A customized dual ascent algorithm for a class of traffic coordination problems

Greyson Daugherty and Spyros Reveliotis
School of Industrial & Systems Engineering
Georgia Institute of Technology

Greg Mohler
Georgia Tech Research Institute
Georgia Institute of Technology

Abstract

Lagrangian relaxation has been pursued as a pertinent methodology for many hard scheduling problems. From standard duality theory, the method is known to provide good-quality bounds to the performance of the optimal solutions of these problems, and in certain cases, the optimal Lagrange multipliers can function as a starting point for the construction good suboptimal solutions for the original (primal) problem. In this work we consider the application of the method on a class of novel scheduling problems that concern the routing of a set of mobile agents over the edges of an underlying guidepath network. Besides the introduction of these new scheduling problems and the proposed Lagrangian relaxations, our primary contribution is an "ascent" algorithm for the corresponding dual problem that is characterized by (i) monotonic improvement of the generated solutions and (ii) finite convergence to an optimal solution.

1 Introduction

The basic problem that motivates the research results that are presented in this paper can be briefly stated as follows: Given a set of agents that circulate on the edges of a connected graph, to be called the "guidepath network" in the following, we want to coordinate the traffic of these agents so that (i) they move from their current edges to some destination edges in the minimum possible time, and at the same time (ii) the generated motion observes a set of rules regarding the edge occupancy by these agents. A typical example of such guidepath-based traffic is provided by the zone-controlled automated guided vehicle (AGV) systems that are used in many modern production and distribution facilities [8]. In a typical configuration of such a system, a fleet of mobile robots is used to facilitate the material transfers that must take place among a set of locations in the underlying facility. The transported material is organized in "standard" loads, and each vehicle can carry one load at a time. Also, for safety and other reasons, the vehicle motion is constrained into certain "corridors" that are defined either physically or virtually,[1] and all these corridors are linked together into the corresponding "guidepath network". Furthermore, in order to avoid any collisions among the AGVs, the links of the "guidepath" network are split up into "zones", and it is required that, at any time point, each zone is occupied by at most one vehicle. This requirement refines the original guidepath network to a larger graph with similar overall topology but with the graph nodes and edges defined by the specified zones and their connectivity. Also, it introduces a need for controlled allocation of the zones of the resulting guidepath network to the traversing vehicles. In addition, in the emergent operational regime, it is physically impossible for any two

[1] e.g., by some sort of painted strip or duct tape that must be scanned by the robot during its motion, or by radio signals that "guide" the robot.

vehicles to “swap” the occupation of two neighboring zones.[2] Finally, in most topologies of the underlying guidepath network, the motion restrictions that are described above can give rise to deadlocks among the circulating AGVs, and raise the need for an appropriate deadlock avoidance policy (DAP) [17].

The operational characteristics and the traffic coordination problems that were described in the previous paragraph for the current AGV systems extend to any other unit-load, guidepath-based automated material-handling system that is employed in the contemporary industrial sectors, like (a) the over-head monorail systems that are used in the current semiconductor fabs [19], and (b) the complex crane systems that are used in many ports and freight terminals [8]. But our work is also motivated more directly by another application that arises in the context of quantum computing [14]. This application concerns the physical medium and operations that implement the elementary logical operations and the corresponding “(logical) gates” in quantum computing, especially as they are represented in the GTRI Quantum Machine Parameterizer (QMP). In this computational paradigm, information is stored in “qubits”, i.e., in discrete elements that obey quantum mechanics (e.g. electron states in trapped ions), and the processing of this information is performed by the transfer of the corresponding qubits in certain locations where they will have their states changed, possibly through their interaction with other qubits that are brought simultaneously to the same location. However, the storage and the transfer of those qubits during this entire operation must be done in a very controlled manner that preserves the quantum state of the qubits involved. An abstracted model of the ion trap implementation is taken as the base case, where ions are steered around an ion trap maze, but it is easily extendable to other technologies. Operations are thus described as storing and guiding ions through a sequence of “ion traps”, that essentially isolate the considered atoms from the rest of their operational environment. Furthermore, it should be obvious that qubits cannot coexist uncontrollably in any of the provided ion traps, since any such uncontrollable coexistence can lead to the uncontrollable interaction between these ions and to the eventual loss of the information stored in them. For similar reasons, the physical swapping of ions is taken as not physically possible, although this constraint can be relaxed in certain cases. Hence, these traffic systems also enforce strict exclusivity and other constraints in the allocation of the various zones (in this case, the ion traps) of the guidepath network to the traveling agents, which can give rise to deadlocking effects, as in the case of the AGV traffic networks.

Furthermore, in, both, the AGV operational setting and that of the GTRI Quantum Machine Parameterizer, the moving agents typically execute mulit-leg trips that require them to visit certain “milestone” edges of the guidepath network in a certain sequence, and possibly interact at those edges with other agents, before they continue with the subsequent legs of their trips. In our work, these multi-leg trips will be coordinated by a “receding horizon” / model predictive control (MPC) scheme [10], that decomposes naturally the overall routing and scheduling problem to a sequence of subproblems that are defined by the immediate destinations of the traveling agents. Hence, the problem described in the opening part of this section constitutes the core subproblem to be repeatedly solved in this MPC framework. Following standard MPC practices,[3] we propose to address this last problem through a deterministic mathematical programming (MP) formulation. Furthermore, the regularity / uniformity of (i) the zones employed in the AGV control setting, and of (ii) the ion traps that constitute the guidepath links in the QMP framework, also enable a time-discretizing approach based on the time that is required for the traversal of any of these guidepath links by the system agents. The resulting MP formulation that characterizes the aforementioned scheduling (sub-)problem for the QMP operational context, is a mixed integer program (MIP) [13]

[2] The physical infeasibility of such a swapping is especially obvious in the case of neighboring zones corresponding to the same original corridor and occupied by vehicles moving in opposite directions in those zones.

[3] These practices result from the notion of “certainty equivalence” in MPC theory [3].

that is presented in detail in the next section.

However, the main focus of this paper is the solution of the (Lagrangian) dual problem [5] of the aforementioned formulation, that is defined by the Lagrangian relaxation of some of the most difficult constraints of the original MIP. From a practical standpoint, the capability to address effectively the dual problem of the considered MIP formulation is significant since (a) the obtained solutions provide bounds for the optimal value of the original MIP formulation, and (b) they might also contain useful information for the synthesis of heuristic (near-)optimal solutions for the original problem. From a more analytical standpoint, our main result is a dual ascent algorithm that is customized to the considered MP formulation and the derived dual problem, and it is further characterized by (i) a monotonic improvement of the solutions for the dual problem that are generated by the algorithm iterations, and by (ii) finite convergence to an optimal solution of this last problem.

In view of the above positioning of our work, the rest of the paper is organized as follows: Section 2 provides a detailed description of the traffic coordination problem that was described at the beginning of this introductory section, and proceeds to the analytical characterization of this problem by means of a mixed integer programming (MIP) formulation. Section 3 introduces the proposed Lagrangian relaxation for the MIP formulation of Section 2, defines the corresponding dual problem, and develops a complete and streamlined methodology for the evaluation of the dual function. Section 4 presents the proposed algorithm for the dual problem that is formulated in Section 3, and establishes the properties of this algorithm that were mentioned in the previous paragraph. Section 5 presents the application of the developed algorithm on an example problem instance, and discusses the algorithm performance on this instance as well as some faced challenges. Section 6 provides some further discussion on the presented results, including the positioning of the presented developments in the context of the existing literature. Finally, Section 7 concludes the paper and discusses directions for future work.

2 The basic scheduling problem considered in this work: Detailed problem description and a MIP formulation

2.1 A detailed description of the considered scheduling problem

In this section we provide a detailed description of the particular scheduling problem that is the focus of this work. For better motivation, but also for more specificity and concreteness, we present this problem as it materializes in the operational context of the QMP operations that were discussed in the introductory section. We remind the reader that, in this setting, we move quantum bits (or "qubits"), manifested as physical particles, in a guidepath network composed of ion traps. These particles must be sent to specific destinations in order to interact with one another, to receive external inputs, or to be stored for later use.

Hence, to proceed with a more analytical formulation of the considered traffic coordination problem, as a first step, we will model the guidepath network as a connected, directed graph $\mathcal{G} = (V, E)$, where each edge $e \equiv (i, j) \in E$, connecting vertex i to vertex j, is assumed to be of unit length. As explained in Section 1, the assumption that all edges $e \in E$ have identical length results from the uniformity of the ion traps that constitute the basic links of the guidepath network, and facilitates the discretization of time in the eventual formulation of the considered problem.[4]

[4] We also notice, for completeness, that the physical edges of the considered guidepath networks do not possess any sense of direction. Since, however, we need to capture the sense of direction in the motion of the agents traversing these edges, we model each edge of the guidepath network by a *pair* of directed edges of opposite connectivity.

The circulating qubits themselves are modeled as a set of agents, $\mathcal{A}$, with each agent $a \in \mathcal{A}$ being located on a particular edge at any given time. Furthermore, each agent a has an immediate destination edge d_a associated with it, and the set collecting the destinations for all agents will be denoted by E^D. In the considered QMP context, E^D includes locations where gate operations can be performed, as well as storage zones.

We assume that every agent $a \in \mathcal{A}$ traverses any given edge $e \in E$ in a consistent length of time, to be considered as the *unit* time in the following. Besides the uniformity of the edge lengths that was discussed in the previous paragraphs, the uniformity of the traversal time of the guidepath edges by the traveling agents is also based on the further assumption that, *when in motion* (i.e., when not blocked or holding at a particular position), each agent moves at an identical, deterministic speed.[5]

We also assume that, unless they share a common destination location,[6] agents cannot cohabitate on a single edge, and they cannot pass one another by swapping edges. And lastly, we will assume that an agent a can pass through a destination edge $e \in E^D$ that is not assigned to it; more formally, if an edge e is an element of E^D but $e \neq d_a$, agent a treats edge e as a normal transit edge.

Given these assumptions, we seek to route all agents $a \in \mathcal{A}$ to their respective destinations d_a in minimal time; that is, we seek (i) a selection of traveling routes for these agents, and (ii) a schedule that will coordinate the agent advancement on their corresponding routes, such that the total time, w, that will take for each each agent a to reach its destination d_a, is minimized.[7]

2.2 Formulation of the considered problem as an Integer Program

Next, we proceed to express the scheduling problem that was defined in Section 2.1 as a MIP formulation. We provide this formulation by detailing, in separate subsections, the problem data and the involved parameters, the decision variables, the constraints, and the objective of the formulation.

2.2.1 Notation

— $V = \{v_1, v_2, ..., v_m\}$: Guidepath-graph vertices

— $E = \{e_1, e_2, ..., e_n\}$, with $e_l = (v_i, v_j)\ \forall l \in \{1, ..., n\}$: Guidepath-graph edges (or "zones")

— $\mathbb{T}$ $(n \times n)$: A binary matrix expressing the agent transitional dynamics on the guidepath graph; $\mathbb{T}_{i,j} = 1$ iff a direct transition from e_i to e_j is allowed. We set $\mathbb{T}_{i,i} = 1\ \forall i$ s.t. $e_i \in E$

— $\bar{e} = (v_j, v_i)$: Complementary edge of edge $e = (v_i, v_j)$

— ${}^{\bullet}e_l = \{e_q \in E : \mathbb{T}_{q,l} = 1 \land q \neq l\}$: The set of input edges for e_l, $\forall e_l \in E$

— $e_l^{\bullet} = \{e_q \in E : \mathbb{T}_{l,q} = 1 \land q \neq l\}$: The set of output edges for e_l, $\forall e_l \in E$

— $\mathcal{A} = \{a_1, a_2, ..., a_K\}$: The set of traveling agents

[5] In the QMP operational setting, this assumption is naturally justified by the basic physics that govern the ion transport through the ion traps that define the guidepath network. In the case of the AGV systems, vehicle speeds sometimes can be dependent on the carrying loads, but given the (typically) small length of the guidepath edges, the effects of these speed variations can be considered as negligible.

[6] In the QMP setting, the corresponding edges are essentially "(logical) gates" that implement specific elementary operations in the underlying computation.

[7] We also emphasize, for completeness, that here we consider only the specification and formulation of the basic subproblem that will be addressed in the context of the broader MPC framework that was indicated in Section 1.

— d_a: Destination edge for agent $a \in \mathcal{A}$

— $E^D = \{e_j \in E : \exists a \in \mathcal{A} \text{ s.t. } d_a = e_j\}$: The set of the current / immediate destinations for all agents

— s_a: Starting edge for agent $a \in \mathcal{A}$; this edge also specifies initial orientation for the agent motion

— T: An upper bound on the required completion time (and therefore a bound on the optimal value of the objective function)

— $t \in \mathcal{T} = \{0, 1, ..., T\}$: Time index

2.2.2 Decision Variables

— $\forall a \in \mathcal{A}$, $\forall e \in E$, $\forall t \in \mathcal{T}$, $x_{a,e,t} \in \{0,1\}$ indicates whether, in the derived schedule, agent a is located on the directed edge e at timestep t; these decision variables constitute the primary decision variables of the scheduling problem. Also, for notational convenience, in the following, we shall denote by $\boldsymbol{x}$ the vector that collects all the variables $x_{a,e,t}$.

— w: An auxiliary variable that will represent the total time to completion – i.e., the "makespan" – of the optimal schedule.

