Public Transport (2023) 15:377–409
https://doi.org/10.1007/s12469-022-00312-5

ORIGINAL RESEARCH

A simulation-based optimization approach for designing transit networks

Obiora A. Nnene[1] · Johan W. Joubert[2] · Mark H. P. Zuidgeest[1]

Accepted: 13 November 2022 / Published online: 18 January 2023

Abstract

Public transport network design deals with finding efficient network solution(s) from a set of alternatives that best satisfies the often-conflicting objectives of stakeholders like passengers and operators. This work presents a simulation-based optimization (SBO) model for designing public transport networks. The work's novelty is in developing such a network design model that fully accounts for the stochastic behavior of commuters on the transit network. The SBO discipline solves decision-based problems like the transit network design problem (TNDP) by combining simulation and optimization models. The proposed model integrates a disaggregated activity-based travel demand simulation with a multi-objective network optimization algorithm. Trip-based travel demand models are commonly used to represent traveler behavior in the literature. The approach limits its ability to accommodate the stochastic realities of traveler behavior in a transit network design solution. Using activity-based simulation instead makes it possible to account for a more realistic traveler behavior, especially real-time decisions made in response to changing network dynamics which ultimately affect the distribution of demand over time on the network. The proposed model is applied to the improved design of the integrated public transport network in the City of Cape Town, South Africa. The results show SBO can design efficient network solutions that reflect the objectives of network stakeholders.

Keywords Simulation-based optimization · Transit network design · Activity-based modeling · Multi-objective optimization · Metaheuristics

✉ Obiora A. Nnene
obiora.nnene@uct.ac.za

Extended author information available on the last page of the article

1 Introduction

Public transport network design deals with finding efficient network solution(s) from a set of alternatives that best satisfies the often-conflicting objectives of stakeholders like passengers and operators. The commuter aims to minimize their total travel time and other associated costs. At the same time, the operator sees costs in terms of the total resources needed to operate a profitable service. Hence, their goal is to minimize this cost even at the risk of serving routes the commuter may consider lengthy and unattractive. Therefore, solving a transit network design problem (TNDP) requires finding a compromise between these conflicting goals. The known methods for solving the TNDP in the literature are broadly classified as analytical and heuristic. The former comprises exact search algorithms which attempt to find the closed form of an objective function in the search for a best possible solution to the problem. Recent research with analytical solutions in the literature are Constantin and Florian (1995), Lee and Vuchic (2005), Chen et al. (2017), Daganzo and Ouyang (2019), and Ranjbari et al. (2020). However, these analytical solutions are limited in solving the TNDP due to the nearly infinite amount of resources needed to find a solution to even relatively small transit network design problems (Chakroborty 2003). On the other hand, the TNDP lends itself to heuristic solution techniques, especially metaheuristics, owing to their relatively simple adaptation to extensive TNDP case studies. Furthermore, metaheuristics are *approximate* algorithms that can find *good* solution(s) in a reasonable amount of time. Some works utilizing heuristic approaches include Pattnaik et al. (1998), Fan and Machemehl (2004), Alrabghi and Tiwari (2015), Huang et al. (2018), Nnene et al. (2019), Yang and Jiang (2020).

In this paper, the authors present a simulation-based optimization (SBO) approach for solving the TNDP. This method combines simulation with optimization models to solve decision-based problems like the TNDP (Gosavi 2015a). The goal of the paper is to develop a so-called simulation-based transit network design model (SBTNDM) that integrates simulation with optimization and in the design of transit networks. The research question revolves around how SBO can be applied to the optimized design of transit networks, leading to the realization of efficient network solutions. In the SBTNDM, an activity-based simulation (ABS) evaluates alternative network solutions by simulating travel demand on them whilst a multi-objective optimization algorithm searches for efficient network solutions. The paper's main contribution is solving the TNDP by integrating a disaggregated activity-based travel demand simulation with a multi-objective metaheuristic network optimization solution framework. Using activity-based simulation makes it possible to fully account for the microscopic behavior of travelers and other agents on the network. Also, the network design process can incorporate temporal fluctuations in demand. The static trip-based travel demand model is commonly used to represent traveler behavior in the literature, as seen in the reviews of Johar et al. (2016), Durán-Micco and Vansteenwegen (2022), Ibarra-Rojas et al. (2015). However, they cannot account for an event such as a route change made in response to unexpected road closures or other stochastic decisions made by agents in real-time while responding to changes in network dynamics. Hence, it is advantageous to use ABS as it offers a more detailed and accurate reflection of demand distribution on the network, which is a critical consideration when operators choose the area to provide a service. Furthermore, a detailed service timetable rather than headway is a crucial input in the ABS. Therefore, the simulation offers a way to solve the TNDP, which allows for a feedback loop between travel demand and service supply. However, using the detailed timetable introduces the need to customize the network decision variable's encoding to facilitate the operations of the optimization algorithm. As a result, the authors define a custom encoding scheme to address this. This custom encoding is considered the other contribution of the paper, since representations like vectors and strings are used more commonly in the literature (Szeto and Wu 2011; Buba and Lee 2018). The proposed solution model is ultimately applied to the design of a transit network in Cape Town, South Africa, which needs improvement in terms of operational cost reduction and ridership increase.

In the remainder of this paper, Sect. 2 presents a theoretical background for the proposed model, and Sect. 3 presents the mathematical model for the problem and Sect. 4 outlines the component algorithms of the proposed SBO network design solution framework. Section 5 presents the results of testing the proposed solution and discusses its application to a large-scale transit network in Cape Town, South Africa, mainly as it affects passengers and service operators. In the final section, possible areas of future research are highlighted.

2 Literature review

This section starts with an overview of the major developments in the TNDP literature, specifically in terms of how previous researchers have tackled the network design problem. The discourse is then narrowed to the applications of SBO in the TNDP literature, which is more relevant to our work. Thereafter, we discuss the key components of the SBTNDM and their operations. With the guiding question of this research focussing on how the SBO is applied to the TNDP, it is important to understand solutions trends both in the literature and how they are evolving. Before the year 2000, many TNDP solution attempts used analytical methods. However, since then, metaheuristic solution models have gained prominence among researchers. Advances in operations research and computational science literature may have aided this development, making implementing and using metaheuristic procedures relatively straightforward. TNDP solution algorithms are classified by the number of objectives in the problem, namely single and multiple-objective optimization. The key difference between single and multi-objective solution algorithms is that in the former, a linear summation of all objectives is used to reduce many objectives to a single one. Furthermore, weights must be defined beforehand for each objective, and the obtained single results reflect the weighted objectives (Mauttone and Urquhart 2009). In contrast, the outcome of multi-objective algorithms is a Pareto frontier (Knowles et al. 2008), which represents the possible trade-offs between a problem's objectives, making it possible to obtain valuable information about these trade-offs and sensitivity for weighting the various objectives in terms of an optimal design solution (Possel et al. 2018). Among other reasons, however, this makes the multi-objective solution approach more complex than the single-objective version. Examples of single-objective solution approaches are Cipriani et al. (2012); Chen et al. (2017); Nnene et al. (2017) and Cipriani et al. (2020), while multi-objective solution models include Brands and van Berkum (2014); Heyken Soares et al. (2019) and Momenitabar and Mattson (2021). Recent review articles on the TNDP include those by Durán-Micco and Vansteenwegen (2022), Iliopoulou et al. (2019), Ibarra-Rojas et al. (2015) and Johar et al. (2016).

SBO applications in transportation planning, especially transit network design, are of interest in this paper. The technique has been applied to different aspects of transportation research like traffic signal design control (Osorio and Selvam 2015; Osorio 2016), in which the authors combine a mathematical model with traffic simulators to identify points on a network with high-level performance in terms of a stated indicator. Song et al. (2013) performed the minimization of generalised cost on a multimodal transport system using a proprietary transport simulation software VISUM that was combined with a genetic algorithm. Furthermore, Yan et al. (2013), in their attempt to solve the robust network design problem, used the Monte Carlo simulation to model travel demand flow with an embedded discrete choice model to represent passenger choices. While Hassannayebi et al. (2021) and Gao et al. (2022) receptively apply a discrete event simulation to the rescheduling and passenger capacity analysis on rail services. Lastly, Bal and Badurdeen (2022) apply SBO to the optimization of circular networks to make location and allocation decisions when implementing a lease and sell strategy. More relevant to this paper are the works of Dandl et al. (2021) and Ma and Chow (2022). This is because the authors use activity-based simulations in their solutions. The former presents an SBO solution framework, which combines Bayesian optimization with an agent-based transport system simulation within a tri-level optimization solution framework. Their research objective was to capture inter-decision dynamics between mobility service operators and commuters which could then be used to optimize and analyse policies that relate to service providers. Within their solution framework, the policymaker represented the highest level, the operator represented the middle level, and the traveler was at the lowest level. The Bayesian optimization algorithm was used to maximize social benefits for the authorities and profit for the operator, while agent-based simulation was used to simulate user behavior on the network. The model was applied to the case of toll and parking costs for automated mobility-on-demand systems in Munich, Germany. Also, Ma and Chow (2022) proposed a bi-level modeling framework for solving the transit frequency setting problem. The authors used an analytical route cost function representing the upper level and a lower level represented by an agent-based market equilibrium function which takes the frequencies of the routes and outputs demand for the transit network representing the lower level. The problem is applied to the case of the Brooklyn bus network in New York, USA, which is done with the idea of understanding how the service performs in competition with dial-a-ride services.

Due to the increased understanding of the power of simulations in evaluating complex stochastic systems and the advancement of computation science, more researchers are using them in transit network design. The two works that use ABS are most relevant to the model proposed in this paper though their problem objective and application context differ. As such, the authors make a valuable contribution in applying and advancing the use of SBO technique to solve the TNDP particularly in the context of public transportation. It is also important to highlight that the research in this paper is a metaheuristic SBO, combining a multi-objective metaheuristic algorithm known as the non-dominated sorting genetic algorithm NSGA-II with an agent-based travel demand simulation known as multi-agent transport simulation (MATSim). The following section discusses these two modeling components for the proposed model.

