

Article

# Individually Optimized Commercial Road Transport: A Decision Support System for Customizable Routing Problems

Max Leyerer \*, Marc-Oliver Sonneberg, Maximilian Heumann, Tim Kammann and Michael H. Breitner

School of Economics and Management, Leibniz University Hannover, 30167 Hannover, Germany; sonneberg@iwi.uni-hannover.de (M.-O.S.); heumann@iwi.uni-hannover.de (M.H.); kammann@iwi.uni-hannover.de (T.K.); breitner@iwi.uni-hannover.de (M.H.B.)

\* Correspondence: leyerer@iwi.uni-hannover.de

Received: 30 August 2019; Accepted: 6 October 2019; Published: 9 October 2019



**Abstract:** The Vehicle Routing Problem (VRP) in its manifold variants is widely discussed in scientific literature. We investigate related optimization models and solution methods to determine the state of research for vehicle routing attributes and their combinations. Most of these approaches are idealized and focus on single problem-tailored routing applications. Addressing this research gap, we present a customizable VRP for optimized road transportation embedded into a Decision Support System (DSS). It integrates various model attributes and handles a multitude of real-world routing problems. In the context of urban logistics, practitioners of different industries and researchers are assisted in efficient route planning that allows for minimizing driving distances and reducing vehicle emissions. Based on the design science research methodology, we evaluate the DSS with computational benchmarks and real-world simulations. Results indicate that our developed DSS can compete with problem-tailored algorithms. With our solution-oriented DSS as final artifact, we contribute to an enhanced economic and environmental sustainability in urban logistic applications.

**Keywords:** vehicle routing problem; decision support system; design science research; green information system; urban logistics

---

## 1. Introduction

In today's competitive and highly demanding environment, there is a growing recognition for cost reduction and customer service to be achieved based on well-organized logistics processes. The efficient distribution of goods and services is crucial for many companies since transportation is an important cost factor. Thus, vehicle routing is considered as an essential source for potential savings in a company's distribution system [1]. Studies demonstrated that optimized routes can lead to significant cost savings ranging up to 20–30% [2,3]. Besides the individual economic importance for companies, the macroeconomic relevance of efficient routing plans is tremendous in terms of ecological sustainability. The avoidance of needless long routes with low degrees of capacity utilization enables a notable reduction of emissions and congestion by relieving the urban and interurban road infrastructure [4].

The determination of the optimal set of routes to be performed by a vehicle fleet to serve given customers is known as Vehicle Routing Problem (VRP) [3]. The VRP and its many variants have been widely studied for more than 50 years [5]. Typically, routing models are developed to support operational as well as tactical decision-making processes in transport and logistics [6]. While a human planner may often be able to conceive a feasible routing schedule within a reasonable time

for small problem sets, such manual solutions are usually far from being optimal when the specific business context requires the consideration of structural constraints such as time windows and capacity restrictions [7]. While researchers have devoted much effort into developing new variants and solution methods for the VRP, much of this work has been criticized of “being too focused on idealized models with non-realistic assumptions for practical applications” [8]. To tackle this research gap, we applied a Design Science Research (DSR) methodology to develop a Decision Support System (DSS) capable of optimizing routing problems in different business contexts of urban logistics. The DSS must transfer objectives as well as constraints into VRP models and it must be able to solve structurally different problems [9]. The DSS as resulting artifact is classified as nascent design theory [10]. Besides individual companies, our society is becoming increasingly aware of environmental and economic sustainability [11]. The attention on sustainability is also recognized in Information System (IS) research, as information is a prerequisite for making appropriate decisions on sustainability actions [12]. The emerged research domain of Green IS addresses the transformative role of IS in the context of a sustainable society and business strategies. The emphases in this field vary by conceptualization, analyses, design, and impact of such systems [12]. Studies examining Green IS research by Malhotra et al. [13] and Gholami et al. [14] reveal that design and impact-oriented research is lacking. Since the IS domain is described as an interdisciplinary space, where discourses of several disciplines are straddled and a variety of topics as well as methodologies are present [15], we combine elements of computer science, operations research, management science, and transportation science in our approach. The described requirements and challenges lead to our research question:

*How can a customizable VRP be implemented into a DSS enabling route optimization for urban logistics applications of several industries?*

Our aim is to gather various vehicle routing attributes into a generic and customizable optimization model. In our final artifact, called Multi-Attribute Vehicle Routing Decision Support System (MAVRDSS), the user describes a practical routing problem based on the underlying business context. To achieve this objective, the article is structured as follows: First, we present the applied DSR methodology for the artifact development. Afterwards, the various VRP attributes are described and classified, referring to related research. Subsequently, we introduce our developed DSS and its components, constituting a prototype web-application. To evaluate MAVRDSS, we present computational benchmarks and simulation results of two real-world application cases in the context of urban logistics. Further, we discuss implications, limitations, recommendations, and contributions of our study. We complete our article with conclusions and an outlook.

## 2. Methodology

Our research methodology is based on DSR principles as proposed by Hevner et al. [16]. This method stands in contrast to behavioral science, because the design science approach systematically seeks to create “new and innovative artifacts” [16]. This means it is the most suitable approach for creating, specifying, and evaluating a particular topic while addressing its relevance and its rigor. Hevner [17] presents three cycles of activities (relevance, rigor, and design) which influence each other.

The topic’s contextual environment and related issues are addressed by the relevance cycle. Our research is motivated by the observation that companies can have large efficiency reserves in transportation processes due to a lack of optimized routes. The amount of goods and services that must be delivered is rising and tackles nearly all kinds of business operations. Together with the ongoing urbanization and the progressing climate change, the need for optimization still rises to reduce incidental burdens. By providing efficient routing support, we address the lack of solution-oriented Green IS and increase the livability of inhabitants by securing needs while at the same time improving air quality, traffic load, and noise pollution [13,14] as already presented within the introduction. Within the rigor cycle, the review of existing scientific knowledge depicts a crucial part of the research process. We carried out a literature review on multi-attribute VRP. To ensure an appropriate focus on the

application domain, the scope of the VRP review is narrowed to real-world application cases, relevant routing attributes, and their modeling approaches. The results of the rigor cycle are presented in Section 3.

The design cycle is defined as an iterative process that uses several build-and-evaluate loops and revises developed design artifacts until they are ready for a real-world application. To develop our technological artifact, we conducted several loops while respecting environmental requirements, scientific methods, and present knowledge. Based on existing taxonomies, we derived a VRP classification scheme as a basis to formulate a basic VRP. Starting with the implementation of only few model attributes, we stepwise expanded the VRP model by several features and functionalities. Together with the application of a (heuristic) solution method, we integrated the multi-attribute vehicle routing model into a DSS prototype (called MAVRDSS), that is classified as nascent design theory [10]. Besides activities of developing a technological artifact (Section 4), the design cycle calls for a continuous evaluation of the resulting research to ensure its adequacy to improve the environment. On one side, the functionality and efficiency of the routing model is monitored continuously throughout the development process. On the other side, the evaluation of the artifact must be based on scientific theories and engineering methods drawn from the knowledge base. Venable et al. [18] propose the comparative evaluation of a new artifact with existing artifacts to determine whether it makes an improvement on the state of the art. For this purpose, we conduct benchmark calculations on different well-known VRP instances to examine the artifact's validity, performance, and functionality. Concurrently, we compare our MAVRDSS with existing problem-tailored solution approaches. Further, the artifact is tested on two real-world instances in the context of urban logistics, namely a small-sized coffee supplier and a medium-sized parcel service provider (Section 5).

### 3. VRP Attributes and Related Work

In view of the large body of scientific works, the VRP is widely considered as a distinct field of knowledge in operations research and computer science [3,19]. Regarding the planning horizon of transport and logistics processes, the VRP assists short-term and daily decisions related to diverse transportation services of goods and passengers. This section aims to give a brief overview of the most relevant VRP attributes or rather constraints. The goal of the following literature overview is to obtain insights on actual real-world problems in transport and logistics and how they can be addressed efficiently within the context of routing models. Following the overall purpose of creating a DSS for a variety of real-world routing problems, these insights provide the basis for building a routing DSS considering aspects that are relevant in practice. VRP research follows a clear trend towards including aspects that are essential to routing in real-life. "Real-world problems are generally characterized by several interacting attributes, which describe their feasibility and optimality structures" [20]. Such routing problems with a multitude of possible attributes and their combinations are often grouped under the denomination of Rich VRP [8]. Those models share the characteristics of including additional constraints and objectives, generalizing, and unifying other independent problems [21]. When referring to various constraints and objectives, some authors use the term attributes and refer to rich-like models as multi-attribute VRP [20,22,23].

Table 1 contains a brief description of real-world VRP attributes which are based on the scheme of Lahyani et al. [6] serving as reference for the developed DSS in the next section. This list does not claim to be complete as there exist even more VRP attributes in the scientific literature. Regarding the explanations and the developed DSS, the diverse types of nodes (e.g., customers, intermediate facilities, central depots, driver homes) are translated into only two distinct node sets: A set of nodes where requests occur (called *customers*) and a set of nodes where no requests occur (called *depots*).

