Delft University of Technology

Delft Center for Systems and Control

Technical report 12-028

An improved method for solving micro-ferry scheduling problems*

M. Burger, B. De Schutter, and H. Hellendoorn

If you want to cite this report, please use the following reference instead:
M. Burger, B. De Schutter, and H. Hellendoorn, "An improved method for solving micro-ferry scheduling problems," *Proceedings of the 1st European Symposium on Quantitative Methods in Transportation Systems (LATSIS 2012)*, Lausanne, Switzerland, 8 pp., Sept. 2012. Paper 56.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

An Improved Method for Solving Micro-Ferry Scheduling Problems

M. Burger*, B. De Schutter and H. Hellendoorn
Delft Center for Systems and Control,
Delft University of Technology, Delft, The Netherlands
*Email: m.burger@tudelft.nl

1 Introduction

In [3, 4] we have introduced a model that can be used to solve micro-ferry scheduling problems. It is a variant of the travelling salesman problem with a focus on scheduling pick-ups and deliveries of passengers for micro-ferries in a harbour. This problem distinguishes itself from other variants by the use of variable speeds of the vehicles, and the explicit consideration of the energy consumption of the ferries. Based on a mixed-integer programming description of the travelling salesman problem [1, 6], the micro-ferry scheduling problem is formulated using additional variables and constraints. The result is a non-linear mixed-integer optimisation problem that can be approximated by a mixed-integer linear program. In this paper a method is introduced to solve this optimisation problem based on constraint optimisation, and its improvement in computation time compared to the method of [3, 4] is shown by simulations.

2 Micro-ferry scheduling problems

At first a brief description of the micro-ferry scheduling problem is provided; more details on the modelling of this problem can be found in [3, 4]. With micro-ferries we denote small, autonomous water-taxis that can travel at different speeds within a certain range.

Throughout the paper the index j is used to denote the transportation request of interest, and the index i to denote its predecessor. A network with M micro-ferries and N unscheduled transportation requests is considered, and combining the requests currently handled by the micro-ferries with the new requests results in $R = M + N$ requests in total. Furthermore, without loss of generality, we use the sets

$$\mathcal{M} = \{1, \ldots, M\}, \quad \mathcal{N} = \{M + 1, \ldots, M + N\}, \quad \mathcal{R} = \{1, \ldots, M + N\}, \tag{1}$$

to denote the currently handled request, the new requests, and all requests respectively.

2.1 Objectives of the problem

The problem of scheduling a group of micro-ferries takes into account a trade-off between

- minimising the energy consumption,
- minimising the difference between scheduled and desired pick-up times,
- minimising the empty-travel distance of the micro-ferries,
- minimising the travel time for passengers.

The objectives can be achieved by minimising the objective function

$$J = \alpha_{\mathrm{ec}} J_{\mathrm{ec}} + \alpha_{\mathrm{tw}} J_{\mathrm{tw}} + \alpha_{\mathrm{et}} J_{\mathrm{et}} + \alpha_{\mathrm{tt}} J_{\mathrm{tt}}, \tag{2}$$

where $\alpha_{\mathrm{ec}}, \alpha_{\mathrm{tw}}, \alpha_{\mathrm{et}}, \alpha_{\mathrm{tt}} \geq 0$ are weights[1]. The *energy consumption* is given by

$$J_{\mathrm{ec}} = \sum_{i\in\mathcal{R}} \sum_{j\in\mathcal{R}} (\mathsf{p}_2 u_j + \mathsf{p}_1 + \mathsf{p}_0 \frac{1}{u_j}) \mathsf{C}_{ij} x_{ij} = \sum_{j\in\mathcal{R}} (\mathsf{p}_2 u_j + \mathsf{p}_1 + \mathsf{p}_0 w_j) \sum_{i\in\mathcal{R}} \mathsf{C}_{ij} x_{ij}, \tag{3}$$

where $\mathsf{p}_0, \mathsf{p}_1, \mathsf{p}_2 \geq 0$ are parameters of the function

$$P(u_j) = \mathsf{p}_2 u_j^2 + \mathsf{p}_1 u_j + \mathsf{p}_0, \tag{4}$$

describing the instantaneous power (in [W]) of the micro-ferries as a function of the speed u_j. The constant C_{ij} reflects the distance the micro-ferry has to travel if it would perform request j directly after request i; the binary variable x_{ij} shows whether ($x_{ij} = 1$) or not ($x_{ij} = 0$) transportation request j will be handled directly after request i. The variable $w_j = u_j^{-1}$ is the reciprocal of the speed, referred to as the pace of the micro-ferry [5].