2.1 Modeling components

2.1.1 Non-dominated sorting genetic algorithm-II

Deb et al. (2000) is credited for the development of the NSGA-II, a typical multi-objective evolutionary algorithm (MOEA). Their operations mimic biological phenomena like genetics and bee or ant colonies, and they are thus called *bio-inspired* algorithms (Branke et al. 2008; Rangaiah and Bonilla-Petriciolet 2013; Elarbi et al. 2017). This class of algorithms works by enabling the realization of newer and presumably better generations of solutions from existing ones. To apply the NSGA-II to the TNDP, an initial population that constitutes the problem's search space must be generated. This population is made up of feasible network alternatives or chromosomes, and each chromosome possesses genes or routes. The best-performing chromosome in the population often represents a near-optimum solution, given that for very difficult problems like the TNDP it is not feasible to know if a solution is *optimum*. The chromosomes or networks must also be encoded in a way that is amenable to the algorithm's operators. In the literature, string and binary encodings have been the most common representations used when solving the TNDP. In Buba and Lee (2018), a string is used to represent the network route, while a tuple is used to represent the route's operational frequency as the number of vehicles operated per hour and the unique identifier for that route. However, in this paper an innovative encoding based on the JavaScript object notation (JSON) data structure (Crockford 2011) is used to facilitate the simultaneous handling of the route network design and frequency setting problems. Details of this encoding and how it is used in the proposed SBTNDM are discussed in Sect. 4.3, where the model's implementation is described. After encoding the network solutions, they are scored or evaluated against the objective function(s) of the problem. After this, the initialized solutions are evaluated against the objective functions and sorted into different Pareto frontiers using the non-dominated sorting procedure. A solution is considered to be non-dominated if it performs better than other solutions in at least one objective and is not worse than the other objectives. Hence, all non-dominated solutions are ranked 1 and temporarily removed, then the next set of non-dominated solutions are identified and ranked till all the solutions are ranked. The rank of each frontier is assigned as their fitness score and used to indicate the dominance of solutions. After ranking the solution, a *binary tournament* selection operator is used to select parents that will be used to reproduce the offspring. The operator randomly chooses two solutions, determines the fitter of both and then adds that one to the mating pool. To achieve this, the binary tournament operator uses a *crowded comparison* procedure, which measures the total distance of both solutions relative to their neighbours. The solution with a larger crowded distance is considered fitter and selected as the larger distance indicates a better spread in the Pareto frontier. First, the selected solutions are compared based on their dominance or rank value. However, if they both belong to the same front, i.e. they do not dominate each other, the crowding distance is then used to obtain the better solution. In the next step, the genetic operators, namely crossover and mutation, are used to create a population of children/offspring of a size equivalent to that of the first parent. These operators are used to generate offspring and introduce diversity in the population, respectively. Thereafter, the procedure is slightly different from the first generation: the generated offspring and parent are merged to form a population that is twice the size of the original population in every subsequent generation. The merged population is evaluated and again ranked according to the non-dominance and crowding distance criteria, and the better-performing half of the merged populations is selected as the new parent population. This process goes on iteratively until a specified termination condition is satisfied. Elitism is introduced in the algorithm by archiving a small percentage of the best-performing, *elite* solutions from both the parent and offspring populations during successive generations, which are reused as part of the parent population in the next generation.

2.1.2 MATSim

MATSim is an activity-based multi-agent simulation framework which models the microscopic demand of travelers by simulating their daily activity schedule and decision-making on a transport network. The modeled travelers are called agents and the simulation is designed to model their travel demand and stochastic decision making in 24-hour periods. In terms modeling public transit systems, MATSim organises data in a format that is commonly used by public transit services worldwide (Horni et al. 2016). A public transport network line modeled in MATSim will therefore comprise two or more transit routes. Each route serves one direction of travel and enables transit vehicles to move to and from the depot at the end and beginning of a day, respectively. The routes also have as an attribute the list of `departures`, which gives information about the time a vehicle starts at the first stop on that route. A route also includes a sequential list of transit stops that are served, alongside operating timetables that indicate when vehicles arrive or leave a stop. The times are specified as offsets in time units from the departure at the first stop so that at each subsequent stop, the offset is added to the initial departure time from the first stop. Each departure contains a vehicle's start time on the route and a reference to the vehicle. As the timing information is part of the route, it becomes possible to have routes with identical stop sequences but different time offsets. Stop locations are described by their coordinates and an optional `name` or `id`. They must be assigned to unique lines of the network for the simulation. The hierarchical tree

