions are stored in the archive. The second paradigm involves simply applying a 2-opt mechanism to the route under consideration while still respecting the sequence visitation constraint. Finally, the third paradigm involves determining which depot is best to act as the home depot for the vehicle under consideration. The three paradigms are selected probabilistically by generating a random number according to a uniform distribution and employing the roulette wheel mechanism based on equal proportions of each paradigm being selected.

A pseudocode description of the entire MACO algorithm is provided in Algorithm 3, highlighting all the relevant components of the algorithm.

Algorithm 3: MACO algorithm

```
Input  : Locations and tiers of facilities, number of ants employed,
         mean and standard deviation for normal distribution,
         maximum travel time of the vehicles for all routes, travel
         time matrix
Output: Non-dominated front of solutions
  population = initialpopulation(locations, autonomy);
  heuristic.travel.time = initial.heuristic.time(locations, population);
  heuristic.arcs = initial.heuristic.arcs(locations, population);
  pheromone.time = pher.time(locations);
  pheromone.arcs = pher.arcs(locations, population);
  pheromone.mixture = 0.5 × pheromon.arcs + 0.5 × pheromone.time;
  for i ← 1 to iterations do
      for cycle ← 1 to 3 do
          if cycle = 1 then
              pheromone = pheromone.time;
              heuristic = heuristic.time;
          end
          else if cycle = 2 then
              pheromone = pheromone.arcs ;
              heuristic = heuristic.arc;
          end
          else
              pheromone = pheromone.mixture ;
              heuristic = heuristic.mixture;
          end
          for m ← 1 to no of ants do
              route = antcolony(locations, autonomy, timewindows);
              route = sequence.fix(route);
              route = post.optimisation(route)
              population = population + route;
          end
          for d ← 1 : length(population) do
              a = autonomy.penalty(population[d]) ×
              sequence.penalty(population[d]);
              x[d] = total.travel.time(population[d]) ×a;
              y[d] = arc.travel.times(population[d]) ×a;
          end
          ranking = NSGA2.ranking((x, y));
          range = (max(x) − min(x), max(y) − min(y));
          front = population[ranking <= pop.keep];
          crowding = crowdingdistance(front, cycle);
          pheromone =
          globalpheromone(front, cycle, pheromone, crowding);
          population = front;
      end
  end
  Return remove.infeasibilities(population)
```

If a vehicle k arrives at a facility i outside the relevant time window, the time-window penalty factor for facility i in the archive is calculated as

$$s(i) = \max\left\{\frac{a_i - T_{ik}}{a_i}, \frac{T_{ik} - g_i}{g_i}\right\} + 1, \tag{10}$$

where T_{ik} is the service starting time of vehicle k at facility i, a_i is the earliest possible service starting time of a vehicle at facility i and g_i is the latest possible service starting time of a vehicle at that facility. The