2.2.3 Constraints

1. Agents must occupy exactly one and only one position at any time period.
   $\forall a \in \mathcal{A}, \forall t \in \mathcal{T}, \quad \sum_{e \in E} x_{a,e,t} = 1$

2. Every agent $a \in \mathcal{A}$ has some initial position, designated by s_a.
   $\forall a \in \mathcal{A}, \quad x_{a,s_a,0} = 1$
   $\forall a \in \mathcal{A}, \forall e \in E \text{ with } e \neq s_a, \quad x_{a,e,0} = 0$

3. Every agent must reach its desitnation edge, d_a, within the provided time horizon.
   $\forall a \in \mathcal{A}, \quad x_{a,d_a,T} = 1$

4. Agents cannot leave their destination nodes after reaching them.
   $\forall a \in \mathcal{A}, \forall t \in \mathcal{T} \setminus \{T\}, \quad x_{a,d_a,t+1} \geq x_{a,d_a,t}$

5. Agents can only transition to adjacent, directionally compatible edges.
   $\forall a \in \mathcal{A}, \forall e \in E, \forall t \in \mathcal{T} \setminus \{T\}, \quad x_{a,e,t} \leq x_{a,e,t+1} + \sum_{e' \in e^\bullet} x_{a,e',t+1}$

6. No two agents may cohabitate on an edge that is not a destination edge.
   $\forall e = (v_i, v_j) \in E \setminus E^D \text{ s.t. } i < j, \forall t \in \mathcal{T}, \quad \sum_{a \in \mathcal{A}} (x_{a,e,t} + x_{a,\bar{e},t}) \leq 1$

7. An agent can only access an edge if that edge was empty during the previous cycle, unless this edge is its destination; an edge of the guidepath network cannot have one agent transitioning into it and another agent transitioning out at the same instant.
   $\forall a \in \mathcal{A}, \forall e \in E \setminus d_a, \forall e' \in e^\bullet \setminus \{d_a\}, \forall t \in \mathcal{T} \setminus \{T\},$
   $$x_{a,e,t} + x_{a,e',t+1} \;\leq\; 2 - \sum_{a' \in \mathcal{A}, a' \neq a} (x_{a',\bar{e'},t} + x_{a',e',t})$$

8. This constraint is linking the auxiliary variable w to the primary decision variables of the problem, $x_{a,e,t}$; when coupled with the objective of this formulation, this constraint ensures that w expresses the makespan of any optimal solution that is returned by the considered formulation.
$\forall a \in \mathcal{A}, \quad w \;\geq\; \sum_{t=0}^{T}(1 - x_{a,d_a,t}) \;=\; T + 1 - \sum_{t=0}^{T} x_{a,d_a,t}$

9. All decision variables, apart from w, are binary.
$\forall a \in \mathcal{A}$, $\forall e \in E$, $\forall t \in \mathcal{T}, \quad x_{a,e,t} \in \{0,1\}$

2.2.4 Objective Function

$$\min w$$

We seek to minimize the makespan w of the derived schedule, under the constraints of Section 2.2.3. That is, we will try to minimize the necessary time for the considered set of agents to reach their very next destinations, given their current positions.

3 A Lagrangian relaxation of the considered IP formulation and the corresponding dual problem

The problem described in Section 2.1 has considerable conceptual affinity with the classical "job shop" scheduling problem [15] that is known to be NP-hard. In particular, in the considered operational setting, the role of the "machines" is played by the edges of the guidepath network, while the traveling agents correspond to the "jobs" running through these "machines". Furthermore, our scheduling problem is characterized by (possibly extensive levels of) routing flexibility for each agent, and also by implicit precedence constraints that are induced by the fact that certain destination edges d_a might constitute necessary "stepping stones" for some other agents $a'(\neq a)$ in their trips to their destinations. Based on these remarks, it is expected that finding a strictly optimal solution for the considered scheduling problem will be computationally intractable for any instances of significant size.[8] This realization motivates the development of some heuristic method able to provide good-quality schedules for this problem but not necessarily optimal. In such an endeavor, it is desirable (i) to exploit the "structure" of the original problem formulation as a "guide" in the search for promising solutions, and (ii) to have a means for assessing the quality of the derived solutions w.r.t. the (generally unknown) optimal schedule. As pointed out in Section 1, MP Duality theory can assist along both of these lines by (a) providing lower bounds to the optimal value of the original MIP formulation, and possibly by (b) providing additional information, by means of the generated dual optimal solutions, that can be useful for the synthesis of the heuristic solutions that were suggested above. Hence, in this section we shift attention to a dual formulation of the MIP formulation of Section 2 that is obtained by relaxing some of the harder constraints of that MIP formulation.

[8] The intractability of the considered scheduling problem can be argued more formally on the basis of a result that is presented in [16] and essentially implies that testing the feasibility of the considered scheduling problem is NP-complete. Therefore, the scheduling problem itself must be NP-hard (since, otherwise, feasibility could be assessed by solving the scheduling problem itself).

3.1 The proposed Lagrangian relaxation and the corresponding dual problem

We start the corresponding developments by noticing that, in the constraint set of Section 2.2.3, Constraints (1) through (5), together with Constraint (9) that enforces the binary nature of the primary decision variables, seek to shape up the behavior of each agent $a \in \mathcal{A}$ in isolation. Characteristically, it is not hard to see that if the MIP formulation of Section 2.2 involved only the aforementioned constraints, the optimal route for any agent $a \in \mathcal{A}$ w.r.t. the objective of Section 2.2.4 would be any "shortest path" from the current location of the agent, s_a, to its destination location, d_a, with the relevant distances measured in terms of the uniform length of the zones in the guidepath graph. Hence, if the MIP formulation of Section 2.2 involved only the aforementioned constraints, it would be solvable through a decomposition of the original problem to the aforementioned "shortest path" problems [2], one for each agent $a \in \mathcal{A}$, which are known to be of polynomial complexity w.r.t. to the "size" of the guidepath graph.[9] But, in reality, things are complicated by the presence of the additional Constraints (6) through (8). These constraints express the additional limitations that are imposed on the agents' schedules due to their interaction through the limited buffering capacity of the edges of the guidepath graph, and, for that reason, they are characterized as the "coupling" constraints of the formulation. Furthermore, it should be evident from the above discussion that these coupling constraints are the primary reason for the very high complexity of the considered scheduling problem.

In the relaxed formulation that is pursued in this section, the complexity that is incurred by Constraints (6)–(8) is mitigated by expressing the logic communicated by these constraints in the MIP formulation of Section 2.2, in a "softened" (or, more "relaxed") manner, in particular, as additional cost to the problem objective that arises from the violation of these constraints. Furthermore, the way that any violation of these relaxed constraints impacts the original objective function is moderated through the association of a "weight" with each of these constraints, that acts as a "(unit) price" for the corresponding violation. The new objective function that results from such a modification of the original objective function is known as the corresponding "Lagrangian" function, and the aforementioned "weights" that are associated with the relaxed constraints are the "Lagrange multipliers" for these constraints [5].

Next, we proceed with a detailed characterization of the Lagrangian function for the MIP formulation of Section 2.2 that results from the relaxation that was suggested in the previous paragraph. We start by defining the three vectors of the Lagrange multipliers that correspond to the three primary subsets of the coupling constraints:

— $\boldsymbol{\lambda}$: vector of Lagrange multipliers for constraint set (6)

— $\boldsymbol{\mu}$: vector of Lagrange multipliers for constraint set (7)

— $\boldsymbol{\nu}$: vector of Lagrange multipliers for constraint set (8)

The reader should notice that the constraint sets (6)–(8) are defined by ranging a certain subset of the problem indices over a particular subset of their possible values; for example, the constraints in constraint set (6) are obtained by ranging the edge index e over the edge set $\{(v_i, v_j) \in E \setminus E^D : i < j\}$ and the time index t over the corresponding index set $\mathcal{T}$. In the following, we shall refer to the elements of the vectors $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ by indexing these elements by the index values that generate the corresponding constraint; hence, the elements of vector $\boldsymbol{\lambda}$ will be represented by $\lambda_{e,t},\ e \in \{(v_i, v_j) \in E \setminus E^D : i < j\},\ t \in \mathcal{T}$. Under this notational convention, the corresponding

[9] For the needs of this discussion, the "size" of a graph can be defined by the numbers of the nodes and the edges in the graph.

Lagrangian function can be written as follows:

$$\begin{aligned}L(\boldsymbol{x}, w; \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \equiv w + \sum_{\{e\in E: v_i < v_j\}} \sum_{t\in\mathcal{T}} \lambda_{e,t} \left[\sum_{a\in\mathcal{A}} (x_{a,e,t} + x_{a,\bar{e},t}) - 1 \right] + \\ + \sum_{a\in\mathcal{A}} \sum_{e\in E\setminus E^D} \sum_{e'\in e^\bullet\setminus\{d_a\}} \sum_{t\in\mathcal{T}\setminus\{T\}} \mu_{a,e,e',t} \left[x_{a,e,t} + x_{a,e',(t+1)} + \sum_{\{a'\in\mathcal{A}: a'\neq a\}} (x_{a',e',t} + x_{a',\bar{e}',t}) - 2 \right] + \\ + \sum_{a\in\mathcal{A}} \nu_a \left[T + 1 - w - \sum_{t\in\mathcal{T}} x_{a,d_a,t} \right]\end{aligned} \tag{1}$$

It is evident that in order to have the last three terms in equation 1 acting as “penalties” for the violation of the corresponding constraints, we need to have

$$\boldsymbol{\lambda} \geq \boldsymbol{0}; \quad \boldsymbol{\mu} \geq \boldsymbol{0}; \quad \boldsymbol{\nu} \geq \boldsymbol{0} \tag{2}$$

Then, for any arbitrary pricing of the Lagrange multipliers $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, the “relaxed” version of the MIP formulation of Section 2.2 can be expressed as follows:

$$\begin{aligned}\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \equiv \min_{\boldsymbol{x}, w} \Bigg\{ w + \sum_{\{e\in E: v_i < v_j\}} \sum_{t\in\mathcal{T}} \lambda_{e,t} \left[\sum_{a\in\mathcal{A}} (x_{a,e,t} + x_{a,\bar{e},t}) - 1 \right] + \\ + \sum_{a\in\mathcal{A}} \sum_{e\in E\setminus E^D} \sum_{e'\in e^\bullet\setminus\{d_a\}} \sum_{t\in\mathcal{T}\setminus\{T\}} \mu_{a,e,e',t} \left[x_{a,e,t} + x_{a,e',(t+1)} + \sum_{\{a'\in\mathcal{A}: a'\neq a\}} (x_{a',e',t} + x_{a',\bar{e}',t}) - 2 \right] + \\ + \sum_{a\in\mathcal{A}} \nu_a \left[T + 1 - w - \sum_{t\in\mathcal{T}} x_{a,d_a,t} \right] \Bigg\}\end{aligned} \tag{3}$$

s.t. the constraint sets (1), (2), (3), (4), (5) and (9).

Function $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ is known as the “dual” function for the original MIP formulation of Section 2.2 (under the particular selection of the relaxed constraints) [5]. Furthermore, since the minimization problem in the right-hand-side (rhs) of Eq. 3, that defines this function, is a relaxation of the original MIP formulation, it follows that function $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ provides a lower bound to the optimal value for the MIP formulation of Section 2.2, for any values of its variables $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$. Naturally, we are interested in obtaining the tightest possible lower bound that can be provided by $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$, by solving the following optimization problem:

$$\max_{\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}} \theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \tag{4}$$

s.t. Eq. (2).

The above MP formulation is known as the “dual” problem in the corresponding Duality theory, while the original problem and the corresponding MIP formulation of Section 2.2 are characterized as the “primal” [5]. Let $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ denote an optimal solution for the dual problem. Then, besides providing the tightest possible lower bound for the original MIP formulation of Section 2.2 that can be obtained from the dual function, in some cases, the availability of the vector $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ can convey useful information for obtaining a (near-)optimal solution for the primal problem. This information is conveyed in vector $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ itself, and also in the vectors $\hat{\boldsymbol{x}}$ that constitute optimal solutions for the corresponding minimization problem in the rhs of Eq. 3, that defines the optimal value $\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ for the dual problem.

Next, we proceed to reveal additional structure in the Lagrangian function of Eq. 1 that will prove particularly useful for the efficient evaluation of the dual function $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$, for any given vector $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$, and for the solution of the corresponding dual problem.