For each request j there is a desired time-window $[t_{\mathrm{a},j}, t_{\mathrm{b},j}]$ for the pick-up time. The variable $s_j \geq 0$ gives the amount of time the pick-up time is scheduled outside the time-window (the misfit). Therefore, the *time-window misfit* becomes

$$J_{\mathrm{tw}} = \sum_{j\in\mathcal{R}} s_j. \tag{5}$$

The *empty-travel distance* (i.e. the distance without passengers aboard) gives undesired costs for the operator, and it can be obtained by

$$J_{\mathrm{et}} = \sum_{i\in\mathcal{R}} \sum_{j\in\mathcal{R}} \mathsf{c}_{ij} x_{ij}, \tag{6}$$

where c_{ij} equals the distance from the delivery location of request i to the pick-up location of request j, referred to as the relocation distance. The travel time can be considered as a measure for the quality of service towards the customers; lower travel times mean faster arrivals. The *total travel time* is given by

$$J_{\mathrm{tt}} = \sum_{j\in\mathcal{R}} \frac{\mathsf{c}_{jj}}{u_j} = \sum_{j\in\mathcal{R}} \mathsf{c}_{jj} w_j, \tag{7}$$

with c_{jj} the distance from the pick-up location to the delivery location of request j, referred to as the transportation distance. The constant C_{ij} in (3) equals the sum of c_{ij} and c_{jj}.

2.2 Constraints on the optimisation variables

Besides the optimisation variables x_{ij}, u_j, w_j, and s_j introduced in the previous section, we also use the following variables to describe the micro-ferry scheduling problem:

- e_j: energy level of the micro-ferry after completion of request j,
- t_j: pick-up time for the passengers of request j,
- k_j: index number of the micro-ferry that handles request j,
- y_j: indicator of whether ($y_j = 1$) or not ($y_j = 0$) to charge at the end of request j,
- τ_j: charging time associated with request j,
- ϵ_j: energy consumed during request j.

[1] For the objectives that are to be taken into account, the corresponding weight should be strictly positive. If some objectives are not important, e.g. the empty travel distance or the total travel time, the associated weights can be set to zero.

2.2.1 Constraints for the exact problem description

Using the variables defined above, the micro-ferry scheduling problem can be stated as [3, 4]

$$\text{minimise} \quad \alpha_{\text{ec}} J_{\text{ec}} + \alpha_{\text{tw}} J_{\text{tw}} + \alpha_{\text{et}} J_{\text{et}} + \alpha_{\text{tt}} J_{\text{tt}} \tag{8a}$$

$$\text{subject to} \quad \sum\nolimits_{i \in \mathcal{R}} x_{ij} = 1; \qquad \sum\nolimits_{i \in \mathcal{R}} x_{ji} = 1 \qquad \forall\, j \in \mathcal{R} \tag{8b}$$

$$t_i - t_j + \mathsf{c}_{ii} w_i + \mathsf{c}_{ij} w_j + \tau_i + \mathsf{t}_{\text{ch}} y_i + \mathsf{T} x_{ij} \leq \mathsf{T} - \mathsf{t}_{\text{de}} \qquad \forall\, i \in \mathcal{R}, j \in \mathcal{N} \tag{8c}$$

$$\mathsf{t}_{\text{a},j} - s_j \leq t_j \leq \mathsf{t}_{\text{b},j} + s_j \qquad \forall\, j \in \mathcal{R} \tag{8d}$$

$$\mathsf{E}(x_{ij} - 1) \leq e_i - e_j + \mathsf{r}_{\text{ch}} \tau_i - \epsilon_j \leq \mathsf{E}(1 - x_{ij}) \qquad \forall\, i \in \mathcal{R}, j \in \mathcal{N} \tag{8e}$$

$$\mathsf{E}(x_{ij} - 1) \leq \epsilon_j - \mathsf{C}_{ij} \left(\mathsf{p}_2 u_j + \mathsf{p}_1 + \mathsf{p}_0 w_j \right) \leq \mathsf{E}(1 - x_{ij}) \qquad \forall\, i, j \in \mathcal{R} \tag{8f}$$

$$\varepsilon y_j \leq \mathsf{r}_{\text{ch}} \tau_j \leq \mathsf{E} y_j \qquad \forall\, j \in \mathcal{R} \tag{8g}$$

$$e_j + \mathsf{r}_{\text{ch}} \tau_j \leq \mathsf{E} \qquad \forall\, j \in \mathcal{R} \tag{8h}$$