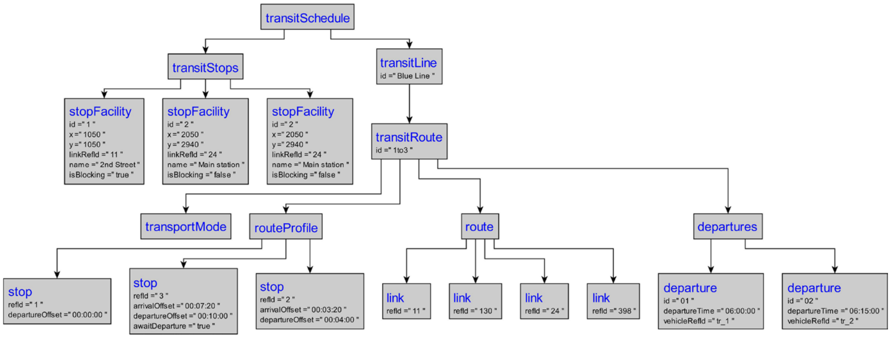


Fig. 1 A hierarchical tree structure for the transit schedule file

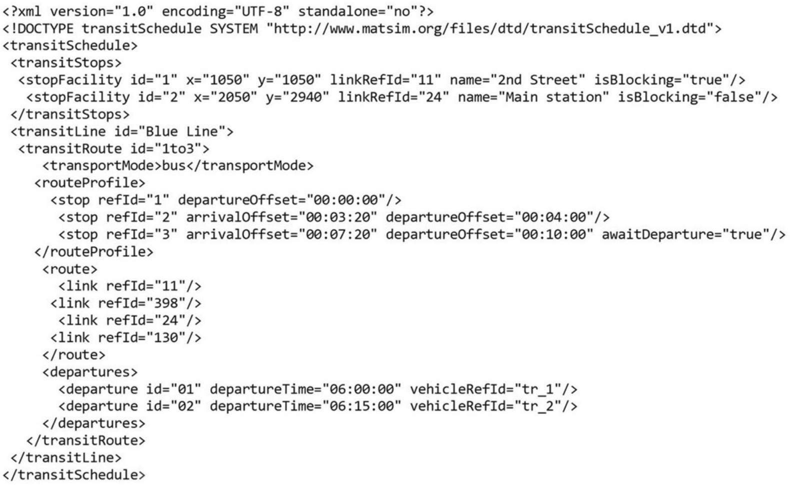

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE transitSchedule SYSTEM "http://www.matsim.org/files/dtd/transitSchedule_v1.dtd">
<transitSchedule>
 <transitStops>
  <stopFacility id="1" x="1050" y="1050" linkRefId="11" name="2nd Street" isBlocking="true"/>
   <stopFacility id="2" x="2050" y="2940" linkRefId="24" name="Main station" isBlocking="false"/>
 </transitStops>
 <transitLine id="Blue Line">
  <transitRoute id="1to3">
     <transportMode>bus</transportMode>
    <routeProfile>
      <stop refId="1" departureOffset="00:00:00"/>
       <stop refId="2" arrivalOffset="00:03:20" departureOffset="00:04:00"/>
       <stop refId="3" arrivalOffset="00:07:20" departureOffset="00:10:00" awaitDeparture="true"/>
    </routeProfile>
     <route>
       <link refId="11"/>
      <link refId="398"/>
       <link refId="24"/>
      <link refId="130"/>
     </route>
     <departures>
       <departure id="01" departureTime="06:00:00" vehicleRefId="tr_1"/>
       <departure id="02" departureTime="06:15:00" vehicleRefId="tr_2"/>
     </departures>
   </transitRoute>
 </transitLine>
</transitSchedule>
```

Fig. 2 The MATSim transit schedule file with routes and their schedules

structure for the schedule file can be seen in Figure 1 while the transit schedule file may be seen in Figure 2.

To model congestion on a public transit network, MATSim adopts a queue-based traffic flow model. This means that vehicles enter a link from an intersection, join the end of a waiting queue and remain there until the time required to travel the link with free flow has lapsed and they are at the head of the waiting queue. In terms of routing transit demand, an events-based public transport router is used in the MATSim simulation environment. Its main input are the commuter's start time, origin and destination pair (OD). The router mimics reality in its ability to compute alternative routes for agents. Their originally scheduled departure can either be due to the late arrival of a vehicle or to it arriving full and being unable to take more passengers. This is achieved by taking the transit service's given schedule as a base in the first iteration, then generating updated information on travel times, vehicle occupancy, and waiting times between subsequent iterations.

Next, the operational steps required to model transport in MATSim are described.

1. Initial demand generation: The initial demand is generated by creating daily activity plans from socioeconomic and demographic data of agents within a given transportation area. The demand is usually generated through sampling or discrete choice modeling and is subsequently converted to activity chains or *plans* for the agents.
2. Execution: This involves simulating the generated demand. The plans are executed sequentially by time of occurrence in a way that respects certain boundary conditions, like the closing hours of a shop or the maximum link and flow capacity of a road. The constraint represents the physical infrastructure where the activities and trips will be undertaken (Meister et al. 2010). Another name for this step is mobility simulation, or *mobsim* for short.
3. Scoring: After executing the agents' plans, the plans are evaluated. A score is obtained by evaluating the plan using a utility function known as a *scoring function*. MATSim uses the scores to measure and compare the quality of a passenger's plan to determine whether it should be dropped or not.
4. Replanning or innovation strategy: Agents adapt their plans in response to changes in the transit network, allowing the agent to modify their plans as they *learn* about prevailing network conditions, making it possible for the agent to maximize their experience on the public transport network. Details of these steps can be found in Horni et al. (2016).
5. Termination and post-analysis MATSim: This specifies a termination criterion that signals the simulation to stop when the condition has been met. Meister et al. (2010) describe this termination point as an agent-based stochastic user equilibrium (SUE). The system runs until the score of the agent's plan does not meaningfully improve, marking the end of the simulation. Post-analysis involves collecting and aggregating network performance indicators, passenger mileage and average trip duration to gain insight into the travel demand and simulated behavior of agents within the study area.

The input data for a MATSim simulation are the network which contains information about nodes and links, plans which is the daily activity chains of all travelers, transit schedules that consist of routes and their departure times, transit vehicles which details the operational fleet and their characteristics and lastly, the configuration file that is a collection of parameter settings needed to run the simulation. These are formatted as Extensible Markup Language (XML) (Bray et al. 2006) data structures. In this work, the transit supply side data such as the network and operational schedules can be extracted from General Transit Feed Specification (GTFS) data of a transit service. On the other hand, the demand data in MATSim is created from sources like travel diary surveys, census data and other passenger usage information sources. In this work the passenger activity chains were derived from automated fare collection data for the network being designed.

3 Models

The overarching optimization goal of the work is to minimize costs for transit users and operators of the network. This is depicted in the objective functions in Equation (1). For the user objective shown in Equation (2), the expression is summation of total travel time, transfers and a penalty for unsatisfied demand. Total travel time is obtained by multiplying a monetary factor for time by the generalised cost of travel of the commuters. Generalised cost is the sum of the access time ($t_a^{r_k}$), waiting time ($t_w^{r_k}$), in-vehicle travel time ($t_{trv}^{r_k}$) and transfer time (n_{tr}) where applicable. Due to the negative perception commuters have of transfers on the transit network (Owais 2015), a time penalty (ϕ_{time}) is applied to trips involving transfers in the model. Lastly, the unsatisfied penalty is applied to each network solution for the amount of travel demand not satisfied by the network. It is obtained by multiplying the total unsatisfied demand ($q_u^{r_k}$) by a time (t_u) and monetary factor (β_{unsat}) for unsatisfied demand. In Equation (3) the operators are concerned with the total operational cost, which is the sum of distance ($d_r^{r_k}$) operated by the vehicle fleet ($n_b^{r_k}$) multiplied by a monetary factor for vehicle mileage and the total vehicle time $t_r^{r_k}$ multiplied by the fleet and its corresponding monetary factor. Operational distance is the cost that accrues from wear and tear on the operator's vehicles as they traverse the designated routes to satisfy passenger demand and is typically measured in kilometres. However, operational time consists of personnel cost elements, such as salaries, that accrue throughout operations. By minimizing these objective functions, the total cost incurred on the network will be optimized for the stakeholders. Thus, the model is formulated as follows:

$$\text{Min} : Z_1, Z_2 \tag{1}$$

$$\begin{aligned} Z_1 = \beta_{\text{time}} \cdot \left(\sum_{r \in R} \sum_{r_k \in R_{m_r}} t_{trv}^{r_k} q_{trv}^{r_k} + \sum_{r \in R} \sum_{r_k \in R_{m_r}} t_a^{r_k} q_a^{r_k} + \sum_{r \in R} \sum_{r_k \in R_{m_r}} t_w^{r_k} q_w^{r_k} \right) \\ + \phi_{\text{time}} \cdot \sum_{n \in N} n_{tr} + \left(\beta_{\text{unsat}} \cdot t_u \cdot \sum_{r \in R} \sum_{r_k \in R_{m_r}} q_u^{r_k} \right) \end{aligned} \tag{2}$$

$$Z_2 = \left(\beta_{\text{dist}} \cdot \sum_{r \in R} \sum_{r_k \in R_{m_r}} d_r^{r_k} n_b^{r_k} + \beta_{\text{op}} \cdot \sum_{r \in R} \sum_{r_k \in R_{m_r}} t_r^{r_k} n_b^{r_k} \right) \tag{3}$$

subject to an agent-based route selection model which is based on the conditional probability of the average route cost for both the user and operator

$$P_n(k) = P_n(k \mid E\{\tau(x(\{r_k^n\}))\}) \tag{4}$$

and some feasibility conditions on route length, frequency and vehicle fleet:

$$n_b^{r_k} < B \tag{5}$$

$$r_{tot} \leq R_{max} \tag{6}$$

$$d_{min} \leq d_r^{r_k} \leq d_{max} \tag{7}$$

3.1 Sets

N = set of nodes on the network (-)
R = set of transit routes (-)
R_{m_r} = set of segments r_k that serves demand on route r (-)

3.1.1 Decision variables

r = route on the network (-);
r_k = segment r_k that serves demand on route r (-);
B = Total fleet size (-);

3.1.2 Parameters

Z_1 = user cost objective function (-);
β_{time} = monetary unit value for user travel time ('000);
$t_{trv}^{r_k}$ = travel time on route segment r_k (hr);
$q_{trv}^{r_k}$ = travel demand on route segment r_k (pax);
$t_a^{r_k}$ = access time on route segment r_k (hr);
$q_a^{r_k}$ = passengers boarding on route segment r_k (pax);
$t_w^{r_k}$ = waiting time on route segment r_k (hr);
$q_w^{r_k}$ = passengers waiting on route segment r_k (pax);
ϕ_{time} = time penalty associated with transfers (-);
n_{tr} = transfers on a route r (-);
β_{unsat} = monetary unit value for unsatisfied travel ('000);
$t_u^{r_k}$ = time penalty for unsatisfied travel r_k (hr);
$q_u^{r_k}$ = volume of unsatisfied travel demand r_k (pax);
Z_2 = operator cost objective function ('000);
β_{dist} = monetary unit value for vehicle mileage ('000);
$d_r^{r_k}$ = length of route segment r_k (km);
$n_b^{r_k}$ = bus operating on a route segment (-);
β_{op} = monetary unit value for vehicle operating time ('000);
n = index of the agent (-);
$P_n(k)$ = agent-based probabilistic route choice model (-);
E = mean traffic conditions on the network (-);
$\tau(x)$ = network costs as a result of x (-);
x = network conditions (-);
$\{r_k^n\}$ = all individual agent route demands on the network (-);

d_{min} = minimum route length (km);
d_{max} = maximum route length (km);
r_{tot} = number of designed routes (-);
R_{max} = maximum number of routes that are allowed on the network (-);