**Table 1.** List of common Vehicle Routing Problem (VRP) attributes, abbreviations, and description.

Attribute	Abbr.	Description
Pickup and delivery	PD	Customers may have pickup and delivery requests to be served in sequential, separate or simultaneous manner
Multiple depots	MD	There is more than one node without a request; vehicle trips depart from and/or arrive at more than one node
Multiple periods	MP	The planning horizon covers more than one period and requests are served according to visitation patterns
Multiple products	MC	Vehicles can carry different types of products/commodities (e.g., dry, liquid, frozen)
Heterogeneous vehicles	HV	Vehicles have different properties (e.g., costs, capacity, maximum travel distance, travel speed and distance)
Capacitated vehicles	CV	Vehicles have limited loading capacities
Limited range	LR	The length of a route is restricted (e.g., limited fuel, emissions)
Compartmentalized vehicles	VC	Different product types are carried in separate vehicle compartments
Loading policy/specific ordering	LP	Routing must respect possible ordering of visits induced by a loading policy (e.g., last-in-first-out)
Driver regulations	DR	Routing must respect legislation on driver working hours and safety (e.g., rest periods)
Inventory considerations	IN	Requests base on customers' inventory levels (typically: avoiding stock-out situations)
Capacitated depots	CD	Depots have limited storing capacities
Multiple time windows	MW	Customers are associated with more than one time window, in which the request can be served (equivalently: The planning period is interrupted by at least one break time)
Time window component		Time restrictions on
-customer/requests	CT	-customer/requests (e.g., opening hours, break times)
-depots	DT	-depots (e.g., closing hours)
-vehicles/drivers	VT	-vehicles/drivers (e.g., working hours, lunch breaks)
-roads	RT	-roads (e.g., time dependent access restriction)
Optional visits	OV	Decision consists in choosing which requests to serve as visits are not mandatory
Multiple trips	MT	Vehicles can perform more than one trip in a planning period (equivalently: A tour includes at least one intermediate depot visit)
Multiple visits/split-deliveries	MV	Vehicles can visit customers more than once and may serve a fraction of demand at each visit
Open routes	OR	Start and end node of a vehicle's tour may not be equal
Incompatibilities	IC	Node sequencing must respect physical inclusion or exclusion restrictions between, e.g., vehicle-customer, vehicle-depot, vehicle-product, or customer-depot
Precedence constraints	PC	Routing must respect a possible visiting order due to physical or service considerations (e.g., dial-a-ride, loading issues)
Balanced routes	BR	The workload of routes must be balanced (e.g., in terms of distance, duration, loaded quantities, or cost)
Objective components		Optimization targets can include
-distance	DO	-distance (e.g., minimize total travel distance or cost of travel)
-times/durations	TO	-times/durations (e.g., minimize travel durations or time windows)
-vehicle/driver	VO	-vehicle/driver (e.g., minimize number of operated vehicles/drivers)
-customer/request	CO	-customer/request (e.g., maximize number of visits)
-facility/depot	FO	-facility/depot (e.g., minimize number of necessary locations)
-loads	LO	-loads (e.g., minimize number or cost of loadings)
Multiple objectives	MO	There is more than one optimization target (e.g., minimize fixed vehicle cost and distance cost); multiple targets are prioritized either by weights or by a hierarchical ordering (includes several weighted sum objectives)
Soft constraints	SC	Problem constraints addressed in objective function (e.g., penalty fee)

The described VRP attributes are considered in diverse combinations by many authors. To identify the state of research in this field, we systematically analyzed and classified existing Rich VRP literature. Regarding the terminologies, the review methodology must consider three crucial issues: First, the definition of the term “rich” to denote a model is vague and its usage may significantly differ among authors. Some authors may not use the term “rich” at all, but the model addresses various constraints aiming at a close representation of the real world. Second, the state-of-the-art in VRP research is continuously evolving. Models explicitly denoted as “rich” by their authors some years ago may not be regarded as such anymore. Third, although the terms “real-world” or “real-life” are used by many authors to describe their model and the problem type, the degree to which an actual application case is considered may significantly differ among the studies. In fact, some authors address a realistic case study and provide comprehensive descriptions of the given problem, whereas other authors merely hint at possible fields of application.

To cope with the mentioned issues, we followed the proposed guidelines of Kitchenham et al. [24]. At first stage, we examined diverse databases, including *Google Scholar*, *Springer Link*, *AISel*, and *Web of Science*. We searched for the keywords “vehicle routing”, “rich”, “multi attribute”, “multi feature”, “real world”, and “case study” in various combinations. In addition to that, we conducted a forward and backward search following the principles of Webster and Watson [25], resulting in a total number of 54 relevant articles. At a second stage, exclusion criteria were used to narrow the research scope on studies providing valuable insights on real-world transport and logistics problems as well as modeling attributes. We considered approaches containing at least four model attributes (routing attributes and objective function elements) and a specific application case. To secure actual high-quality research and a solid peer-reviewing, we excluded articles of conference proceedings as well as journal articles published before 2009. As a result, 25 relevant articles were comprised.

As shown by the table, several routing attributes (e.g., heterogeneous vehicles (HV), capacitated vehicles (CV), time window component: Customer/requests (CT)) and objective function elements (e.g., objective component: Distance (DO), objective component: Vehicle/driver (VO), multiple objectives (MO)) are already intensively investigated, but no model is able to comprise almost all model features. Regarding the presented list, the average number of included model attributes is 7.4. The most elements were implemented by Armas et al. [26], who considered 14 of 32 model attributes within their approach. In general, many articles deal with the VRP and its diverse specifications. Eksioglu et al. [27] present the development of VRP-research over the past years. They identified over 1000 articles in academic journals. Compared to the small number of multi-attribute VRP presented in Table 2, most of the existing VRP articles are problem-tailored and lack generality. Less implemented attributes and especially the option to combine different constraints constitute a research gap we address.

To provide an actual benefit for service and delivery companies of all sizes and sectors, the VRP must be embedded into a user-friendly DSS. Yet, many idealized models are frequently unable to capture the variety of problem attributes found in practical routing settings. Recently, the research community has turned the attention to incorporating more different constraints and objectives into optimization models to tackle the complexities of real-world applications. Apart from that, several commercial web-based VRP tools exist which offer only limited usability due to the small choice of problem specifications. To be applicable in a multitude of business contexts, a valuable DSS must keep its generality while offering as many attributes as possible. A user must be able to choose among the described attributes which best reflect the problem at hand. To address these challenges, we developed MAVRDSS presented in the next section.



## 4. Multi-Attribute Vehicle Routing Decision Support System (MAVRDSS)

### 4.1. Requirements

Routing software is mainly employed by companies producing industrial goods that need to plan local pickup and/or delivery routes, long-haul truckload routes, as well as service companies, such as couriers, buses, or technicians, to name a few [1]. One finding is that most of these companies are relatively large or specialized on deliveries (shipping companies). To provide relevant support for decision makers of diverse industries in urban logistics, we developed MAVRDSS from scratch, using the web-application framework *Ruby on Rails*. The tool's target groups are all small- and medium-sized enterprises that participate in the urban commercial traffic and that operate an own vehicle fleet, e.g., newspaper delivery, solid waste collection, or craftsmen-, parcel-, pharmacy-, e-grocery-, and care-services. As most of those companies behave according to their gathered experience and best practices, only few use appropriate software to optimize their daily vehicle routing. Consequently, there are considerable saving potentials in terms of travel distance, travel time, travel expenses, vehicle emissions, and the resulting traffic volume. To achieve these unexploited potentials and to provide relevant assistance for especially small- and medium-sized enterprises without existing routing optimization, MAVRDSS must offer high usability and quick results for as many VRP attributes as possible.

Regarding the computation time, MAVRDSS must provide quick routing recommendations for the decision maker. The increasing computer power enables researchers to tackle ever-larger combinatorial optimization problems. This led to the emergence of various solutions methods that have been widely applied on the VRP, which is proven to be NP (non-deterministic polynomial-time)-hard [51]. Methods for solving optimization problems can be classified into two fundamental approaches: Exact methods based on mathematical programming on the one hand, and approximate methods based on heuristics and metaheuristics on the other hand [52]. Exact methods can guarantee optimal solutions. In this context, mixed integer programming (MIP) is widely regarded as one of the most powerful tools in operations research. However, when faced with large-scale combinatorial problems, even the best commercial MIP solvers are unable to produce high quality solutions in acceptable running times. As a result, researchers commonly use heuristics for solving VRP [6,53]. Approximate methods obtain good solutions on large-size problem instances but do not have information about the solution's optimality. Heuristics are most often designed to solve a specific problem. Systematically following several steps, they focus on quickly obtaining a feasible and acceptable solution [54]. In contrast to that, metaheuristics are general-purpose algorithms performing a more thorough search of the solution space. They aim to provide acceptable solutions in reasonable running times while allowing inferior and sometimes infeasible moves. They are applicable on a broader class of problems and may be employed as a guiding strategy in tailoring underlying heuristics [53,55]. Metaheuristics can be categorized further into two main classes: Local search and population search. Local search based methods, for instance Tabu Search, intensively explore the solution space by iteratively moving from the current solution to another promising solution in its neighborhood. Population search based algorithms, for instance Genetic Algorithms, maintain a pool of good parent solutions and recombine them to produce promising offspring that updates the pool. The performance of a local search algorithm largely depends on its capability to diversify the search, that is, to "escape" from locally optimal solutions [56]. Creating problem-independent algorithms for designing an efficient local search is an active topic of research. Recently, some authors have turned their attention towards such unified solution approaches to solve a variety of VRP [23,56,57]. However, applying these approaches usually requires strong algorithmic expertise and a considerable amount of user involvement (e.g., defining neighborhoods, adapting layers, etc.), which is normally not present in small- and medium-sized enterprises within the described industries. In general, Bräysy and Gendreau [58] recommend heuristics which possess the following characteristics: High solution quality in terms of objective function value, speed, simplicity

of implementation, flexibility, and robustness. MAVRDSS addresses these requirements and enables users without optimization knowledge the determination of best possible solutions.

