Redesigning Midday Meal Logistics for the Akshaya Patra Foundation: OR at Work in Feeding Hungry School Children

B. Mahadevan, S. Sivakumar, D. Dinesh Kumar, K. Ganeshram

To cite this article:

Full terms and conditions of use: http://pubsonline.informs.org/page/terms-and-conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article’s accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2013, INFORMS

Please scroll down for article—it is on subsequent pages

INFORMS is the largest professional society in the world for professionals in the fields of operations research, management science, and analytics.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org
Redesigning Midday Meal Logistics for the Akshaya Patra Foundation: OR at Work in Feeding Hungry School Children

B. Mahadevan, S. Sivakumar
Indian Institute of Management, Bangalore 560076, India {b.mahadevan@iimb.ernet.in, sivakumar.s@iimb.ernet.in}
D. Dinesh Kumar
Unisys Corporation, Bangalore 560066, India, dduraisamy@gmail.com
K. Ganeshram
Flipkart India Private Limited, Bangalore 560034, India, ganeshram.kandappan@gmail.com

Midday meal programs at schools are prevalent in countries such as India. The Akshaya Patra Foundation, a not-for-profit organization, operates such a program in India for about 1.3 million children in more than 9,000 schools in nine states. The foundation faced a logistics problem in efficiently distributing food within the available time window. This paper discusses the challenges it faced, how we used OR modeling to overcome them, and how we designed, developed, and implemented a software solution. To address the logistics problem, we proposed a three-stage decomposition heuristic solution, which consists of clustering schools, assigning appropriate distribution vehicles to clusters, and routing vehicles within the clusters. We implemented our solution, AMRUTA, on a pilot basis in one location. Based on this pilot, the projected annual cost savings are US$75,000, which would enable the foundation to add 2,400 more children. When the program is fully implemented, we estimate that the annual cost savings will be about US$1.96 million. This project demonstrates how operations research can be useful for solving social sector problems in a developing country such as India.

Key words: distribution logistics; OR/MS implementation; heuristics; case study; software implementation.

History: This paper was refereed.
The TAPF operating model involves setting up a cooking infrastructure in a city that can cater to the demands of a number of rural schools in the surrounding area using delivery vans. The capacity of a kitchen and the size of a delivery fleet are determined based on the estimated demand in a region. Once a facility is established, TAPF determines the routing schedule for each delivery van and dispatches cooked food from the kitchen to the schools as per the schedule. Loading food onto the vans starts at about 8 AM each day, and the vans must complete their delivery schedule before 12:30 PM. The schools break for lunch at 1 PM; therefore, any delay in delivery could result in students going back to their afternoon sessions without having had lunch. Therefore, maintaining a strict cooking-to-consumption time is critical.

TAPF faced certain challenges with respect to its operations because of the increasing complexity of the logistics of distributing the cooked food. As more schools were added to its network, new clusters of schools were created on an incremental basis. This resulted in stretching the existing delivery fleet to cover additional schools, with route extensions drawn on an ad-hoc basis. In addition, the existing distribution model was unable to handle variations in traffic and road conditions. This led to persistent delays in food delivery to schools, forcing children to miss lunch, thereby challenging TAPF’s very objective.

The average cost to cook a meal is 10 cents, with two-thirds of the cost subsidized by the government. However, the daily distribution cost for a cooked meal is in the range of three to four cents depending on the area. This disproportionate distribution cost has a critical bearing on the efficiency of operations, and limits expansion of coverage. TAPF’s senior management felt that it could service the existing demand with fewer vehicles if it could improve its logistics planning; using the capacity released, it could potentially serve additional children. However, TAPF lacked a formal method of logistics planning. Management’s expectation was that a formal logistics planning solution would enable it to (1) keep the cooking-to-consumption time within the desired level, and (2) optimize the distribution cost, given the strict time-window constraint. The Vasanthapura kitchen in South Bangalore illustrates the need for a formal logistics solution. As of 2010, this kitchen delivered food to about 530 schools using a fleet of 35 vehicles. If these schools could be served with one less vehicle without violating the time-window constraints, an opportunity to expand coverage existed.

Thus, the TAPF logistics problem is to find a solution that minimizes the deployed fleet capacity while meeting the time-window requirements. This problem belongs to the general class of vehicle routing problems (VRPs) known as the heterogeneous fixed fleet VRP with time windows (HFFVRPTW). This is a more challenging version of an earlier work (Bartholdi et al. 1983) in which the authors discuss a program that provides meals on wheels; this problem is classified as NP-hard and has no known exact solution algorithms. In this paper, we propose a three-stage decomposition heuristic for solving the industrial-grade version of the problem. We developed and implemented the solution algorithm as a software solution, which we call the Akshaya Patra midday meal routing and transportation algorithm (AMRUTA); in Sanskrit, AMRUTA means the nectar that bestows immortality to the person who consumes it. The three-stage heuristic solution implemented in AMRUTA addresses the clustering of schools, assignment of appropriate vehicles to clusters, and vehicle routing within the clusters.

We did a pilot deployment of AMRUTA at TAPF’s Vasanthapura kitchen. It resulted in significant savings in monthly operating costs because of reductions in both the number of delivery vans and trip length. The annualized cost savings from this implementation (US$75,000—18.61 percent of the monthly operating cost) can enable TAPF to provide lunch to an additional 2,400 children. Once AMRUTA is implemented in all TAPF kitchens across India, we estimate that the cost savings will be about US$1.96 million, enabling TAPF to expand its services nationally to an additional 62,000 children at prevailing costs.

The research contribution in this paper is the modeling and development of a heuristic solution and the adaptations made to solve an industrial-grade logistics problem. The model development, implementation, and results reinforce the power of operations research and management science (OR/MS) to address problems pertaining to substantially improving societal welfare. Furthermore, the solution and the software developed could address similar capacitated
distribution problems with time-window constraints, especially in the food and beverage industries.