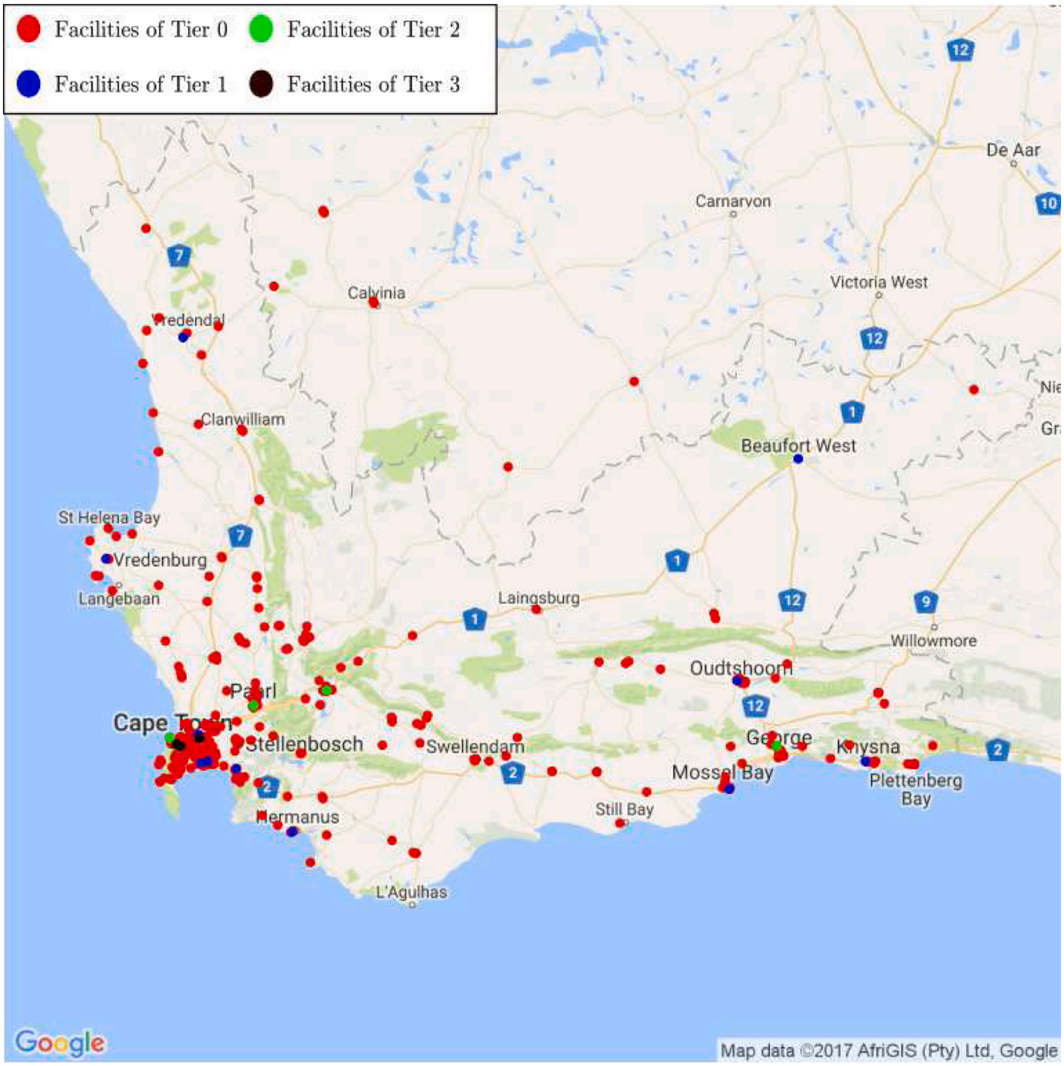


Fig. 4. Geographical distribution of 388 healthcare facilities in the regional network of a Southern African pathology service provider.

autonomy penalty factor for solution j within the archive is

$$e(j) = \max\left\{1, 1 + \frac{L_{mat} - \mu}{\mu}\right\}, \tag{11}$$

where L_{mat} is the maximum travel time of all the routes contained within the solution and μ is the maximum allowable travel time for vehicles.

Suitable values for the parameters employed in the MACO algorithm (as determined during a sensitivity analysis) are finally summarised in Table 3. The parameters α, β and φ are employed in (4) to select the next vertex to visit during route construction probabilistically. The number of ants employed in the MACO algorithm determines how many different potential routes are built during each iteration. As mentioned, a probabilistic rule is utilised to determine which vertices to append to each route. If the random number (generated according to a uniform distribution) is below the best-select parameter value, then the vertex associated with the largest probability is selected; otherwise, the roulette wheel mechanism is employed. The best-select parameter also affects the sequence-fix function, as it determines which of the four routes proposed by the function *sequencefix* is implemented. The practice of including a best-select parameter is popular in the literature (Bell and McMullen, 2004; Wang et al., 2016). It is, in fact, a key component in maintaining solution diversity. The parameter δ serves as an elitist-related parameter, ensuring that solutions of a higher quality are associated with a larger pheromone deposit among

Table 3
The parameter values employed in the MACO algorithm for solving a realistic instance of the MTVRPGC, based on a sensitivity analysis.

Parameter	α	ants	β	best-select	δ	diversity	φ	ℓ_b	ρ	γ_{pen}
Value	2	20	5	1	300	0.7	2	0.6	0.8	1.5

the relevant arcs during the local pheromone update mechanism. The diversity parameter determines the percentage of the archive for that specific colony which consists of solutions with a large crowding distance. The parameter ℓ_b is employed to allow the MACO algorithm the necessary flexibility, after having constructed the initial routes, in order to adhere to the tier-visitation constraint with respect to maximum travel time for vehicles. The parameter ρ is employed in both the local and global pheromone update mechanisms, where it affects the rate of decay within the local pheromone update mechanism and the rate of convergence within the global pheromone update mechanism. Finally, the parameter γ_{pen} determines the magnitude of the penalty function implemented in the MACO algorithm.