Based on the VRP taxonomy of Eksioglu et al. [27] and our literature review in Section 3, we derived the following scheme to classify relevant real-world VRP-attributes (Figure 1). We reduced the complexity of the VRP taxonomy presented by Eksioglu et al. [27] because not all categories are appropriate for the developed MAVRDSS. For instance, we do not distinguish between different solution methods as we focus on a heuristic approach. This complexity reduction is also suggested by Lahyani et al. [6], who assess existing taxonomies as too ambiguous due to its high level of detail. The goal is not to stress the particularities and differences between all VRP variants but rather to list all relevant model attributes that constitute a specific VRP variant, while keeping a moderate level of granularity.

I. Information Characteristics		II. Problem Physical Characteristics	
<i>input data</i>	<i>depots</i>	<i>vehicles</i>	<i>customers</i>
+input evolution	└ single	+type	└ unrestricted
└ static	└ multiple	└ homogenous	└ inventory considerations
└ dynamic		└ heterogeneous	
+input quality	<i>periods</i>	+capacity	<i>depots</i>
└ deterministic	└ single	└ unrestricted	└ unconstrained
└ stochastic	└ multiple	└ capacitated	└ capacitated
<i>travel mode</i>	<i>product types</i>	+range	<i>times</i>
└ unimodal	└ single	└ unlimited	└ unconstrained
└ multimodal	└ multiple	└ limited	└ time dimension
		+structure	+time window structure
		└ no compartments	└ single time window
		└ compartmentalized	└ multiple time windows
<i>number of vehicles</i>		+loading policy	+time window components
└ unlimited number		└ no policy	└ customers/requests
└ bounded number		└ specific ordering	└ depots
<i>customer requests</i>		<i>drivers</i>	└ vehicles/drivers
└ pickup or delivery		└ no regulations	└ roads
└ pickup and delivery		└ regulations	
III. Route Characteristics		IV. Objective Characteristics	
<i>node covering</i>	<i>incompatibilities</i>	<i>structure</i>	<i>components</i>
└ complete	└ all compatible	└ single objective	└ distance
└ optional visits	└ incompatibilities	└ multiple objectives	└ times
		└ weighted sum	└ vehicle/driver
<i>visit frequency</i>	<i>node coupling</i>	└ hierarchical ordering	└ customer/request
└ single visit	└ no precedence		└ depot
└ multiple visits / load splitting	└ precedence constraints	<i>hardness of constraints</i>	└ load
		└ hard constraints	
<i>depot connectivity</i>	<i>routes balancing</i>	└ soft constraints	
└ closed routes	└ no balancing		
└ open routes	└ balanced routes		
<i>vehicle use</i>			
└ single trip			
└ multiple trips			

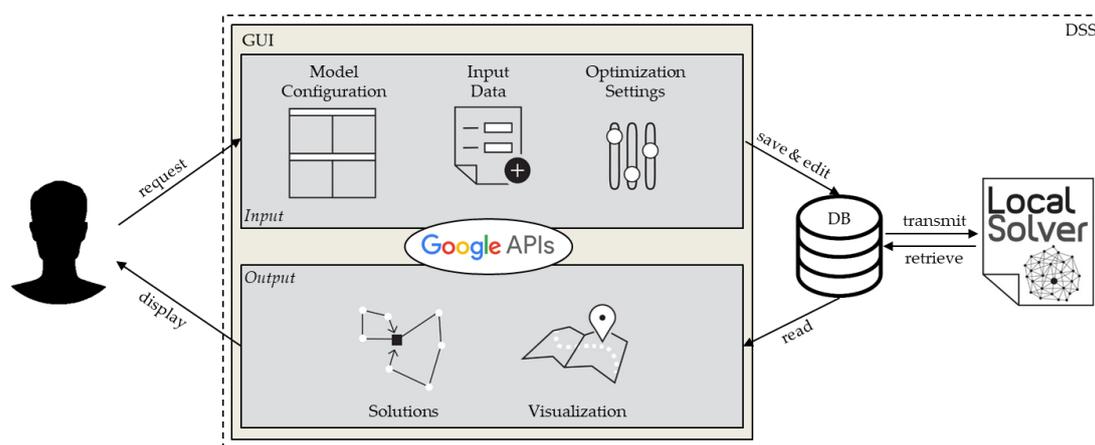
Figure 1. Classification of the implemented VRP-attributes (based on Lahyani et al. [6]).

The possible VRP attributes are classified by four categories to systematically model the given problem characteristics. The MAVRDSS-user describes the case-specific problem characteristics and its constraints in the presented order (I–IV). Figure 1 also indicates which VRP-attributes are implemented within MAVRDSS (black font). Referring to the VRP-attributes presented in Table 1, we implemented 26 of the 32 described attributes, offering high flexibility and a wide range of applications. Due to inconsistencies, the gray marked attributes are not implemented. The requirement of preserving the model’s generality is fulfilled for nearly all attribute combinations. Only few incompatibilities between

attributes exist, for instance, the open routes are incompatible with the single depot attribute. Further, the multiple trips attribute is incompatible with the pickup and delivery attribute and the load-splitting attribute cannot be combined with multiple trips.

#### 4.2. Functionality and User Interaction

MAVRDSS is built as prototype web-application to run the computationally intensive optimization on a remote server, while a MAVRDSS-user only needs to operate a device supporting a web-browser. All user actions are executed via the graphical user interface (GUI). The software comprises five main components as depicted in Figure 2: Model configuration, input data, optimization settings, solutions, and visualization. The model configuration represents the customizable part of the routing model based on the attributes in the classification scheme. With the manifold configuration opportunities shown in Figure 1, diverse urban logistic applications can be simulated, considering the industries' specific constraints. Within the whole configuration process, the user is served with descriptive information about the model attributes. The input data refers to the customers, depots, vehicles, and their associations. A user creates customers, depots, etc., and associates parameter values by filling out input forms. Optimization settings refer to user-sided configurations of the optimization process, which can be set before an optimization run (e.g., running time limit). MAVRDSS provides an interface-to-database system that allows writing and reading relevant data. A step counter variable keeps track of the configuration status. After saving the model configuration, input data, and optimization settings in the database (DB), the user can start an optimization run. All information is transformed into a case-specific optimization model and passed to the software *LocalSolver* for resolution. *LocalSolver* is a mathematical programming solver based on local search techniques following the model-and-run paradigm that offers the possibility to tackle a large class of different problems. In this way, the complex processes of defining neighborhoods, moves, and the search strategy are automated. After an optimization run, the solutions (e.g., customer sequencing) are saved in the DB. The user can access and visualize them via the GUI, using *Google APIs* (application programming interface) for appropriate presentation in an embedded map. A user interacts with these described components when operating the web-application (see Figure 2).

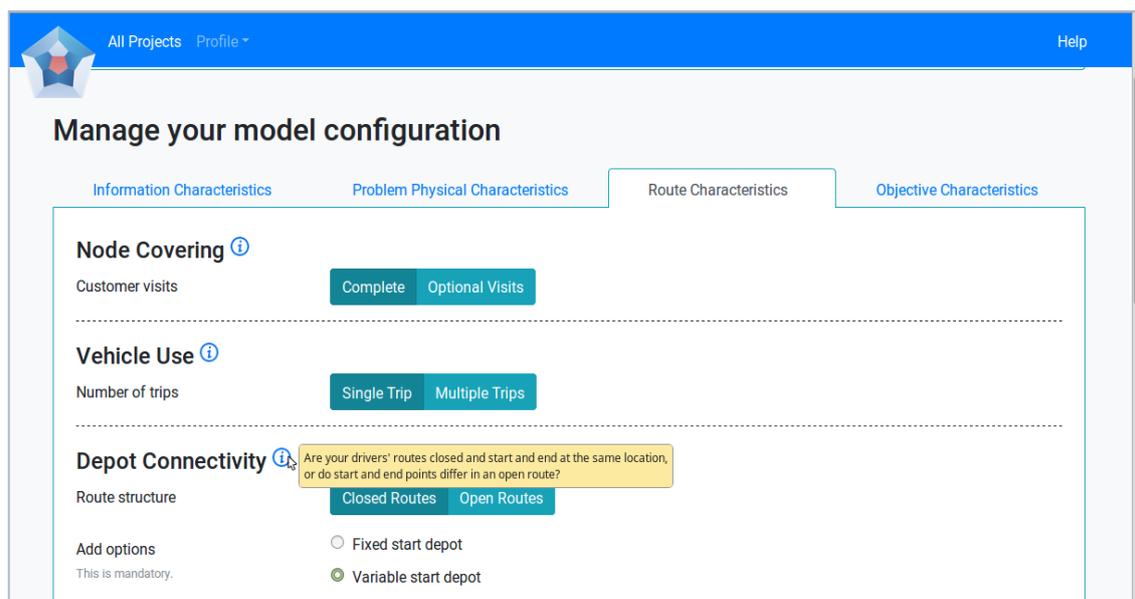


**Figure 2.** User interaction and system architecture of Multi-Attribute Vehicle Routing Decision Support System (MAVRDSS).