**Midday Meal Logistics**

We will use Vasanthapura MDMS program to illustrate the scale of the distribution problem. The Vasanthapura kitchen has a cooking capacity of 50,000 meals a day. Schoolchildren prefer hot, freshly cooked food. To serve this preference and also meet the objective of participation in schools, TAPF developed its operating model around distributing hot freshly cooked food. The cooking process uses mechanized steam-heated cauldrons that are built specifically for mass-producing food on the scale that TAPF requires. Although the cooking preparations start late at night on the previous day, the cooking process typically starts at 4 AM, and the first batch of cooked food is available around 7 AM. The cooked food must be delivered to schools before their scheduled lunch breaks, allowing a total transportation window of four to six hours for the various batches. The food is packed in stainless steel containers for transportation. The Vasanthapura menu consists of three food items—cooked rice, sambar (lentil soup), and curd (yoghurt)—that are loaded in large, medium, and small containers, respectively. The other 18 centralized kitchens follow a similar operating model of cooking and distributing the food, except that the food menu is tailored to the local taste. For example, sambar and steamed rice are replaced by dal (lentil
soup) and chapattis (wheat bread) in the northern states of India.

Vasanthapura has a fleet of 35 delivery vans of varying capacities. These delivery vans have a three-tier rack structure suitable for loading the three sizes of containers. The staffing pattern for each delivery van includes a driver, a supervisor, and one or two delivery personnel who load and unload the food. The Vasanthapura kitchen serves about 530 schools; each school has an average of 210 children. The schools are at an average distance of 17 kilometers (km) from the kitchen, with the farthest school 43 km away. The daily demand at each school is fairly stable in contrast to the meals-on-wheels problem (Bartholdi et al. 1983). There is ample demand within all the regions to expand the service coverage; however, TAPF is constrained by its fixed-fleet capacity and the available time window.

The capital investment for acquiring a delivery van is about US$28,000. Each year, TAPF’s central office makes the acquisition decisions, which are beyond the operating budgets of the regions (kitchen locations). The central office annually assigns a limited fleet of delivery vans to each region; only the operating costs are managed at the regional level. The objective of the manager in each region is to make the best use of the available resources to maximize the service coverage. TAPF incurs costs for monthly maintenance of the delivery vans, fuel costs, and staff salaries. The total daily trip length of the vehicles in the Vasanthapura region in its existing routing was about 1,400 km and took more than five hours to complete the food distribution. More details on TAPF’s operating model are available in Upton et al. (2007) and on the TAPF website (http://www.akshayapatra.org).

AMRUTA: OR Modeling, Solution, and Software Design

Operations Research Model

An abundance of OR literature is available on VRP variants that somewhat resemble the TAPF problem. For example, Dell’Amico et al. (2007) and Bräysy et al. (2008) study the fleet-size-mix VRP with time windows. In their problem variant, they endogenously determine the optimal fleet size and mix from a heterogeneous set of available vehicles, in addition to the best routing solution. However, in our problem, the choice of number and type of vehicles is strictly limited to the fixed fleet available within each location. Tarantilis et al. (2003) and Li et al. (2007) study the fixed-fleet heterogeneous VRP, which also resembles the TAPF problem. However the TAPF problem is additionally constrained on time windows. Privé et al. (2005) discuss a practical case of a soft drink manufacturer’s distribution problem, which involves a heterogeneous fleet in a capacitated environment with time windows. Unlike their situation, TAPF need not plan for the return logistics; instead, it faces the constraint of a fixed fleet size. Overall, we can generalize the TAPF problem as a HFFVRPTW.

This problem can be categorized among the hardest of NP-hard VRPs. Appendix A shows the mixed-integer formulation of the TAPF problem. According to this formulation, a problem size of 500 schools and 30 vehicles would involve 7.56 million binary variables and 15,000 real variables. For bigger kitchens, the problem size grows exponentially. Therefore, because of its intractability and execution complexity, it is not amenable to exact solutions or partial enumeration-based solution techniques. Hence, we propose a heuristic solution approach.

Solution Approach

Solutions for industrial-grade transportation problems with large number of nodes typically require a combination of heuristics and multistage decomposition (Fisher and Jaikumar 1981, Thangiah et al. 1994, Tan et al. 2001). We follow a similar solution approach in AMRUTA. Dondo and Cerdà (2007) study the heterogeneous fleet capacitated VRP with time windows in the context of multiple depots. They propose a three-stage solution in which the clusters of nodes around the depots are determined in the first stage. The assignment of vehicles to clusters is done in the second stage, and the ordering of nodes within the clusters is determined in the third stage. AMRUTA broadly follows a similar decomposition approach; however, the TAPF problem is a single-depot problem.

We can logically decompose the solution approach into three stages, as follows:

1. Stage 1: creating $k$ clusters of demand nodes;
Stage 2: assigning \( k \) vehicles to the \( k \) clusters; 
(3) Stage 3: ordering nodes within the clusters.

The solution techniques proposed for the respective stages are as follows:

(1) Stage 1: modified version of K-means clustering.
(2) Stage 2: a greedy heuristic initial solution, followed by a series of two-opt interchanges.
(3) Stage 3: a self-organizing map (SOM)-based genetic algorithm heuristic.

Figure 2 presents the schematic model of the proposed solution approach. One TAPF requirement is to create a range of solutions for different fleet sizes. We refer to the number of vehicles deployed in a particular solution by the variable \( k \). Each cluster of schools is served by a vehicle. Determining the bounds for the number of vehicles is a variable cost and size bin-packing problem (VSBPP) and is NP-hard. Several heuristics are available for obtaining these bounds (Crainic et al. 2011, Haouari and Serairi 2009). We adopt a simple heuristic to identify the bounds on the number of vehicles. The lower bound on the number of clusters (\( k \)) is the minimum number of vehicles (arranged in the descending order of respective capacities) that meets the demand. Because we assume that the available fleet has adequate capacity to service the complete demand, we therefore set the upper bound to be the total number of vehicles available. We execute the entire three-stage heuristic for each \( k \) value between these bounds, starting with a \( k \) value set at the lower bound and then iterating progressively. The use of bin-packing heuristics can provide tighter bounds; however, establishing tighter bounds is less critical to our solution approach because it is enumerative.