3.2 The separability of the dual function and its implications

3.2.1 A revised characterization of the dual function

The Lagrangian function $L(\boldsymbol{x}, w; \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$, defined in Eq. 1, is an affine linear function of the primary decision variables $\boldsymbol{x}$, w. To reveal more clearly this structure, next we rewrite function L by collecting its various terms so that no primary decision variable is shown in the resulting expression more than once:

$$
\begin{aligned}
&L(\boldsymbol{x}, w; \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) = \\
&\quad \sum_{a \in \mathcal{A}} \nu_a (T+1) - \left[\left(\sum_{\{e \in E : v_i < v_j\}} \sum_{t \in \mathcal{T}} \lambda_{e,t} \right) + \left(\sum_{a \in \mathcal{A}} \sum_{e \in E \setminus E^D} \sum_{e' \in e^{\bullet} \setminus \{d_a\}} \sum_{t \in \mathcal{T} \setminus T} 2\mu_{a,e,e',t} \right) \right] + \\
&\quad + w\Big(1 - \sum_{a \in \mathcal{A}} \nu_a\Big) + \sum_{a \in \mathcal{A}} \Bigg\{ \sum_{e \in E \setminus E^D} \sum_{t \in \mathcal{T} \setminus \{0,T\}} \Bigg[\lambda_{e,t} + \sum_{e' \in e^{\bullet} \setminus \{d_a\}} \mu_{a,e,e',t} + \sum_{e' \in {}^{\bullet}e \setminus E^D} \mu_{a,e',e,t-1} + \\
&\quad\quad + \sum_{\{a' \in \mathcal{A} : a' \neq a\}} \sum_{e' \in {}^{\bullet}e \setminus E^D} (\mu_{a',e',e,t} + \mu_{a',e',\bar{e},t}) \Bigg] x_{a,e,t} + \sum_{e \in E \setminus E^D} \Bigg[\lambda_{e,0} + \sum_{e' \in e^{\bullet} \setminus \{d_a\}} \mu_{a,e,e',0} + \\
&\quad\quad\quad \sum_{\{a' \in \mathcal{A} : a' \neq a\}} \sum_{e' \in {}^{\bullet}e \setminus E^D} (\mu_{a',e',e,0} + \mu_{a',e',\bar{e},0}) \Bigg] x_{a,e,0} + \sum_{e \in E \setminus E^D} \lambda_{e,T} x_{a,e,T} - \sum_{t \in \mathcal{T}} \nu_a x_{a,d_a,t} \Bigg\} \qquad (5)
\end{aligned}
$$

Then, setting

$$
\begin{aligned}
\Delta_{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}} &\equiv -\left[\left(\sum_{\{e \in E : v_i < v_j\}} \sum_{t \in \mathcal{T}} \lambda_{e,t} \right) + 2 \left(\sum_{a \in \mathcal{A}} \sum_{e \in E \setminus E^D} \sum_{e' \in e^{\bullet} \setminus \{d_a\}} \sum_{t \in \mathcal{T} \setminus T} \mu_{a,e,e',t} \right) \right]; && (6) \\
C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} &\equiv \lambda_{e,t} + \sum_{e' \in e^{\bullet} \setminus \{d_a\}} \mu_{a,e,e',t} + \sum_{e' \in {}^{\bullet}e \setminus E^D} \mu_{a,e',e,t-1} + \sum_{\{a' \in \mathcal{A} : a' \neq a\}} \sum_{e' \in {}^{\bullet}e \setminus E^D} (\mu_{a',e',e,t} + \mu_{a',e',\bar{e},t}), \\
&\quad \forall a \in \mathcal{A},\ \forall e \in E \setminus E^D,\ \forall t \in \mathcal{T} \setminus \{0, T\}; && (7) \\
C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,0} &\equiv \lambda_{e,0} + \sum_{e' \in e^{\bullet} \setminus \{d_a\}} \mu_{a,e,e',0} + \sum_{\{a' \in \mathcal{A} : a' \neq a\}} \sum_{e' \in {}^{\bullet}e \setminus E^D} (\mu_{a',e',e,0} + \mu_{a',e',\bar{e},0}), \\
&\quad \forall a \in \mathcal{A},\ \forall e \in E \setminus E^D; && (8) \\
C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,T} &\equiv \lambda_{e,T},\ \forall a \in \mathcal{A},\ \forall e \in E \setminus E^D; && (9) \\
C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,d_a,t} &\equiv -\nu_a,\ \forall a \in \mathcal{A},\ \forall t \in \mathcal{T}, && (10)
\end{aligned}
$$

we obtain the following expression for the Lagrangian function:

$$
\begin{aligned}
L(\boldsymbol{x}, w; \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \; = & \\
& \sum_{a\in\mathcal{A}} \nu_a(T+1) + \Delta_{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}} + w(1 - \sum_{a\in\mathcal{A}} \nu_a) \; + \sum_{a\in\mathcal{A}} \Bigg\{ \sum_{e\in E\setminus E^D} \; \sum_{t\in\mathcal{T}\setminus\{0,T\}} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} x_{a,e,t} + \\
& \quad + \sum_{e\in E\setminus E^D} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,0} x_{a,e,0} + \sum_{e\in E\setminus E^D} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,T} x_{a,e,T} + \sum_{t\in\mathcal{T}} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,d_a,t} x_{a,d_a,t} \Bigg\} = \\
& \quad\quad = \sum_{a\in\mathcal{A}} \nu_a(T+1) + \Delta_{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}} + w(1 - \sum_{a\in\mathcal{A}} \nu_a) \; + \sum_{a\in\mathcal{A}} \sum_{e\in E} \sum_{t\in\mathcal{T}} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} x_{a,e,t} \qquad (11)
\end{aligned}
$$

Also, from Eq. (11), we get the following representation of the dual function:

$$
\begin{aligned}
\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \; \equiv \; & (T+1) \sum_{a\in\mathcal{A}} \nu_a + \Delta_{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}} \; + \\
& + \min_{\boldsymbol{x}, w} \Big\{ w(1 - \sum_{a\in\mathcal{A}} \nu_a) + \sum_{a\in\mathcal{A}} \sum_{e\in E} \sum_{t\in\mathcal{T}} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} x_{a,e,t} \Big\} \text{ s.t. Primal Constraints (1)-(5) and (9)} \qquad (12)
\end{aligned}
$$

In the above definition of the dual function, a complication arises from the fact that the (primal) variable w is a free variable in the MIP formulation of Section 2.2. This implies that for those cases where $(1 - \sum_{a\in\mathcal{A}} \nu_a) \neq 0$, the minimization problem that appears in the second line of Eq. 12 is ill-defined, since we can always drive the optimal objective value to $-\infty$ by setting w to an arbitrarily large or small value depending on the sign of the difference $(1 - \sum_{a\in\mathcal{A}} \nu_a)$. To avoid this complication, we restrict the formulation of the dual problem by adding the constraint

$$
\sum_{a\in\mathcal{A}} \nu_a = 1 \qquad (13)
$$

Under this condition, the dual function $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ can be written as

$$
\begin{aligned}
\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \; \equiv \; & (T+1) + \Delta_{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}} \; + \\
& \sum_{a\in\mathcal{A}} \left[\min_{\boldsymbol{x}_a} \Big\{ \sum_{e\in E} \sum_{t\in\mathcal{T}} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} x_{a,e,t} \Big\} \text{ s.t. Primal Constraints (1)-(5) and (9)} \right] \qquad (14)
\end{aligned}
$$

where the vector $\boldsymbol{x}_a$ collects all the primary variables $x_{a,e,t}$ with fixed index a. Furthermore, the corresponding dual problem reduces to:

$$
\max_{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}} \tilde{\theta}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \;\; \equiv \;\; \theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \; - \; (T \; + \; 1), \;\; \text{s.t.} \;\; \boldsymbol{\lambda} \geq 0, \;\; \boldsymbol{\mu} \geq 0, \;\; \boldsymbol{\nu} \geq 0, \;\; \sum_{a\in A} \nu_a = 1 \qquad (15)
$$

3.2.2 An efficient evaluation of the dual function

We shall address the solution of the dual problem of Eq. (15) in a following section. In this section we address the problem of evaluating the dual function $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ at any given point $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$, based on the definition of this function that is provided in Eq. (14). A nice feature of this definition is that the minimization problem involved in the evaluation of function $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ decomposes to a number of subproblems, with each subproblem concerning the routing of a single agent $a \in \mathcal{A}$

and involving only the corresponding primary variables $x_{a,e,t}$.[10] More specifically, each of these subproblems can be expressed by the following integer program:

$$\min_{x_{a,e,t}} \Big\{ \sum_{e \in E} \sum_{t \in \mathcal{T}} C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} x_{a,e,t} \Big\}, \quad \text{s.t. the Primal Constraints (1)–(5) and (9)} \tag{16}$$

In order to understand the nature of feasibility and optimality for the above MIP formulation, the following remarks are in order: First of all, Constraint (3) in this formulation implies that any feasible solution for it must have agent a eventually placed at its destination edge d_a. At the same time, the remaining (primal) constraint sets (1), (2), (4), (5) and (9) ensure that any feasible solution can be interpreted as a "route" from the starting edge s_a to the destination edge d_a that is realizable in the underlying guidepath graph.

Furthermore, the reader should notice that the non-negativity of the Lagrange multipliers that is enforced by Eq. (15), together with the definition of the various cost coefficients in Eqs (6)–(10), imply that all the immediate cost accrued during the execution of any of the aforementioned feasible routes will be nonnegative, except for the immediate cost corresponding to a state where the agent has reached its destination edge d_a; indeed, the coefficients that correspond to this particular state are all equal to $-\nu_a$ and, therefore, they constitute an immediate "gain" that is collected by the considered agent at any time point that it is located in its destination edge. Hence, the optimal routes in a feasible formulation of Eq. (16) establish a trade-off between the agent's desire to reach the destination edge d_a as soon as possible and start collecting its gain, and the need to avoid any routes that might get the agent to its destination fast but also accrue a very high cost (manifested by the values of the corresponding cost coefficients $C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t}$). This realization suggests the re-interpretation of the MIP formulation of Eq. (16) as a dynamic program (DP) [4] with the following value function $V_a(e,t)$:

For $t = T$:

$$V_a(e,t) = \begin{cases} -\nu_a & \text{if } e = d_a \\ \infty & \text{otherwise} \end{cases} \tag{17}$$

For $t \in \{0, 1, 2, ..., T-1\}$:

$$V_a(e,t) = \begin{cases} -\nu_a + V_a(e,t+1) & \text{if } e = d_a \\ C^{\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\nu}}_{a,e,t} + \min\limits_{e' \in e^{\bullet} \cup \{e\}} \{V_a(e',t+1)\} & \text{if } e \in E \setminus d_a \end{cases} \tag{18}$$

According to standard DP theory, the value function $V_a(e,t)$ can be interpreted as the total cost (including any collected gain) that will be accumulated by the traveling agent a if it starts from edge e at time t and tries to reach its destination edge d_a by the specified time T while following an optimal routing policy. In particular, the presence of the '∞' value in Eq. (17), that provides the definition of $V_a(e,T)$, ensures that any optimal solution to the considered DP will never fail to place the agent in its destination edge, if there exists such a route that is realizable within time T. If this is not the case, the DP solver will return an infinite optimal cost for the considered problem, and this result implies that either T must be extended, or there is some condition that prohibits the agent from reaching its destination (i.e., the connectivity of the guidepath graph renders some subset of its edges inaccessible to the considered agent).

For feasible problem instances, the value $V_a(s_a, 0)$ will be finite, and an optimal route can be obtained by the, so called, "greedy" policy. This policy, at time t, routes agent a in a way that

[10] The careful perusal of the previous developments will reveal that the separability that is present in the definition of the dual function of Eq. (14) is the result of (i) the selection of the constraints to be relaxed and (ii) the linearity of these constraints in terms of the primary decision variables $x_{a,e,t}$.

minimizes the restriction $V_a(\cdot, t+1)$ of the corresponding value function. Generally, it is possible that there are more than one optimal routes, all sharing the same optimal cost $V_a(s_a, 0)$. In such a case, it is also possible to obtain a compact representation of all the optimal routes in the form of an acyclic digraph $\mathcal{G}_a$ consisting of edges labeled by pairs (e, t) for some subset of $E \times \mathcal{T}$. Digraph $\mathcal{G}_a$ can be constructed very efficiently by a "forward reaching" algorithm that starts from edge $(s_a, 0)$ and, for every visited edge (e, t), it processes that edge by appending to its end (and to the constructed graph) all the edges $(e', t+1)$ that are the minimizers in the evaluation of $V_a(e, t)$ through the second branch in the rhs of Eq. (18). The algorithm will terminate its exploration in any of the, thus, constructed paths, once the destination edge d_a has been reached; at that point, the corresponding path will be completed with the appendage of the subpath consisting of the labeled edges (d_a, t) where t spans from the current time to the end of the considered time horizon T.

Having developed a complete methodology for the evaluation of the dual function and for the solution of the corresponding minimization problem, next we turn to the solution of the dual problem that is defined in Eq. (15).

4 Solving the dual problem

4.1 An efficient representation of the dual problem and its implications for the problem structure and its solution

In this section we shall develop an algorithm for the solution of the dual problem of Eq. (15). In order to address this issue, we start by taking a closer look at the structure of the maximized function $\tilde{\theta}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$. From Eqs (3), (13) and (15), this function can be written as follows:

$$\begin{aligned}\tilde{\theta}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \equiv \min_{\boldsymbol{x}} \Bigg\{ & \sum_{\{e\in E: v_i<v_j\}} \sum_{t\in\mathcal{T}} \lambda_{e,t} \Bigg[\sum_{a\in\mathcal{A}} (x_{a,e,t} + x_{a,\bar{e},t}) - 1 \Bigg] + \\ & + \sum_{a\in\mathcal{A}} \sum_{e\in E\setminus E^D} \sum_{e'\in e^\bullet\setminus\{d_a\}} \sum_{t\in\mathcal{T}\setminus\{T\}} \mu_{a,e,e',t} \Bigg[x_{a,e,t} + x_{a,e',(t+1)} + \sum_{\{a'\in\mathcal{A}: a'\neq a\}} (x_{a',e',t} + x_{a',\bar{e}',t}) - 2 \Bigg] + \\ & - \sum_{a\in\mathcal{A}} \nu_a \sum_{t\in\mathcal{T}} x_{a,d_a,t} \Bigg\}\end{aligned} \tag{19}$$

s.t. the constraint sets (1), (2), (3), (4), (5) and (9).