Upon extensive numerical experimentation, the algorithmic parameter values in Table 3 were found to perform well in respect of small to medium sized TVRPGC instances.

The MACO algorithm was applied to the same sixteen instances as the CPLEX implementation, as described in Section 5.2, and was able to

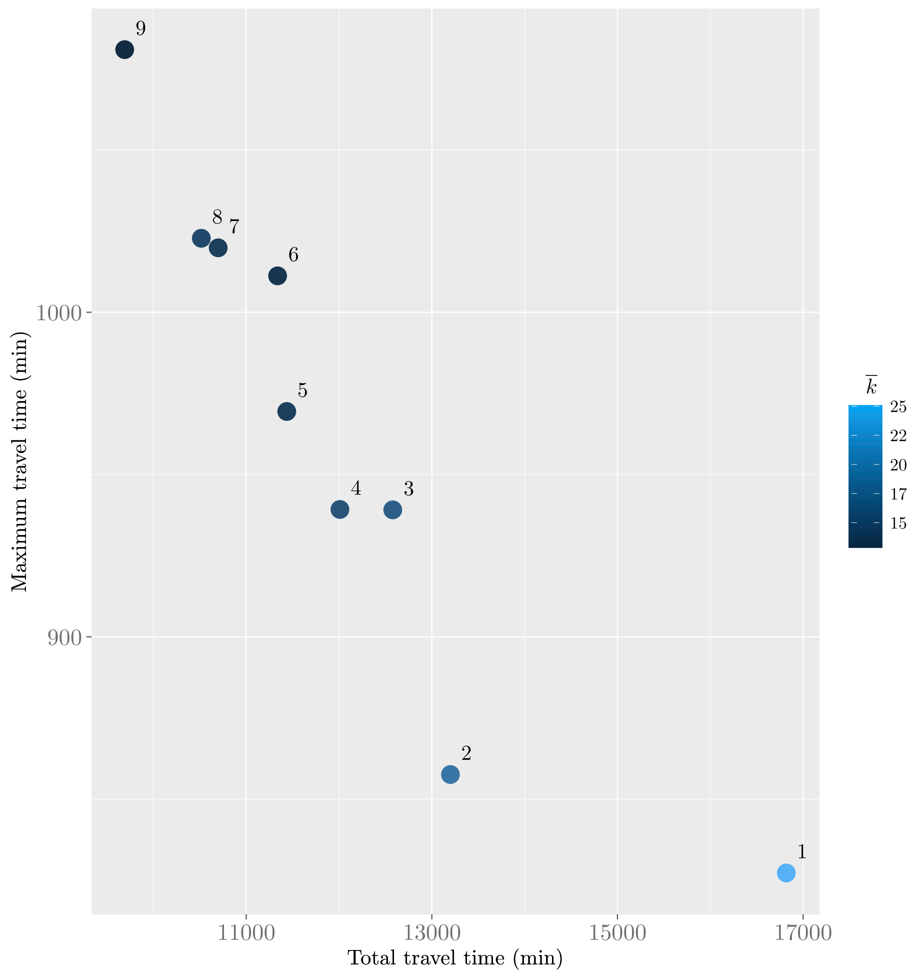


Fig. 5. Approximate Pareto front returned by the MACO algorithm for the case study MTVRPGC instance, utilising $\bar{k}$ vehicles.

find the optimal solutions for all the instances in a fraction of the time required by the CPLEX model. The results may be seen in Appendix B.

7. A real case study

The results returned by the MACO algorithm (described in the previous section) are presented in this section for a real case study involving 388 healthcare facilities within the South African Western Cape. Of these, there are three facilities of tier three, four facilities of tier two, eleven facilities of tier one and 359 facilities of tier zero, as well as eleven vehicle depots located at the facilities of tier one. The locations of these facilities are illustrated in Fig. 4 and their (longitude, latitude) coordinates, offset by a constant value for anonymisation purposes, are available online (Smith, 2019b). The matrix of expected travel times between the facilities shown in Fig. 4 is also available online (Smith, 2019b). The travel times were calculated using estimates provided by *Open Source Routing Machine* (OSRM) (OSRM, 2017).