Stage 1

The objective of the first stage is to create \( k \) clusters that minimize the intracluster travel distance for vehicles. This reduces the problem to a capacitated clustering problem (CCP) in which the \( n \) nodes are to be clustered into \( k \) clusters with the objective of minimizing route cost or distance within a specified cluster capacity constraint. The CCP could be solved using the K-means clustering algorithm, which is a simple and computationally efficient method for partitioning a set of points with the objective of minimizing the distances of the points to the centroid. The quality of solutions generated by the K-means clustering algorithm usually depends on the quality of the choice of initial seeds (or initial centroids) (Lattin et al. 2003, pp. 290–291). Several approaches are available for solving the CCP problem (Krishna and Narasimha Murty 1999, Lu et al. 2004, Zalik 2008, Geetha et al. 2009).

The modified K-means solution technique in AMRUTA is similar to that of Geetha et al. (2009). Our modification over the standard K-means algorithm ensures that large demand points are kept separate as much as possible; thus, the smaller demand points can be packed into the nearest clusters without violating the capacity constraints. We achieve this by using priority values (as opposed to just distances) for clustering. The top \( k \) large demand nodes are chosen as the initial centroids. The remaining nodes are clustered based on their priority values. The priority value of a node is computed as the ratio of the distance of the node from the cluster centroid to the demand at the node. A lower value indicates higher priority and vice versa. An unassigned node is assigned to the nearest cluster that has adequate capacity to serve the demand at the node. The time-window consideration is relaxed during this stage, because using it before completing the vehicle assignment makes little sense. The output from this stage is a set of \( k \) clusters of nodes that satisfy the capacity constraint.

Stage 2

AMRUTA initially follows a greedy heuristic in which the first \( k \) vehicles from the fixed fleet with the lowest monthly operating cost are initially assigned to the \( k \) clusters. Because of the heterogeneity in the vehicle capacities, slack or surplus may exist in the individual clusters. At this stage, we improve the allocation by refining cluster membership without violating the time-window and vehicle-capacity constraints. Several studies use a similar approach (Thangiah et al. 1994, Tan et al. 2001, Ferland and Michelon 1988, Tarantilis et al. 2003). We follow a two-opt interchange mechanism, which is qualitatively similar to the \( \lambda \)-interchange mechanism followed by Thangiah et al. (1994) with \( \lambda = 1 \). In this process, nodes belonging to one cluster are evaluated for interchange with another cluster to determine if the solution quality improves.
If the new solution is feasible, then we consider it to be better if the total cost decreases. This process leads to the merger and demerger of clusters, finally yielding the best solution that minimizes the monthly operating cost. Nodes that could not be accommodated in any cluster are left unclustered for manual consideration.

**Stage 3**

In the third stage, we use a route-planning heuristic to arrive at the visitation sequence of the nodes in each cluster. This problem resembles the travelling salesman problem (TSP) for each of the $k$ petals of the overall route map. However, we still need to preserve the constraints on capacities and time windows in the final solution. See Laporte et al. (2002) and Nilsson (2003) for a survey of exact and approximate solution techniques for the TSP problem. Gendreau et al. (2002) provide a good overview of applying meta-heuristics to the capacitated VRP. Kohonen (1990) first proposed the genetic algorithm called SOM for solving industrial-grade TSP problems. Several studies discuss the application of the SOM algorithm for solving the TSP and VRP problems (Budinich 1996, Modares et al. 1999, Bai et al. 2006, Brocki and Korzynke 2007, Créput et al. 2007). The SOM algorithm is computationally efficient and generates high-quality heuristic solutions.

SOM extends the basic concept of a solution to a single-dimensional TSP problem, where the greedy nearest-neighbor heuristic of jumping to the next unvisited node generates good quality solutions. This is done by projecting points from two-dimensional Euclidean space onto a circular ring and applying the nearest-neighbor principle on the projected images. Because perfect preservation of neighborhoods is not possible with projection, the genetic algorithm needs clever choices of policy parameters to preserve the neighborhood as much as possible. The SOM algorithm needs two policy parameters, that is, the learning rate $\alpha$ and the neighborhood function variance $\sigma$ (Bai et al. 2006). For more details, see Fröhlich (2004) and the list of online SOM resources hosted on the University of North Carolina website (Bauers 2010). Appendix B includes the high-level pseudocode of the complete algorithm.

**AMRUTA Implementation**

**Software Design Considerations**

We identified four key considerations for the design and development of the algorithm and the software: meeting management expectations, end-user acceptance, low-cost, and ease of maintenance. TAPF’s management preferred a range of best solutions based on the available fleet, as opposed to a single optimal solution as the software proposed. Its rationale is twofold. Not all factors that influence day-to-day operational decision making can be modeled upfront. Management wanted a series of what-if scenarios based on the available fleet. Its expectation was to use this range of solutions to determine the best routing plan for the day, by manually considering exogenous factors not included in the model (e.g., staff absenteeism, vehicle unavailability). Management also preferred to receive a graphical representation of the solution and to generate multiple management information system (MIS) reports using
Section 3: User Acceptance of the Software

Figure 3: This route map of the distribution model that TAPF used prior to implementing AMRUTA shows crisscrossing and overlapping routes that highlight the inefficiencies in the model.

Finally, user acceptance of the software was an important design consideration. User acceptance refers to much more than simply signing off on the software. It also refers to a commitment from the user community to actively use the software on a day-to-day basis to make operational decisions. Because the TAPF user base is not technologically savvy, ease of use was important in achieving user acceptance. TAPF management preferred less-expensive software and upgrade costs. Therefore, low license cost and usability were also included as design considerations.

Software Implementation

With the previously mentioned design considerations in mind, we developed AMRUTA using Microsoft.
Visual Basic, and programmed the three-stage heuristic routines as Visual Basic macros. We used Microsoft Excel to enter input data and generate output reports. Because Excel is part of the standard desktop configuration at TAPF, every user has access to it and some prior experience using it. Additionally, Excel’s ease of data maintenance and graphical reporting features makes it suitable for users.