$$k_i - k_j + (M - 1)(x_{ij} + x_{ji}) \leq M - 1 \qquad \forall\, i, j \in \mathcal{R} \tag{8i}$$

$$t_j = \mathsf{t}_{\text{o},j}; \quad e_j = \mathsf{e}_{\text{o},j}; \quad k_j = \mathsf{k}_{\text{o},j} \qquad \forall\, j \in \mathcal{M} \tag{8j}$$

$$u_j w_j = 1 \qquad \forall\, j \in \mathcal{R} \tag{8k}$$

$$x_{ij}, y_j \in \{0, 1\} \qquad \forall\, i, j \in \mathcal{R} \tag{8l}$$

where E is the upper bound on the energy levels e_j, and T should be chosen larger than the latest expected pick-up time (conform the big-M method [7]). The constants r_{ch}, t_{ch}, and t_{de} represent the charging rate, the fixed charging time, and the disembarking plus embarking time respectively. The initial conditions for the pick-up times, the energy levels, and the index numbers of the micro-ferries are represented by $\mathsf{t}_{\text{o},j}$, $\mathsf{e}_{\text{o},j}$, and $\mathsf{k}_{\text{o},j}$ respectively.

Equalities (8b) are the assignment constraints ensuring that every request is handled once and only once, (8c) ensures consistency in the pick-up times, and (8d) assigns values to the slack variables s_j representing the misfit to the desired time-window $[\mathsf{t}_{\text{a},j}, \mathsf{t}_{\text{b},j}]$. Inequalities (8e) set the energy levels after delivery for request j equal to the energy level after delivery for request i, plus the charged energy during request i, minus the energy consumption during request j, when $x_{ij} = 1$. By using (8f) the energy consumption variables ϵ_j are enforced to satisfy

$$\epsilon_j = (\mathsf{p}_2 u_j + \mathsf{p}_1 + \mathsf{p}_2 w_j) \sum_{i \in \mathcal{R}} \mathsf{C}_{ij} x_{ij} \quad \forall\, j \in \mathcal{R}, \tag{9}$$

by providing an upper bound and lower bound that both equal the energy consumed when request i precedes request j, since then $x_{ij} = 1$. This method to enforce equality constraints such as (9) with multiplications between continuous variables (u_j and w_j) and binary variables (x_{ij}) is based on the methods described in [2]. Inequalities (8g), with $\varepsilon > 0$ the minimum amount of energy to charge, ensure the relationship $y_j = 0 \Leftrightarrow \tau_j = 0$, and (8h) prohibits the micro-ferries from over-charging. Due to (8i) each request is assigned a unique index number, and the initial conditions are set by (8j). Equality constraint (8k) forces w_j to be the reciprocal of u_j.

For the formulation above to hold the assumption is made that the maximum energy a single transportation request can cost will be smaller than the maximum energy a micro-ferry can contain. More specifically, for the constraints (8f) we assume that

$$\mathsf{C}_{ij} \left(\mathsf{p}_2 u_j + \mathsf{p}_1 + \mathsf{p}_0 w_j \right) \leq \max_{i,j \in \mathcal{R}} \mathsf{C}_{ij} \cdot \max\{ \mathsf{p}_2 \overline{u} + \mathsf{p}_1 + \mathsf{p}_0 \overline{u}^{-1};\ \mathsf{p}_2 \underline{u} + \mathsf{p}_1 + \mathsf{p}_0 \underline{u}^{-1} \} \leq \mathsf{E}, \tag{10}$$

where it is noted that the function $\mathsf{p}_2 u + \mathsf{p}_1 + \mathsf{p}_0 u^{-1}$ is convex for $\mathsf{p}_0, \mathsf{p}_1, \mathsf{p}_2 > 0$, and hence it reaches its maximum value at one of the bounds $\underline{u}$ or $\overline{u}$. Solving the optimisation problem (8) results in feasible schedules with respect to the energy levels. If initially the energy level of a micro-ferry is not sufficient to execute a request, it is either not scheduled to handle the request, or it is scheduled to charge before handling the request.

2.3 Speed approximation

Notice that (8k) is a non-linear constraint, as it contains the multiplication of two optimisation variables. The optimisation problem can be transformed into a mixed-integer linear programming (MILP) problem by approximating the speed $u_j > 0$ based on the value of $w_j > 0$, i.e., we want to approximate the convex function

$$u_j = w_j^{-1}. \tag{11}$$

This function can be approximated using multiple affine functions, where an arbitrary precision can be obtained by increasing the number of affine functions.