The routes currently employed by a Southern African pathology service provider to collect and deliver specimens within this regional network result in a total travel time of 15 175.7 minutes and a maximum travel time of the vehicles of 1 080.9 minutes. The aforementioned routes are followed by a total of 64 vehicles, although the motivation behind the choice of this number of vehicles remains unclear as it would seem from the data that two or more routes may often be serviced by a single vehicle. It is possible that the vehicles in question may also have been used for other purposes. In order to generate a more reasonable estimate of the number of vehicles required to service the currently implemented routes for the purpose of comparison with the results generated by the MACO algorithm, the following logic was applied: The maximum travel time of a vehicle was set to 1 081 minutes and routes that end at a facility at which another route begins were allocated to a single vehicle if the maximum travel time for a vehicle constraint was not violated. This process resulted in an estimated 36 vehicles required to service the routes currently employed by the pathology service provider.

Time window information was not available. Accordingly, all the time window model information was specified in a manner so as not to be limiting constraints or result in any penalisation of the objective function values during the algorithmic implementation. The maximum travel time of a vehicle was taken as 1 100 minutes.

The MACO algorithm was applied to the MTVRPGC instance described above and was allowed a limit of 1 000 search iterations after which the incumbent solutions were recorded. All the numerical work reported here was performed on an i7-4770 processor running at 3.40 GHz with 8GB of memory within the Windows 7 operating system after having implemented the MACO algorithm of Section 6 in R.

The approximate Pareto front returned by the MACO algorithm for the case study MTVRPGC instance may be seen in Fig. 5. The non-dominated front contains nine feasible solutions. During execution of the algorithm, approximately thirty non-dominated "solutions" were, however, uncovered, with the majority being removed from final consideration due to constraint violations. This highlights the nature of the MTVRPGC—its solution space is typically tightly constrained, allowing for even heavily penalised infeasible solutions to be competitive with feasible solutions.