Tables 1(a) and 1(b) show snapshots of the input data pertaining to the demand points and fleet size, respectively. We represent the geographical distribution of the demand nodes using global positioning system (GPS) coordinates. This spreadsheet format is intuitive and easy to use for the user community to enter and update data. AMRUTA’s processing time on a standard Windows desktop is about three minutes. The complete set of reports is generated in an additional 10 minutes. These are consistent with our requirement to execute the software again if the input data change.

The output from AMRUTA includes a master solution summary, which gives an overview of the range

Figure 4: This route map, which AMRUTA generated for a 27-vehicle fleet, shows no crisscrossing flows or overlaps; this solution reduced the number of vehicles from 35 to 27 and the total trip length by 75 km per day.
Input data (demand details)

Please enter the name, GPS coordinates, demand for every school in separate rows. The kitchen should always be the first entry in the following table. Do not leave any blanks between the data.

<table>
<thead>
<tr>
<th>S. no.</th>
<th>School name</th>
<th>Demand</th>
<th>Latitude</th>
<th>Longitude</th>
<th>Rice (big vessel)</th>
<th>Sambar (medium vessel)</th>
<th>Curd (small vessel)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Vasanthapura Kitchen</td>
<td>0</td>
<td>12</td>
<td>53</td>
<td>17.23</td>
<td>77</td>
<td>32</td>
</tr>
<tr>
<td>1</td>
<td>RV Girls HS—Jayanagar</td>
<td>1</td>
<td>12</td>
<td>56</td>
<td>26.19</td>
<td>77</td>
<td>34</td>
</tr>
<tr>
<td>2</td>
<td>CNS—Jayanagar</td>
<td>2</td>
<td>12</td>
<td>56</td>
<td>34.61</td>
<td>77</td>
<td>34</td>
</tr>
<tr>
<td>3</td>
<td>Parikrama—Jayanagar (P)</td>
<td>3</td>
<td>12</td>
<td>56</td>
<td>28.59</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>Rani HPS—Jayanagar</td>
<td>4</td>
<td>12</td>
<td>56</td>
<td>18.73</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>5</td>
<td>Rani HS—Jayanagar</td>
<td>5</td>
<td>12</td>
<td>56</td>
<td>18.73</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>6</td>
<td>SESS HPS—LalBhagh Siddapura</td>
<td>6</td>
<td>12</td>
<td>56</td>
<td>47.60</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>St. Andrews KHPS—L.Siddapura</td>
<td>7</td>
<td>12</td>
<td>56</td>
<td>41.02</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>GHPS—LalBhagh</td>
<td>8</td>
<td>12</td>
<td>56</td>
<td>44.04</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>9</td>
<td>s/c—Dayanada Nagar</td>
<td>9</td>
<td>12</td>
<td>56</td>
<td>39.14</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>Hombegowda BHS—W.Garden</td>
<td>10</td>
<td>12</td>
<td>56</td>
<td>49.05</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>11</td>
<td>GHPS—W.Garden</td>
<td>11</td>
<td>12</td>
<td>56</td>
<td>54.17</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>12</td>
<td>GHS—W.Garden</td>
<td>12</td>
<td>12</td>
<td>56</td>
<td>54.01</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>13</td>
<td>Ganganma GHS—W.Garden</td>
<td>13</td>
<td>12</td>
<td>56</td>
<td>52.15</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>14</td>
<td>CNS—Lakkasandra</td>
<td>14</td>
<td>12</td>
<td>56</td>
<td>49.00</td>
<td>77</td>
<td>35</td>
</tr>
<tr>
<td>15</td>
<td>CPS—Lakkasandra</td>
<td>15</td>
<td>12</td>
<td>56</td>
<td>49.00</td>
<td>77</td>
<td>35</td>
</tr>
</tbody>
</table>

Table 1(a): This table shows a snapshot of input demand points that users enter and maintain in a spreadsheet format. The input variables are school names, GPS coordinates, and demand quantities for food items.

Input data (vehicle details)

Please enter the vehicle details in descending order of capacity i.e., large vehicles first followed by smaller vehicles. Do not leave any lines blank between the data. Assumption: There are three racks: bottom, middle, and top. Big vessels sit in the bottom rack, medium vessels in the middle, and small vessels at the top.

<table>
<thead>
<tr>
<th>S. no.</th>
<th>Vehicle type</th>
<th>No. of vehicles</th>
<th>Rack capacity</th>
<th>No. of personnel</th>
<th>Salary per month (INR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Eicher</td>
<td>9</td>
<td>60 60 60 60 60 60</td>
<td>3.00 420</td>
<td>1 1 2</td>
</tr>
<tr>
<td>2</td>
<td>Tata</td>
<td>11</td>
<td>60 60 60 60 60 60</td>
<td>3.00 420</td>
<td>10,723 1 1 2</td>
</tr>
<tr>
<td>3</td>
<td>Load King</td>
<td>5</td>
<td>40 40 40 40 40 40</td>
<td>4.00 280</td>
<td>9,844 1 1 2</td>
</tr>
<tr>
<td>4</td>
<td>Svaraj Mazda</td>
<td>5</td>
<td>40 40 40 40 40 40</td>
<td>4.00 280</td>
<td>9,844 1 1 2</td>
</tr>
<tr>
<td>5</td>
<td>Max Pikup</td>
<td>5</td>
<td>24 24 24 24 24 24</td>
<td>6.00 168</td>
<td>2,825 1 1 1</td>
</tr>
</tbody>
</table>

Table 1(b): This table shows a snapshot of input data on vehicle details that users enter and maintain in spreadsheet format. The data input fields are vehicle type, number of vehicles, capacity, mileage, and cost and staffing parameters.

of solutions for different deployed fleet sizes, including their trip elapsed times, trip distances, and total operating costs (see Table 2). AMRUTA also generates a series of solution-level output reports, including the solution summary (see Table 3), the graphical route map (see Figure 4), and the routing plan for an individual vehicle (i.e., the trip sheet) in both tabular format and graphical view for each route within a solution (see Figure 5).