While only the input data and the model configuration are treated as own resources, all components are part of a project resource. In terms of the model associations, each user can have different projects and each project can contain many vehicles, customers, etc. A user can create new instances of the resources, and read, update, or delete them. A project is runnable if the user has selected the model attributes, has created at least one vehicle, one depot, and two customer locations. Moreover, the travel distances and durations between all customers and depots must be generated. HTML elements are

used to display the required content based on the current controller action. The client's browser must support *JavaScript* to run the web-application.

As the routing model's versatility and customizability are the fundamental features of MAVRDSS, its utilization must be supported through an adequate design and functionality of the web-application. In this regard, the main requirement is the applicability of MAVRDSS by end-users with neither programming skills nor knowledge of optimization methods. While the development of a user-friendly interface with meaningful help systems is a long-winded process, this article offers a prototype-tool for the multi-attribute VRP. Information buttons with hover effect provide a brief description of each attribute to further support the user. For instance, Figure 3 illustrates an extract of the web-application's GUI during the model configuration.



**Figure 3.** Screenshot of the MAVRDSS-GUI during the model configuration.

#### 4.3. System Components and Implementation

MAVRDSS is developed using *Ruby on Rails*, which is an open-source web development framework written in the higher object-orientated programming language *Ruby*. The framework focuses on productivity and aims to accelerate the development of web-applications [59]. Apart from *LocalSolver*, all tools used to implement MAVRDSS are managed within the *Ruby on Rails* framework.

The spatial component of MAVRDSS comprises the following features: generating distance and duration matrix, autocomplete form for creating addresses, displaying locations on maps, and visualizing route solutions. For these purposes, MAVRDSS interacts with several *Google APIs*. All API functions are accessed through the *Google Maps JavaScript API*. *Google Maps Embed API* is used to render maps in different HTML views. For example, a user can see the location of a created address in the embedded map. When a user has created a new address, the *Google Geocoding API* service converts the address into geographic coordinates, which is further used to place markers on a map and to center the map based on addresses. If permitted, the *Google Geolocation API* is used to retrieve the current position of the user operating MAVRDSS and to adjust the autocomplete address towards that position. When typing the first letters of a street name into the address form, a user gets proposals of nearby street names starting with the same letters. The *Waypoints API* is used to display routes based on solution values from an optimization run. The distance and duration matrix are generated using the *Google Distance Matrix API*.

As stated above, *LocalSolver* is used to integrate all components of a local search heuristic and to solve the user-specific VRP. The search strategy is a simulated annealing heuristic with restarts. The

selection of the next move is a choice among existing moves. The strategy collects statistics on each neighborhood and adapts the search continuously. For those moves initially identified as infeasible, the selection probability is set to zero. The moves are general-purpose transformations of neighborhoods striving to maintain the feasibility at each iteration. The solver is not limited to linear constraints but can handle highly nonlinear models. The *LocalSolver* software is implemented in C++ language. Its compiler accepts models written in C++, C#, Python, Java, or the built-in modeling language LSP (*Local Search Programming*). The latter is used in version 7.0 to implement the described multi-attribute VRP. In the LSP language, it is possible to manipulate data structures as in imperative programming and to define the optimization model using mathematical operators, built-in variables and functions. LSP has all characteristics of a scripting language but is dedicated to modeling and solving a problem [60]. Within MAVRDSS, the efficiency of the resulting customized model bases on choices of appropriate constraints, well-defined objective functions, and the right solver parameterization. Besides user-sided efficiency considerations, the implementation of the multi-attribute VRP in *LocalSolver* must stay lightweight and exclude superfluous computations as much as possible to remain efficient. The multi-attribute VRP is implemented in a single LSP model based on polyvalent components. This means that most basic functions are used for different configuration setups and only the function's constituents are evaluated individually with respect to the selected attributes. The model configuration is translated into 77 Boolean variables relating to specific functions in the LSP file. These values are used to determine the final model that is passed to the solver. To minimize the computation time and to attain an efficient model, only the selected attributes are passed to *LocalSolver*. This is achieved with an extensive set of conditional statements in the LSP model file, which activate or deactivate attributes and attribute combinations.

To provide an exemplary insight into the modeling approach, both an objective and a constraint implementation are explained using mathematical formulation and the resulting LSP-code. The vehicle capacity restriction is modeled as a hard and problem structuring constraint. According to the previously explained efficient and narrow programming in LSP, such simple but mandatory limitation can be tied in a relationship of served demand quantity to vehicle capacity shown in equations (1–3). Equation (1) defines the load of the first sequence as the sum of the demand quantity at the first customer location ( $n$ ) and an initial load of the vehicle  $k$  ( $q_k^{init}$ ), whereas the demand  $q_{n(k,1)}$  can be positive in case of a pickup node or negative for a delivery node. For every following node  $p$  in the tour, the current load is the recursively defined function (2). The size of  $Load_{k,p}$  varies for each vehicle according to the number of associated visits in a tour  $c_k$ . Constraint (3) ensures that the current load does not exceed the vehicle specific capacity ( $\bar{q}_k^{vehicle}$ ) and that it is positive.

$$Load_{k,1} = q_k^{init} + q_{n(k,1)} \quad \forall k \quad (1)$$

$$Load_{k,p} = Load_{k,p-1} + q_{n(k,p)} \quad \forall k, p \in 2, \dots, c_k \quad (2)$$

$$0 \leq Load_{k,p} \leq \bar{q}_k^{vehicle} \quad \forall k, p \in 1, \dots, c_k \quad (3)$$

As visible in the following code, the corresponding function is defined as a condition for single or multiple trips. In case of a single trip (line 2 and 3) the variable load `currentLoad[k][i]` for each vehicle  $k$  and stop  $i$  is bound to the maximum capacity `truckCapacity[k]` of the allocated truck and ensures a value greater than zero.

```

1  if (currentLoadConstraint) {
2    if (singleTrip)constraint and(0..c-1, i => 0 <= currentLoad[k][i] && currentLoad[k][i]
3      <= truckCapacity[k]);
4    if (multipleTrips)constraint and(0..c-1, i => currentLoad[k][i] <= truckCapacity[k]);}

```

When mapping the underlying problem, modeling hard constraints that are not necessary to structure the problem should be avoided to prevent early optimization stops and local optima. A

complementary approach for an optional condition, such as the even distribution of the route duration for all drivers, is therefore to describe an imbalance function that is to be minimized as an objective. Equation (4) shows the corresponding function, where the imbalance is defined as the maximum value of zero and the difference between the minimum and maximum tour duration ( $Dur^{difference}$ ) minus the average duration weighted by the tolerance factor  $\beta \cdot \left( \sum_k \frac{Dur_k^{tour}}{Z_k} \right)$ .

$$\min \left( imbalance = \max \left\{ 0, Dur^{difference} - \beta \cdot \sum_k \frac{Dur_k^{tour}}{Z_k} \right\} \right) \quad (4)$$

The minimization is implemented and performed by the minimize command of the LSP library in line 21 after the relations of the imbalance function are defined in lines 5–14. Line 6 defines the maximum tour length for each vehicle  $k$ . Lines 7 to 11 define the shortest tour as the longest tour in advance and check in a for loop iterating for each vehicle whether the respective tour is between zero and the longest tour to define the shortest tour for these cases as routeMin. Line 13 then defines the imbalance including the weighting factor beta for a variable tolerance limit of the deviations in route duration. In this example, the code is shown for the case the user chooses a hierarchical weighting of attributes. Thereby, the imbalance minimization is included in the objective function according to the selected target hierarchy.

```

5  if (balanceRouteDurations) {
6    routeMax <- max[k in 0..nbTrucks-1](routeDurations[k]);
7    routeMin <- routeMax;
8    for [k in 0..nbTrucks-1] {
9      routeMin <- ((routeDurations[k] > 0 && routeDurations[k] < routeMin)
10       ? routeDurations[k] : routeMin);
11    }
12    averageDuration <- totalDuration / nbTrucksUsed;
13    imbalance <- max(0, (routeMax - routeMin) - beta*averageDuration);
14  }
15  function readObjectives() {
16    if (hierarchicalObjectives) {
17      local P = hierarchical.split("_");
18      for [obj in 0..P.count()-1] {
19        [...]
20        else if (toInt(P[obj]) == 18) {
21          minimize imbalance;}
22      }
23    }
24  }

```

## 5. Results

### 5.1. Benchmark Calculations

To analyze the tool's performance and to evaluate our solution approach, we provide a proof-of-concept. This involves testing MAVRDSS across a range of available problem instances. The performance of MAVRDSS is tested on three routing problem variants, namely the split-delivery VRP, the multiple-depot VRP, and the Location Routing Problem (LRP). Various authors addressed these VRP variants and tested their proposed solution method on different standard instances in comparison with other studies who considered the same instances. For each problem variant, there is a set of instances chosen and two or three studies serve as benchmarks to compare the performance of the solution methods on each considered instance. To the best of our knowledge, the selected reference studies are among the best performing solution methods on the respective instances.