The objectives are subject to an agent-based SUE (Horni et al. 2016) which describes the individual traveler's behavior on a public transportation network. Equation (4) is a probabilistic choice model that is used as a proxy for the agent-based SUE with the assumption that travelers base their route choice on the average route costs on the network. This traffic assignment method is based on decoupling the steady flow of passengers on a network in the static and dynamic contexts to that of the individual traveler. Flötteröd and Rohde (2011) and Zhou and Taylor (2014) show that it is challenging to model traffic flow dynamics in complex networks, but disaggregating the OD matrix into individual trip makers allows for vehicle assignment to each trip maker. On this premise, the user equilibrium (UE) and SUE can then be extended to a so-called *disaggregate* or *particle* case, where the particle represents the *microscopic* or single traveler with their route choices replaced with random variables. Hence, each traveler can draw routes from this choice distribution and the resulting distribution of traffic conditions regenerates the choice distribution. This method when combined with stochastic network loading that uses time-dependent trip departures and an extension of choice dimensions beyond the traditional ones (route and mode choice) used in UE to accommodate destination choice and others, leads to the realization of an agent-based model that describes fully the disaggregate behavior of agents on the network. However, the complexity of the model means that it rather lends itself to simulation rather than an analytical solution. Hence, simulation ensures that each agent can optimize their plans on the network by modifying either their departure time, route, mode and destination choices. These choice dimensions define the variation that occurs in the agent's plans during simulation. The process is repeated till the average score of the population is stabilized or attains equilibrium which is also SUE as the optimization is performed in terms of individual scoring functions and within each traveler's set of plans. This is achieved based on a co-evolutionary algorithm (Meneghini et al. 2016) which optimizes each agent's plan in competition for network resources with other agents, while respecting defined constraints.

In the description of the SBTNDM it is important to highlight that the route and fleet size are used as the problem's decision variables. The latter serves as a proxy for the operator's total budget. In the TNDP literature, a decision variable is a resource that is subject to the transit stakeholders' choice in terms of its allocation (Curtin 2004). The limits or bounds of their availability are usually defined by a feasibility constraint, which is a parameter that defines the limiting conditions of the decision variable(s) in a TNDP. They generally define the feasibility of the optimization problem and ensure that solutions are obtained within reasonable resource limitations. The feasibility constraints for the model are those on vehicle fleet size, number of routes and total route length as seen in Equations (5) through Equation (7). These constraints are used to set the allowed limiting conditions for the allocation of resources on the transit network. Equation (5) is the fleet size constraint that represents the limits of the operator's resources. This ensures that an optimal network does not utilize more vehicles than the available number vehicles. Furthermore, the resources at an operator's disposal determines what service frequency they can provide. Hence, the constraint on fleet size significantly affects the level of service that can be provided with a transit network design solution. Equation (6) defines a constraint on the maximum number of routes in the designed solution. This ensures that the maximum number of routes determined according to the current vehicle fleet size is not exceeded. The maximum number of routes has a big impact on fleet size and driver scheduling.

Lastly, Equation (7) is a feasibility constraint on transit service route length. Usually, public transit operators will not run a service on routes that users can conveniently traverse by walking. Operators also avoid developing excessively long routes (Cipriani et al. 2012), as they make schedule adherence difficult and may require too many transfers, which users find unappealing (Walker 2011).

3.1.3 Modeling assumptions

The following assumptions are made in the model development:

1. At the level of the network, a fixed total travel demand context is assumed.
2. A complete trip or satisfied demand may be in two forms: *boarding–alighting* (B–A) or *boarding–connection–alighting* (B–C–A). The former is a direct trip without transfer, while the latter is a trip satisfied with one transfer required. This specification aligns with how demand coverage is defined in this article: demand that is satisfied with zero or one transfer. It is assumed that commuters generally find a trip less attractive beyond one transfer and that this would lead them to search for alternative, more direct routes or even in some cases to change their mode of travel (Owais 2015).
3. In this work automated fare collection data is used to create the daily trip chains of the commuters which is subsequently converted to the initial demand used in the MATSim simulation.
4. In agent-based travel demand models, demand is generated from people's activities at different locations based on various land uses; however, in this work it was not possible to obtain information concerning activities or activity locations outside the transit network. Consequently, activities refer strictly to transactions like passenger boarding, alighting transfers and others that occur on the network.

4 Solution procedure

Three steps are taken in the solution framework to realize the SBTNDM. The first is a heuristic route network generation algorithm (NGA), which is used to generate initial candidate transit networks. Secondly, an agent-based simulation route network evaluation procedure (NEP) is used to score the quality of each generated transit network. Finally, an NSGA-II network search algorithm (NSA) is used to search for the Pareto-optimal set of network solutions. The reader is referred to Figure 3, in which the interaction between the three components of the model are shown.

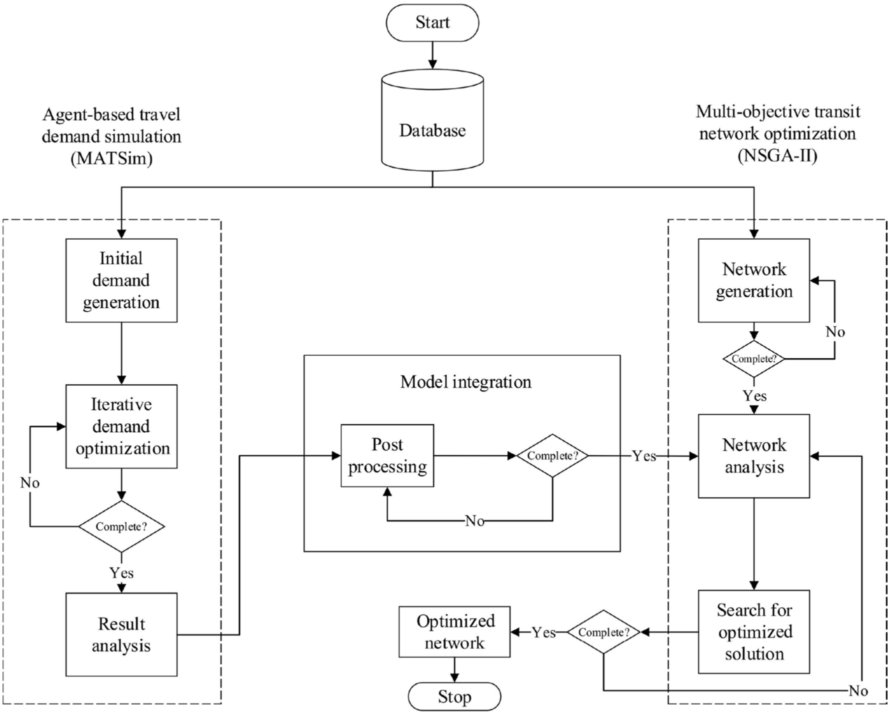

Start
Database
Agent-based travel demand simulation (MATSim)
Multi-objective transit network optimization (NSGA-II)
Initial demand generation
Iterative demand optimization
No
Complete?
Yes
Result analysis
Model integration
Post processing
Complete?
No
Yes
Network generation
No
Complete?
Yes
Network analysis
Search for optimized solution
Complete?
Yes
No
Optimized network
Stop

Fig. 3 Flow diagram of SBTNDM

4.1 Network generation algorithm

The first stage of the SBTNDM involves creating a pool of feasible transit networks from which the first population of solutions will be initialized. An ad-hoc heuristic algorithm was developed for the network generation exercise. Its inputs include: (1) an existing transit network and its constituent routes, (2) the network size parameter (number of routes) and (3) feasibility criteria for route length (r_{len}), route directness (r_{dir}) – minimum deviation from the shortest path between a given origin destination pair and route overlap $r_{overlap}$ considered in this work as the maximum coincidence between the links of a route and the shortest path. These parameters are used to define the feasibility conditions for acceptable routes. The network generation heuristic is developed with the Java programming language (Arnold and Gosling 2000), JGrapht (Michail et al. 2019)—an open-source graph creation and manipulating library—and XML (Bray et al. 2006). The existing transit network data is presented as one of the outputs of a General Transit Feed Specification (GTFS) feed (Wong 2013), which involves extracting and reformatting the transit network and routes from the GTFS data. The network is then converted into a GraphML file (Brandes et al. 2002), a unique XML format for graphs. The conversion makes it possible to read the network as a graph with nodes, links and their attributes, and the graph can be manipulated with the JGrapht tool and graph theory operations. As part of the reformatting activity, the OD stops for existing network routes are extracted and used in the NGA. The steps taken to generate the feasible candidate networks with the NGA are:

1. Read in OD pair data: The algorithm starts by reading in the OD pairs extracted from the existing network routes.
2. Generate multiple paths between each OD pair: Next, the *k*-shortest paths algorithm by Yen (1971) is used to create a user-specified number of paths for all the OD or node pairs so that multiple routes can be enumerated between each OD pair. The *k*-shortest path algorithm typically generates multiple paths in increasing order of magnitude relative to a weighted cost factor. In this work, the path length in kilometres for each route is used as the cost factor. Therefore, if x paths are generated between an OD pair, the first path corresponds to the Dijkstra shortest path (SP) (Johnson 1973), and its length is equal to the beeline distance between the node pairs. The created paths, which will hereafter be referred to as *alternate* paths, are usually longer than the shortest path in increasing order of magnitude.
3. Check route length feasibility conditions for all routes: At this stage, the route length r_{len} feasibility is checked for both the shortest path and alternate paths to verify that a maximum and minimum route length condition is satisfied.
4. Check other feasibility conditions on the alternate paths: After satisfying the route length feasibility, other checks are carried out only on the alternate paths. The first one verifies the directness r_{dir} of the route, checking that an alternate path does not deviate excessively from the geometry of the shortest path.

   (a) Check for route directness: This is important because users consider route deviations unappealing; hence, the deviation should be very small. However, it is sometimes necessary for a route to deviate to adjoining areas where a major transit route does not run to help cover demand in those areas. A factor of 1.2 is used in this work.
   (b) Check for route overlap: The second feasibility condition is for route overlap $r_{overlap}$, checking whether there is a similarity between the links of the shortest path and the alternate path. A minimum value of 0.5 has been used in this work, implying that each satisfactory alternate route must contain at least half of the shortest path's constituent links.
   (c) Check if the route exists already: Lastly, a final check is made to ensure that the alternate path does not currently exist in the list of stored routes.

5. Save the feasible routes for the current OD pair: If all the above-stated conditions are met, the alternate path is saved as a candidate route in a list created for the specific OD pair. This process is then repeated for all the OD pairs, with each OD pair having its own unique list wherein the routes generated for that OD are saved.