The TAPF operations manager uses the master solution summary report to determine the best daily routing plan. This report helps the manager make
vehicle solution, and took over five hours to complete 20–35 vehicles. The minimal-cost solution was the 20-Vasanthapura. It generated a range of solutions with values and the projected trip duration in minutes, trip length in kilometers, and the associated monthly operating costs.

<table>
<thead>
<tr>
<th>Sl. no.</th>
<th>No. of vehicles</th>
<th>Elapsed time at the last node (minutes)</th>
<th>Distance covered (circular) in kms</th>
<th>Total distance covered (kms)</th>
<th>Total number of unclustered nodes</th>
<th>Total distribution cost (rupees)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution #1</td>
<td>20</td>
<td>125.58</td>
<td>305.73</td>
<td>216.93</td>
<td>35.57</td>
<td>101.96</td>
</tr>
<tr>
<td>Solution #2</td>
<td>20</td>
<td>119.14</td>
<td>307.53</td>
<td>215.21</td>
<td>28.46</td>
<td>102.71</td>
</tr>
<tr>
<td>Solution #3</td>
<td>22</td>
<td>126.30</td>
<td>294.00</td>
<td>196.16</td>
<td>28.79</td>
<td>97.29</td>
</tr>
<tr>
<td>Solution #4</td>
<td>21</td>
<td>119.14</td>
<td>309.55</td>
<td>203.69</td>
<td>26.87</td>
<td>103.55</td>
</tr>
<tr>
<td>Solution #5</td>
<td>22</td>
<td>83.90</td>
<td>286.67</td>
<td>192.60</td>
<td>28.64</td>
<td>94.20</td>
</tr>
<tr>
<td>Solution #6</td>
<td>23</td>
<td>96.38</td>
<td>286.67</td>
<td>185.31</td>
<td>21.48</td>
<td>94.20</td>
</tr>
<tr>
<td>Solution #7</td>
<td>25</td>
<td>48.99</td>
<td>268.34</td>
<td>174.78</td>
<td>17.83</td>
<td>94.20</td>
</tr>
<tr>
<td>Solution #8</td>
<td>26</td>
<td>48.99</td>
<td>261.78</td>
<td>173.81</td>
<td>17.83</td>
<td>91.48</td>
</tr>
<tr>
<td>Solution #9</td>
<td>26</td>
<td>48.99</td>
<td>261.98</td>
<td>172.57</td>
<td>17.83</td>
<td>94.06</td>
</tr>
<tr>
<td>Solution #10</td>
<td>26</td>
<td>49.23</td>
<td>244.60</td>
<td>172.15</td>
<td>17.93</td>
<td>95.65</td>
</tr>
<tr>
<td>Solution #11</td>
<td>27</td>
<td>39.15</td>
<td>246.67</td>
<td>165.92</td>
<td>14.49</td>
<td>94.06</td>
</tr>
<tr>
<td>Solution #12</td>
<td>28</td>
<td>39.15</td>
<td>245.27</td>
<td>161.98</td>
<td>14.06</td>
<td>91.66</td>
</tr>
<tr>
<td>Solution #13</td>
<td>29</td>
<td>39.15</td>
<td>247.50</td>
<td>158.58</td>
<td>13.63</td>
<td>91.48</td>
</tr>
<tr>
<td>Solution #14</td>
<td>30</td>
<td>41.57</td>
<td>245.45</td>
<td>154.02</td>
<td>11.72</td>
<td>91.48</td>
</tr>
<tr>
<td>Solution #15</td>
<td>31</td>
<td>41.57</td>
<td>247.33</td>
<td>149.69</td>
<td>12.34</td>
<td>91.66</td>
</tr>
<tr>
<td>Solution #16</td>
<td>32</td>
<td>53.01</td>
<td>246.67</td>
<td>147.33</td>
<td>12.87</td>
<td>91.48</td>
</tr>
<tr>
<td>Solution #17</td>
<td>33</td>
<td>38.60</td>
<td>242.72</td>
<td>142.96</td>
<td>12.34</td>
<td>91.69</td>
</tr>
<tr>
<td>Solution #18</td>
<td>34</td>
<td>38.60</td>
<td>245.57</td>
<td>140.20</td>
<td>9.96</td>
<td>96.85</td>
</tr>
<tr>
<td>Solution #19</td>
<td>35</td>
<td>24.46</td>
<td>244.38</td>
<td>136.13</td>
<td>7.79</td>
<td>97.52</td>
</tr>
</tbody>
</table>

Table 2: A snapshot of the master solution summary report presents a range of best solutions for different \( k \) values and the projected trip duration in minutes, trip length in kilometers, and the associated monthly operating costs.

Dynamic decisions about the best routing alternative for the day, considering exogenous factors, such as cooking completion times, local school holidays, traffic conditions, staff absenteeism, and vehicle availability. In addition, it also allows the manager to drill down to view the routing plan and the cost of a particular solution, and aids in making dynamic trade-offs between cost and service time. Each day, the trip sheet is printed and given to the vehicle’s driver and (or) supervisor as a travel aid. Comparing the generated trip sheet and the actual trip data is a valuable exercise to improve the policy parameters for executing AMRUTA. These reports were also useful during user-acceptance testing and pilot implementation for critically reviewing the working of the algorithm and fine-tuning the policy parameters. Finally, they also help TAPF management estimate the savings generated by AMRUTA.