From the discussion of the previous sections, we know that, for every vector $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) \geq \mathbf{0}$ and a sufficiently large time length T, the minimization problem that appears in the rhs of Eq. (19) is well-defined, having a finite optimal value and a finite set of optimal solutions. More specifically, these optimal solutions are defined by any selection among the optimal routes that constitute the optimal solutions for the MIP formulations of Eq. (16) that are defined for each agent $a \in \mathcal{A}$. Let $\{\boldsymbol{x}_a,\ a \in \mathcal{A}\}$ denote such a selection of optimal routes for the various agents, and $\boldsymbol{x}$ denote the vector that results from the concatenation of the vectors $\boldsymbol{x}_a$. Also, for the economy of the subsequent discussion, let us set $\boldsymbol{\pi} \equiv (\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$, and denote by m the dimensionality of this vector. Furthermore, for any $i \in \{1, \ldots, m\}$, we shall denote by $g_i(\boldsymbol{x})$ the quantity that multiplies the Lagrange multiplier $\boldsymbol{\pi}[i]$ in the rhs of Eq. (19), and we shall also use the notation $\boldsymbol{g}(\boldsymbol{x})$ for the vector that collects all the $g_i(\boldsymbol{x})$ for $i = 1, \ldots, m$.

From the above definitions it is evident that, for any given vector $\boldsymbol{\pi}$, the set of the optimal vectors $\boldsymbol{x}$ for the minimization problem of Eq. (19) is a finite set; let us consider an arbitrary enumeration of its elements, say $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$. This induces the enumeration $\boldsymbol{g}_j,\ j = 1, \ldots, k$, with $\boldsymbol{g}_j \equiv \boldsymbol{g}(\boldsymbol{x}_j)$. Let us also denote by X the entire set of vectors $\boldsymbol{x}$ that satisfy the constraint sets (1), (2), (3), (4), (5) and (9), i.e., the set of vectors that constitute feasible solutions to the MP formulation that defines function $\tilde{\theta}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$. Then, under the previously introduced notation, function $\tilde{\theta}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ can be expressed as follows:

$$\tilde{\theta}(\boldsymbol{\pi}) \ = \ \min_{\boldsymbol{x} \in X} \{\boldsymbol{\pi}^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \tag{20}$$

Since the elements of the set X are binary vectors, this set is finite. And the fact that $\tilde{\theta}(\boldsymbol{\pi})$ is defined as the minimum of a finite set of linear functions in $\boldsymbol{\pi}$ implies that it is a concave polyhedral function.[11] This last remark, when combined with the structure of the constraints in Eq. (15), imply that the dual problem of Eq. (15) concerns the maximization of a concave function over a convex set. Hence, our problem falls in a class of rather easy problems in optimization theory [5], but its solution is complicated by the fact that the objective function $\tilde{\theta}(\boldsymbol{\pi})$ is not differentiable at every point of the feasible region; in particular, it is not differentiable at the points where the minimization problem in the rhs of Eq. (20) will have many optimal solutions. At these points, one must work with the local "subgradients" of the function in order to identify an improving direction – or a "direction of ascent" – for any optimizing algorithm.

4.2 A brief introduction to subgradients and subdifferentials for concave functions and their role in the characterization of the maximal points of these functions

In this section we review some key results pertaining to the notion of "subgradient" and its role in convex optimization; the presented discussion is based on the corresponding material in [5]. We start by reminding the reader that, for any concave function $f : \ \mathbb{R}^m \to \mathbb{R}$, a vector $\boldsymbol{d} \in \mathbb{R}^m$ is a subgradient of f at a point $\boldsymbol{x} \in \mathbb{R}^m$ if

$$\forall \boldsymbol{z} \in \mathbb{R}^m, \quad f(\boldsymbol{z}) \leq f(\boldsymbol{x}) + (\boldsymbol{z} - \boldsymbol{x})^T \cdot \boldsymbol{d} \tag{21}$$

The above definition of the subgradients of f further implies that a vector $\boldsymbol{d}$ is a subgradient for a concave function $f : \ \mathbb{R}^m \to \mathbb{R}$ at some point $\boldsymbol{x} \in \mathbb{R}^m$ if and only if (iff) for every directional derivative $f'(\boldsymbol{x}; \boldsymbol{y})$ of f at $\boldsymbol{x}$ along some direction $\boldsymbol{y}$, it holds that

$$f'(\boldsymbol{x}; \boldsymbol{y}) \leq \boldsymbol{y}^T \cdot \boldsymbol{d} \tag{22}$$

The set of all the subgradients for the concave function f at $\boldsymbol{x} \in \mathbb{R}^m$ is called the subdifferential of f at $\boldsymbol{x}$ and it is denoted by $\partial(\boldsymbol{x})$. The subdifferential $\partial(\boldsymbol{x})$ is a nonempty, convex and compact set, and for any directional derivative $f'(\boldsymbol{x}; \boldsymbol{y})$ it holds that

$$f'(\boldsymbol{x}; \boldsymbol{y}) = \min_{\boldsymbol{d} \in \partial f(\boldsymbol{x})} \boldsymbol{y}^T \cdot \boldsymbol{d} \tag{23}$$

In particular, f is differentiable at $\boldsymbol{x}$ with gradient $\nabla f(\boldsymbol{x})$ iff it has $\nabla f(\boldsymbol{x})$ as the unique subgradient at $\boldsymbol{x}$. Then, it follows from Eqs (22) and (23) that, for every direction $\boldsymbol{y}$,

$$f'(\boldsymbol{x}; \boldsymbol{y}) = \boldsymbol{y}^T \cdot \nabla f(\boldsymbol{x}) \tag{24}$$

Finally, point $\boldsymbol{x}$ maximizes f over a convex set $X \subset \mathbb{R}^m$ iff there exists a subgradient $\boldsymbol{d} \in \partial f(\boldsymbol{x})$ such that

$$\forall \boldsymbol{z} \in X, \quad \boldsymbol{d}^T \cdot (\boldsymbol{z} - \boldsymbol{x}) \leq 0 \tag{25}$$

[11] In fact, concavity is a more general property of dual functions [5].

4.3 Characterizing the subgradients and the subdifferential of $\tilde{\theta}(\boldsymbol{\pi})$

Returning to the concave function $\tilde{\theta}(\boldsymbol{\pi})$ that is defined by Eq. (20), consider any vector $\tilde{\boldsymbol{\pi}}$ satisfying the constraints of Eq. (15) and let the vector $\tilde{\boldsymbol{x}}$ be one of the minimizers of the rhs of Eq. (20). Then, it holds that for all $\boldsymbol{\pi} \in \mathbb{R}^m$,

$$\begin{aligned} \tilde{\theta}(\boldsymbol{\pi}) &= \min_{\boldsymbol{x} \in X} \{\boldsymbol{\pi}^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \\ &\leq \boldsymbol{\pi}^T \cdot \boldsymbol{g}(\tilde{\boldsymbol{x}}) \\ &= \tilde{\boldsymbol{\pi}}^T \cdot \boldsymbol{g}(\tilde{\boldsymbol{x}}) + (\boldsymbol{\pi} - \tilde{\boldsymbol{\pi}})^T \cdot \boldsymbol{g}(\tilde{\boldsymbol{x}}) \\ &= \tilde{\theta}(\tilde{\boldsymbol{\pi}}) + (\boldsymbol{\pi} - \tilde{\boldsymbol{\pi}})^T \cdot \boldsymbol{g}(\tilde{\boldsymbol{x}}) \end{aligned} \tag{26}$$

Eq. (26) together with Eq. (21) imply that the vector $\boldsymbol{g}(\tilde{\boldsymbol{x}})$ is a subgradient of function $\tilde{\theta}(\boldsymbol{\pi})$ at $\tilde{\boldsymbol{\pi}}$. In fact, it can be shown that the subdifferential of $\tilde{\theta}(\boldsymbol{\pi})$ at $\tilde{\boldsymbol{\pi}}$ is given by

$$\partial\tilde{\theta}(\tilde{\boldsymbol{\pi}}) = \Big\{ \tilde{\boldsymbol{g}} : \ \tilde{\boldsymbol{g}} = \sum_{j=1}^{k} \alpha_j \tilde{\boldsymbol{g}}_j; \ \alpha_j \geq 0, \ \sum_{j=1}^{k} \alpha_j = 1 \Big\} \tag{27}$$

where, according to the previously introduced convention, the vectors $\tilde{\boldsymbol{g}}_j$ are induced by the set of the minimizers $\{\tilde{\boldsymbol{x}}_j, \ j = 1, \ldots, k\}$ of the expression $\min_{\boldsymbol{x} \in X}\{\tilde{\boldsymbol{\pi}}^T \cdot \boldsymbol{g}(\boldsymbol{x})\}$.

Furthermore, in the considered case, the functions $g_i(\boldsymbol{x})$ that define the components of the vector function $\boldsymbol{g}(\boldsymbol{x})$, for all $i = 1, \ldots, m$, are linear affine functions of $\boldsymbol{x}$; i.e., each $g_i(\boldsymbol{x})$ can be written as

$$g_i(\boldsymbol{x}) = \boldsymbol{p}_i^T \cdot \boldsymbol{x} + q_i \tag{28}$$

where the corresponding parameters $(\boldsymbol{p}_i, q_i)$ are determined from Eq. (19). Hence, the subdifferential of Eq. (27) can be rewritten as:

$$\begin{aligned} \partial\tilde{\theta}(\tilde{\boldsymbol{\pi}}) &= \Big\{ \tilde{\boldsymbol{g}} : \ \forall i = 1, \ldots, m, \ \ \tilde{\boldsymbol{g}}[i] = \sum_{j=1}^{k} \alpha_j (\boldsymbol{p}_i^T \cdot \boldsymbol{x}_j + q_i); \ \alpha_j \geq 0, \ \sum_{j=1}^{k} \alpha_j = 1 \Big\} \\ &= \Big\{ \tilde{\boldsymbol{g}} : \ \forall i = 1, \ldots, m, \ \ \tilde{\boldsymbol{g}}[i] = \boldsymbol{p}_i^T \cdot \sum_{j=1}^{k} (\alpha_j \boldsymbol{x}_j) + q_i; \ \alpha_j \geq 0, \ \sum_{j=1}^{k} \alpha_j = 1 \Big\} \end{aligned} \tag{29}$$

Eq. (29) implies that, at any point $\boldsymbol{\pi}$, the subgradients of the function $\tilde{\theta}(\boldsymbol{\pi})$ are defined by the convex hull [5] of the minimizers $\boldsymbol{x}_j, \ j = 1, \ldots, k$, of the expression $\min_{\boldsymbol{x} \in X}\{\boldsymbol{\pi}^T \cdot \boldsymbol{g}(\boldsymbol{x})\}$, through the mapping that is defined by the vector function $\boldsymbol{g}(\boldsymbol{x})$. In the following, we shall denote the convex hull of a given set of points S by $Conv(S)$. Hence, a more concise expression of Eq. (29) is as follows:

$$\partial\tilde{\theta}(\boldsymbol{\pi}) = \Big\{ \boldsymbol{g}(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in Conv\Big(\arg\min_{\boldsymbol{x} \in X} \{\boldsymbol{\pi}^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \Big) \Big\} \tag{30}$$

One complication for the representations of the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$ that is provided by Eqs (27) and (29) arises from the fact that the cardinality of the vector set $\{\boldsymbol{x}_j, \ j = 1, \ldots, k\}$ can be very large. To demonstrate this effect, for any given vector $\boldsymbol{\pi}$, let us denote the coresponding vector set $\{\boldsymbol{x}_j, \ j = 1, \ldots, k\}$ more concisely by $X(\boldsymbol{\pi}) \subseteq X$. From the previous discussion, we know that the elements of the set $X(\boldsymbol{\pi})$ can be perceived as a selection of an optimal route for each agent $a \in \mathcal{A}$ for the corresponding subproblem of Eq. (16). These optimal routes for any given agent a can be represented by an appropriate pricing of the binary vector $\boldsymbol{x}_a$ that collects the components of the vector $\boldsymbol{x}$ corresponding to agent a; let us denote by $X_a(\boldsymbol{\pi})$ the set of vectors $\boldsymbol{x}_a$ that correspond to optimal routes for agent a under $\boldsymbol{\pi}$. Then $X(\boldsymbol{\pi}) = \prod_{a\in\mathcal{A}} X_a(\boldsymbol{\pi})$, and, therefore, the representational complexity of the set $X(\boldsymbol{\pi})$ is exponential w.r.t. $|\mathcal{A}|$.

Fortunately, this complexity problem can be circumvented by providing a more distributed representation for the sets $X_a(\boldsymbol{\pi})$, $a \in \mathcal{A}$, and for their product set $X(\boldsymbol{\pi})$. This new representation is obtained by capitalizing upon the graphical representation of the sets $X_a(\boldsymbol{\pi})$ that was described in the closing part of Section 3.2.2, that dealt with the computation of the corresponding routes. We remind the reader that according to that representation, the set $X_a(\boldsymbol{\pi})$ is represented by an acyclic connected digraph $\mathcal{G}_a(\boldsymbol{\pi})$ with a single "source" edge labeled by $(s_a, 0)$ and a single "terminal" edge labeled by (d_a, T). Every other edge of the digraph $\mathcal{G}_a(\boldsymbol{\pi})$ is labeled by (e, t) for some $e \in E$ and some t with $0 < t < T$, indicating that for some optimal route in $X_a(\boldsymbol{\pi})$, agent a will occupy the edge e of the guidepath graph at time t. Moreover, it is easy to see that the immediate successors of an edge labeled by (e, t) in $\mathcal{G}_a(\boldsymbol{\pi})$ will be some edges labeled $(e', t+1)$ and with $e' \in e^{\bullet} \cup \{e\}$. As already remarked in Section 3.2.2, any optimal route in $X_a(\boldsymbol{\pi})$ is represented in $\mathcal{G}_a(\boldsymbol{\pi})$ as a path leading from the source edge $(s_a, 0)$ to the terminal edge (d_a, T). Even more interestingly for the needs of the subsequent developments, every vector in $Conv(X_a(\boldsymbol{\pi}))$ can be represented as a "flow" [2] transferring a unit of fluid from the source edge of $\mathcal{G}_a(\boldsymbol{\pi})$ to its terminal edge. Similarly, the convex hull of the set $X(\boldsymbol{\pi})$ is represented by a set of flows, one for each graph $\mathcal{G}_a(\boldsymbol{\pi})$, $a \in \mathcal{A}$, and therefore, independent from each other.