6. Perform stratified sampling of routes in all saved lists of routes: After generating the feasible routes, candidate network solutions are created by first using a stratified sampling technique to select routes from each OD and combining them into a network. In stratified sampling, a population is divided into various

sub-populations, and individuals are then selected from each group or strata to make up a random sample. See further details of stratified sampling in Dorofeev and Grant (2006). In this work, drawing from this sampling technique, the list of routes generated for each OD pair is considered a stratum. The sampling is then achieved by randomly choosing the routes from each stratum and combining them into networks to ensure that the order of the existing network OD pairs is retained after sampling. Through this process, a pool of feasible networks is generated. From this pool of feasible networks, the first population is initialized in the NSGA-II. In cases where it is not paramount to retain the order of the routes, the feasible networks generated for all the OD pairs can be placed in a single pool. Other sampling techniques, like *random sampling*, may then be used to generate the required number of networks.

7. Convert the sampled routes to a network transit schedule input file: The final step in the route generation process is to convert the candidate route networks to MATSim transit schedule files, the appropriate input format for the optimization algorithm. However, for the NSGA-II to operate on the solutions, a unique encoding will be defined. Details of this will be revealed when the NSA is discussed.

4.2 Network evaluation procedure

In this step of the model, MATSim is used to evaluate the generated network solutions, and a parallel implementation of a MATSim public transit scenario is set up for this purpose. This is called by the SBTNDM during the evaluation process, and MATSim is called each time a new solution is to be evaluated. Inputs for the NEP include:

1. the initialized population of network alternatives,
2. a synthetic population of agents and their travel demands for a 24-hour activity plan created from the automated fare collection data,
3. an initial schedule of transit operations on the routes of the network, comprising a timetable with its detailed fleet schedule and vehicle departures and
4. a fleet of transit vehicles that will operate the schedules.

The network evaluation step outputs an objective function value or score, which is mapped to each evaluated network. The optimization algorithm then uses the score to rank the performance of each solution in the next step - NSA. Before evaluating a new solution, the subsisting transit schedule data file is overwritten, as it would have been altered during the NSGA-II reproduction. The MATSim simulation process then begins by executing and optimizing the users' initial demand. At the end of the simulation, the resulting events files are analyzed to evaluate the objective functions in Equations (2) and (3), respectively, with parameter values obtained from the events file. A score or objective function value is obtained from the analysis and that score is assigned to the current network solution, which is returned to the optimization module for further processing. The MATSim scenario used in this paper was parallelized. The parallel implementation of the simulation is discussed next.

One way to account for the randomness associated with stochastic processes is to simulate the process multiple times and use the mean result of the different simulation runs. In this research, the simulation experiment ran multiple instances of MATSim in each evaluation of the candidate network solutions. To satisfactorily describe the stochastic behavior of passengers on the transit network, multiple runs of the simulation are required in each iteration of the optimization process. MATSim has multi-threading capabilities, which means that it can run in parallel when extensive simulations or a high number of iterations are required. The parallelization is achieved by setting MATSim's `numberOfThreads` feature in the `global` module within the configuration file. Internally, each *simulation* or *run* is comprised of a user-specified number of MATSim *iterations*. Therefore, the number of iterations required to achieve equilibrium in every run of the simulation, was experimentally determined to be 80 iterations. Figure 4 shows the number of iterations for this model.

It should be noted that the iterations operate sequentially and not in parallel, because in the simulation each new iteration uses the results of the previous one as input. In essence, succeeding iterations *learn* from preceding ones until an equilibrium is achieved in the simulation. However, as multiple *runs* are required in this case, they can be set up in parallel. Each parallel MATSim simulation is executed in its own Java virtual machine (JVM) (Arnold and Gosling 2000), because each simulation needs to use a unique pseudo-random number generator (PRNG) (Rahimov et al. 2011; Matsumoto and Nishimura 1998). In the end, the various results are averaged and used. The collection of multiple *runs* is referred to as an *ensemble of runs*, and counts as one *evaluation* of the candidate networks. The MATSim *ensemble* can be seen in Figure 5.

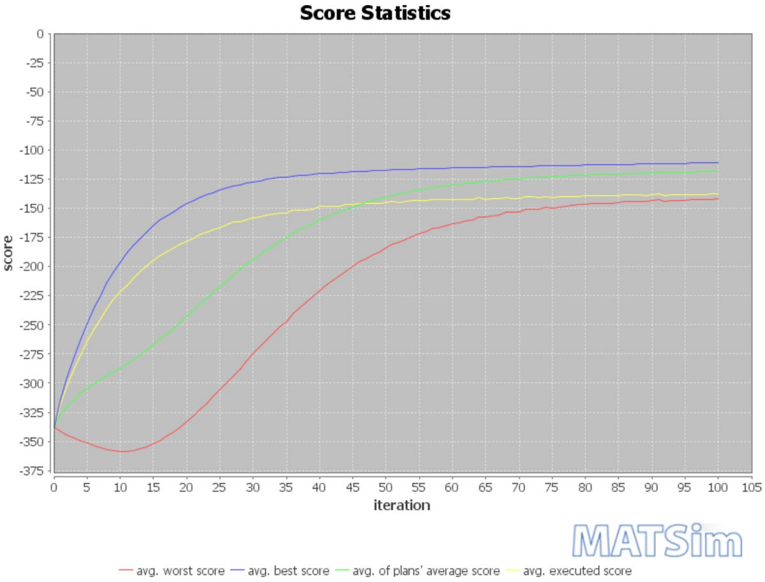


Fig. 4 Number of MATSim iterations for the SBTNDM; convergence occurs after 80 iterations

In this work, 30 runs of the simulation are used in each ensemble. This value is obtained by experimentation. This means that to evaluate each network, 30 parallel runs of MATSim are executed.

4.3 Network search algorithm

The final stage of the model describes how the network optimization progresses and how a Pareto set of transit network solutions is realized. The main inputs used here are the feasible candidate solutions from the NGA and the objective function scores from the NEP, implying that at different stages of its operation, the NSA will call the NGA and NEP sub-routines. The generated network routes are converted to MATSim transit schedule files, which contain both the transit routes and their schedules. The format of these files is Extensible Markup Language (XML) (Bray et al. 2006).

An important step in evolutionary algorithms is to encode the phenotype of each solution. To this end, the transit schedule file which is initially in XML format, is converted to a JSON data structure, facilitating the efficient manipulation of the transit schedules with the genetic operators (selection, crossover and mutation) during the reproduction process. However, this encoding scheme makes it necessary to customize the NSGA-II operators to enable them to manipulate the JSON format. As stated previously, the major advantage of this approach is that it accommodates the encoding of each network with a detailed operational schedule, thereby facilitating the simultaneous handling of the route network design and frequency setting sub-problems of the TNDP. The optimization process then starts with initializing the pool of feasible solutions in the NSGA-II. The initial population is evaluated with the NEP, thereafter, the NSGA-II's crowding comparison operator is used to rank the solutions, based on the objective function scores obtained from the evaluation step. Subsequently, pairs of the best performing solutions are selected from the population and encoded as JSON files to serve as parents in the reproduction

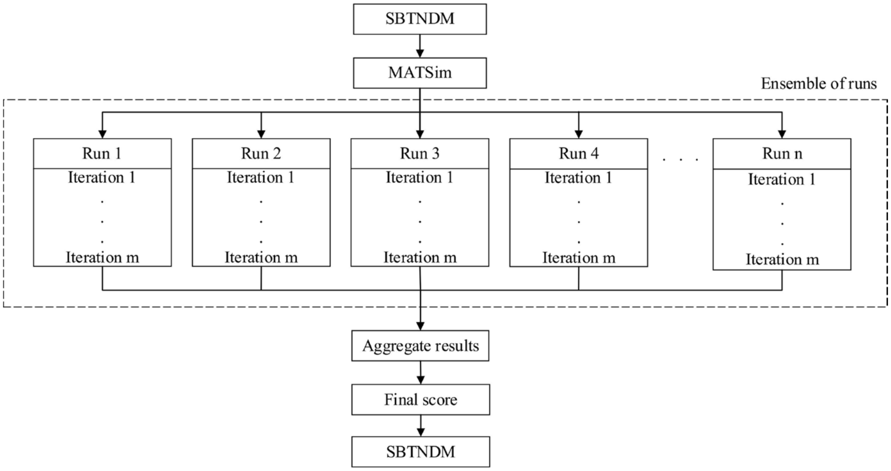
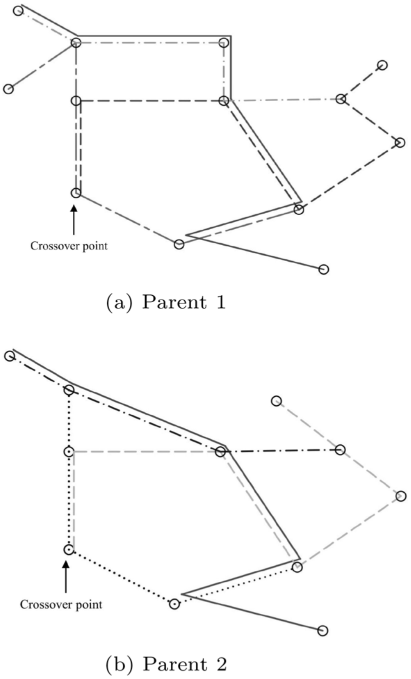


Fig. 5 The parallel implementation of MATSim used in the SBTNDM

process. The *single point crossover* and *polynomial mutation* operators are then used to perform the actual reproduction of offspring. The crossover and mutation operators adapt the genetic programming (GP) strategy which solves the problem of fixed length solution encoding commonly used in genetic algorithms by defining non-linear structures with different sizes and shapes. This is applicable to the XML and JSON decision variables, since the latter are tree-like structures. The method allows the direct manipulation of the encoded network variable with a crossover or mutation point corresponding to a node on the network while the genetic material or routes are swapped between nodes as demonstrated in Figures 6a, 7, 8a.

The flexibility of this approach entails that further customization, such as multiple point operations is possible. The crossover and mutation operators are controlled by probabilities set to 0.75 and 0.25, respectively. For each offspring created, a check is done of its topology to ensure it is logical. MATSim allows for the check to be done using its network cleaner function. After creating a new population of offspring solutions, the offspring are merged with the parent population. The process continues iteratively with the better-performing solutions selected in each generation, thereby guaranteeing continuous improvement of the solutions until the termination criterion (number of generations) is reached. The latter is set to ensure that the algorithm stops once the criterion is satisfied. Lastly, the set of solutions obtained in the final generation are decoded by converting them from the JSON format back to the MATSim network and schedule files for further analysis.

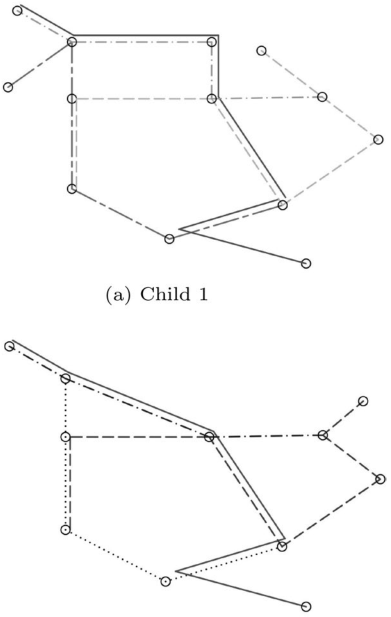


(a) Parent 1

(b) Parent 2

Fig. 6 Parent networks chosen from the pool of feasible network solutions

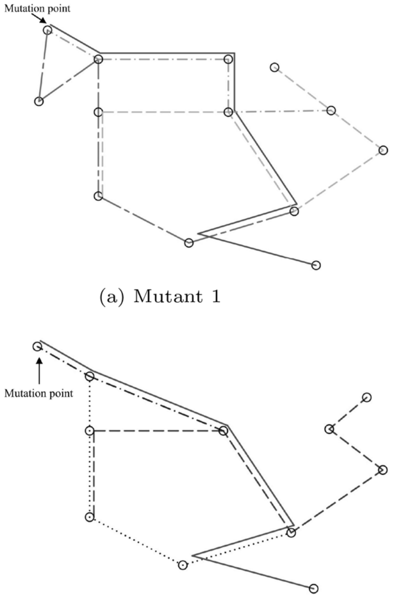
Fig. 7 Offspring networks after crossover



(a) Child 1

(b) Child 2

Fig. 8 Offspring networks after mutation



(a) Mutant 1

(b) Mutant 2

5 Results and discussion

In this section two tests are performed on the SBTNDM. The first tests are that of computational time and the performance of the algorithm in terms of its result quality. Two indicators – hypervolume and generational distance are used for the latter test. The numerical tests, on the other hand, involve the application of the model to a real network case study in the city of Cape Town, South Africa. These tests will demonstrate the robustness of the model and its practical application in the improvement of large-scale networks. The tests are conducted with the national Centre for High Performance Computing's Lengau cluster, which is a Dell Linux HPC cluster with a total of 1368 nodes and 32832 cores and allowing access to 240 nodes at a time. On this resource, one experiment took approximately 15 minutes to run. To show how the SBTNDM scales in terms of the network size, the number of routes in the network is varied from 10 to 50. The plot for computational time of the model can be seen in Figure 9. Computation time is observed to increase as the number of routes or network size increases.

5.1 Algorithm performance