In the next three tables, the reference algorithms are stated at the table head. The first column describes the problem instance and size of the respective VRP variant. The Gap-column indicates the relative percentage gap of the heuristic solution to the best-known solution. The best-known solutions are taken from the most recently published reference study. Boldface entries indicate if the best-known solution is reached (gap = 0.0%). The CT-column states the computation time in seconds. *LocalSolver* indicates the running time only in full seconds. The best result of three runs is stated for each instance and the number of customer subsets is increased iteratively until no more feasible solution is obtained after 120 seconds. This short time span is deployed to meet the requirement of real-world routing applications where fast decisions are essential. The simulated annealing level is set to 2 and the number of threads is set to 2 for all problem instances. All calculations using MAVRDSS are run on a computer with an *Intel Core i7 QuadCore CPU* (2.2 GHz) and 8 GB memory.

Table 3 presents the computational results for the split-delivery VRP on seven benchmark instances provided by Belenguer et al. [61]. The solutions are compared to the SplitILS heuristic from Silva et al. [62] and to an unnamed heuristic from Aleman et al. [63]. According to Silva et al. [62], the SplitILS outperformed previous heuristics and obtained new best-known solutions on almost all instances (gap = 0.0% for all). Computations of MAVRDSS are stopped after 120 seconds and best values are reported.

**Table 3.** Computational results for the Split-Delivery VRP.

Number of Customers	Aleman et al.		SplitILS		MAVRDSS	
	Gap	CT	Gap	CT	Gap	CT
22	<b>0.0</b>	2.6	<b>0.0</b>	0.2	<b>0.0</b>	19
23	0.1	1.6	<b>0.0</b>	0.1	0.4	12
30	<b>0.0</b>	7.5	<b>0.0</b>	0.2	0.4	9
33	0.1	8.4	<b>0.0</b>	0.5	1.5	22
51	0.1	49.8	<b>0.0</b>	1.8	1.2	17
76	0.2	91.4	<b>0.0</b>	51.8	4.2	45
101	0.2	298.2	<b>0.0</b>	38.9	8.8	117

MAVRDSS is able to reliably produce feasible solutions on the given problem sets, i.e., all customer demands are served and capacity restrictions hold. Although the solutions are generally not bad since they mostly stay below the 5% optimality gap for instance sizes with up to 76 customers, they are inferior compared to the problem-tailored heuristics. On one side, the results show that *LocalSolver* generates the solutions in rather short running time. On the other side, these solutions are hardly improved in the remaining running time. This indicates that the search strategy is likely to lead to locally optimal solutions and that a search diversification is not always successful.

Table 4 presents the computational results for the Multi-Depot VRP as compared to three algorithms that have shown the best performance as stated by Cordeau and Maischberger [64]. The instances for the Multi-Depot VRP are taken from Cordeau et al. [65], where the number of vehicles is given. The three algorithms are ALNS [56], CGL [65], and ITS/1 [64]. The gaps reported by Cordeau et al. [65] are based on best-known values. The gap values are updated using the best-known values reported in the reference paper by Cordeau and Maischberger [64]. The latter authors do not provide information on the running time of their algorithm. Instead, they report the best result after 100,000 iterations, whereby an average computation time of 110 seconds is stated. However, computations of MAVRDSS are stopped after 120 seconds and best values are reported.

**Table 4.** Computational results for the Multi-Depot VRP.

N  ×  M  ×  K	ALNS		CGL		ITS/1		MAVRDSS	
	Gap	CT	Gap	CT	Gap	CT	Gap	CT
50 × 4 × 4	0.0	29	0.0	15	0.0	-	0.0	12
50 × 4 × 2	0.0	28	0.1	50	0.0	-	0.0	14
75 × 5 × 3	0.0	64	0.6	17	0.1	-	0.0	22
80 × 2 × 5	0.1	75	0.1	1	0.0	-	0.0	42
100 × 2 × 5	0.3	120	0.4	79	0.2	-	3.4	31
100 × 2 × 8	0.5	88	0.6	336	0.5	-	1.5	32
100 × 3 × 6	0.7	93	0.2	293	0.4	-	3.4	48
100 × 4 × 4	0.8	88	1.1	427	0.7	-	1.9	40
160 × 4 × 5	0.0	179	0.4	539	0.0	-	7.1	28
240 × 6 × 5	0.3	315	0.0	10	0.3	-	5.5	54
249 × 2 × 14	0.8	333	2.5	1175	2.2	-	7.0	102
249 × 3 × 12	0.5	361	1.6	419	1.7	-	5.4	88
249 × 4 × 8	0.5	363	2.3	984	1.7	-	5.2	89
249 × 5 × 6	0.8	357	1.0	1040	1.0	-	3.5	110
360 × 9 × 5	0.5	582	1.1	1786	0.7	-	0.9	78

Note: |N|: Number of customer nodes; |M|: Number of depot nodes; |K|: Number of vehicles/tours.

The obtained objective function values are mostly inferior compared to those of the three reference studies. However, the objective function values obtained by MAVRDSS are around the 5% optimality gap (to best-known solution). The results lead to a similar conclusion as for the Split-Delivery VRP: *LocalSolver* can produce decent solutions in short running times, but these solutions are hardly improved in the remaining time.

Table 5 compares the MAVRDSS solutions on the classic LRP (simultaneous determination of optimal depot locations and vehicle routes) to three algorithms that have shown the best performances in terms of solution quality and running time among six algorithms tested in a comparative study by Yu et al. [66].

**Table 5.** Computational results for the Location Routing Problem (LRP).

N  ×  M	MAPM		LRGTS		SALRP		MAVRDSS	
	Gap	CT	Gap	CT	Gap	CT	Gap	CT
21 × 5	0.0	0.3	0.0	0.2	0.0	18.3	0.0	1
22 × 5	4.6	0.3	0.4	0.2	0.0	16.6	0.0	1
27 × 5	0.0	1.0	0.1	0.3	0.0	23.3	0.0	2
29 × 5	0.0	0.8	0.0	0.4	0.0	23.9	0.0	1
32 × 5	6.0	1.0	0.1	0.5	0.0	25.1	0.0	2
36 × 5	5.4	1.4	3.5	0.7	0.0	31.7	0.0	2
50 × 5	0.0	3.8	3.7	2.4	0.0	52.8	0.0	18
75 × 10	8.4	9.4	8.1	10.1	0.4	126.8	2.2	94
88 × 8	1.5	34.2	5.2	17.5	0.0	226.9	0.8	65
100 × 10	3.9	44.5	3.0	28.2	0.6	330.8	2.2	86
134 × 8	9.7	110.5	7.1	48.3	0.0	552.4	3.4	9
150 × 10	1.4	255.0	2.3	119.2	2.7	577.0	6.0	80

Note: |N|: Number of customer nodes; |M|: Number of depot candidate sites.

These three heuristics are: MAPM [67], LRGTS [68], and SALPR [66]. The instance set for the LRP is gathered by Barreto [69]. These instances consider the classical Location-Routing constraints of capacitated depots and vehicles. Computations in MAVRDSS are stopped after 120 seconds and best values are reported. The comparative results indicate that MAVRDSS can compete with

problem-tailored LRP algorithms. For small datasets, the solutions reached best-known values within reasonable running times. For larger datasets, high quality solutions were reliably produced within seconds and further improved.

To conclude, the presented results demonstrate the effectiveness of MAVRDSS when applied on weakly constrained models. Especially with relatively few hard constraints, for instance present in a LRP, the risk of being stuck in locally optimal solutions is rather low. Although the computational results for the Multi-Depot VRP as well as for the Split-Delivery VRP show that *LocalSolver* does not always successfully diversify the search, it is able to produce valuable results in short computation times. For some instances, the obtained results even outperform the specialized reference heuristics. In summary, MAVRDSS enables to model various routing problems and achieve high quality solutions which can often compete with problem-tailored algorithms.

## 5.2. Real-World Simulations

After presenting a comparative evaluation based on different standard instances and problem-tailored solution approaches, two real-world application cases in the context of urban logistics are described to analyze the added value of MAVRDSS. The first case covers three working days of a freelance courier employed at a small-sized coffee supplier. The second case contains a day trip of a medium-sized parcel service provider. We compare the tool's solution with the actual distance driven by the couriers in both cases. Considering the presented classification scheme, relevant attributes of the routing problem in both cases are given depot locations, given customer locations, a deterministic demand, and only one product type. Capacitated vehicles with a limited range, closed routes, and single customer visits are further specifications. In addition, the first case comprises customer time windows that must be met when supplying customers. Minimizing the total distance while visiting all customers exactly once constitutes the central objective in both cases. In line with this efficiency maximization, the total vehicle emissions and the resulting transport costs are minimized leading to an optimized resource utilization. As many couriers just obtain their assigned customer destinations, they must plan the route on their own. Accordingly, the driver decides independently and directly on the ordering of destinations based on personal experience.