A detailed representation of the aforementioned flows for any given digraph $\mathcal{G}_a(\boldsymbol{\pi})$, $a \in \mathcal{A}$, can be obtained as follows: Let $\mathcal{N}_a(\boldsymbol{\pi})$ denote the "transshipment" nodes of the digraph $\mathcal{G}_a(\boldsymbol{\pi})$, $a \in \mathcal{A}$, i.e., all the nodes of this graph except for the "tail" node of the edge $(s_a, 0)$ and the "head" node of the edge (d_a, T). Also, for every node $n \in \mathcal{N}_a(\boldsymbol{\pi})$, denote by $IN(n)$ the set collecting its incoming edges and by $OUT(n)$ the set collecting its outgoing edges. Furthermore, associate the nonnegative real variables $q_a(e, t)$ with the edges of the graph $\mathcal{G}_a(\boldsymbol{\pi})$. Then, the considered flows for agent a are expressed by any pricing of the variables $\{q_a(e, t)\}$ that satisfy the following set of constraints:

$$q_a(s_a, 0) \;=\; 1.0 \tag{31}$$

$$\forall n \in \mathcal{N}_a(\boldsymbol{\pi}), \quad \sum_{(e,t)\in OUT(n)} q_a(e, t) \;-\; \sum_{(e,t)\in IN(n)} q_a(e, t) \;=\; 0.0 \tag{32}$$

Indeed, it is easy to see that each constraint in Eq. (32) expresses a "flow balance" condition for the corresponding node $n \in \mathcal{N}_a(\boldsymbol{\pi})$. On the other hand, Eq. (31) expresses the fact that the total amount of flow conveyed through the graph $\mathcal{G}_a(\boldsymbol{\pi})$ is equal to one unit.

For any solution of Eqs (31) and (32), we obtain an element $\tilde{\boldsymbol{x}}_a(\boldsymbol{\pi})$ of the set $\tilde{X}_a(\boldsymbol{\pi}) \equiv Conv(X_a(\boldsymbol{\pi}))$ by setting

$$\begin{aligned} &\tilde{\boldsymbol{x}}_{a,s_a,0}(\boldsymbol{\pi}) = \tilde{\boldsymbol{x}}_{a,d_a,T}(\boldsymbol{\pi}) = 1; \\ &\quad \tilde{\boldsymbol{x}}_{a,e,t}(\boldsymbol{\pi}) = q_a(e, t) \text{ for every intermediate edge } (e, t) \text{ of the digraph } \mathcal{G}_a(\boldsymbol{\pi}); \\ &\qquad \tilde{\boldsymbol{x}}_{a,e,t}(\boldsymbol{\pi}) = 0 \text{ otherwise.} \end{aligned} \tag{33}$$

The representation of the set $Conv(X(\boldsymbol{\pi})) \equiv \tilde{X}(\boldsymbol{\pi})$ involves the employment of a set of Eqs (31) and (32) for each agent $a \in \mathcal{A}$, with an independent set of variables for each such system of equations, and the construction of the corresponding vectors $\tilde{\boldsymbol{x}}_a(\boldsymbol{\pi})$ from these variables through the logic expressed by Eq. (33); then, $\tilde{X}(\boldsymbol{\pi}) = \prod_{a\in\mathcal{A}} \tilde{X}_a(\boldsymbol{\pi})$.

Finally, Eqs (19), (29), (30) and (33) imply that, for every element $\tilde{\boldsymbol{x}} \in \tilde{X}(\boldsymbol{\pi})$, the corresponding element $\boldsymbol{g}(\tilde{\boldsymbol{x}}) \in \partial\tilde{\theta}(\boldsymbol{\pi})$ can be obtained as follows:

$$\forall e \in E \text{ with } v_i < v_j, \quad \forall t \in \mathcal{T}, \quad g(e, t; \tilde{\boldsymbol{x}}) \;:=\; \sum_{a\in\mathcal{A}} (\tilde{\boldsymbol{x}}_{a,e,t} + \tilde{\boldsymbol{x}}_{a,\bar{e},t}) - 1 \tag{34}$$

$$\forall a \in \mathcal{A}, \ \ \forall e \in E \setminus E^D, \ \ \forall e' \in e^\bullet \setminus \{d_a\}, \ \ \forall t \in \mathcal{T} \setminus \{T\},$$
$$g(a, e, e', t; \tilde{\boldsymbol{x}}) \ := \ \tilde{\boldsymbol{x}}_{a,e,t} + \tilde{\boldsymbol{x}}_{a,e',(t+1)} + \sum_{\{a' \in \mathcal{A}: a' \neq a\}} (\tilde{\boldsymbol{x}}_{a',e',t} + \tilde{\boldsymbol{x}}_{a',\bar{e'},t}) - 2 \tag{35}$$

$$\forall a \in \mathcal{A}, \quad g(a; \tilde{\boldsymbol{x}}) \ := \ \sum_{t \in \mathcal{T}} \tilde{\boldsymbol{x}}_{a,d_a,t} \tag{36}$$

Next, we employ the above characterization of the subdifferential of the function $\tilde{\theta}(\boldsymbol{\pi})$ in order to develop a subgradient-based method for the optimization of the MP of Eq. (15). The presented method constitutes an "ascent" method for the considered optimization problem [5]. We break down the presentation of the proposed algorithm into two parts: The first part concerns the provision of an optimality test and a method for generating directions of improvement at any point $\boldsymbol{\pi}$ that is visited by the algorithm. The second part provides a methodology that determines the size of the steps to be taken by the algorithm in those improving directions so that the incurred improvements are maximized.

4.4 The optimality test and the generator of improving directions employed by the proposed algorithm

In the context of the dual problem that is defined by Eq. (15), Eq. (25) implies that, at any optimal solution $\tilde{\boldsymbol{\pi}}$ for this problem, the corresponding subdifferential $\partial\tilde{\theta}(\tilde{\boldsymbol{\pi}})$ contains either (i) the $\mathbf{0}$ vector, or (ii) a subgradient $\tilde{\boldsymbol{g}} \neq \mathbf{0}$ that is perpendicular to the hyperplane defined by the constraint $\sum_{a \in A} \nu_a = 1$, or, finally, (iii) a subgradient $\tilde{\boldsymbol{g}} \neq \mathbf{0}$ that has non-zero projection on the hyperplane $\sum_{a \in A} \nu_a = 1$, but for any $\boldsymbol{\pi}$ that is feasible w.r.t. the constraints of Eq. (15), it holds that $\tilde{\boldsymbol{g}}^T \cdot (\boldsymbol{\pi} - \tilde{\boldsymbol{\pi}}) \leq 0$. Next, we provide a linear program (LP) [11] that tests this optimality condition at any feasible point $\boldsymbol{\pi}$ of the dual problem of Eq. (15).

More specifically, this LP is constructed in a way that its optimal value will be equal to zero if the aforementioned optimality condition is satisfied at $\boldsymbol{\pi}$, and strictly positive otherwise. The decision variables and the constraints of this LP (i) will provide a characterization of the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$ by means of Eqs (31)–(36), and, together with the structure of the objective function, (ii) will determine the optimal value of this function so that it reflects the testing logic that was defined above. With these specifications in mind, a detailed description of the considered LP is as follows:

The column vector collecting the variables of the considered LP is the vector $\boldsymbol{\psi} \equiv [\boldsymbol{q}^T \boldsymbol{y}^T \boldsymbol{z}^T \delta]^T$, where: The vector $\boldsymbol{q}$ collects all the nonnegative real variables $q_a(e,t)$ that appear in the flow characterizations of Eqs (31)–(32), for all $a \in \mathcal{A}$, and it will be employed in the representation of the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$. On the other hand, the vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ are nonnegative and of dimensionality m; i.e., their dimensionality is equal to the number of the relaxed constraints of the original MIP formulation of Section 2.2, and thus, equal to the dimensionality of the vectors $\boldsymbol{\pi}$ and $\boldsymbol{g}$. Finally, the variable δ is a free scalar. Collectively, $\boldsymbol{y}$, $\boldsymbol{z}$ and δ provide a set of real variables that will facilitate the expression of the test logic that is implemented by the considered LP.

The first set of the "technological" constraints in the considered LP takes the form:

$$\forall a \in \mathcal{A}, \quad F_a(\boldsymbol{\pi}) \cdot \boldsymbol{q}_a \ = \ \boldsymbol{\beta}^1_a(\boldsymbol{\pi}) \tag{37}$$

Eq. (37) is meant as a matrix-based representation of the content of Eqs (31)–(32), where the information that is provided by Eq. (31) has been inserted, through substitution, in Eq. (32). Hence, the variable vector $\boldsymbol{q}_a$ collects the "flow" variables $q_a(e,t)$ corresponding to a single agent

$a \in \mathcal{A}$. The matrix $F_a(\boldsymbol{\pi})$ has one row for every "transshipment" node n of the corresponding digraph $\mathcal{G}_a(\boldsymbol{\pi})$, and this row acts as a "constituency" vector for the set of edges in $\mathcal{G}_a(\boldsymbol{\pi})$ that are incident to node n. Furthermore, the sign of the non-zero (unit) elements of this row indicates whether the corresponding edge is an incoming or an outgoing edge w.r.t. node n, with the positive sign indicating an outgoing edge. However, edge $(s_a, 0)$ is excepted from this representation since, as mentioned above, the flow of this edge is already priced to 1.0 and it does not constitute a free variable. In a similar spirit, the rhs vector $\boldsymbol{\beta}_a^1(\boldsymbol{\pi})$ in the above equation has all its components equal to zero, except for the component that corresponds to the single transshipment node n that is incident to the edge $(s_a, 0)$; this component is equal to 1.0. For the following developments, it is also useful to collect all the constraints that are expressed by Eq. (37) in the single structure

$$F(\boldsymbol{\pi}) \cdot \boldsymbol{q} \;=\; \boldsymbol{\beta}^1(\boldsymbol{\pi}) \tag{38}$$

where the matrix $F(\boldsymbol{\pi})$ is block-diagonal with its diagonal blocks being the matrices $F_a(\boldsymbol{\pi})$ of Eq. (37), and the vectors $\boldsymbol{q}$ and $\boldsymbol{\beta}^1(\boldsymbol{\pi})$ are obtained from the concatenation of the corresponding sets of vectors $\{\boldsymbol{q}_a\}$ and $\{\boldsymbol{\beta}_a^1(\boldsymbol{\pi})\}$, for $a \in \mathcal{A}$.

The second set of the "technological" constraints employed by the considered LP involves the representation of the elements $\boldsymbol{g}$ of the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$ that is based on the "flow" vector $\boldsymbol{q}$ of Eq. (38) and the construction logic of Eq. (33)–(36). This construction logic will be expressed in the considered LP by the elements of a matrix B and a rhs vector $\boldsymbol{\beta}^2$ that collect, respectively, the coefficients that multiply the vector $\tilde{\boldsymbol{x}}$ and the constants appearing in the rhs of Eqs (34)–(36). Hence, matrix B is a $dim(\boldsymbol{g}) \times dim(\tilde{\boldsymbol{x}})$ binary matrix, while $\boldsymbol{\beta}^2$ is a vector of dimensionality equal to the dimensionality of vector $\boldsymbol{g}$ and it contains three strings of components with corresponding values 1, 2 and 0. Furthermore, under this specification of B and $\boldsymbol{\beta}^2$,

$$\boldsymbol{g} \in \partial\tilde{\theta}(\boldsymbol{\pi}) \iff \exists\, \boldsymbol{q} \geq \boldsymbol{0} \text{ with } F(\boldsymbol{\pi}) \cdot \boldsymbol{q} \;=\; \boldsymbol{\beta}^1(\boldsymbol{\pi}) \text{ such that } B \cdot \boldsymbol{q} \;-\; \boldsymbol{\beta}^2 \;=\; \boldsymbol{g} \tag{39}$$

Next, we employ Eq. (39) and the remaining variables $\boldsymbol{y}$, $\boldsymbol{z}$ and δ to formulate the required optimality test; i.e., we shall use these variables to test whether the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$ contains the vector $\boldsymbol{0}$, or a vector $\boldsymbol{g} \neq \boldsymbol{0}$ that is perpendicular to the hyperplane $\sum_{a \in A} \nu_a = 1$, or, finally, a vector $\boldsymbol{g} \neq \boldsymbol{0}$ that has non-zero projection on this hyperplane, but for any $\boldsymbol{\pi}$ that is located in that hyperplane and is feasible w.r.t. the nonnegativity constraints of Eq. (15), it holds that $\tilde{\boldsymbol{g}}^T \cdot (\boldsymbol{\pi} - \tilde{\boldsymbol{\pi}}) \leq 0$. We start the construction of this test with the following remark:

The simpler test of assessing whether $\boldsymbol{0} \in \partial\tilde{\theta}(\boldsymbol{\pi})$ can be performed straightforwardly by the following LP:

$$\min_{\boldsymbol{q},\boldsymbol{y},\boldsymbol{z}} \sum_{i=1}^{m} y_i + \sum_{i=1}^{m} z_i \tag{40}$$

s.t.

$$F(\boldsymbol{\pi}) \cdot \boldsymbol{q} \;=\; \boldsymbol{\beta}^1(\boldsymbol{\pi}) \tag{41}$$

$$B \cdot \boldsymbol{q} \;+\; \boldsymbol{y} \;-\; \boldsymbol{z} \;=\; \boldsymbol{\beta}^2 \tag{42}$$

$$\boldsymbol{q} \geq \boldsymbol{0};\; \boldsymbol{y} \geq \boldsymbol{0};\; \boldsymbol{z} \geq \boldsymbol{0} \tag{43}$$

Indeed, if $\boldsymbol{0} \in \partial\tilde{\theta}(\boldsymbol{\pi})$, then it follows from Eq. (39) that the constraint of Eq. (42) is satisfied with $\boldsymbol{y} = \boldsymbol{z} = \boldsymbol{0}$, and given the nonnegativity of the variables $\boldsymbol{y}$ and $\boldsymbol{z}$, the above LP has an optimal value of zero. If, on the other hand, $\boldsymbol{0} \notin \partial\tilde{\theta}(\boldsymbol{\pi})$, then the satisfaction of the constraint of Eq. (42)

will necessitate the usage of some of the "slack" and/or "surplus" variables $\boldsymbol{y}$ and $\boldsymbol{z}$, and therefore, the optimal value of the above LP will be strictly positive.