In the computational tests, attributes of the model's solutions such as their spread and convergence are measured. The tests also give parameter values that can be used when the model is applied to design scenarios. The result of an MOEA is a near-optimal solution set, which is also called an *approximation set* and considered as an approximation of the often unknown Pareto front which is also called the *reference set*. The quality of MOEA solutions is therefore measured based on the proximity of the approximation set to the *reference set* in the search space (Coello et al. 2007), if the latter is known or available. When the reference set is not known as is the case with many TNDPs, it is possible to measure the quality of an MOEA's solutions by checking factors like the convergence or the spread of solutions across the obtained

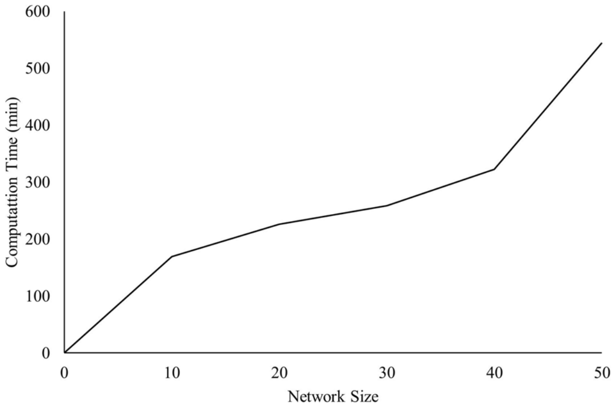


Fig. 9 Plot of the computational time vs. network size

Pareto front. Two performance indicators, namely hypervolume and generational distance, are used in this paper to measure the quality of the obtained solutions.

5.1.1 Hypervolume

The Hypervolume indicator measures the volume of a problem's search space that is dominated by the approximation set (Bringmann and Friedrich 2013). The indicator is calculated relative to a reference point known as the *nadir* point which is usually the worst-case objective value for each objective function (Hadka 2017). Some advantages of the hypervolume indicator are that:

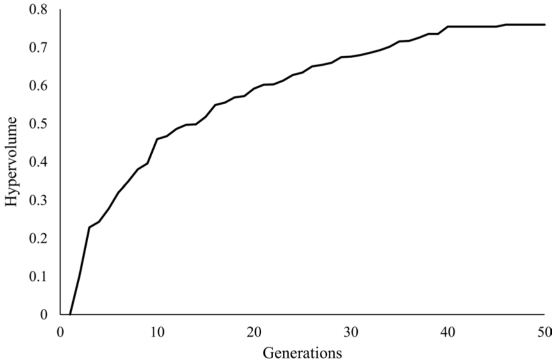
1. it is easily adapted to problems with many objectives,
2. it is a measure of both convergence and diversity in an MOEA and
3. it does not require prior knowledge of the Pareto front to guide the search for a solution that approximates the former.

The main limitation of this indicator is that it is computationally expensive. In terms of its behavior, a higher hypervolume value indicates a better solution or approximation set, because it dominates a greater portion of the search space. Figure 10 shows a plot of the indicator after 50 generations of the SBTNDM. The figure shows that the value of the indicator steadily increases as the algorithm's generations increase, implying that the SBTNDM's solutions improve in successive generations, which matches the known behavior of the hypervolume indicator. The results also show that the indicator converges close to 50 generations; hence, the number of generations required to get near-optimal network solutions with the SBTNDM is 50.

5.1.2 Generational distance

The generational distance (GD) indicator is used to measure the convergence of the solution set obtained from an MOEA (Liu et al. 2019). It is obtained by measuring the average distance between each solution in the approximation set and the nearest one in an MOP's reference set. Smaller values of the indicator are considered better.

Fig. 10 Plot of the hypervolume indicator



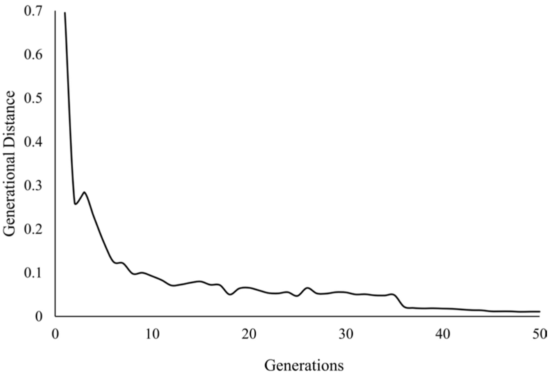


Fig. 11 Plot of the generational distance indicator

When the approximated set is a subset of the reference set, the GD is equal to zero. A plot of GD against the number of generations is shown in Figure 11.

If the approximation set contains a single solution that is too close to the reference set relative to other solutions in the set, the GD measurements may be unrealistically low, and for this reason GD is often combined with other quality or performance indicators. The results show a convergence of indicator values after 45 generations, and the behavior of the indicator observed in the figure is in line with its expected behavior.

5.2 Numerical results

The results discussed here were obtained by applying the SBTNDM to the design of the integrated Public transit network (IPTN) in Cape Town, South Africa. The IPTN which is a public transit network planned in anticipation of the future effect of urban growth on travel demand in Cape Town. It is a long term plan which is expected to be implemented in phases and intended to be fully functional by 2032. The plan involves a significant expansion of the city's existing public transportation network. This is logical as the population of the city is expected to grow by approximately 37% by the target year. The current network comprises a bus network known as the Golden Arrow Bus Service (GABS), a bus rapid transit (BRT) network and a rail network. When completed, it is expected that BRT and rail would form the backbone of the IPTN. This work focuses on the improvement of the BRT service. The network consists of 472 nodes and about 46 operational routes. The service is intended as the backbone for a planned larger and fully Integrated Rapid Transit Network (IRTN) in Cape Town, which comprised of other land-based public transport modes like a bus and rail service that currently operate with low efficiency. The BRT system offers a restricted tap-in and tap-out access to passengers at the terminals and with dedicated bus lines in high congestion areas like the central business district. However, it shares network links with other road-based public transport modes like GABS in other areas within the network. Some inefficiencies have been identified, as the service experiences low ridership on some routes and there is also

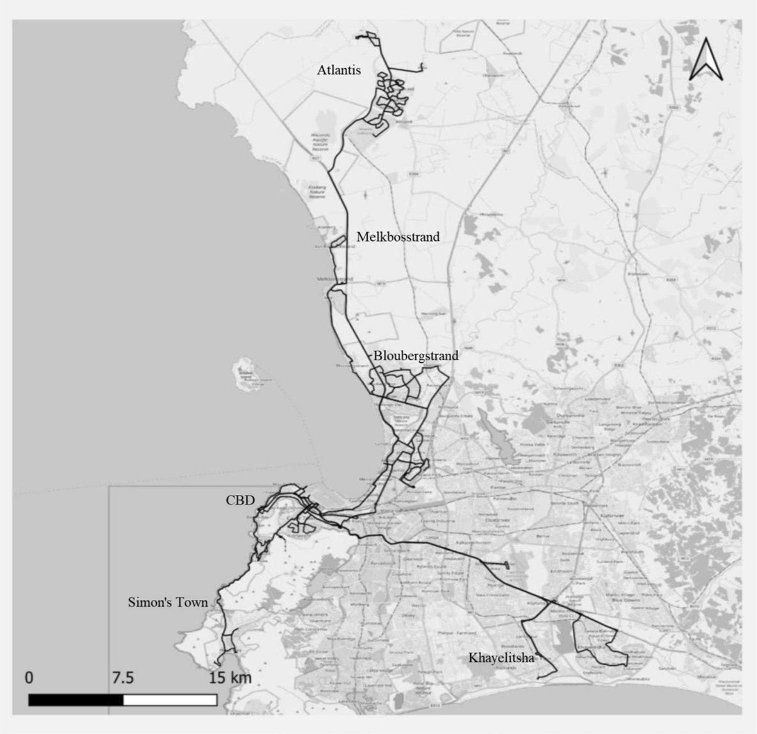


Fig. 12 The MyCiTi BRT network

a need to reduce the total operational costs of the system. An image of the network is given in Figure 12.

The application of the SBTNDM to this network optimization problem will, therefore, focus on the respective objectives of reducing user costs to attract more commuters and reducing the service operator's total costs. Two main data sources are utilized, being the automated fare collection data from which the agent population and their attributes are obtained and the GTFS data for the service from which the network, schedules and transit vehicles are obtained. One of the disadvantages of evolutionary algorithms is the randomness and uncertainty in the final solutions, meaning, that the produced solutions may vary with each run of the algorithm. This further means that though the solution scores might be the same, the detailed route structures and timetable schedules within the solutions might be different and, in actual operations, such small details can matter. From the perspective of transit operators who use the proposed optimization algorithm, it would be difficult to rely on the sets of solutions, since, they vary with each run. To prevent this, the results discussed here are from seeded runs of the model. In computational science, random seeds are used to generate a series of pseudo-random numbers that can replicate the state of an experiment or simulation (Gosavi 2015b). It implies that if all input parameters are kept constant, a simulation's results would be the same if set to run with a given seed, and different if the random seed changes. The resulting Pareto set

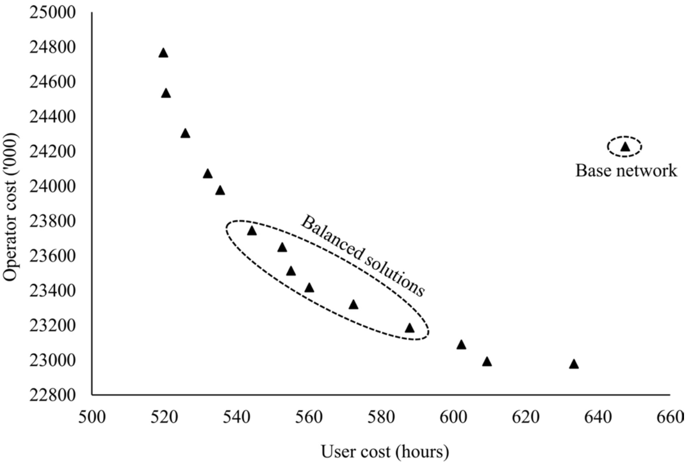


Fig. 13 Pareto front plotted with the base case network

of solutions obtained by the SBTNDM is further evaluated, and the evaluation of each network solution is done with MATSim. The Pareto front can be seen in Figure 13. The plot shows the solutions plotted with the base case of the MyCiTi BRT network.

A visual observation of the plot reveals that among the solutions in the Pareto front, the network with the highest travel time also has the lowest operator cost and vice versa, which is indicative of the *trade-off* between both the users' and operators' perspectives. Users prefer direct trips that reduce their travel time, while operators prefer longer and slightly more circuitous routes that increase the volume of demand they can *potentially* satisfy while reducing their average service costs. *Balanced* network solutions occur within the marked cluster in the middle of the plot in Figure 13. In the context of this work, the networks are considered to be balanced, as they exhibit the least conflict between the previously mentioned objectives, i.e, they are the best compromise solutions between the stated objectives. These solutions are, therefore, regarded as the best *trade-off* network solutions to the problem. The above discussion, however, contrasts with the current situation of the MyCiTi BRT network, indicated as base network in Figure 13, which shows both higher user and higher operator objective cost values compared to the models' solutions. This is indicative of the earlier mentioned operational issues of low ridership and high operational costs on the network. In Table 1 the objective function scores for the Pareto solution set and the base case are presented. The base network is clearly inferior to the Pareto set of solutions as it performs worse than all solutions in the user cost objective.

Among the solutions obtained from the SBTNDM, network 1 has the lowest *user cost* or *objective 1* score and will be referred to as the *user-centric* solution. This network also has the highest *operator cost* or *objective 2* score. By contrast, network 14 has the lowest operator cost and will be called the *operator-centric* solution. As indicated earlier, the best compromise between the objectives occurs between solutions

Table 1 Raw objective function values

Network	User cost (hours)	Operator cost ('000)
1	519.79	24,767.52
2	520.51	24,536.12
3	525.85	24,304.72
4	532.02	24,073.32
5	535.50	23,976.82
6	544.26	23,745.42
7	552.67	23,648.92
8	555.06	23,514.02
9	560.16	23,417.52
10	572.36	23,321.01
11	587.91	23,186.12
12	602.20	23,089.61
13	609.32	22,993.11
14	633.37	22,979.07
Base network	647.55	24,228.14

6 through 11. However, network solution 8 shows the least difference in the values of both objectives. Hence, the balanced network is considered as one that shows the least conflict or the best compromise between the commuter's and operator's perspectives. To depict this clearly, the above objective scores are normalized by rescaling them to a scale of 1 to 10 using the standard formula below.

$$z = y_{min} + (x_i - x_{min}) \cdot (y_{max} - y_{min})/(x_{max} - x_{min}) \tag{8}$$

where:

x_{min} = minimum objective function value
x_{max} = maximum objective function value
y_{min} = normalized scale minimum value
y_{max} = normalized scale maximum value
x_i = objective function value to normalize
z = expected normalized objective function value

Subsequently, the normalized scores are ordered and plotted against one another. Table 2 shows a plot of the normalized objective function scores.

This facilitates the results being plotted on a similar scale, and the normalized scores are ordered and plotted against one another. Figure 14 shows a plot of the normalized objective function scores for the Pareto solutions.