In the first case, a courier had to supply up to 12 customers in urban surroundings each day. The distance travelled by the courier ranged from 28.03 km to 42.07 km per trip. With an optimized order of destinations using MAVRDSS, the routes are reduced by distances from 4.61 km to 11.89 km per trip. This corresponds to a range of savings from 16.14% to 42.75%. In the second case, a courier had to deliver 119 parcels to 97 customers in urban surroundings, resulting in a total driving distance of 44.27 km in one trip of one day. Taking the same customer locations and solving the described optimization problem with MAVRDSS and the hardware described above (Intel Core i7 QuadCore CPU 2.2 GHz, 8GB memory), it takes 45 seconds to receive a solution with a total driving distance of 27.52 km. Reducing the distance by 16.75 km implies an efficiency enhancement of 37.83%. When using the same hardware and an exact solution method (solver: CPLEX) for the identical problem instance, it takes approximately 24 hours to receive a similar solution (28.94 km). Again, this difference regarding the computation time demonstrates the requirement of a (local search) heuristic when solving routing problems.

## 6. Discussion and Contributions

With MAVRDSS, we created a research artifact to provide relevant assistance for VRP of diverse industries in urban logistics. Guided along the three cycles of a DSR process, this section critically reviews our approach and discusses limitations as well as implications of the resulting artifact with respect to the above-mentioned research goal.

As part of the rigor cycle, we conducted a literature review focused on multi-attribute and real-world VRP. Our study integrates the existing knowledge on VRP and their solution approaches together with computer-aided decision support. In doing so, we combine scientific literature of

transportation, logistics, operations research, management science, as well as computer science. Previous research activities in these fields demonstrate the lack of decision support for generic vehicle routing planning. While most of the literature focuses on problem-tailored solution approaches, we developed a multi-attribute VRP allowing for the implementation of several problem-specific business contexts. A finding of the literature review is that many models explicitly denoted as rich or real-world VRP by their authors offering only few or no descriptions about addressed real-world issues. To keep track of the various model attributes considered in the research community, we derive a classification scheme for VRP. The development of the classification scheme is motivated by the observation that existing classification schemes are either too detailed (suffering from ambiguity), or too imprecise (leaving voids and being undescriptive when classifying the various model features). Our scheme of existing VRP attributes is capable of providing a fast overview by grouping attributes within problem information, physical, route, and objective characteristics. With the primary goal of an artifact's utility in the DSR process, the classification scheme is used as a navigating structure for the customizable part of the developed DSS. In this regard, the attribute classification aims to remain descriptive and easy to apply by users without prior knowledge of VRP nor optimization methods.

The identified model attributes are brought together in a prototype IS-artifact, called MAVRDSS. The main requirement is to provide a customizable VRP including these attributes to preserve the model's generality, which is met with only few exceptions as mentioned above. The translation of the attributes selected by the user into the optimization model components is a major challenge. Regarding the modeling environment of our DSS, a benefit of using *LocalSolver* and the built-in modeling language *Local Search Programming*, is the interface-to-model system enabling an integration of new model attributes. Since writing or adjusting existing heuristic algorithms is a long-winded process, a developer using *LocalSolver* can profit from its model-and-run approach, considering the specific limitations of the local search based resolution techniques. Consequently, MAVRDSS allows for a great variability of attribute combinations. On one side, the generality is a major benefit in terms of its applicability by a wide range of practitioners and researchers facing different distribution problems. On the other side, the model's generality is also a point of criticism as some company-specific scenarios are possibly not represented sufficiently by the pool of model attributes. Without a doubt, there are specifications that cannot be modeled with MAVRDSS, e.g., the integration of the parking situation at customers [70] or the avoidance of left-turning and zigzagging [36]. Hence, the prototype is not a "one-size-fits-all" solution, but rather a useful tool for most of decision makers with an optimization potential in terms of vehicle utilization efficiency and the resulting impacts.

The right choice of attributes represents the most crucial part regarding the performance of MAVRDSS. Since the MAVRDSS-users can specify a practical problem by choosing from a wide pool of model attributes, there are potential user-sided error sources which must be identified. On one side, adding too many constraints to the model complicates the move from one feasible solution to another. On the other side, the optimization model may lose its practical sense without real problem structuring constraints. MAVRDSS must offer appropriate guidance on various combinations of objectives, especially with respect to possible conflicts. While much effort is devoted to eliminating user-sided errors during the model configuration stage, there are only few validations on the input data that is passed to the optimization model.

Another essential element is the method to determine travel times. As described, the *Google* API is used to retrieve the estimated travel times as well as distances between the nodes of the system. In the present status of the tool, this estimation is based on the current traffic situation of the road network. As a potential extension of MAVRDSS, the user could choose whether the current or the average traffic situation should be taken as basis of the travel time and distance calculation. Thereby, the accuracy as well as the optimality of the solution could be enhanced in some application cases. In general, the travel time variability is a limitation of our approach. For instance, Shao et al. [71] present a VRP where the travel times between nodes depend on time periods. These result as a day can be subdivided into an adjustable number and length of time periods. Corresponding travel times result for each

period to account for variations of the road network conditions and the corresponding travel speed over the day. However, besides changing traffic conditions, there are several external events that can affect the schedule of a service provider during the day, e.g., a customer cancels his/her appointment, a customer requests a new time window, extreme weather conditions occur. Those events require a fast reevaluation of the changed conditions but as demonstrated by the benchmarks, MAVRDSS produces high quality results in short computation times so that a fast reoptimization is possible.

As part of the design cycle, MAVRDSS was evaluated on well-known academic VRP benchmarks to ensure validity and functionality of the model formulation as well as the chosen solution method. Since each study typically uses different hardware, the comparability of the presented benchmarks is limited. However, the presented results of the comparative evaluation generally document the efficiency of *LocalSolver* when applied on weakly constrained models. Regarding the resulting computation time and especially the potential attribute combinations, the formulation of the model attributes in the *Local Search Programming* language is crucial. For the considered instances, our implementation of formulating the model attributes can compete with problem-tailored heuristics. Solver specifications (simulated annealing level, number of threads, etc.) were kept constant during the benchmark tests leaving room for further variation.

As advised for the methodology of DSR, deeper empirical evaluation in the field forms a major part of the relevance cycle and will increase practicality, rigor, and generalizability of our approach. As 86.5% of the DSS related DSR artifacts [72], no complete field trial has been realized yet. In this regard, the MAVRDSS prototype should be tested in practice to validate its real-world utility. To apply the developed web-application, a company must own valid business licenses for the integrated software *LocalSolver* as well as the *Google* APIs. Instead of *Google* APIs, an open-source API (e.g., *OpenStreetMap*) could be used as an alternative. However, we recommend a cooperation with existing transport companies and other services to further validate and evaluate our developed DSS.

As stated in the introduction, the artifact development of MAVRDSS is also motivated by the potential to decrease the road traffic's impacts (e.g., emissions) when optimizing routes. In addition to economic costs, the environmental effects can be taken into consideration while constructing transportation routes. Many authors have dealt with the environmental sensitivity of routing aspects by including broader constraints and objectives in VRP models [52]. As the vehicle emissions are directly related to the travel distance and the according fuel consumption [73], emissions are currently not stated separately in MAVRDSS.

With our artifact classified as nascent design theory, we reacted to the call of Malhotra et al. [13] and Gholami et al. [14], who point out the overrepresentation of conceptualization and analyses compared to solution-oriented research. We combined transportation, logistics, operations research, management science, computer science, and especially Green IS research to promote the transformative role of IS in contributing to enhanced economic, social, and environmental sustainability. With the help of MAVRDSS, we enable better decision-making through easy usability for diverse companies in finding appropriate solutions for the delivery of goods and services, especially in cities.

By implementing the problem into a DSS prototype, we answer our research question by enabling generic route optimization leading to efficient vehicle utilization. The possibility to model a wide range of routing problems allows for a broad application in logistics problems of diverse industries. Due to the integration of *LocalSolver*, decent solutions in terms of running time and solution quality can be achieved. When comparing the obtained results to problem-tailored approaches, MAVRDSS is often able to achieve results with similar quality. Even if the frequently occurring transport problems are covered by existing VRP and problem-tailored solution approaches, small- and medium-sized companies often lack appropriate decision support (e.g., newspaper and grocery delivery, or craftsmen-, pharmacy-, and care-services). We tackle this gap with our prototype, because special VRP cannot be developed for each application case. In doing so, we contribute to DSR applications for environmental sustainability as well as the Green IS domain as our research addresses relevant issues regarding the efficient supply of people with goods and services. The use of MAVRDSS minimizes driving

distances resulting in emission reductions, especially compared to non-optimized routes. Because many companies do not use routing software yet, there are considerable saving potentials regarding driving distances, costs, and resulting emissions. Based on the prospering e-commerce, cost pressure continues to arise for companies, especially within the transport sector. By using MAVRDSS, companies can optimize their planning processes and minimize the distances of delivery processes for goods as well as services towards a more cost-efficient operation mode. When run on an efficient server, MAVRDSS can achieve even more advantages for decision makers through the use of a web-application.