In March–April 2011, we piloted AMRUTA in Vasanthapura. It generated a range of solutions with 20–35 vehicles. The minimal-cost solution was the 20-vehicle solution, and took over five hours to complete the distribution. However, TAPF’s management preferred the 27-vehicle solution because its total trip duration was four hours. Table 4 shows a comparison of the existing routing structure with the minimal-cost (i.e., 20-vehicle) solution and the preferred (i.e., 27-vehicle) solution. This table illustrates the critical trade-off that must be made between cost and responsiveness in determining the best routing plan for the day. For example, the monthly operating cost of the 20-vehicle solution is 35 percent lower than TAPF’s existing routing plan, whereas the cost of the 27-vehicle solution is only 18.6 percent lower. However, the trip duration (responsiveness) of the 20-vehicle solution is about 24 percent higher than that of the 27-vehicle solution. This led TAPF management to prefer the 27-vehicle solution because it provides a good balance between cost and responsiveness. Following the AMRUTA implementation, management reduced the fleet size from 35 to 27. The annualized cost savings realized in this region during the pilot period was US$75,000 (18.61 percent savings in monthly operating cost). This would enable TAPF to expand service
coverage in the region by about 4.8 percent (i.e., provide service to an additional 2,400 children). The solution is also ecofriendly because it reduces the total trip length from 1,400 km to 1,325 km, thereby saving about 443 litres of diesel per month. TAPF is currently implementing AMRUTA in all its kitchens. Assuming cost savings of similar proportions, it projects annual savings across India to be in the range of US$1.96 million. At the prevailing costs, this will enable TAPF to expand services nationally by an additional 62,000 children.

### Implementation Adaptations

Given the scale of the TAPF logistics problem, AMRUTA required several heuristic adjustments and adaptations to standard OR algorithms. The challenge of solving large real-life problems lies in incorporating realistic conditions without compromising the solution quality or increasing the algorithm’s complexity. During AMRUTA development, we addressed these issues in several ways.

To improve processing time and mathematical tractability, we aggregate the demand for the food items into an equivalent demand unit. For example, one vessel of steamed rice is equivalent to two vessels of sambar or four vessels of yoghurt. We also must consider several aspects to model vehicle travel time. We apply a correction factor to the calculated Euclidean distance to account for road quality, traffic conditions, and available connectivity between the nodes. Furthermore, the model uses two speed bands (i.e., one for the travel between the kitchen and the

---

**Table 3:** A snapshot of the solution summary report presents the details of the number of schools covered by each vehicle, the estimated trip duration, trip length, and associated cost details.
first node in the cluster, and the second for travel between successive nodes within the cluster). This adjustment helps account for the variations in vehicle speed between urban and rural roads and the frequent stop-start sequence within the cluster. We divide the service time at a school into a fixed component and a variable component, which we link to the number of containers to be unloaded at the site, thus making service-time modeling realistic.

Similarly, in the three-stage decomposition structure, the first stage of clustering checks only for violations of the capacity constraint, and we relax the check on time windows. To reduce the reorganization effort in the next stage, the clusters are packed up to only 80 percent of capacity. If unclustered nodes are still left after this packing, then clusters are packed to 100 percent capacity. In Stage 2, if certain clusters are filled to 150 percent of capacity, they are divided into two clusters with the top-two demand nodes as their initial centroids. Those filled to less than 25 percent of capacity are discarded and the relevant nodes moved
Table 4: A comparison of the existing routing plan with two of the solutions (20- and 27-vehicle) generated by AMRUTA illustrates the nature of the trade-off between cost and responsiveness.

<table>
<thead>
<tr>
<th>Comparison parameters</th>
<th>Existing AMRUTA solution with 20 vehicles</th>
<th>Minimal cost</th>
<th>Savings</th>
<th>% savings</th>
<th>AMRUTA solution with 27 vehicles</th>
<th>Preferred solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of vehicles used</td>
<td>35</td>
<td></td>
<td>20</td>
<td>15</td>
<td>20</td>
<td>27</td>
</tr>
<tr>
<td>Total trip length in kilometres</td>
<td>1,400(^1)</td>
<td></td>
<td>204.31</td>
<td>14.6</td>
<td>1,325.11</td>
<td>74.89</td>
</tr>
<tr>
<td>Elapsed time to complete distribution (minutes)</td>
<td>n.a</td>
<td></td>
<td>n.a</td>
<td>n.a</td>
<td>246.67</td>
<td>n.a</td>
</tr>
<tr>
<td>Monthly operating cost (in US(^\ast))</td>
<td>$33,818</td>
<td></td>
<td>$11,682</td>
<td>34.5</td>
<td>$27,523</td>
<td>$6,295</td>
</tr>
</tbody>
</table>

\(^1\) Estimated based on available data.
\(^\ast\) Estimated at a Diesel price of Rs. 45 per litre, 22 working days per month, and exchange rate of 1 USD = 50 INR.

to nearest neighbor. This heuristic adjustment reduces the workload for the two-opt interchange process.

Checking on the time-window constraint stresses the system resources. To overcome this challenge, we developed an adaptation. Of the total time available, the transportation and service time are approximated to be split in a 60–40 ratio. For example, if 240 minutes is available for a trip, nodes are added up to only a total travel time of 144 minutes. This adaptation reduces the processing overhead to compute the exact service time for evaluating a feasible move. The high-level pseudocode in Appendix B explains these adaptations. Thus, during the Stage 2 process, clusters are merged and demerged and the inefficient clusters are split or eliminated. Only during Stage 3, after the optimal visitation sequences are computed, are the exact time-window constraints checked to determine feasibility. These adjustments help improve the processing time and tractability of the solution.

We modeled these heuristic adjustment factors (percentages and ratios) as input parameters to AMRUTA. Because the quality of a solution is a function of the input data, such a parametric design approach provides us the ability to continuously review and fine-tune these parameters. During the pilot implementation, this approach facilitated user acceptance of the proposed solution.

Resistance to change is not uncommon in organizations. TAPF operations managers have been making day-to-day routing decisions to the best of their abilities, taking into account several exogenous factors that are not easy to model. In developing AMRUTA, these organizational dynamics influenced certain design choices. The choice of providing a range of solutions for different deployed fleet sizes, rather than a single take-it-or-leave-it solution was a critical decision. This approach gave TAPF’s management a greater understanding of the behavior of various cost structures, helped it gravitate toward a workable solution that satisfied its overall requirements, and provided an opportunity for management and the users to actively participate in the decision making. Multiple solutions for various fleet sizes enabled both management and users to understand the critical trade-off between cost and responsiveness.