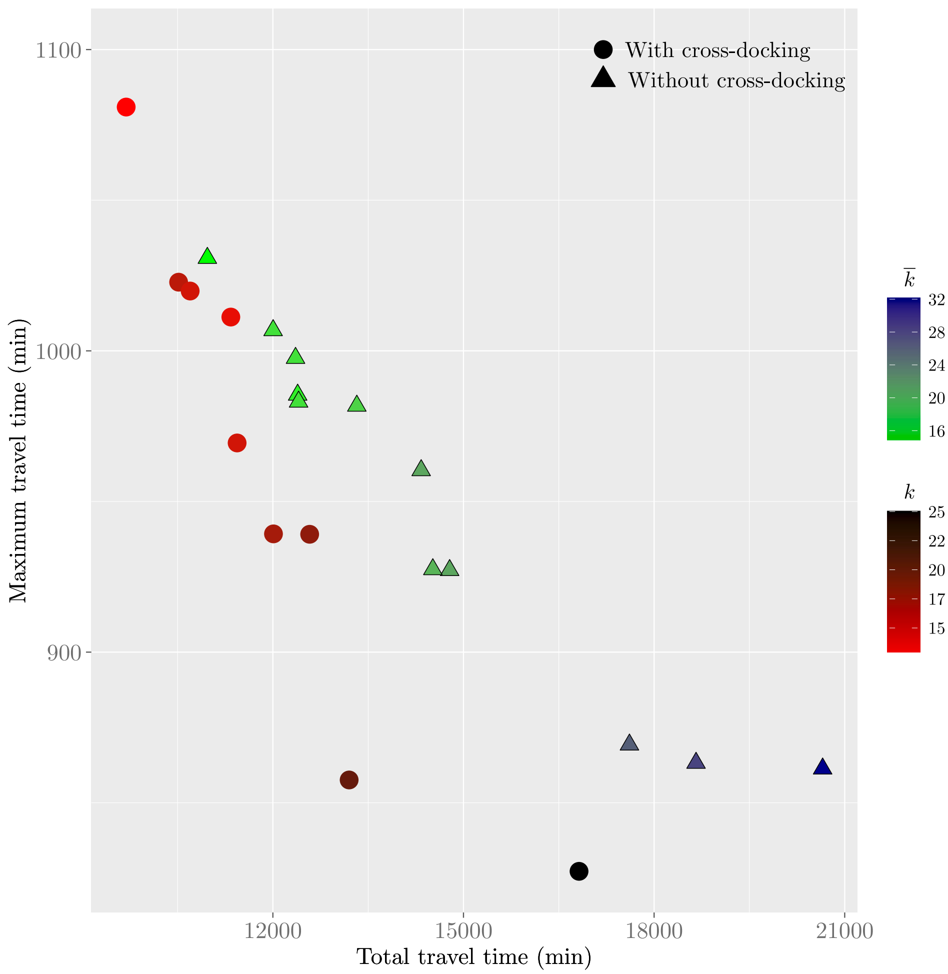


Fig. 6. Approximate Pareto fronts returned by the MACO algorithm utilising $\bar{k}$ vehicles with global cross-docking and without global cross-docking for the case study MTVRPGC instance.

The algorithmic implementation took eight hours and sixteen minutes to execute a thousand iterations. This computational burden is deemed acceptable in terms of current industry practice as the routes employed in the case study are typically fixed and repeated on a daily basis. Additionally, the planning period adopted by the pathology healthcare service provider is typically a monthly schedule, with minimal alterations affected to the routes within a single planning period. The non-dominated front of Fig. 5 exhibits a considerable trade-off in solution choices with respect to the total travel time and the number of vehicles utilised. The variation in maximum travel time of the vehicles is, however, relatively small (the trade-off with respect to maximum travel time for a vehicle is limited to approximately 230 min). The routes associated with the points in Fig. 5 are presented in Tables C.13–C.21 of Appendix C.

The improvement potentially to be experienced if Solution 9 of Fig. 5 were to be adopted on a daily basis is summarised in Table 4 with respect to the three objectives pursued in the MTVRPGC. These improvements are considerable—such improvements are typically not achievable when comparing optimisation results against industry standards. The operations in the case study are, however, not presently coordinated centrally, but are instead independently managed in various sub-regions of the case study. Some of the improvements presented here may therefore be attributed to the efficiency gains that one would expect to see when managing the logistics of a network of this size and type in a centralised rather than a decentralised manner. As such, the results reported in this section may be valuable, as they may be cited in support of a case to adopt centralised logistics management instead of a decentralised approach in an attempt to save on specimen transportation costs.

Table 4
Comparison between vehicle routes currently employed by the pathology service provider and Solution 9 of Fig. 5 for the MTVRPGC case study instance.

Objective	Current	MACO	Percentage improvement
Total travel time (min)	15 175.7	9 690.8	36.1
Maximum travel time of the vehicle (min)	1 080.9	992.1	8.1
Number of vehicles	36	13	63.8

The largest potential improvement realises in the reduction of the number of vehicles required to perform the necessary collection and delivery of pathological specimens between the respective facilities. A reduction of over sixty-three percent was achievable when considering all facilities collectively and exploiting the cross-docking capability of the MACO algorithm, which is absent in the current vehicle routing implementation of the pathology healthcare service provider.

The significant improvements of Table 4 are to be expected for three main reasons, namely centralised management, utilisation of an effective metaheuristic algorithm and the possibility of exploiting the global cross-docking component. A further investigation was performed to determine the contribution of the global cross-docking component towards the improvements attained by the MACO algorithm with the results elucidated in Fig. 6. The incorporation of global cross-docking has a significant impact, as highlighted in Fig. 6, on both the number of vehicles employed and the total travel time of the vehicles within the MTVRPGC network.

8. Conclusion

A new rich type of VRP, called the MTVRPGC, was introduced in this paper. It is an extension of the celebrated VRP in which commodities have to be collected from a number of facilities and which facilitates global cross-docking (*i.e.* cross-docking that can occur at any vertex within a subset of vertices). The problem involves the segregation of intermediate facilities into a variety of tiers, arranged according to unique commodity processing capabilities and allows for the possibility of the spill-over of unmet demand for commodity collection into a next planning period. An MILP formulation was proposed for finding optimal solutions to small hypothetical MTVRPGC test instances.