## 7. Conclusions and Outlook

We present a technological artifact enabling generic route optimization taking present knowledge on VRP and related solution approaches into account. Previous research activities in these fields demonstrate the lack of decision support for customizable vehicle routing planning. In the context of urban logistics, this adaptability is important due to the broad applications in diverse industries. The developed DSS integrates a multi-attribute VRP and allows for optimizing transportation problems of several problem-specific business contexts. The presented results indicate that our DSS can compete with problem-tailored algorithms. The use of MAVRDSS minimizes driving distances resulting in emission reductions, especially compared to non-optimized routes. Companies can optimize their planning processes for goods transportation as well as services towards a more cost-efficient operation mode. As many companies do not use routing software, the saving potentials in terms of road traffic and its impacts are manifold. With the deployment of MAVRDSS, we enable better decision-making through easy usability for diverse companies in finding appropriate solutions for the delivery of goods and services, especially within urban surroundings.

**Author Contributions:** Conceptualization, M.L. and M.-O.S.; methodology, M.L. and M.-O.S.; software, M.L., M.-O.S., M.H., and T.K.; validation, M.L., M.H., and T.K.; writing—original draft preparation, M.L., M.-O.S., and T.K.; writing—review and editing, M.H.B.; visualization, M.L. and M.H.; supervision, M.H.B.

**Funding:** The Open Access fund of Leibniz Universität Hannover.

**Acknowledgments:** The publication of this article was funded by the Open Access fund of Leibniz Universität Hannover.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Drexel, M. Rich vehicle routing in theory and practice. *Logist. Res.* **2012**, *5*, 47–63. [[CrossRef](#)]
2. Hasle, G.; Kloser, O. Industrial vehicle routing. In *Geometric Modelling, Numerical Simulation, and Optimization*; Hasle, G., Lie, K.-A., Quak, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; pp. 397–436.
3. Toth, P.; Vigo, D. *The Vehicle Routing Problem*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2001.
4. Cattaruzza, D.; Absi, N.; Feillet, D.; González-Feliu, J. Vehicle routing problems for city logistics. *Eur. J. Transp. Logist.* **2017**, *6*, 51–79. [[CrossRef](#)]
5. Laporte, G. Fifty years of vehicle routing. *Transp. Sci.* **2009**, *43*, 408–416. [[CrossRef](#)]
6. Lahyani, R.; Khemakhem, M.; Semet, F. Rich vehicle routing problems. From a taxonomy to a definition. *Eur. J. Oper. Res. Soc.* **2015**, *241*, 1–14. [[CrossRef](#)]
7. Rademeyer, A.L.; Lubinsky, D.J. A decision support system for strategic, tactical and operational visit planning for on-the-road personnel. *S. Afr. J. Ind. Eng.* **2017**, *28*, 57–72.
8. Hartl, R.F.; Hasle, G.; Janssens, G.K. Special issue on rich vehicle routing problems. *Cent. Eur. J. Oper. Res.* **2006**, *14*, 103–104. [[CrossRef](#)]
9. Lahyani, R.; Coelho, L.C.; Khemakhem, M.; Laporte, G.; Semet, F. A multi-compartment vehicle routing problem arising in the collection of olive oil in Tunisia. *Omega* **2014**, *51*, 1–10. [[CrossRef](#)]
10. Gregor, S.; Hevner, A.R. Positioning and presenting design science research for maximum impact. *MIS Q* **2013**, *37*, 337–355. [[CrossRef](#)]

11. Dedrick, J. Green IS: Concepts and issues for information systems research. *Commun. Assoc. Inf. Syst.* **2010**, *27*, 173–184. [[CrossRef](#)]
12. Watson, R.T.; Boudreau, M.C.; Chen, A.J. Information systems and environmentally sustainable development: Energy informatics and new directions for the IS community. *Manag. Inf. Syst. Q.* **2010**, *34*, 23–38. [[CrossRef](#)]
13. Malhotra, A.; Melville, N.P.; Watson, R.T. Spurring impactful research on information systems for environmental sustainability. *Manag. Inf. Syst. Q.* **2013**, *37*, 1265–1274. [[CrossRef](#)]
14. Gholami, R.; Watson, R.T.; Molla, A.; Hasan, H.; Bjørn-Andersen, N. Information systems solutions for environmental sustainability: How can we do more? *J. Assoc. Inf. Syst.* **2016**, *17*, 521–536. [[CrossRef](#)]
15. Hassan, N.R. Value of IS research: Is there a crisis? *Commun. Assoc. Inf. Syst.* **2014**, *34*, 801–816. [[CrossRef](#)]
16. Hevner, A.R.; March, S.T.; Park, J.; Ram, S. Design science in information systems research. *Manag. Inf. Syst. Q.* **2004**, *28*, 75–105. [[CrossRef](#)]
17. Hevner, A.R. A three cycle view of design science research. *Scand. J. Inf. Syst.* **2007**, *19*, 87–92.
18. Venable, J.; Pries-Heje, J.; Baskerville, R. FEDS: A framework for evaluation in design science research. *Eur. J. Inf. Syst.* **2016**, *25*, 77–89. [[CrossRef](#)]
19. Baldacci, R.; Battarra, M.; Vigo, D. Routing a heterogeneous fleet of vehicles. In *The Vehicle Routing Problem: Latest Advances and New Challenges*; Golden, B.L., Raghavan, S., Wasil, E.A., Eds.; Operations Research/Computer Science Interfaces, Series; Springer: New York, NY, USA, 2008; Volume 43, pp. 3–27.
20. Crainic, T.G.; Crisan, G.C.; Gendreau, M.; Lahrichi, N.; Rei, W. A concurrent evolutionary approach for rich combinatorial optimization. In Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference, Montreal, QC, Canada, 8–12 July 2009; ACM: New York, NY, USA; pp. 2017–2022.
21. Caceres-Cruz, J.; Riera, D.; Juan, A.; Arias, P.; Guimarans, D. Rich vehicle routing problem: Survey. *ACM Comput. Surv.* **2015**, *47*, 1–29. [[CrossRef](#)]
22. Crainic, T.G.; Crisan, G.C.; Gendreau, M.; Lahrichi, N.; Rei, W. Multi-thread integrative cooperative optimization for rich combinatorial problems. In Proceedings of the 23rd IEEE International Parallel and Distributed Processing Symposium, Rome, Italy, 23–29 May 2009; IEEE: New York, NY, USA; p. 8.
23. Vidal, T.; Crainic, T.G.; Gendreau, M.; Lahrichi, N.; Rei, W. A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Oper. Res.* **2012**, *60*, 611–624. [[CrossRef](#)]
24. Kitchenham, B.; Brereton, O.P.; Budgen, D.; Turner, M.; Bailey, J.; Linkman, S. Systematic literature reviews in software engineering—A systematic literature review. *Inf. Softw. Technol.* **2009**, *51*, 7–15. [[CrossRef](#)]
25. Webster, J.; Watson, R. Analyzing the past to prepare for the future: Writing a literature review. *Manag. Inf. Syst. Q.* **2002**, *26*, 13–23.
26. Armas, J.; Melián-Batista, B.; Moreno-Pérez, J.A.; Brito, J. GVNS for a real-world rich vehicle routing problem with time windows. *Eng. Appl. Artif. Intell.* **2015**, *42*, 45–56. [[CrossRef](#)]
27. Eksioglu, B.; Vural, A.V.; Reisman, A. The vehicle routing problem. A taxonomic review. *Comput. Ind. Eng.* **2009**, *57*, 1472–1483. [[CrossRef](#)]
28. Amorim, P.; Parragh, S.N.; Sperandio, F.; Almada-Lobo, B. A rich vehicle routing problem dealing with perishable food: A case study. *TOP* **2014**, *22*, 489–508. [[CrossRef](#)]
29. Cattaruzza, D.; Absi, N.; Feillet, D. The multi-trip vehicle routing problem with time windows and release dates. *Transp. Sci.* **2016**, *50*, 676–693. [[CrossRef](#)]
30. Ceselli, A.; Righini, G.; Salani, M. A column generation algorithm for a rich vehicle-routing problem. *Transp. Sci.* **2009**, *43*, 56–69. [[CrossRef](#)]
31. Coelho, V.N.; Grasas, A.; Ramalhinho, H.; Coelho, I.M.; Souza, M.J.F.; Cruz, R.Z. An ILS-based algorithm to solve a large-scale real heterogeneous fleet VRP with multi-trips and docking constraints. *Eur. J. Oper. Res. Soc.* **2016**, *250*, 367–376. [[CrossRef](#)]
32. Derigs, U.; Kurowsky, R.; Vogel, U. Solving a real-world vehicle routing problem with multiple use of tractors and trailers and EU-regulations for drivers arising in air cargo road feeder services. *Eur. J. Oper. Res. Soc.* **2011**, *213*, 309–319. [[CrossRef](#)]
33. Derigs, U.; Pullmann, M.; Vogel, U. A short note on applying a simple LS/LNS-based metaheuristic to the rollon–rolloff vehicle routing problem. *Comput. Oper. Res.* **2013**, *40*, 536–546. [[CrossRef](#)]
34. Goel, A. A column generation heuristic for the general vehicle routing problem. In Proceedings of the International Conference on Learning and Intelligent Optimization, Venice, Italy, 18–22 January 2010; p. 9.