2.3.1 Piece-wise affine approximation

The first method to approximate (11) is by using a piece-wise affine (PWA) function, as is used in [3, 4]. Figure 1 shows an example where $u_j \in [1, 5]$ ($w_j \in [0.2, 1]$) with $P = 3$ segments.

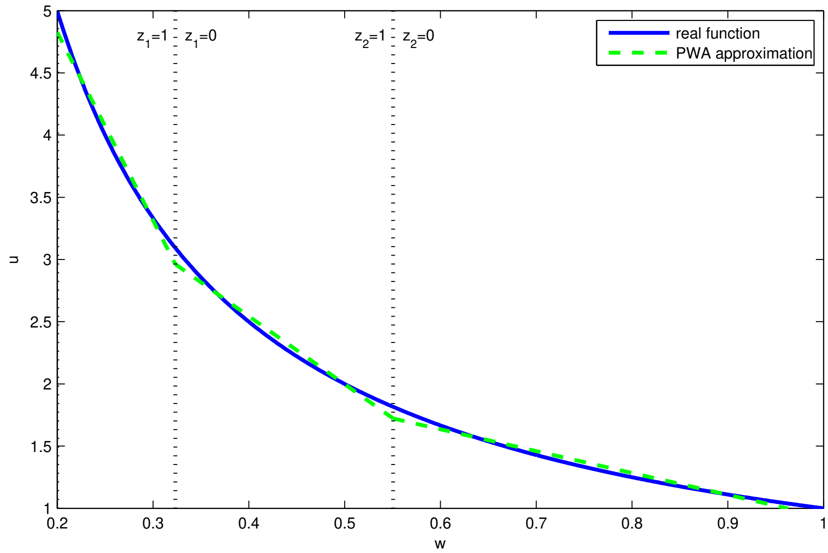

Figure 1: *Graphical example of approximating the speed function (11) by a PWA function*

Using ω_0 and ω_P to denote the lower and upper bound on w_j respectively, and constants scalars ω_p satisfying $\omega_p < \omega_{p+1}$ for all $p \in \{1, \ldots, P\text{–}1\}$ for an approximation with P segments, the approximation $\hat{u}_j$ of the speed u_j can be written as

$$\hat{u}_j = \begin{cases} \mathrm{a}_1 w_j + \mathrm{b}_1, & \omega_0 \leq w_j \leq \omega_1 \\ \quad\vdots & \quad\vdots \\ \mathrm{a}_P w_j + \mathrm{b}_P, & \omega_{P-1} \leq w_j \leq \omega_P \end{cases} \tag{12}$$

The constants $\omega_1, \ldots, \omega_{P-1}$, $\mathrm{a}_1, \ldots, \mathrm{a}_P$, and $\mathrm{b}_1, \ldots, \mathrm{b}_P$ can be found by minimising the error $u_j - \hat{u}_j$ in a least-squares sense. The PWA function (12) can be transformed into a single function by introducing $R \cdot P$ binary variables z_{jp} satisfying

$$[z_{jp} = 1] \;\Leftrightarrow\; [w_j \leq \omega_p]. \tag{13}$$

The relation (13) can be enforced by the linear constraints

$$w_j - \omega_p \leq \mathsf{W}(1 - z_{jp}) \;\; \forall j \in \{1, \ldots, R\}, p \in \{1, \ldots, P\}, \tag{14a}$$

$$\omega_p - w_j \leq \mathsf{W} z_{jp} \qquad \forall j \in \{1, \ldots, R\}, p \in \{1, \ldots, P\}, \tag{14b}$$

with $\mathsf{W} := \omega_P - \omega_0$. The PWA function (12) can then be written as

$$\begin{aligned}\hat{u}_j &= (\mathsf{A}_1 z_{j1} + \mathsf{A}_2 z_{j2} + \cdots + \mathsf{A}_P z_{jP}) w_j + (\mathsf{B}_1 z_{j1} + \mathsf{B}_2 z_{j2} + \cdots + \mathsf{B}_P z_{jP}), \\ &= (\mathsf{A}_1 w_j + \mathsf{B}_1) z_{j1} + \cdots + (\mathsf{A}_P w_j + \mathsf{B}_P) z_{jP} = \sum_{p=1}^{P} (\mathsf{A}_p w_j + \mathsf{B}_p) z_{jp},\end{aligned} \tag{15}$$

where $\mathsf{A}_1, \ldots, \mathsf{A}_P$ and $\mathsf{B}_1, \ldots, \mathsf{B}_P$ are constants given as

$$\mathsf{A}_p = \mathrm{a}_p - \mathrm{a}_{p+1} \quad \forall p \in \{1, \ldots, P-1\}, \qquad \mathsf{A}_P = \mathrm{a}_P, \tag{16a}$$
$$\mathsf{B}_p = \mathrm{b}_p - \mathrm{b}_{p+1} \quad \forall p \in \{1, \ldots, P-1\}, \qquad \mathsf{B}_P = \mathrm{b}_P. \tag{16b}$$