A novel MACO algorithm for the solution of larger MTVRPGC instances was also introduced in Section 6. The MACO algorithm was validated against a real MTVRPGC instance within the Western Cape province of South Africa in which 388 healthcare facilities are present. The vehicle routes returned by the MACO algorithm realised significant improvements in all three objectives of the MTVRPGC relative to the routes currently employed by the pathology healthcare provider in question.

CRediT authorship contribution statement

A. Smith: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualisation. **P. Toth:** Conceptualization, Methodology, Validation, Formal analysis, Writing - review & editing, Supervision. **L. Bam:** Conceptualization, Investigation, Supervision. **J.H. van Vuuren:** Conceptualization, Methodology, Validation, Formal analysis, Resources, Writing - review & editing, Supervision, project administration, Funding acquisituion.

Acknowledgement

We would like to thank Carlos Rodrigo Rey Barra (University of Bologna) for his continued support in completing the numerical computations.

Appendices. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cor.2021.105526.

References

Baldacci, R., Mingozzi, A., Roberti, R., Wolfler Calvo, R., 2013. An exact algorithm for the two-echelon capacitated vehicle routing problem. Oper. Res. 61 (2), 298–314.

Battarra, M., Cordeau, J.F., Iori, M., 2014. Pickup and delivery problems for goods transportation. In: Toth, P., Vigo, D. (Eds.), Vehicle Routing: Problems, Methods, and Applications. In: MOS-SIAM Series on Optimization, vol. 18. Philadelphia, pp. 161–191, (Chapter 6).

Bell, J.E., McMullen, P.R., 2004. Ant colony optimization techniques for the vehicle routing problem. Adv. Eng. Inform. 18 (1), 41–48.

Braekers, K., Ramaekers, K., Van Nieuwenhuyse, I., 2016. The vehicle routing problem: State of the art classification and review. Comput. Ind. Eng. 99, 300–313.

Bräysy, O., Gendreau, M., 2005a. Vehicle routing problem with time windows, part I: Route construction and local search algorithms. Transp. Sci. 39 (1), 104–118.

Bräysy, O., Gendreau, M., 2005b. Vehicle routing problem with time windows, Part II: Metaheuristics. Transp. Sci. 39 (1), 119–139.

Buijs, P., Vis, I.F., Carlo, H.J., 2014. Synchronization in cross-docking networks: A research classification and framework. European J. Oper. Res. 239 (3), 593–608.

Bullnheimer, B., Hartl, R.F., Strauss, C., 1997. A new rank based version of the ant system: A computational study. In: SFB Adaptive Information Systems and Modelling in Economics and Management Science. WU Vienna University of Economics and Business, in CEJOR.

Clarke, G.U., Wright, J.W., 1964. Scheduling of vehicles from a central depot to a number of delivery points. Oper. Res. 12 (4), 568–581.

Cuda, R., Guastaroba, G., Speranza, M.G., 2015. A survey on two-echelon routing problems. Comput. Oper. Res. 55, 185–199.

Dantzig, G., Ramser, J., 1959. The truck dispatching problem. Manage. Sci. 6 (1), 80–91.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6 (2), 182–197.

Desaulniers, G., Desrosiers, A., Erdmann, J., Solomon, M., Soumis, F., 2002. Vehicle routing problem with pickup and delivery. In: Toth, P., Vigo, D. (Eds.), The Vehicle Routing Problem. SIAM, Philadelphia, pp. 225–242, (Chapter 9).

Desaulniers, G., Madsen, O.B.G., Ropke, S., 2014. The vehicle routing problem with time windows. In: Toth, P., Vigo, D. (Eds.), Vehicle Routing: Problems, Methods, and Applications. In: MOS-SIAM Series on Optimization, vol. 18. Philadelphia, pp. 119–159, (Chapter 5).

Doerner, K., Salazar-Gonzalez, J.J., 2012. Pickup and delivery routing problems for people transportation. In: Toth, P., Vigo, D. (Eds.), Vehicle Routing: Problems, Methods, and Applications. In: MOS-SIAM Series on Optimization, vol. 18. Philadelphia, pp. 193–212, (Chapter 7).