For example, if we reduce the number of vehicles from 27 to 26 in the Vasanthapura example, the trip duration increases by eight minutes and the average distance traveled per vehicle by 2.73 km. The total distance covered increases by 22 km, but the operating cost decreases by about US$800 per month. On a given day, managers can determine the appropriate routing plan by comparing costs and responsiveness. The monetary value of time (responsiveness) is a function of the exogenous factors that prevail on that day. Hence, flexibility to perform dynamic trade-offs is critical when faced with factors such as unexpected delays in the cooking, prevalence of unusual weather and (or) traffic conditions, staff absenteeism, or vehicle unavailability. Thus, empowering managers to make dynamic decisions based on scientifically generated data improves both the operational decisions and the user community’s confidence in the solution.
Conclusion

MDMS programs are a prevalent mechanism to boost enrollment, retention, and participation of children at schools. TAPF operates such a program in India. TAPF faced the problem of using a fixed fleet of vehicles to efficiently distribute food within the available time window. We implemented an OR/MS-based logistics planning solution that has shown significant promise for reducing costs. This paper discusses the challenges faced and how we overcame them by using OR modeling in the design, development, and implementation of a software solution.

The heuristic solution discussed in this paper should interest researchers and practitioners working on large-scale capacitated VRPs with time-window constraints. This problem is representative of the logistic problem of distributing perishable food and beverages. Therefore, the model could serve as a case study for applying OR algorithms in the distribution problems that the food and beverage industries face. It can also be extended to other industries that have similar constraints of fixed-fleet capacity and strict time windows.

Better solution algorithms, especially those based on adaptive learning techniques, can further improve the quality of the solutions that AMRUTA generates. Stochastic demand and service times are areas for additional research. Integrated distribution planning of all the kitchens in a multidepot distribution model is another possible extension. TAPF’s vision—that no child should be deprived of education because of hunger—is a significant aspect of social welfare and societal transformation in countries such as India. This application illustrates the potential for using OR/MS algorithms to address such serious real-world issues, and how OR can be useful for solving social sector problems in a developing country such as India.

Appendix A. Mathematical Formulation of the TAPF Problem

For ease of cross-referencing, we present the mixed-integer formulation of the TAPF problem using the notations followed by Bräysy et al. (2008).

Let $C$ be a set of all the customer demand nodes. Let node 0 signify the central kitchen from which the transportation begins. Define $N$ as the set union $\{0\} \cup C$. Let $V$ be the set of vehicle types. Factors such as capacity limit and mileage offered are specific to the vehicle type. Let $L_k$ be the set of individual vehicles available within each type of the fixed fleet $k$.

The definitions of the decision variables used in the formulation are as follows:

- $X_{ij}^{kl}$: An indicator variable that is set to 1 if a vehicle of type $k$ with vehicle number $l$ directly travels from node $i$ to node $j$ in the undirected graph.
- $Y_{ij}^{kl}$: A nonnegative real number that captures the arrival time of a vehicle of type $k$ with vehicle number $l$, at node $i$.

The definitions of the constants used in the formulation are as follows:

- $d_{ij}$: Equivalent demand of food at node $i$; it is a composite demand of the three food items in an equivalence scale.
- $q_k$: Capacity limit of a vehicle of type $k$ expressed in equivalent demand units.
- $s_i$: Service time at node $i$; it can contain a fixed component and a variable component per container.
- $t_{ij}$: Vehicle independent travel time between node $i$ and node $j$.
- $a_i$: Earliest acceptable arrival time at node $i$.
- $b_i$: Latest acceptable arrival time at node $i$.
- $M_{ij}$: A large integer $M_i$, which acts as an upper-bound; according to Bräysy et al. (2008), we can set the bound for this term as $\max(b_i + s_i + t_{ij} - a_j, 0)$ for all $i \in N$, for all $j \in N$.
- $C_{ij}^k$: Travel cost for transportation between node $i$ and node $j$ using a vehicle of type $k$; it is a function of $t_{ij}$ and the mileage offered by a vehicle of type $k$.
- $M_C$: Monthly operating cost of a vehicle of type $k$; it includes the vehicle maintenance cost and staff salaries of vehicle operators.

$$\min \left\{ \sum_{k \in V} \sum_{l \in L_k} \sum_{i \in N} C_{ij}^k \cdot X_{ij}^{kl} + \sum_{k \in V} \sum_{l \in L_k} \sum_{j \in C} M_C \cdot X_{ij}^{kl} \right\}$$

subject to

$$\sum_{k \in V} \sum_{l \in L_k} \sum_{j \in C} X_{ij}^{kl} = 1 \quad \forall i \in C. \quad (A1)$$

$$\sum_{j \in N} \left[ d_{ij} \cdot \sum_{l \in L_k} X_{ij}^{kl} \right] \leq q_k \quad \forall k \in V, \forall l \in L_k, \quad (A2)$$

$$\sum_{j \in N} X_{ij}^{kl} = 1 \quad \forall k \in V, \forall l \in L_k, \quad (A3)$$
\[ \sum_{i \in N} x_{ih}^k - \sum_{j \in N} x_{ij}^k = 0 \quad \forall h \in C, \forall k \in V, \forall l \in L_k, \] (A4)

\[ \sum_{i \in N} x_{ij}^k = 1 \quad \forall k \in V, \forall l \in L_k, \] (A5)

\[ y_{ij}^k + s_i + t_{ij} - y_{ij}^k \leq (1 - x_{ij}^k) M_{ij}, \] (A6)

\[ a \sum_{j \in N} x_{ij}^k \leq y_{ij}^k \quad \forall i \in N, \forall k \in V, l \in L_k, \] (A7)

\[ b_l \sum_{j \in N} x_{ij}^k \geq y_{ij}^k \quad \forall i \in N, \forall k \in V, \forall l \in L_k, \] (A8)

\[ x_{ij}^k \in \{0, 1\} \]

\[ y_{ij}^k \in \mathbb{R}; \quad y_{ij}^k \geq 0. \]

The objective function attempts to minimize the sum of the transportation cost and the monthly operating cost of the fixed fleet. Constraint (1) ensures that each node is visited exactly once and by only one vehicle. Constraint (2) is the heterogeneous capacity constraint posed by the vehicle. Constraint (3) ensures that each vehicle is assigned to exactly one route from the kitchen. Constraint (4) is the flow-conservation constraint at all intermediate nodes of a route. Constraint (5) ensures that every vehicle ends back at the kitchen exactly once. This also helps to eliminate subtours. Constraint (6) preserves the integrity of the arrival times at successive nodes on a route. Constraints (7) and (8) are the constraints on earliest and latest arrival times, respectively. Constraint (9) ensures that invalid arcs are eliminated from consideration.