2.3.2 Constraint-based approximation

An alternative to the piece-wise affine approximation discussed above is to minimise the values of the optimisation variables u_j, constrained by equations that depend on the value of w_j. Figure 2 gives an example of this method, based on the same values as before.

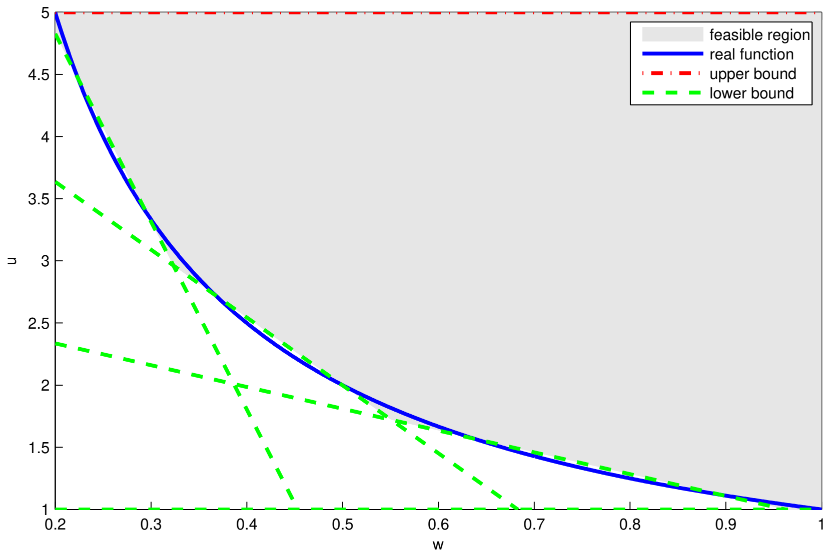

Figure 2: *Graphical example of approximating the speed by constraint minimisation*

To obtain the approximate value of u_j, we integrate the optimisation problem

$$\text{minimise} \quad \hat{u}_j \tag{17a}$$
$$\text{subject to} \quad \mathrm{a}_p w_j + \mathrm{b}_p \leq \hat{u}_j \quad \forall\, p \in \{1, \ldots, P\} \tag{17b}$$
$$\underline{u} \leq \hat{u}_j \leq \overline{u} \tag{17c}$$

with the original optimisation problem, where the parameters a_p and b_p are the same as defined in (12), and $\underline{u}$ and $\overline{u}$ denote the minimum and maximum allowed speed respectively.

Note: Since the variables $\hat{u}_j$ are already minimised due to the energy consumption term (3) —with the approximate speed $\hat{u}_j$ substituted for the real speed u_j— there is no need to add an extra objective function for minimising the variables of $\hat{u}_j$ as stated in (17a). Therefore, the objective functions for both the PWA-based approximation method and the constraint-based approximation method are the same, as will be the values for $\hat{u}_j$ since their minimum values —given a value of w_j— will coincide with the values from (12). For an optimal value of $\hat{u}_j$, the active lower bound for w_j in Figure 2 corresponds to the affine function that is used when $\omega_{p-1} \leq w_j \leq \omega_p$ in Figure 1.

3 Simulation Results

The method based on speed approximation using PWA functions as described in Section 2.3.1 adds binary variables to the optimisation problem, which in general increases the computation time. Furthermore, it requires the use of several additional constraints (see [4] for more details) that are based on the big-M method, notorious for its poor performance in combination with branch-and-bound methods as used in MILP solvers.

The constraint-based approximation method described in Section 2.3.2 does not introduce additional binary variables, and the constraints in (17) are expected to produce strong cuts (i.e. cuts that are efficient in branch-and-bound algorithms). Therefore, the constraint-based method is expected to solve problems faster than the PWA-based method.

To compare the computation time of both methods, the CPU time[2] it takes to solve both MILPs to optimality is determined for an example problem with 4 locations and 3 micro-ferries. The used requests with their associated pick-up and delivery location, the desired time-window, and the transportation distance, are given in Table 1.