35. Hemmelmayr, V.; Doerner, K.F.; Hartl, R.F.; Rath, S. A heuristic solution method for node routing based solid waste collection problems. *J. Heuristics* **2013**, *19*, 129–156. [[CrossRef](#)]
36. Holland, C.; Levis, J.; Nuggehalli, R.; Santilli, R.; Winters, J. UPS optimizes delivery routes. *INFORMS J. Appl. Anal.* **2017**, *47*, 8–23. [[CrossRef](#)]
37. Ibeas, A.; Moura, J.L.; Dell’Olio, L. Planning school transport: Design of routes with flexible school opening times. *Transp. Plan. Technol.* **2009**, *32*, 527–544. [[CrossRef](#)]
38. Kovacs, A.A.; Golden, B.L.; Hartl, R.F.; Parragh, S.N. The generalized consistent vehicle routing problem. *Transp. Sci.* **2015**, *49*, 796–816. [[CrossRef](#)]
39. López-Sánchez, A.D.; Hernández-Díaz, A.G.; Vigo, D.; Caballero, R.; Molina, J. A multi-start algorithm for a balanced real-world open vehicle routing problem. *Eur. J. Oper. Res. Soc.* **2014**, *238*, 104–113. [[CrossRef](#)]
40. Oppen, J.; Løkketangen, A.; Desrosiers, J. Solving a rich vehicle routing and inventory problem using column generation. *Comput. Oper. Res.* **2010**, *37*, 1308–1317. [[CrossRef](#)]
41. Osaba, E.; Yang, X.-S.; Diaz, F.; Onieva, E.; Masegosa, A.D.; Perallos, A. A discrete firefly algorithm to solve a rich vehicle routing problem modelling a newspaper distribution system with recycling policy. *Soft Comput.* **2017**, *21*, 5295–5308. [[CrossRef](#)]
42. Prescott-Gagnon, E.; Desaulniers, G.; Rousseau, L.-M. Heuristics for an oil delivery vehicle routing problem. *Flexible Serv. Manuf. J.* **2014**, *26*, 516–539. [[CrossRef](#)]
43. Rais, A.; Alvelos, F.; Carvalho, M.S. New mixed integer-programming model for the pickup-and-delivery problem with transshipment. *Eur. J. Oper. Res. Soc.* **2014**, *235*, 530–539. [[CrossRef](#)]
44. Rasmussen, M.S.; Justesen, T.; Dohn, A.; Larsen, J. The home care crew scheduling problem. Preference-Based Visit Clustering and Temporal Dependencies. *Eur. J. Oper. Res. Soc.* **2012**, *219*, 598–610. [[CrossRef](#)]
45. Rieck, J.; Zimmermann, J. A new mixed integer linear model for a rich vehicle routing problem with docking constraints. *Ann. Oper. Res.* **2010**, *181*, 337–358. [[CrossRef](#)]
46. Santillán, C.G.; Reyes, L.C.; Rodriguez, M.L.M.; Barbosa, J.J.G.; López, O.C.; Zarate, G.R.; Hernández, P. Variants of VRP to optimize logistics management problems. In *Logistics Management and Optimization through Hybrid Artificial Intelligence Systems*; Zezzatti, O., Ochoa, C.A., Chira, C., Hernandez, A., Basurto, M., Eds.; IGI Global: Hershey, PA, USA, 2012; pp. 207–237.
47. Schmid, V.; Doerner, K.F.; Laporte, G. Rich routing problems arising in supply chain management. *Eur. J. Oper. Res. Soc.* **2013**, *224*, 435–448. [[CrossRef](#)]
48. Soysal, M.; Bloemhof-Ruwaard, J.M.; Bektaş, T. The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations. *Int. J. Prod. Econ.* **2015**, *164*, 366–378. [[CrossRef](#)]
49. Stenger, A.; Schneider, M.; Goetze, D. The prize-collecting vehicle routing problem with single and multiple depots and non-linear cost. *Eur. J. Transp. Logist.* **2013**, *2*, 57–87. [[CrossRef](#)]
50. Wen, M.; Cordeau, J.-F.; Laporte, G.; Larsen, J. The dynamic multi-period vehicle routing problem. *Comput. Oper. Res.* **2009**, *37*, 1615–1623. [[CrossRef](#)]
51. Lenstra, J.K.; Kan, A.H.G. Complexity of vehicle routing and scheduling problems. *Networks* **1981**, *11*, 221–227. [[CrossRef](#)]
52. Lin, C.; Choy, K.L.; Ho, G.T.S.; Chung, S.H.; Lam, H.Y. Survey of green vehicle routing problem. Past and future trends. *Expert Syst. Appl.* **2014**, *41*, 1118–1138. [[CrossRef](#)]
53. Cordeau, J.-F.; Gendreau, M.; Laporte, G.; Potvin, J.V.; Semet, F. A guide to vehicle routing heuristics. *J. Oper. Res. Soc.* **2002**, *53*, 512–522. [[CrossRef](#)]
54. Talbi, E.G. *Metaheuristics: From Design to Implementation*, 74th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2009.
55. Laporte, G. The vehicle routing problem. An overview of exact and approximate algorithms. *Eur. J. Oper. Res. Soc.* **1992**, *59*, 345–458. [[CrossRef](#)]
56. Pisinger, D.; Ropke, S. A general heuristic for vehicle routing problems. *Comput. Oper. Res.* **2007**, *34*, 2403–2435. [[CrossRef](#)]
57. Kritzinger, S.; Doerner, K.F.; Hartl, R.F.; Kiechle, G.; Stadler, H.; Manohar, S.S. Using traffic information for time-dependent vehicle routing. *Procedia Soc. Behav. Sci.* **2012**, *39*, 217–229. [[CrossRef](#)]
58. Bräysy, O.; Gendreau, M. Vehicle routing with time windows, part 1: Route construction and local search algorithms. *Transp. Sci.* **2005**, *39*, 104–118. [[CrossRef](#)]
59. Hartl, M. Ruby on Rails Tutorial: Learn Web Development with Rails. Available online: <https://www.railstutorial.org/book/> (accessed on 10 August 2019).

60. Gardi, F.; Benoist, T.; Darlay, J.; Estellon, B.; Megel, R. Local Search for 0-1 Nonlinear Programming. In *Mathematical Programming Solver Based on Local Search*; Gardi, F., Benoist, T., Darlay, J., Estellon, B., Megel, R., Eds.; John Wiley & Sons: Hoboken, NJ, USA, 2014; pp. 29–51.
61. Belenguer, J.M.; Martinez, M.C.; Mota, E. A lower bound for the split delivery vehicle routing problem. *Oper. Res.* **2000**, *48*, 801–810. [[CrossRef](#)]
62. Silva, M.M.; Subramanian, A.; Ochi, L.S. An iterated local search heuristic for the split delivery vehicle routing problem. *Comput. Oper. Res.* **2015**, *53*, 234–249. [[CrossRef](#)]
63. Aleman, R.E.; Zhang, X.; Hill, R.R. A ring-based diversification scheme for routing problems. *Int. J. Math. Oper. Res.* **2009**, *1*, 163–190. [[CrossRef](#)]
64. Cordeau, J.-F.; Maischberger, M. A parallel iterated tabu search heuristic for vehicle routing problems. *Comput. Oper. Res.* **2010**, *39*, 2033–2050. [[CrossRef](#)]
65. Cordeau, J.-F.; Gendreau, M.; Laporte, G. A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks* **1997**, *30*, 105–119. [[CrossRef](#)]
66. Yu, V.F.; Lin, S.-W.; Lee, W.; Ting, C.-J. A simulated annealing heuristic for the capacitated location routing problem. *Comput. Ind. Eng.* **2010**, *58*, 288–299. [[CrossRef](#)]
67. Prins, C.; Prodhon, C.; Wolfler Calvo, R. A memetic algorithm with population management (MA| PM) for the capacitated location-routing problem. In *Proceedings of the European Conference on Evolutionary Computation in Combinatorial Optimization*, Budapest, Hungary, 10–12 April 2006; Gottlieb, J., Raidl, G.R., Eds.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 183–194.
68. Prins, C.; Prodhon, C.; Ruiz, A.; Soriano, P.; Wolfler Calvo, R. Solving the capacitated location-routing problem by a cooperative lagrangean relaxation-granular tabu search heuristic. *Transp. Sci.* **2007**, *41*, 470–483. [[CrossRef](#)]
69. Barreto, S.S. Analysis and modelling of location-routing problems. Ph.D. Thesis, Aveiro University, Aveiro, Portugal, October 2004. (In Portuguese).
70. Holguin-Veras, J.; Hodge, S.; Wojtowicz, J.; Singh, C.; Wang, C.; Jaller, M.; Aros-Vera, F.; Ozbay, K.; Weeks, A.; Replogle, M.; et al. The New York City off-hour delivery program: A business and community-friendly sustainability program. *INFORMS J. Appl. Anal.* **2018**, *48*, 70–86. [[CrossRef](#)]
71. Shao, S.; Guan, W.; Ran, B.; He, Z.; Bi, J. Electric vehicle routing problem with charging time and variable travel time. *Math. Probl. Eng.* **2017**, *5098183*, 1–13. [[CrossRef](#)]
72. Arnott, D.; Pervan, G. Design science in decision support systems research: An assessment using the Hevner, March, Park, and Ram guidelines. *J. Assoc. Inf. Syst.* **2012**, *13*, 923–949. [[CrossRef](#)]
73. Figliozzi, M. Vehicle routing problem for emissions minimization. *Transp. Res. Rec.* **2010**, *2197*, 1–7. [[CrossRef](#)]