We note that even for a moderately-sized problem, the number of variables could be very large. For a network with \( n \) schools served from a single kitchen, the number of potential paths that can be served by \( k \) vehicle classes of \( L_k \) vehicles each is \((n + 2)^2\). Thus, the number of binary decision variables is \( kL_k(n + 2)^2 \). The arrival time of a vehicle at a particular school is captured as a positive real variable \((Y)\). The number of \( Y \) variables is \( kL_k(n + 2) \).

Appendix B. High Level Pseudocode of the Solution Algorithm

1. Compute the lower bound (KLB) and upper bound (KUB) of the deployed fleet size.

2. Do for every \( k \) from KLB to KUB;

   \( \text{Stage 1: Create } K \text{ clusters that satisfy the capacity constraint.} \)
   - Assign the \( k \) nodes (schools) with highest demand as initial centroids.
   - Compute the priority values for each remaining node, and cluster them using the modified \( K \)-means algorithm based on their priority values.
   
   \( \text{Stage 2: Assign vehicles and refine cluster membership.} \)
   - Assign the \( k \) vehicles with the lowest monthly cost to the \( k \) clusters.
   - Compute capacity required for each cluster;
     - If required capacity in a cluster is more than desired maximum, split to form two new clusters with the top-two demand nodes as initial centroids;
     - If required capacity is less than desired minimum, merge the cluster with its nearest neighbor.
   - Use a greedy heuristic to create an initial feasible visitation sequence for each cluster.
   - Compute the trip duration for each cluster and compare against the available window for travel time.
   - Use the two-opt interchange process to assess feasible moves of nodes across clusters. Perform the following checks on the destination cluster to evaluate feasible moves;
     - Ensure that interchanging the node does not exceed the time window;
     - Ensure that after the interchange, the vehicle capacity is not exceeded.
   - \( \text{Stage 3: Establish the visitation sequence of the nodes in a cluster.} \)
     - Apply the SOM algorithm to compute the optimal visitation sequence of nodes within each cluster.
     - Recompute the trip duration using actual travel times and service times and check for violations of the time-window constraint;
       - If violations exist, make a feasible move (satisfying the time-window and capacity constraints) to the nearest-neighbor cluster;
       - else, mark it as unclustered node.
     - End of Do loop.
     - Generate the MIS reports for the range of solutions.

Acknowledgments
We acknowledge the support of Prabhu Madhu Pandit Dasa, TAPF chairman, Prabhu Chanchalpathi Dasa, TAPF
vice chairman, and Shridhar Venkat, TAFT executive director, to study this problem. We also acknowledge the significant role played by PN Seshadri of Kaul Associates in developing the solution and AMRUTA.

References


Verification Letter

Shridhar Venkat, Executive Director, The Akshaya Patra Foundation, Bangalore 560022, India, writes:

“We are pleased to certify that the Akshya Patra Transportation and Routing Algorithm (AMRUTA) software that was developed under the guidance of Professor Shri. B. Mahadevan has been successfully deployed as a part of our mid-day meals distribution operations in Vasanthapura kitchen. We are currently in the process of rolling out this software to our other kitchens across India.

“Some of the important design aspects of AMRUTA that makes it highly suitable for our operations are its ability to provide a range of best solutions along with their associated cost structure and the ability to maintain/modify input data with ease and re-execute the solution. The transparency and management insight offered by this design is
extremely valuable for our decision making. This design has also helped improve the uptake and active use of the software among our logistics planners and managers.

“Since implementation, AMRUTA has made a significant impact on the efficiency of our logistics and distribution process. We have verified the financial figures quoted in the research paper and they are in-line with our internal estimates.

“As a leading non-profit organization that is committed toward eradication of hunger among school-going children, we have always believed in adopting professional management practices in order to make best application of the public donations we receive. AMRUTA is one such initiative that has brought professionalism into the management of our logistics and distribution process. The savings generated by AMRUTA are extremely valuable in our ongoing endeavor to improve our service coverage and reach out to as many more school children as possible.”

B. Mahadevan is a professor of operations management at the Indian Institute of Management Bangalore, where he has been teaching since 1992. Professor Mahadevan was previously the EADS–SMI chair professor for sourcing and supply management at IIM Bangalore. He received his MTech and PhD from the Industrial Engineering and Management Division of IIT Madras. Professor Mahadevan is a member of the editorial board of the Production and Operations Management Journal and the International Journal of Business Excellence. He served in the editorial board of Six Sigma and Competitive Advantage.

S. Sivakumar is a doctoral student in production and operations management in Indian Institute of Management, Bangalore. His research interests are in service operations management, particularly on customer-centric operations and capacity management problems. He is a mechanical engineer from the National Institute of Technology, Trichy, India and has a post-graduate diploma in management from IIM, Bangalore. He has worked with Wipro Technologies for 14 years, serving several European and U.S.-based customers belonging to diverse service industries.

D. Dinesh Kumar is a marketing manager for Mobility Solutions at Unisys Corporation. He is passionate about technology and is keen on solving complex real-life problems using innovative technology solutions. He received his post-graduate diploma in management from IIM Bangalore. After obtaining his graduate degree in computer engineering, he worked in Tata Consultancy Services (TCS), a leading Indian IT firm.

K. Ganeshram is an associate director with Flipkart India Private Limited, one of India’s leading e-commerce firms. An electrical engineer by qualification, he received his post-graduate diploma in management from IIM Bangalore. With around five years of experience, he has worked in areas of supply chain, procurement, warehousing, and project management.