Table 1: *Requests that need to be scheduled*

request	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
pick-up	s_1	s_4	s_4	s_1	s_4	s_2	s_2	s_4	s_2	s_1	s_3	s_1	s_2	s_3	s_4	s_2	s_1	s_4
delivery	s_1	s_2	s_3	s_2	s_3	s_1	s_4	s_2	s_1	s_3	s_4	s_3	s_3	s_4	s_2	s_1	s_4	s_2
$t_{\mathrm{a},j}$	-120	-60	0	0	180	240	360	720	720	660	780	960	1020	1080	1080	1440	1440	1500
$t_{\mathrm{b},j}$	-60	0	60	60	240	300	420	780	780	720	840	1020	1080	1140	1140	1500	1500	1560
transp. dist.	0	450	350	400	350	400	450	450	400	500	350	500	250	350	450	400	300	450

To determine the influence of the average number of requests per micro-ferry, and the number of sections P used to approximate (11), both parameters are varied. Figure 3 shows the resulting CPU times in seconds on a logarithmic scale; in the appendix more details on the obtained results are given. Simulations are stopped when reaching 1 hour of real computation time, visible in the figure as the black bars representing 10.000 [s].

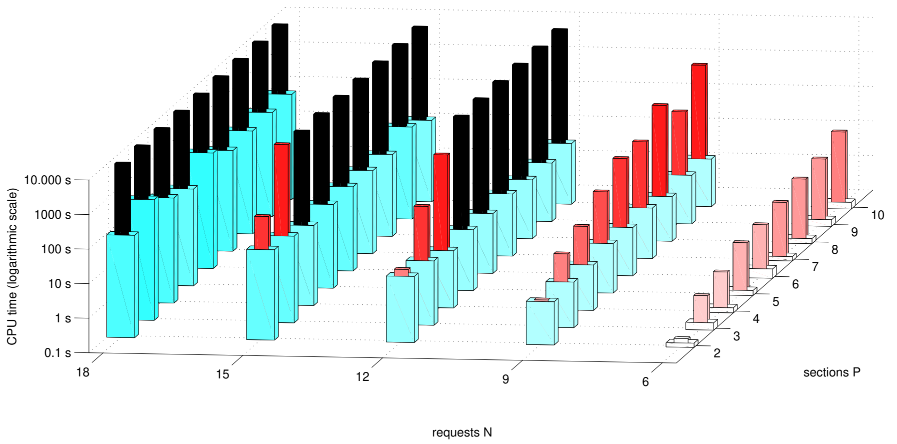

Figure 3: *CPU times for the constraint-based method (thick, blue bars) and PWA-based method (thin, red bars) when using different numbers of requests N and sections P*

The results show that using the constraint-based method results in improved computation times compared to the PWA-based method, up to several orders of magnitude. The former method never reaches the 1 hour time limit, whereas the latter method reaches it for larger numbers of requests and larger numbers of sections. The influence of adding more sections when using the constraint-based method is small; adding more requests results in increased CPU times for both methods.

[2]A desktop computer is used with an Intel Core2 Quad Q8400 processor and 4GB of RAM, running 64-bit versions of SUSE Linux Enterprise Desktop 11, MATLAB R2012a, and TOMLAB 7.8 using CPLEX 12.2.

4 Conclusions

A method for scheduling transportation requests, while taking into account the energy consumption of micro-ferries has been presented. By using the speed of a micro-ferry as an optimisation variable, the scheduler has the flexibility to minimise energy consumption while also serving the passengers on time. The resulting optimisation problem becomes non-linear, but two methods to approximate the problem (with arbitrary precision) by a mixed-integer linear program are provided.

The piece-wise affine approximation method introduces additional binary variables, resulting in a problem that is computationally expensive. The constrained-based approximation method does not introduce additional binary variables, while providing the same solution. Using the latter method, speed improvements of several orders of magnitude have been obtained. The speed approximation can be improved by using more segments to model it, without notable increases of computation times.

Appendix

Table 2 contains the values of the simulation results for the example discussed in Section 3. The results are ordered by the number of new requests N and the number of segments P of the speed approximation. The table contains several measures:

- J_{ec}: total energy consumption (3) in $[\text{W} \cdot \text{s}]$,
- J_{et}: total empty travel distance (6) in [m],
- J_{tt}: total travel time (7) in [s],
- J_{tw}: total time-window misfit (5) in [s],
- T_{r}: real simulation time in [s],
- T_{c}: CPU simulation time in [s],
- δ: binary variable indicating whether ($\delta = 1$) or not ($\delta = 0$) optimality is reached.