Dondo, R., Méndez, C.A., Cerdá, J., 2011. The multi-echelon vehicle routing problem with cross docking in supply chain management. Comput. Chem. Eng. 35 (12), 3002–3024.

García-Martínez, C., Cordón, O., Herrera, F., 2007. A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. European J. Oper. Res. 180 (1), 116–148.

Grangier, P., Gendreau, M., Lehuédé, F., Rousseau, L.M., 2017. A matheuristic based on large neighborhood search for the vehicle routing problem with cross-docking. Comput. Oper. Res. 84, 116–126.

Irnich, S., Toth, P., Vigo, D., 2014. The family of vehicle routing problems. In: Toth, P., Vigo, D. (Eds.), Vehicle Routing: Problems, Methods, and Applications. In: MOS-SIAM Series on Optimization, vol. 18. Philadelphia, pp. 1–33, (Chapter 5).

Jozefowiez, N., Semet, F., Talbi, E.-G., 2008. Multi-objective vehicle routing problems. European J. Oper. Res. 189 (2), 293–309.

Kallehauge, B., 2008. Formulations and exact algorithms for the vehicle routing problem with time windows. Comput. Oper. Res. 35 (7), 2307–2330.

Kolen, A.W.J., Rinnooy Kan, A.H.G., Trienekens, H.W.J.M., 1987. Vehicle routing with time windows. Oper. Res. 35 (2), 266–273.

Liao, C.J., Lin, Y., Shih, S.C., 2010. Vehicle routing with cross-docking in the supply chain. Expert Syst. Appl. 37 (10), 6868–6873.

Maknoon, Y., Laporte, G., 2017. Vehicle routing with cross-dock selection. Comput. Oper. Res. 77, 254–266.

Montoya-Torres, J.R., Franco, J.L., Isaza, S.N., Jiménez, H.F., Herazo-Padilla, N., 2015. A literature review on the vehicle routing problem with multiple depots. Comput. Ind. Eng. 79, 115–129.

OSRM, 2017. Open source routing machine. [Online], Available at: http://www.http://project-osrm.org/. (Accessed 2 May 2016).

Perboli, G., Tadei, R., Vigo, D., 2011. The two-echelon capacitated vehicle routing problem: Models and math-based heuristics. Transp. Sci. 45 (3), 364–380.

Rais, A., Alvelos, F., Carvalho, M.S., 2014. New mixed integer-programming model for the pickup-and-delivery problem with transshipment. European J. Oper. Res. 235 (3), 530–539.

Savelsbergh, M.W., Sol, M., 1995. The general pickup and delivery problem. Transp. Sci. 29 (1), 17–29.

Smith, A., 2019a. Elserivier. [Online], Available at: https://drive.google.com/open?id=0BwVjk2uwIjGld0dSOTFSZ1lSTm8. (Accessed 7 August 2019).

Smith, A., 2019b. TVRPGC Travel times. [Online], Available at: https://drive.google.com/drive/folders/0BwVjk2uwIjGlcUZYSmxJOGhwWjQ. (Accessed 7 August 2019).

Tan, W.F., Lee, L.S., Abdul Majid, Z., Seow, H.V., 2012. Ant colony optimization for capacitated vehicle routing problem. J. Comput. Sci. 8 (6), 846–852.

Toth, P., Vigo, D. (Eds.), 2014. Vehicle Routing: Problems, Methods, and Applications. In: MOS-SIAM Series on Optimization, vol. 18, SIAM, Phildelphia (PA).

Van Belle, J., Valckenaers, P., Cattrysse, D., 2012. Cross-docking: State of the art. Omega 40 (6), 827–846.

Wang, X., Choi, T.M., Liu, H., Yue, X., 2016. Novel ant colony optimization methods for simplifying solution construction in vehicle routing problems. IEEE Trans. Intell. Transp. Syst. 17 (11), 3132–3141.

WHO, 2008. Consultation on technical and operational recommendation for clinical laboratory testing harmonization and standardization. [Online], Available at: http://www.who.int/healthsystems/round11_9.pdf. (Accessed 11 May 2015).

Wolfram, 2016. Wolfram mathematica. [Online], Available at: https://www.wolfram.com/mathematica/. (Accessed 13 March 2016).