Testing whether there exists a subgrandient $\boldsymbol{g} \in \partial\tilde{\theta}(\boldsymbol{\pi})$ that is perpendicular to the hyperplane $\sum_{a\in A}\nu_a = 1$ can still be performed by an LP similar to that of Eqs (40)–(43), but with the objective function and the corresponding Constraint (42) modified as follows:

$$\min_{\boldsymbol{q},\boldsymbol{y},\boldsymbol{z},\delta} \sum_{i=1}^{m} y_i + \sum_{i=1}^{m} z_i \tag{44}$$

$$B\cdot\boldsymbol{q} \;+\; \boldsymbol{y} \;-\; \boldsymbol{z} \;+\; \delta\mathbf{1}_{\boldsymbol{\nu}} \;=\; \boldsymbol{\beta}^2 \tag{45}$$

The vector $\mathbf{1}_{\boldsymbol{\nu}}$ that appears in Constraint (45) is a binary vector of dimensionality m and with its unit elements corresponding to the ν-type of elements in the vector $\boldsymbol{\pi}$ (and $\boldsymbol{g}$). Hence, if the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$ contains an element $\boldsymbol{g}$ that is perpendicular to the hyperplane $\sum_{a\in A}\nu_a = 1$, and therefore, collinear to the vector $\mathbf{1}_{\boldsymbol{\nu}}$, Constraint (45) can be satisfied through the usage of the newly added terms while keeping the vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ at $\mathbf{0}$. In other words, besides the original condition of $\mathbf{0} \in \partial\tilde{\theta}(\boldsymbol{\pi})$, the modified LP that is defined by Eqs (44), (41), (45) and (43) has an optimal value of zero when the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$ contains a subgradient $\boldsymbol{g}$ that is perpendicular to the hyperplane that is defined by the constraint $\sum_{a\in A}\nu_a = 1$.

Next, we shall modify further the LP of Eqs (44), (41), (45) and (43) in order to develop a test for the existence of a subgradient $\boldsymbol{g} \neq \mathbf{0}$ in $\partial\tilde{\theta}(\boldsymbol{\pi})$ that has non-zero projection on the hyperplane $\sum_{a\in A}\nu_a = 1$, but for any $\boldsymbol{\pi'}$ that is feasible w.r.t. the constraints of Eq. (15), it holds that $\boldsymbol{g}^T\cdot(\boldsymbol{\pi'}-\boldsymbol{\pi}) \le 0$. For this, first we notice that any such subgradient $\boldsymbol{g}$ will have a representation in terms of the Constraints (41), (45) and (43) with $||\boldsymbol{y}|| + ||\boldsymbol{z}|| > 0$, where the operator $||\cdot||$ returns the l_1 norm of its argument. More specifically, in the aforementioned representation,

$$\boldsymbol{g} \;=\; B\cdot\boldsymbol{q} \;-\; \boldsymbol{\beta}^2 \;=\; (-\boldsymbol{y} \;+\; \boldsymbol{z}) \;+\; \delta\mathbf{1}_{\boldsymbol{\nu}} \tag{46}$$

and the vector $(-\boldsymbol{y} \;+\; \boldsymbol{z})$ is necessary for expressing the non-zero projection of $\boldsymbol{g}$ in the hyperplane $\sum_{a\in A}\nu_a = 1$ that contains the feasible vectors $\boldsymbol{\pi}$. Also, it is easy to see that, under the objective function of Eq. (44), (i) the vector $(-\boldsymbol{y} \;+\; \boldsymbol{z})$ will be exactly equal to the projection of $\boldsymbol{g}$ in the hyperplane $\sum_{a\in A}\nu_a = 1$ and (ii) $\boldsymbol{y}[i]\boldsymbol{z}[i] = 0$, $\forall i = 1,\ldots,m$ (since the quantity $\sum_{i=1}^{m} y_i + \sum_{i=1}^{m} z_i$ would be unnecessarily increased, otherwise). Finally, if the vector $\boldsymbol{\pi}$ is in the interior of the feasible region of the dual problem of Eq. (15), then, there will exist a point $\boldsymbol{\pi'} \ge \mathbf{0}$ in the hyperplane $\sum_{a\in A}\nu_a = 1$ such that $\boldsymbol{g}^T\cdot(\boldsymbol{\pi'}-\boldsymbol{\pi}) = (-\boldsymbol{y} \;+\; \boldsymbol{z})^T\cdot(\boldsymbol{\pi'}-\boldsymbol{\pi}) > 0$; i.e., in this case, any vector $\boldsymbol{g}$ with a non-zero projection in the hyperplane $\sum_{a\in A}\nu_a = 1$ will fail to meet the corresponding optimality condition.

The only way that the considered optimality condition can be satisfied by a subgradient $\boldsymbol{g}$ with a nonzero projection $(-\boldsymbol{y} + \boldsymbol{z})$ in the hyperplane $\sum_{a\in A}\nu_a = 1$, is by having (i) the vector $\boldsymbol{\pi}$ binding a set of the non-negativity constraints for its elements, and (ii) $\boldsymbol{g}^T\cdot(\boldsymbol{\pi'}-\boldsymbol{\pi}) = (-\boldsymbol{y} + \boldsymbol{z})^T\cdot(\boldsymbol{\pi'}-\boldsymbol{\pi}) \le 0$ for every point $\boldsymbol{\pi'} \ge \mathbf{0}$ in the hyperplane $\sum_{a\in A}\nu_a = 1$. For a given vector $\boldsymbol{\pi}$, let $\Omega(\boldsymbol{\pi}) \equiv \{i : \boldsymbol{\pi}[i] > 0\}$ and $\Omega^C(\boldsymbol{\pi}) \equiv \{i : \boldsymbol{\pi}[i] = 0\}$. Also, let $\mathbf{1}_i$ denote the unit vector in the i-th coordinate. Then, condition (ii) above becomes:

$$\forall i \in \Omega(\boldsymbol{\pi}),\;\; \boldsymbol{g}^T\cdot\mathbf{1}_i \;=\; (-\boldsymbol{y} \;+\; \boldsymbol{z})^T\cdot\mathbf{1}_i \;=\; 0 \tag{47}$$

$$\forall i \in \Omega^C(\boldsymbol{\pi}),\;\; \boldsymbol{g}^T\cdot\mathbf{1}_i \;=\; (-\boldsymbol{y} \;+\; \boldsymbol{z})^T\cdot\mathbf{1}_i \;\le\; 0 \tag{48}$$

In view of Eqs (44) and (43), Eqs (47) and (48) further imply that

$$\forall i \in \Omega(\boldsymbol{\pi}),\;\; \boldsymbol{y}[i] = 0 \;\wedge\; \boldsymbol{z}[i] = 0 \tag{49}$$

$$\forall i \in \Omega^C(\boldsymbol{\pi}),\;\; \boldsymbol{y}[i] \ge 0 \;\wedge\; \boldsymbol{z}[i] = 0 \tag{50}$$

In plain terms, Eqs (49) and (50) imply that the sought subgradient $\boldsymbol{g}$ will satisfy the third optimality condition discussed above iff its projection in the hyperplane $\sum_{a\in A}\nu_a = 1$ can be expressed in terms of the elements of the vector $\boldsymbol{y}$ corresponding to the binding coordinates of $\boldsymbol{\pi}$. But then, the existence of such a subgradient can be detected by a test similar to that defined by the LP of Eqs (44), (41), (45) and (43), but with the objective function of that LP modified to:

$$\min_{\boldsymbol{q},\boldsymbol{y},\boldsymbol{z},\delta} \sum_{i\in\Omega(\boldsymbol{\pi})} y_i + \sum_{i=1}^{m} z_i \tag{51}$$

The next proposition recapitulates all the above discussion.

Proposition 4.1. *Consider the following LP that is defined according to the logic and the notation that were introduced in the previous part of this section.*

$$\min_{\boldsymbol{\psi}} \ \boldsymbol{c}^T(\boldsymbol{\pi})\cdot\boldsymbol{\psi} \ \equiv \ \sum_{i\in\Omega(\boldsymbol{\pi})} y_i + \sum_{i=1}^{m} z_i \tag{52}$$

s.t.

$$A(\boldsymbol{\pi})\cdot\boldsymbol{\psi} \ \equiv \ \begin{bmatrix} F(\boldsymbol{\pi}) & 0 & 0 & \mathbf{0} \\ B & I_{m\times m} & -I_{m\times m} & \mathbf{1}_{\boldsymbol{\nu}} \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{y} \\ \boldsymbol{z} \\ \delta \end{bmatrix} = \begin{bmatrix} \boldsymbol{\beta}^1(\boldsymbol{\pi}) \\ \boldsymbol{\beta}^2 \end{bmatrix} \ \equiv \ \boldsymbol{b}(\boldsymbol{\pi}) \tag{53}$$

$$\boldsymbol{q} \geq \mathbf{0}; \ \boldsymbol{y} \geq \mathbf{0}; \ \boldsymbol{z} \geq \mathbf{0} \tag{54}$$

Then, for any vector $\boldsymbol{\pi}$ satisfying the constraints of Eq (15), the above LP will have a nonnegative finite optimal solution $\mathcal{P}(\boldsymbol{\pi})$. Furthermore, $\mathcal{P}(\boldsymbol{\pi}) = 0$ iff $\boldsymbol{\pi}$ is an optimal solution for the MP of Eq. (15).

In the rest of this section, we consider the dual LP [11] for the LP of Eqs (52)–(54). From LP Duality theory [11], it is well known that this new LP has the same optimal value as the primal, and therefore, it can be used in the aforementioned optimality test in lieu of the primal LP. Furthermore, as we shall show in the following, working with the dual LP for this test has an advantage over working with the primal LP, because in the case of a negative outcome, the optimal solution of the dual LP can be employed towards the construction of an improving direction for the underlying dual problem.

We start these developments by providing the dual LP formulation: Hence, let $\boldsymbol{\eta}$ denote the dual variables for the upper part of the constraints that are defined by the matrix $A(\boldsymbol{\pi})$, and $\boldsymbol{\rho}$ denote the dual variables for the constraints that are defined by the lower part of $A(\boldsymbol{\pi})$. Then, the considered LP formulation can be written as follows:

$$\max_{\boldsymbol{\eta},\boldsymbol{\rho}} \ (\boldsymbol{\beta}^1(\boldsymbol{\pi}))^T\cdot\boldsymbol{\eta} \ + \ (\boldsymbol{\beta}^2)^T\cdot\boldsymbol{\rho} \tag{55}$$

s.t.

$$(F(\boldsymbol{\pi}))^T\cdot\boldsymbol{\eta} \ + \ B^T\cdot\boldsymbol{\rho} \ \leq 0 \tag{56}$$

$$\forall i = 1,\ldots,m, \ \ \boldsymbol{\rho}[i] \ \leq \ \begin{cases} 1 & \text{if } i\in\Omega(\boldsymbol{\pi}) \\ 0 & \text{if } i\in\Omega^C(\boldsymbol{\pi}) \end{cases} \tag{57}$$

$$\forall i = 1,\ldots,m, \ \ -\boldsymbol{\rho}[i] \ \leq \ 1 \iff \boldsymbol{\rho}[i] \ \geq \ -1 \tag{58}$$

$$\mathbf{1}_{\boldsymbol{\nu}}^T \cdot \boldsymbol{\rho} \;=\; 0 \tag{59}$$

In the above LP formulation, Eqs (57)–(58) essentially define the range of the feasible values for the dual variables $\rho[i],\ i = 1, \ldots, m$. A more concise representation of these constraints is as follows:

$$\forall i = 1, \ldots, m, \quad \left\{ \begin{array}{lllll} -1 & \leq & \boldsymbol{\rho}[i] & \leq & 1 \quad \text{if } i \in \Omega(\boldsymbol{\pi}) \\ -1 & \leq & \boldsymbol{\rho}[i] & \leq & 0 \quad \text{if } i \in \Omega^C(\boldsymbol{\pi}) \end{array} \right. \tag{60}$$

On the other hand, the elements of the vector $\boldsymbol{\eta}$ are free variables, and this vector is constrained only by the constraint of Eq. (56).