For the measures above the approximation method that is used is indicated by a superscript, using c for the constraint-based approach and p for the PWA-based approach. The differences between the two approaches while solving the same optimisation problem are indicated by the following error terms:

- $\tilde{x}$: difference in the schedule given by $\tilde{x} = \sum_{i\in\mathcal{R}} \sum_{j\in\mathcal{R}} |x^{\text{c}}_{ij} - x^{\text{p}}_{ij}|$,
- $\tilde{u}$: difference in the speeds given by $\tilde{u} = \sum_{j\in\mathcal{R}} |u^{\text{c}}_{j} - u^{\text{p}}_{j}|$,
- $\tilde{s}$: difference in time-window mismatch given by $\tilde{s} = \sum_{j\in\mathcal{R}} |s^{\text{c}}_{j} - s^{\text{p}}_{j}|$,
- $\tilde{t}$: difference in pick-up times given by $\tilde{t} = \sum_{j\in\mathcal{R}} |t^{\text{c}}_{j} - t^{\text{p}}_{j}|$.

The results show that *similar* results are obtained for both the PWA-based approach and the constraint-based approach; the cost functions are all the same whenever both methods reach the optimum (i.e. when $\delta^{\text{c}} = \delta^{\text{p}} = 1$), indicating that both methods reached a solution with the same, minimal value for the objective function. However, the optimum is not always unique, as can be seen by the non-zero error terms under $\tilde{x}$ and $\tilde{t}$.

Non-uniqueness of the variables x_{ij} for the optimal solution means that either the order of handling the requests can be changed or that some of the requests can be handled by different micro-ferries, without influencing the values of the objective functions. Non-uniqueness of the pick-up times is to be expected, as the values of the pick-up times do not influence the values of the objective functions, as long as the pick-up time is within its desired time-window. Since there is no preference for the pick-up time within the time-window, it is to be expected that the pick-up times can differ for each simulation run, even when using the same approach.