Having identified these three network solutions – 1 (user-centric), 8 (balanced) and 14 (operator-centric) – as proxies for the perspectives mentioned above, they are then isolated for further analysis. The analysis is carried out to measure their performance in terms of different network performance indicators. The indicators used include total satisfied travel demand, total operational cost, network utilization percentages, unsatisfied demand, vehicle mileage and vehicle hours. The results of the analysis are presented in Table 3.

Table 2 Normalized objective function values

Network	User cost	Operator cost
1	1.00	10.00
2	1.06	8.84
3	1.48	7.67
4	1.97	6.51
5	2.24	6.02
6	2.94	4.86
7	3.61	4.37
8	3.79	3.69
9	4.20	3.21
10	5.17	2.72
11	6.40	2.04
12	7.53	1.56
13	7.59	1.97
14	8.09	1.07
15	10.00	1.00

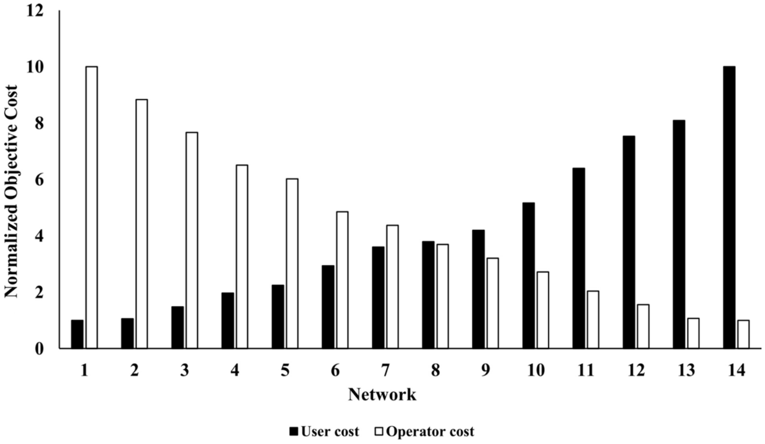


Fig. 14 Network solutions on the Pareto front showing different compromise solutions

Table 3 Aggregate transit network performance indicators for the identified scenarios

Indicators	Base network	Solution 1 (User)	Solution 8 (Balanced)	Solution 14 (Operator)
Satis. demand (pax)	24,928	34,216	31,694	29,590
utilization (%)	64.63	88.71	82.17	76.72
Veh. dist (km)	52,619.35	48,567.20	45,215.15	42,452.99
Veh. time (hr)	2,057.47	1,618.91	1,507.17	1,348.43
Op. cost ('000)	24228.14	24767.52	23514.02	22979.07

In the table, the existing base network satisfies the smallest amount of demand, though its operational cost is less than that of solution 1 but more than the balanced and operator solutions. It should be noted that the increased effectiveness in terms of travel demand satisfaction, which is visible in the other networks when compared to the base network is attributable to latent demand that arises as a result of the network improvements achieved from optimizing the BRT system. On the other hand, solution 1 has the highest satisfied demand and network utilization, as well as the highest operational cost. This is similar to an optimization scenario in which the users' objectives are prioritized and more passenger demand on direct routes is served. Therefore, circuitous routes and those running through transfer points will be minimal or excluded where necessary. This also means that, on average, passengers will travel shorter distances to their destination, which will encourage more people to use the service. However, the increase in ridership leads to an attendant increment in the operational frequencies, because operators would like to maintain the attractiveness of their service and encourage continued patronage from commuters by sustaining a good level of service in terms of travel time. Typically, increased operational frequency is a major cost driver for operators, as they must deploy more resources (personnel and fleet) on the network. In contrast, network solution 14 shows an opposite trend to that of solution 1, as it shows the lowest operator cost and lowest total network utilization and is similar to a case in which the operator's objective is prioritized. The results show that the operator has less vehicle mileage and operational hours than the user-centric solution, but it also satisfies less demand. This may be because trying to maximize network coverage by using circuitous routes may ultimately discourage passengers who prefer the direct routes. A network that is skewed in favor of the operator will primarily contain routes that are longer than those preferred by users. Lastly, an optimal transit network solution would contain a mix of direct routes and other, more circuitous ones. Hence, the solutions earlier referred to as the best compromise solutions, respectively, represent a balance between the user and operator perspectives. As direct routes reduce operators' ability to cover demand along more circuitous paths, an optimized solution must compromise between the needs of commuters and service operators, which is reflected in the middle column of Table 3 for solution 8, where the indicators have values between the user and operator perspectives. The results show that the solution does indeed offer the best compromise because it minimizes costs for all stakeholders. The outcomes discussed above are reinforced in Table 4, where the balanced network solution has indicator values that show a compromise between the users' and operator's perspectives.

Table 4 Average performance indicators at route level for the identified scenarios

Indicators	Base network	Solution 1 (User)	Solution 8 (Balanced)	Solution 14 (Operator)
Route density (pax/route)	541.92	743.83	689.00	643.26
Avg. op. cost ('000)	526.70	538.42	511.17	499.55
Avg. veh. time (hr/route)	44.73	35.19	32.76	29.31
Avg. veh. dist (km/route)	1,143.90	1,055.81	982.94	922.89

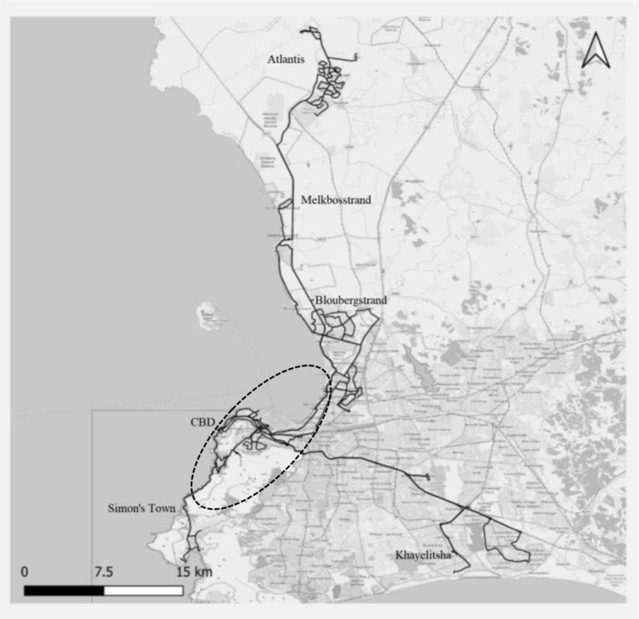
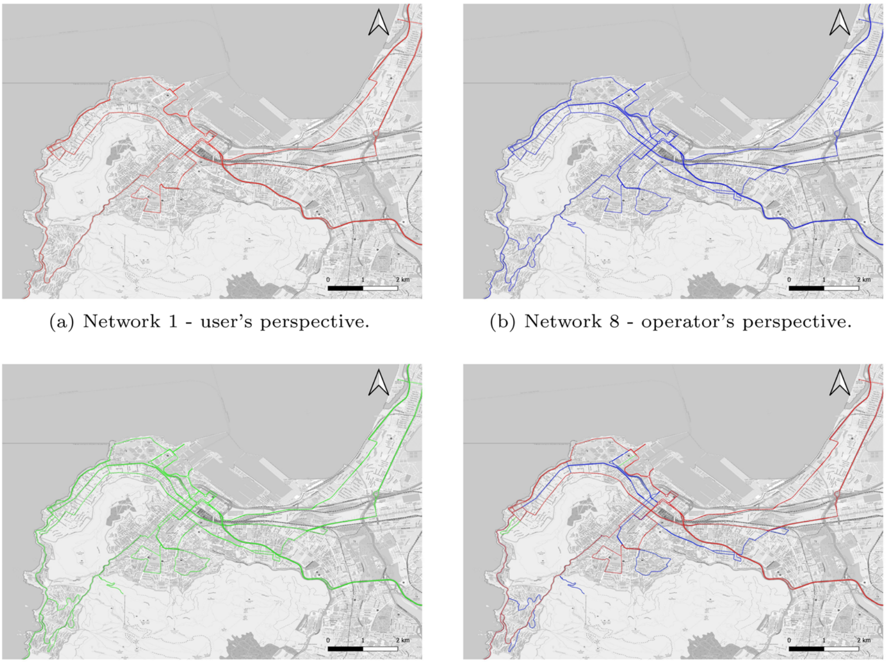


Fig. 15 Network map showing the local area around the CBD



(a) Network 1 - user's perspective.

(b) Network 8 - operator's perspective.

(c) Network 14 - balanced perspective

(d) Overlay of all three networks.

Fig. 16 Network design results

To see the details of the earlier discussed network results at a sizeable scale, a local area of the network results, covering the Central Business District in Cape Town and its surroundings are highlighted in the map (see Figure 15). The details are then shown in Figure 16a through Figure 16c, respectively, while an overlay of all three networks is also visible in Figure 16d. Lastly, these results show that significant design improvements have been achieved with the SBTNDM during the optimization process. Furthermore, the balanced solution is the most attractive for all stakeholders, and it also offers better access to public transit services. Ultimately, depending on what a policymaker wishes to achieve, they can easily apply decision support tools such as a multi-criteria decision analysis to the obtained results to arrive at other trade-off solutions from the set of non-dominated solutions to match their priority. These results show that SBO can indeed yield reasonable network solutions when used in the TNDP process and that ABS can play an essential role in the process. In conclusion, the potential of the SBTNDM to use big data and other technological advances currently unfolding in the transit sector make it viable for modeling large-scale and complex transport scenarios.

6 Limitations

One limitation of the model presented in this work is the fact that an ABM is data-intensive and computationally expensive in terms of the resources required to simulate the model, due to the microscopic level of data needed to build the models. However, future improvements in the computational performances and speeds will most likely address this limitation. Another limitation of this work is that passenger activities are derived strictly from automated fare collection data on the system. While being sufficient for the work, it is useful to expand the activities beyond network-related activities.

7 Conclusion

Simulation-based optimization offers a new approach to tackle the TNDP. It solves the problem through the combination of optimization and simulation models. Though simulations are computationally resource-hungry, given the number of evaluations required to solve a problem, they have the distinct advantage that the stochastic behavior of agents and other random and realistic occurrences on the network can now be simulated within the network design solution model. Furthermore, advances in computational sciences entails that increasingly large scenarios can now be simulated in shorter time frames. In this work the authors present the so-called SBTNDM which combines activity-based travel simulation known as MATSim with the NSGA-II. Results of applying the model to the MyCiTi BRT in Cape Town, South Africa reveals that the designed network solutions perform better than the network in terms of travel time reduction for users and operational cost reduction for operators of the service. Practically, the SBTNDM improves public transport networks in line with the objectives of the network user and operator. As a decision support tool, the model will be useful in guiding policymakers in Cape Town in making policy decisions that are relevant to the transportation context and realities of today. Overall, the use of SBO in solving the TNDP makes it possible to broaden the potential design objectives, variables and performance measures that can be used in the network design and optimization process. This enables the implementation of network optimization studies in increasingly complex scenarios, such as those that are sensitive to time of day, pricing elasticities and/or that require detailed traveler behavior. In terms of research directions that might extend from this work in the future, the primary consideration should be to extend the application of the SBTNDM to a multi-modal network context to study modal integration. There is also potential to study the transit network frequency setting problem with SBTNDM.

Acknowledgements The authors acknowledge the Centre for High Performance Computing (CHPC), South Africa, and the University of Cape Town's ICTS High Performance Computing team for providing computational resources for this research project.

References

Alrabghi A, Tiwari A (2015) State of the art in simulation-based optimisation for maintenance systems. Comput Ind Eng 82:167–182. https://doi.org/10.1016/j.cie.2014.12.022

Arnold K, Gosling J (2000) The Java programming language. Addison-Wesley

Bal A, Badurdeen F (2022) A simulation-based optimization approach for network design: the circular economy perspective. Sustain Prod Consum 30:761–775. https://doi.org/10.1016/j.spc.2021.12.033

Brandes U, Eiglsperger M, Herman I, Himsolt M, Marshall MS (2002) GraphML progress report structural layer proposal. In: Mutzel P, Jünger M, Leipert S (eds) Graph Drawing Lecture notes in computer science, vol 2265. Springer, Berlin, pp 501–512

Brands T, van Berkum E (2014) Performance of a genetic algorithm for solving the multi-objective, multimodal transportation network design problem.. Int J Transp 2(1):1–20. https://doi.org/10.14257/ijt.2014.2.1.01

Branke J, Deb K, Miettinen, Slowinski R (eds) (2008) Multiobjective optimization - interactive and evolutionary approaches. Lecture notes in computer science, vol 5252, Springer, Berlin

Bray T, Jean P, Sperberg-McQueen CM, et al (2006) Extensible Markup Language (XML) 1.1 (Second Edition) W3C Recommendation 16 Aug 2006. Tech. Rep. August

Bringmann K, Friedrich T (2013) Approximation quality of the hypervolume indicator. Artif Intell 195:265–290. https://doi.org/10.1016/j.artint.2012.09.005

Buba AT, Lee LS (2018) A differential evolution for simultaneous transit network design and frequency setting problem. Exp Sys Appl 106:277–289. https://doi.org/10.1016/j.eswa.2018.04.011

Chakroborty P (2003) Genetic algorithms for optimal urban transit network design. Comp-aided Civil Eng 18(3):184–200. https://doi.org/10.1111/1467-8667.00309