The next proposition establishes the ability of the dual LP formulation of Eqs (55), (56), (59) and (60) to provide ascending directions at points $\boldsymbol{\pi}$ that are feasible but not optimal solutions of the dual problem.

Proposition 4.2. *Consider an optimal solution $(\boldsymbol{\eta}^{*T}, \boldsymbol{\rho}^{*T})^T$ of the dual LP of Eqs (55), (56), (59) and (60) for some vector $\boldsymbol{\pi}$ that constitutes a feasible solution to the dual problem of Eq. 15, and let $\mathcal{D}(\boldsymbol{\pi})$ $(= \mathcal{P}(\boldsymbol{\pi}))$ denote the corresponding optimal value. Furthermore, suppose that $\mathcal{D}(\boldsymbol{\pi}) > 0$, and therefore, $\boldsymbol{\pi}$ is a suboptimal point for the dual problem of Eq. (15). Then, the vector $-\boldsymbol{\rho}^*$ defines a "feasible direction of ascent" at $\boldsymbol{\pi}$ for the MP formulation of Eq. (15); i.e., moving from point $\boldsymbol{\pi}$ in the direction of $-\boldsymbol{\rho}^*$ by a carefully selected step s can lead to an increase of the dual function $\tilde{\theta}(\boldsymbol{\pi})$ without violating the constraints of Eq. (15).*

Proof: The feasibility of such a movement is guaranteed by Constraints (59) and (60). More specifically, Constraint (59) ensures that any movement in the direction of $-\boldsymbol{\rho}^*$ will take place in the hyperplane $\sum_{a \in A} \nu_a = 1$. On the other hand, Constraint (60) guarantees that the coordinates of the vector $-\boldsymbol{\rho}^*$ that correspond to binding non-negativity constraints at $\boldsymbol{\pi}$ will be nonnegative, and therefore, it is feasible to move in the direction of $-\boldsymbol{\rho}^*$ while respecting the nonnegativity constraints of Eq. (15).

To show that $-\boldsymbol{\rho}^*$ is a direction of ascent for $\tilde{\theta}(\boldsymbol{\pi})$, it suffices to show that

$$\forall \boldsymbol{g} \in \partial\tilde{\theta}(\boldsymbol{\pi}), \quad (-\boldsymbol{\rho}^*)^T \cdot \boldsymbol{g} > 0 \tag{61}$$

since the above result, when combined with Eq. (23), will imply that the directional derivative of $\tilde{\theta}(\boldsymbol{\pi})$ at $\boldsymbol{\pi}$ in the direction of $-\boldsymbol{\rho}^*$ is strictly positive.

So, let us consider a subgradient $\boldsymbol{g} \in \partial\tilde{\theta}(\boldsymbol{\pi})$, and notice that, according to Eq. (39), there exists a nonnegative vector $\boldsymbol{q}$ such that

$$F(\boldsymbol{\pi}) \cdot \boldsymbol{q} \;=\; \boldsymbol{\beta}^1(\boldsymbol{\pi}) \tag{62}$$

and

$$\boldsymbol{g} \;=\; B \cdot \boldsymbol{q} \;-\; \boldsymbol{\beta}^2 \tag{63}$$

Then, for the considered vector $\boldsymbol{g}$,

$$\begin{aligned} (-\boldsymbol{\rho}^*)^T \cdot \boldsymbol{g} &= (-\boldsymbol{\rho}^*)^T \cdot (B \cdot \boldsymbol{q} \;-\; \boldsymbol{\beta}^2) \\ &= (-\boldsymbol{\rho}^*)^T \cdot B \cdot \boldsymbol{q} \;+\; (\boldsymbol{\rho}^*)^T \cdot \boldsymbol{\beta}^2 \\ &= -\boldsymbol{q}^T \cdot B^T \cdot \boldsymbol{\rho}^* \;+\; (\boldsymbol{\beta}^2)^T \cdot \boldsymbol{\rho}^* \end{aligned} \tag{64}$$

Since, by the working assumption, $\mathcal{D}(\boldsymbol{\pi}) > 0$, from Eq. (55) we also have

$$\begin{aligned} (\boldsymbol{\beta}^2)^T \cdot \boldsymbol{\rho}^* &> -(\boldsymbol{\beta}^1(\boldsymbol{\pi}))^T \cdot \boldsymbol{\eta}^* \\ \text{(from Eq. (62))} \quad &= -(F(\boldsymbol{\pi}) \cdot \boldsymbol{q})^T \cdot \boldsymbol{\eta}^* \\ &= -\boldsymbol{q}^T \cdot (F(\boldsymbol{\pi}))^T \cdot \boldsymbol{\eta}^* \end{aligned} \tag{65}$$

Eqs (64) and (65) when combined with Eq. (56) and the non-negativity of the vector $\boldsymbol{q}$ further imply that

$$\begin{aligned}(-\boldsymbol{\rho}^*)^T \cdot \boldsymbol{g} \quad &> \quad -\boldsymbol{q}^T \cdot B^T \cdot \boldsymbol{\rho}^* \; - \; \boldsymbol{q}^T \cdot (F(\boldsymbol{\pi}))^T \cdot \boldsymbol{\eta}^* \\ &= \quad -\boldsymbol{q}^T \cdot (B^T \cdot \boldsymbol{\rho}^* \; + \; (F(\boldsymbol{\pi}))^T \cdot \boldsymbol{\eta}^*) \\ &\geq \quad 0 \end{aligned} \tag{66}$$

and establish the above claim regarding the ascending nature of the direction that is defined by $-\boldsymbol{\rho}^*$. □

Recapitulating the above discussion, at any feasible point $\boldsymbol{\pi}$ of the MP formulation of Eq. (15), the solution of the dual LP formulation of Eqs (55), (56), (59) and (60) provides an optimality test for $\boldsymbol{\pi}$ by means of the optimal value $\mathcal{D}(\boldsymbol{\pi})$, and for non-optimal vectors $\boldsymbol{\pi}$, it also provides a feasible direction of ascent for the dual function $\tilde{\theta}(\boldsymbol{\pi})$ by means of the optimal value of the dual variable $\boldsymbol{\rho}$. In the next section we provide a procedure that will return a step size $\hat{s}$ for the identified improving direction $-\boldsymbol{\rho}^*$ that will maximize the incurred increase for the dual value function $\tilde{\theta}(\boldsymbol{\pi})$.

4.5 The step-size optimizer employed by the proposed algorithm

The problem addressed in this section can be stated as follows: Given a vector $\boldsymbol{\pi}$ that constitutes a suboptimal feasible point for the dual problem of Eq. (15) and a feasible direction of ascent defined by a vector $\boldsymbol{\rho}$, determine a step $\hat{s} > 0$ in the direction of $\boldsymbol{\rho}$ that will incur the largest possible improvement in the dual function and at the same time it will maintain the non-negativity requirement for the components of the variable vector π. In more formal terms, we need to solve the following optimization problem:

$$\max_{s \geq 0} \; \tilde{\theta}(\boldsymbol{\pi} + s\boldsymbol{\rho}) \tag{67}$$

s.t.

$$\boldsymbol{\pi} + s\boldsymbol{\rho} \;\geq\; \mathbf{0} \tag{68}$$

Furthermore, in view of Eq. (20), the objective function of the above MP formulation can be rewritten as:

$$u(s; \boldsymbol{\pi}, \boldsymbol{\rho}) \;\equiv\; \min_{\boldsymbol{x} \in X} \{(\boldsymbol{\pi} + s\boldsymbol{\rho})^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \tag{69}$$

Hence, similar to the case of the dual problem of Eq. 15, the MP formulation of Eqs (67)–(68) involves the maximization of a piecewise linear, concave function in the single variable s; this function is continuous but it is not differentiable at the points where the minimum in the rhs of Eq. (69) is attained by more than one $\boldsymbol{x} \in X$. Working as in Section 4.3, it can be shown that, at any point s, the subdifferential $\partial u(s; \boldsymbol{\pi}, \boldsymbol{\rho})$ for this new function is defined by

$$\partial u(s; \boldsymbol{\pi}, \boldsymbol{\rho}) \;=\; \Big\{ \boldsymbol{\rho}^T \cdot \boldsymbol{g}(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in Conv\Big(\arg\min_{\boldsymbol{x} \in X} \{(\boldsymbol{\pi} + s\boldsymbol{\rho})^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \Big) \Big\} \tag{70}$$

and given the scalar nature of the elements of $\partial u(s; \boldsymbol{\pi}, \boldsymbol{\rho})$, the above expression simplifies to

$$\begin{aligned}\partial u(s; \boldsymbol{\pi}, \boldsymbol{\rho}) \quad = \quad & \big[\min \big\{ \boldsymbol{\rho}^T \cdot \boldsymbol{g}(\hat{\boldsymbol{x}}) : \; \hat{\boldsymbol{x}} \in \arg\min_{\boldsymbol{x} \in X} \{(\boldsymbol{\pi} + s\boldsymbol{\rho})^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \big\}, \\ & \max \big\{ \boldsymbol{\rho}^T \cdot \boldsymbol{g}(\hat{\boldsymbol{x}}) : \; \hat{\boldsymbol{x}} \in \arg\min_{\boldsymbol{x} \in X} \{(\boldsymbol{\pi} + s\boldsymbol{\rho})^T \cdot \boldsymbol{g}(\boldsymbol{x})\} \big\} \big] \end{aligned} \tag{71}$$

Since $\boldsymbol{\rho}$ is an ascending direction for function $\tilde{\theta}(\boldsymbol{\pi})$ at $\boldsymbol{\pi}$, the interval corresponding to the subdifferential $\partial u(0; \boldsymbol{\pi}, \boldsymbol{\rho})$ is to the right of the origin of the real axis; we shall denote this fact by writing $\partial u(0;\boldsymbol{\pi},\boldsymbol{\rho}) > 0$. Furthermore, Eq. (25) implies that, in the absence of Constraint (68), an optimal solution for the MP formulation that is defined by Eq. (67) is any s for which the subdifferential $\partial u(s;\boldsymbol{\pi},\boldsymbol{\rho})$, defined by Eq. (71), contains the origin. On the other hand, to account for the constraints of Eq. (68), s must be constrained to the smallest value $\bar{s}$ that will lead to the binding of some of these constraints. This value is straightforwardly obtained as

$$\bar{s} = \begin{cases} \min\big\{ -\frac{\boldsymbol{\pi}[i]}{\boldsymbol{\rho}[i]},\ \forall i \in \{1,\ldots,m\} \text{ with } \boldsymbol{\rho}[i] < 0\big\} & \text{if } \exists\ i \text{ with } \boldsymbol{\rho}[i] < 0 \\ \infty & \text{otherwise} \end{cases} \tag{72}$$

Then, the considered algorithm will start by computing $\bar{s}$ through Eq. (72) and, assuming that Eq. 72 will return a finite $\bar{s}$, the algorithm will check whether $\partial u(\bar{s};\boldsymbol{\pi},\boldsymbol{\rho}) \geq 0$. If this inequality is satisfied, it can be concluded that, at $\bar{s}$, the function $u(s;\boldsymbol{\pi},\boldsymbol{\rho})$ has not reached its decreasing part yet, and therefore, the algorithm returns $\hat{s} = \bar{s}$.

If, on the other hand, $\bar{s} = \infty$, or $\bar{s} < \infty$ but $\partial u(\bar{s};\boldsymbol{\pi},\boldsymbol{\rho}) < 0$ (i.e., the entire interval that defines the subdifferential $\partial u(\bar{s};\boldsymbol{\pi},\boldsymbol{\rho})$ is to the left of the origin of the real axis, and the point $\bar{s}$ is in the decreasing part of the function $u(s;\boldsymbol{\pi},\boldsymbol{\rho})$), the MP formulation of Eq. (67)–(68) reduces to the unconstrained MP that is defined by Eq. (67) only.

Next we present an algorithm for identifying an unconstrained maximum for function $u(s;\boldsymbol{\pi},\boldsymbol{\rho})$. This algorithm will start with two points: point $s_1 = 0$ and another point s_2 selected such that $\partial u(s_2;\boldsymbol{\pi},\boldsymbol{\rho}) < 0$; practically, the last condition implies that s_2 exceeds the set of the optimal step-sizes and reaches the decreasing section of the function $u(s;\boldsymbol{\pi},\boldsymbol{\rho})$. Hence, in the case that $\bar{s} < \infty$, we can set $s_2 = \bar{s}$; otherwise, such a point s_2 can be identified by a "trial-&-error" procedure on some increasing sequence of s.