Table 2: *Simulation results for the example of Section 3*

N	P	$J_{\mathrm{ec}}^{\mathrm{c}}$	$J_{\mathrm{et}}^{\mathrm{c}}$	$J_{\mathrm{tt}}^{\mathrm{c}}$	$J_{\mathrm{tw}}^{\mathrm{c}}$	$T_{\mathrm{r}}^{\mathrm{c}}$	$T_{\mathrm{c}}^{\mathrm{c}}$	δ^{c}	$J_{\mathrm{ec}}^{\mathrm{p}}$	$J_{\mathrm{et}}^{\mathrm{p}}$	$J_{\mathrm{tt}}^{\mathrm{p}}$	$J_{\mathrm{tw}}^{\mathrm{p}}$	$T_{\mathrm{r}}^{\mathrm{p}}$	$T_{\mathrm{c}}^{\mathrm{p}}$	δ^{p}	$\tilde{x}$	$\tilde{u}$	$\tilde{s}$	$\tilde{t}$
6	2	309	1100	951	0	0	0	1	309	1100	951	0	0	1	1	8	0	0	120
6	3	311	1100	1039	0	0	1	1	311	1100	1039	0	1	3	1	0	0	0	120
6	4	311	1100	951	0	0	0	1	311	1100	951	0	2	4	1	8	0	0	120
6	5	311	1100	1011	0	0	0	1	311	1100	1011	0	3	7	1	0	0	0	10
6	6	311	1100	1004	0	0	1	1	311	1100	1004	0	3	8	1	0	0	0	120
6	7	311	1100	982	0	0	0	1	311	1100	982	0	3	10	1	0	0	0	120
6	8	312	1100	1019	0	0	0	1	312	1100	1019	0	5	15	1	8	0	0	126
6	9	312	1100	990	0	0	1	1	312	1100	990	0	6	17	1	0	0	0	60
6	10	312	1100	968	0	0	1	1	312	1100	968	0	9	30	1	0	0	0	60
9	2	394	950	1334	0	2	4	1	394	950	1334	0	2	5	1	0	0	0	185
9	3	397	950	1466	0	3	5	1	397	950	1466	0	9	25	1	0	0	0	120
9	4	397	950	1334	0	3	5	1	397	950	1334	0	14	45	1	0	0	0	272
9	5	397	950	1425	0	3	6	1	397	950	1425	0	34	122	1	8	0	0	143
9	6	397	950	1414	0	3	6	1	397	950	1414	0	83	308	1	0	0	0	77
9	7	397	950	1381	0	3	7	1	397	950	1380	0	81	298	1	0	0	0	104
9	8	397	950	1436	0	3	5	1	397	950	1436	0	247	898	1	0	0	0	185
9	9	397	950	1393	0	3	6	1	397	950	1407	0	64	228	1	0	0	0	157
9	10	397	950	1360	0	3	6	1	397	950	1360	0	343	1249	1	0	0	0	185
12	2	468	950	1642	0	8	17	1	468	950	1642	0	9	24	1	8	0	0	33
12	3	471	950	1815	0	7	15	1	471	950	1815	0	93	338	1	0	0	0	141
12	4	471	950	1642	0	5	10	1	471	950	1642	0	653	2438	1	8	0	0	231
12	5	471	950	1761	0	6	13	1	471	950	1761	0	3603	13446	0	0	0	0	190
12	6	471	950	1746	0	6	12	1	471	950	1746	0	3856	8086	0	0	0	0	183
12	7	471	950	1703	0	7	13	1	471	950	1703	0	3604	13348	0	8	0	0	184
12	8	472	950	1775	0	6	11	1	472	950	1775	0	3604	13820	0	8	0	0	125
12	9	471	950	1720	0	5	11	1	471	950	1720	0	3604	13429	0	8	0	0	210
12	10	472	950	1675	0	6	12	1	472	950	1675	0	3603	13894	0	0	0	0	73
15	2	572	1250	1965	0	25	69	1	572	1250	1965	0	120	444	1	8	0	0	300
15	3	577	1250	2176	0	20	55	1	577	1250	2183	0	2727	10513	1	0	0	0	112
15	4	577	1250	1966	0	15	38	1	577	1250	1966	0	3603	13498	0	0	0	0	180
15	5	577	1250	2114	0	18	47	1	577	1250	2114	0	3647	12788	0	8	0	0	284
15	6	577	1250	2096	0	20	48	1	577	1250	2096	0	3602	13520	0	8	0	0	164
15	7	577	1250	2042	0	18	45	1	577	1250	2041	0	3602	13492	0	8	0	0	224
15	8	578	1250	2130	0	16	41	1	578	1250	2132	0	3602	13868	0	0	0	0	186
15	9	578	1250	2063	0	26	75	1	578	1250	2063	0	3719	11590	0	8	0	0	146
15	10	578	1250	2007	0	17	41	1	578	1250	2007	0	3713	12041	0	0	0	0	145
18	2	684	1450	2412	0	48	137	1	684	1450	2412	0	3602	13917	0	0	0	0	279
18	3	689	1450	2695	0	117	398	1	689	1450	2695	0	3602	13558	0	0	0	0	277
18	4	689	1450	2413	0	53	160	1	689	1450	2413	0	3602	13606	0	0	0	0	240
18	5	689	1450	2606	0	35	99	1	689	1450	2606	0	3602	13516	0	8	0	0	340
18	6	689	1450	2582	0	90	294	1	689	1450	2582	0	3661	12170	0	8	0	0	124
18	7	690	1450	2512	0	41	124	1	690	1450	2511	0	3601	13281	0	0	0	0	307
18	8	690	1450	2629	0	47	142	1	690	1450	2629	0	3801	10749	0	0	0	0	240
18	9	690	1450	2539	0	51	152	1	690	1450	2539	0	3601	13282	0	0	0	0	374
18	10	690	1450	2466	0	51	160	1	690	1450	2467	0	3601	13469	0	0	0	0	319

References

[1] Bektaş, T. "The multiple traveling salesman problem: an overview of formulations and solution procedures", *Omega* 34(3), 209–219, 2006.

[2] Bemporad, A., and Morari, M. "Control of systems integrating logic, dynamics, and constraints", *Automatica* 35(3), 407–427, 1999.

[3] Burger, M., De Schutter, B., and Hellendoorn, H. "Micro-ferry scheduling problem with time-windows", in *Proc. of the 2012 American Control Conference*, Montréal, Canada, 3998–4003, 2012.

[4] Burger, M., De Schutter, B., and Hellendoorn, H. "Micro-ferry scheduling problem with charging and embarking times", in *Proc. of the IFAC Symposium on Control in Transportation Systems*, Sofia, Bulgaria, 2012.

[5] Daganzo, C. "Fundamentals of transportation and traffic operations", Pergamon Press, United Kingdom, 1997.

[6] Laporte, G. "The traveling saleman problem: an overview of exact and approximate algorithms", *European Journal of Operational Research* 20(1), 231–247, 1992.

[7] Taha, H.A., "Operations research: An introduction", Macmillan Publishing Company, New York (USA), 4th edition, 1987.