Chen J, Wang S, Liu Z, Wang W (2017) Design of suburban bus route for airport access. Transportmetrica A Transp Sci 13(6):568–589. https://doi.org/10.1080/23249935.2017.1306896

Cipriani E, Gori S, Petrelli M (2012) Transit network design: a procedure and an application to a large urban area. Transp Res Part C: Emerg Technol 20(1):3–14

Cipriani E, Fusco G, Patella SM, Petrelli M (2020) A particle swarm optimization algorithm for the solution of the transit network design problem. Smart Cities 3(2):541–555. https://doi.org/10.3390/smartcities3020029

Coello CAC, Lamont GB, Van Veldhuizen DA (2007) Evolutionary algorithms for solving multi-objective problems, 2nd edn. Springer, New York

Constantin I, Florian M (1995) Optimizing frequencies in a transit network: a nonlinear bi-level programming approach. Int Trans Oper Res 2(2):149–164

Crockford D (2011) Introducing JSON. https://www.json.org/json-en.html

Curtin KM (2004) Operations research. In: Kempf-Leonard K (ed) Encyclopedia of social measurement. Elsevier, Amsterdam, pp 925–931

Daganzo CF, Ouyang Y (2019) A general model of demand-responsive transportation services: from taxi to ridesharing to dial-a-ride. Transp Res Part B: Methodol 126:213–224. https://doi.org/10.1016/j.trb.2019.06.001

Dandl F, Engelhardt R, Hyland M, Tilg G, Bogenberger K, Mahmassani HS (2021) Regulating mobility-on-demand services: tri-level model and Bayesian optimization solution approach. Transp Res Part C: Emerg Technol 125:103075. https://doi.org/10.1016/j.trc.2021.103075

Deb K, Agrawal S, Pratap A, Meyarivan T (2000) A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer M, Deb K, Rudolph G et al (eds) Parallel Problem Solving from Nature PPSN VI. Lecture notes in computer science, vol 1917. Springer, Berlin, pp 849–858

Dorofeev S, Grant P (2006) Statistics for real life sample surveys. Cambridge University Press, Cambridge, UK

Durán-Micco J, Vansteenwegen P (2022) A survey on the transit network design and frequency setting problem. Public Transp 14:155–190. https://doi.org/10.1007/s12469-021-00284-y

Elarbi M, Bechikh S, Ben Said L, Datta R (2017) Multi-objective optimization: classical and evolutionary approaches. In: Bechikh S, Datta R, Gupta A (eds) Recent advances in evolutionary multi-objective optimization. Springer, Cham, pp 1–30

Fan W, Machemehl RB (2004) Optimal transit route network design problem: algorithms. Center for Transportation Research University of Texas at Austin, Texas

Flötteröd G, Rohde J (2011) Operational macroscopic modeling of complex urban road intersections. Transp Res Part B: Methodol 45(6):903–922. https://doi.org/10.1016/j.trb.2011.04.001

Gao G, Lu W, Zhou C, Huang Z (2022) Simulation-based passenger-carrying capacity analysis for urban rail transit transfer stations. In: Zeng Q (eds) Sixth International Conference on Electromechanical Control Technology and Transportation (ICECTT 2021), International Society for Optics and Photonics, vol 12081. SPIE, pp 842–852, https://doi.org/10.1117/12.2623980

Gosavi A (2015) Background. In: Simulation based optimization: parametric optimization and reinforcement learning, 2nd edn. Springer, New York, pp 1–12

Gosavi A (2015) Simulation basics. In: Simulation based optimization: parametric optimization and reinforcement learning, 2nd edn. Springer, New York, pp 13–27

Hadka D (2017) Beginner's guide to the MOEA framework, 2nd edn. CreateSpace Independent Publishing Platform, Raleigh-Durham, http://www.lulu.com/shop/http://www.lulu.com/shop/david-hadka/beginners-guide-to-the-moea-framework/ebook/product-23015046.html

Hassannayebi E, Sajedinejad A, Kardannia A, Shakibayifar M, Jafari H, Mansouri E (2021) Simulation-optimization framework for train rescheduling in rapid rail transit. Transportmetrica B Transp Dyn 9(1):343–375. https://doi.org/10.1080/21680566.2020.1854896

Heyken Soares P, Mumford CL, Amponsah K, Mao Y (2019) An adaptive scaled network for public transport route optimisation. Public Transp 11(2):379–412. https://doi.org/10.1007/s12469-019-00208-x

Horni A, Nagel K, Axhausen K (2016) Agent-based traffic assignement. In: Horni A, Nagel K, Axhausen KW (eds) The multi-agent transport simulation MATSim. Ubiquity Press, London, pp 315–326

Huang D, Liu Z, Fu X, Blythe PT (2018) Multimodal transit network design in a hub-and-spoke network framework. Transportmetrica A: Transp Sci 14(8):706–735. https://doi.org/10.1080/23249935.2018.1428234

Ibarra-Rojas OJ, Delgado F, Giesen, Munoz JC (2015) Planning, operation, and control of bus transport systems: a literature review. Transp Res Part B: Methodol 77(3):38–75. https://doi.org/10.1016/j.trb.2015.03.002

Iliopoulou C, Kepaptsoglou K, Vlahogianni E (2019) Metaheuristics for the transit route network design problem: a review and comparative analysis. Public Transp 11:487–521. https://doi.org/10.1007/s12469-019-00211-2

Johar A, Jain SS, Garg PK (2016) Transit network design and scheduling using genetic algorithm - a review. Int J Optim Control Theor Appl 6(1):9–22

Johnson DB (1973) A note on Dijkstra's shortest path algorithm. J Assoc Comput Machin 20(3):385–388

Knowles J, Corne D, Kalyanmoy D (eds) (2008) Multiobjective problem solving from nature. Springer, Berlin, Heidelberg

Lee Y-J, Vuchic VR (2005) Transit network design with variable demand. J Transp Eng 131(1):1–10. https://doi.org/10.1061/(ASCE)0733-947X(2005)131:1(1)

Liu Y, Wei J, Li X, Li M (2019) Generational distance indicator-based evolutionary algorithm with an improved niching method for many-objective optimization problems. IEEE Access 7:63881–63891. https://doi.org/10.1109/ACCESS.2019.2916634

Ma Z, Chow JY (2022) Transit network frequency setting with multi-agent simulation to capture activity-based mode substitution. Transp Res Record J Transp Res Board 2676(4):41–57. https://doi.org/10.1177/03611981211056909

Matsumoto M, Nishimura T (1998) Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Trans Model Comput Simul 8(1):3–30. https://doi.org/10.1145/272991.272995

Mauttone A, Urquhart ME (2009) A multi-objective metaheuristic approach for the transit network design problem. Public Trans 1(4):253–273. https://doi.org/10.1007/s12469-010-0016-7

Meister K, Balmer M, Ciari F, Horni A, Rieser M, Waraich RA, Axhausen KW (2010) Large-scale agent-based travel demand optimization applied to Switzerland, including mode choice. Paper presented at the 12th World Conference on Transportation Research

Meneghini IR, Guimaraes FG, Gaspar-Cunha A (2016) Competitive coevolutionary algorithm for robust multi-objective optimization: The worst case minimization. 2016 IEEE Congress on Evolutionary Computation. CEC 2016:586–593. https://doi.org/10.1109/CEC.2016.7743846

Michail D, Kinable J, Naveh B, Sichi JV (2019) JGraphT – a Java library for graph data structures and algorithms. ACM Trans Math Softw 46(2):16. https://doi.org/10.1145/3381449

Momenitabar M, Mattson J (2021) A multi-objective meta-heuristic approach to improve the bus transit network: a case study of fargo-moorhead area. Sustainability 13(19):10885. https://doi.org/10.3390/su131910885

Nnene OA, Zuidgeest MHP, Beukes EA (2017) Application of metaheuristic algorithms to the improvement of the MyCiTi BRT network in Cape Town. J South Afr Inst Civil Eng 59(4):56–63

Nnene OA, Joubert JW, Zuidgeest MH (2019) Transit network design with meta-heuristic algorithms and agent based simulation. IFAC-PapersOnLine 52(3):13–18. https://doi.org/10.1016/j.ifacol.2019.06.003

Osorio C (2016) Simulation-based optimization algorithms that enable the efficient use of inefficient traffic simulators for large-scale network optimization. Tech. rep, Massachusetts Institute of Technology, Boston

Osorio C, Selvam KK (2015) Solving large-scale urban transportation problems by combining the use of multiple traffic simulation models. Transp Res Procedia 6:272–284. https://doi.org/10.1016/j.trpro.2015.03.021

Owais M (2015) Issues related to transit network design problem. Int J Comput Appl 120(8):40–45

Pattnaik SB, Mohan S, Tom VM (1998) Urban bus transit route network design using genetic algorithm. J Transp Eng 124(4):368–375

Possel B, Wismans LJ, Van Berkum EC et al (2018) The multi-objective network design problem using minimizing externalities as objectives: comparison of a genetic algorithm and simulated annealing framework. Transportation 45(2):545–572. https://doi.org/10.1007/s11116-016-9738-y

Rahimov H, Babaei M, Farhadi M (2011) Cryptographic PRNG based on combination of LFSR and chaotic logistic map. Appl Math 02(12):1531–1534. https://doi.org/10.4236/am.2011.212217

Rangaiah GP, Bonilla-Petriciolet A (eds) (2013) Multi-objective optimization in chemical engineering. John Wiley and Sons Ltd, Oxford, UK

Ranjbari A, Hickman M, Chiu YC (2020) A network design problem formulation and solution procedure for intercity transit services. Transportmetrica A: Transp Sci 16(3):1156–1175. https://doi.org/10.1080/23249935.2020.1719547

Song M, Yin M, Chen XM, Zhang L, Li M (2013) A Simulation-based approach for sustainable transportation systems evaluation and optimization: theory, systematic framework and applications. Procedia Soc Behav Sci 96:2274–2286. https://doi.org/10.1016/j.sbspro.2013.08.257

Szeto WY, Wu Y (2011) A simultaneous bus route design and frequency setting problem for Tin Shui Wai, Hong Kong. Eur J Op Res 209(2):141–155

Walker J (2011) Human transit. Island Press, Washington, DC

Wong J (2013) Leveraging the general transit feed specification for efficient transit analysis. Transp Res Record J Transp Res Board 2338:11–19. https://doi.org/10.3141/2338-02

Yan Y, Liu Z, Meng Q, Jiang Y (2013) Robust optimization model of bus transit network design with stochastic travel time. J Transp Eng 139(6):625–634. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000536

Yang J, Jiang Y (2020) Application of modified NSGA-II to the transit network design problem. J Adv Transp, Article ID 3753601. https://doi.org/10.1155/2020/3753601

Yen JY (1971) Finding the K shortest loopless paths in a network. Manag Sci 17(11):712–716

Zhou X, Taylor J (2014) DTAlite: a queue-based mesoscopic traffic simulator for fast model evaluation and calibration. Cogent Eng 1(1):1–19. https://doi.org/10.1080/23311916.2014.961345

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Obiora A. Nnene[1] · **Johan W. Joubert**[2] · **Mark H. P. Zuidgeest**[1]

Johan W. Joubert
johan.joubert@up.ac.za

Mark H. P. Zuidgeest
mark.zuidgeest@uct.ac.za

[1] Department of Civil Engineering, Centre for Transport Studies, University of Cape Town, Rondebosch, Cape Town 7700, Western Cape, South Africa

[2] Department of Industrial and Systems Engineering, Centre for Transport Development, Lynwood Road, Pretoria 0002, Gauteng, South Africa