At each of the two selected points, the algorithm picks an element ζ_k, $k = 1,2$, from the corresponding subdifferential, and defines the linear function $u(s_k;\boldsymbol{\pi},\boldsymbol{\rho})+\zeta_k(s-s_k)$. Since $\zeta_1\cdot\zeta_2 < 0$, these two straight lines cross at some point s_3, computed by

$$s_3 \ = \ \frac{u(s_1;\boldsymbol{\pi},\boldsymbol{\rho}) - \zeta_1 \cdot s_1 - u(s_2;\boldsymbol{\pi},\boldsymbol{\rho}) + \zeta_2 \cdot s_2}{\zeta_2 - \zeta_1} \tag{73}$$

Next we show that $s_3 \in (s_1, s_2)$. Hence, consider the difference

$$\begin{aligned} s_3 - s_1 \ &= \ \frac{u(s_1;\boldsymbol{\pi},\boldsymbol{\rho}) - \zeta_1 \cdot s_1 - u(s_2;\boldsymbol{\pi},\boldsymbol{\rho}) + \zeta_2 \cdot s_2}{\zeta_2 - \zeta_1} \ - \ s_1 \\ &= \ \frac{u(s_2;\boldsymbol{\pi},\boldsymbol{\rho}) + \zeta_2 \cdot (s_1 - s_2) - u(s_1;\boldsymbol{\pi},\boldsymbol{\rho})}{\zeta_1 - \zeta_2} \ \geq \ 0 \end{aligned} \tag{74}$$

where the last inequality holds because, by the working assumption, $\zeta_1 > 0 > \zeta_2$ and ζ_2 is a subgradient for the concave function u at point s_2. Furthermore, $s_3 \neq s_1$ because, in the opposite case, the entire line segment of the line $u(s_2;\boldsymbol{\pi},\boldsymbol{\rho}) + \zeta_2 \cdot (s - s_2)$ between the points $u(s_1;\boldsymbol{\pi},\boldsymbol{\rho})$ and $u(s_2;\boldsymbol{\pi},\boldsymbol{\rho})$ must be part of the function $u(s,\boldsymbol{\pi},\boldsymbol{\rho})$, and this last result negates the assumption that the vector $\boldsymbol{\rho}$ defines an ascending direction at $\tilde{\theta}(\boldsymbol{\pi})$ (since the subdifferential $\partial u(s_1 = 0;\boldsymbol{\pi},\boldsymbol{\rho})$ would contain the negative element ζ_2 in that case). The fact that $s_3 < s_2$ can be proved in a similar manner.

With $s_3 \in (s_1, s_2)$, the considered algorithm for the unconstrained maximization of the function $u(s,\boldsymbol{\pi},\boldsymbol{\rho})$ proceeds to check whether $0 \in \partial u(s_3;\boldsymbol{\pi},\boldsymbol{\rho})$. If this is true, then s_3 is an optimal step-size for the unconstrained problem and the algorithm returns $\hat{s} = s_3$. If, on the other hand, $0 < \partial u(s_3;\boldsymbol{\pi},\boldsymbol{\rho})$ (resp., $0 > \partial u(s_3;\boldsymbol{\pi},\boldsymbol{\rho})$), the algorithm sets $s_1 = s_3$ (resp., $s_2 = s_3$) and repeats the previous computation with the updated pair of points (s_1, s_2). Since the number of slopes that

```
Input: Data defining the considered traffic coordination problem as specified in Section 2.2.1;
    $\boldsymbol{\pi}$; $\boldsymbol{\rho}$
Output: $\hat{s}$
Compute $\bar{s}$ according to Eq. (72);
if $\bar{s} \neq \infty \ \wedge \ \partial u(\bar{s}; \boldsymbol{\pi}, \boldsymbol{\rho}) \geq 0$ then
    $\hat{s} := \bar{s}$;
else
    $s_1 := 0$;
    if $\bar{s} \neq \infty$ then
        $s_2 := \bar{s}$;
    else
        $s_2 :=$ any value with $\partial u(s_2; \boldsymbol{\pi}, \boldsymbol{\rho}) < 0$;
    end if
    for $k = 1, 2$ do
        Pick $\zeta_k$ from the subdifferential $\partial u(s_k; \boldsymbol{\pi}, \boldsymbol{\rho})$;
    end for
    Compute $s_3$ according to Eq. (73);
    while $0 \notin \partial u(s_3; \boldsymbol{\pi}, \boldsymbol{\rho})$ do
        if $\partial u(s_3; \boldsymbol{\pi}, \boldsymbol{\rho}) > 0$ then
            $s_1 := s_3$;
        else
            $s_2 := s_3$;
        end if
        Compute $s_3$ according to Eq. (73);
    end while
    $\hat{s} := s_3$;
end if
return $\hat{s}$
```

Figure 1: The step-size selection algorithm

are involved in the definition of the piecewise linear function $u(s, \boldsymbol{\pi}, \boldsymbol{\rho})$ is finite, the algorithm will terminate in finite time.

The complete algorithm for the determination of the sought step size s is formally stated in Figure 1. Furthermore, it must be pointed out that the tests regarding the "sign" of the subdifferential $\partial u(s; \boldsymbol{\pi}, \boldsymbol{\rho})$ at the different points s visited by the algorithm, can be performed systematically and efficiently by means of the subgraphs $\mathcal{G}_a$, $a \in \mathcal{A}$, that were introduced in Section 3.2.2 and the corresponding flows $\boldsymbol{q}_a$ that were introduced in Section 4.3. In particular, Eq. (70) implies that the subdifferential $\partial u(s; \boldsymbol{\pi}, \boldsymbol{\rho})$ can be expressed by means of the aforementioned entities through a development similar to that presented in Section 4.4 for the representation of the subdifferential of the dual function $\widetilde{\theta}(\boldsymbol{\pi})$. Then, the minimal and the maximal elements of this subdifferential can be obtained by solving, respectively, a minimization and a maximization problem over the aforementioned representation where the corresponding objective function is set similar to the inner product that appears in Eq. 71 but with the vector $\hat{\boldsymbol{x}}$ replaced by a vector $\tilde{\boldsymbol{x}}$ that is induced by the flow vector $\mathbf{q}$; this replacement simplifies the two optimization problems as it relaxes the corresponding integrality requirement for vector $\boldsymbol{x}$, and turns these two problems into LPs. Furthermore, it is easy to see that these two LPs are separable to a number of simpler LPs that are defined for each agent $a \in \mathcal{A}$.

The next section epitomizes all the developments in Section 4 up to this point, organizing them in a complete algorithm for the solution of the MP formulation of Eq. (15).

4.6 The complete algorithm for the solution of the dual problem

Here we outline the complete "ascent" algorithm for the solution of the MP formulation of Eq. (15) that is implied by the previous developments of Section 4. We also show that the algorithm will converge in finite time to an optimal solution of this problem.

A description of the proposed algorithm is as follows:

1. **Initialization:**
   (a) Initialize vector $\boldsymbol{\pi}$ to a set of feasible values w.r.t. the constraints of Eq. 15. A possible such initialization will set $\boldsymbol{\lambda} = \boldsymbol{\mu} = \mathbf{0}$ and it will distribute the unit value of $||\boldsymbol{\nu}||$ equally among the components of the vector $\boldsymbol{\nu}$ that correspond to the agents $a \in \mathcal{A}$ that have the longest shortest paths from their initial edge s_a to their destination edge d_a.[12]

   (b) Use Eqs (6)–(10) to compute the corresponding coefficients of the Lagrangian function $L(\boldsymbol{x}; \boldsymbol{\pi})$ for the vector $\boldsymbol{\pi}$ selected in Step (1a) above.

   (c) For each agent $a \in \mathcal{A}$, solve the subproblem of Eq. (16) through the DP algorithm described in Section 3.2.2. Also, construct the corresponding digraphs $\mathcal{G}_a$, $a \in \mathcal{A}$, representing the set of the optimal routes for each agent a.

2. **Main Iteration:**
   (a) Use the current set of digraphs $\mathcal{G}_a$, $a \in \mathcal{A}$, and Eqs (31)–(36) to set up the ("dual") LP of Eqs (55), (56), (59) and (60), and solve this LP.

   (b) If the optimal solution of the "dual" LP in step (2a) is equal to zero, then terminate the algorithm returning the current vector $\boldsymbol{\pi}$ and the current set of digraphs $\mathcal{G}_a$, $a \in \mathcal{A}$. Also, compute and return the value of the function $\theta(\boldsymbol{\pi})$ at the returned vector $\boldsymbol{\pi}$; this computation can be based on Eq. (14) and the information that is provided in the returned vector $\boldsymbol{\pi}$ and the digraphs $\mathcal{G}_a$, $a \in \mathcal{A}$.

   (c) Else (i.e., if the optimal solution of the "dual" LP in step (2a) is greater than zero) do the following:

      i. Compute the direction of ascent $-\boldsymbol{\rho}^*$ from the optimal solution of the "dual" LP.

      ii. Compute the optimal step size $\hat{s}$ in the direction $-\boldsymbol{\rho}^*$ using the procedure that is stated in Figure 1. Also, update the set of the digraphs $\mathcal{G}_a$, $a \in \mathcal{A}$, with those that were obtained during the construction of the subdifferential $\partial u(\hat{s}; \boldsymbol{\pi}, -\boldsymbol{\rho}^*)$ by the aforementioned procedure.

      iii. Update $\boldsymbol{\pi}$ to $\boldsymbol{\pi} - \hat{s}\boldsymbol{\rho}^*$.

      iv. Go to Step (2a).

Next we show that the above algorithm converges finitely to an optimal solution of the dual problem. To establish this result, first we need to prove the following technical lemma.

[12] It can be shown that the proposed selection of the $\boldsymbol{\nu}$ vector is a maximizer of the dual function $\theta(\boldsymbol{\pi})$ among all the feasible vectors $\boldsymbol{\pi}$ with $\boldsymbol{\lambda} = \boldsymbol{\mu} = \mathbf{0}$.

Lemma 4.3. *Let $\boldsymbol{\psi}^* = [\boldsymbol{q}^{*T}, \boldsymbol{y}^{*T}, \boldsymbol{z}^{*T}, \delta^*]^T$ and $[\boldsymbol{\eta}^{*T}, \boldsymbol{\rho}^{*T}]^T$ denote respective optimal solutions for the primal LP formulation of Eqs (52)–(53) and its dual at some feasible point $\boldsymbol{\pi}$ of the dual problem of Eq 15. Also, let $\boldsymbol{g}(\boldsymbol{q})$ denote the subgradient in $\partial\tilde{\theta}(\boldsymbol{\pi})$ that is defined through Eq. (39) by a flow vector $\boldsymbol{q}$ among the set of flows that are defined by Eq. 38. Then, we have that*

$$-\boldsymbol{\rho}^*[i] = \begin{cases} 1, & \text{if } \boldsymbol{g}(i;\boldsymbol{q}^*) > 0 \\ -1, & \text{if } \boldsymbol{g}(i;\boldsymbol{q}^*) < 0 \ \wedge \ i \in \Omega(\boldsymbol{\pi}) \\ 0, & \text{if } \boldsymbol{g}(i;\boldsymbol{q}^*) < 0 \ \wedge \ i \in \Omega^C(\boldsymbol{\pi}) \end{cases} \tag{75}$$

Also,

$$-\boldsymbol{\rho}^{*T} \cdot \boldsymbol{g}(\boldsymbol{q}^*) \ = \ \mathcal{P}(\boldsymbol{\pi}) \tag{76}$$

and for every flow vector $\boldsymbol{q}$ that is a feasible solution to Eq. (38), it holds that

$$-\boldsymbol{\rho}^{*T} \cdot \boldsymbol{g}(\boldsymbol{q}^*) \ \le \ -\boldsymbol{\rho}^{*T} \cdot \boldsymbol{g}(\boldsymbol{q}) \tag{77}$$

Proof: To prove Eq. (75), consider first the case where $\boldsymbol{g}(i;\boldsymbol{q}^*) > 0$. Then, from the definition of $\boldsymbol{\rho}^*$, it follows that $\boldsymbol{\rho}^*[i]$ expresses the sensitivity of the primal optimal value $\mathcal{P}(\boldsymbol{\pi})$ w.r.t. the rhs parameter $\boldsymbol{\beta}^2[i]$ in the primal LP. But from the structure of the constraint of Eq. (53) we can see that any sufficiently small variation $\Delta\boldsymbol{\beta}^2[i]$ can be accommodated by a variation $\Delta\boldsymbol{z}[i] = -\Delta\boldsymbol{\beta}^2[i]$, resulting in a variation for the optimal objective value equal to $\Delta\mathcal{P}(\boldsymbol{\pi}) = -\Delta\boldsymbol{\beta}^2[i]$. Hence, $\boldsymbol{\rho}^*[i] = -1$ and the result follows. Furthermore, the other two cases of Eq. (75) can be argued in a similar manner.

To prove the second part of Lemma 4.3, first we notice that Eq. (76) is an immediate consequence of the definition of $\mathcal{P}(\boldsymbol{\pi})$ through Eq. (52) and the result of Eq. (75).

For the result of Eq. (77), consider a flow vector $\boldsymbol{q}$ that is a feasible solution to Eq. (38). Then, we have:

$$\begin{aligned} -\boldsymbol{\rho}^{*T} \cdot \boldsymbol{g}(\boldsymbol{q}) \ &= \ -\boldsymbol{\rho}^{*T} \cdot (B \cdot \boldsymbol{q} - \boldsymbol{\beta}^2) \quad \text{from Eq. (39)} \\ &= \ -\boldsymbol{\rho}^{*T} \cdot B \cdot \boldsymbol{q} \ + \ \boldsymbol{\rho}^{*T} \cdot \boldsymbol{\beta}^2 \\ &\ge \ \eta^{*T} \cdot F(\boldsymbol{\pi}) \cdot \boldsymbol{q} \ + \ \boldsymbol{\rho}^{*T} \cdot \boldsymbol{\beta}^2 \quad \text{from Eq. (56) and } \boldsymbol{q} \ge \boldsymbol{0} \\ &= \ \eta^{*T} \cdot \boldsymbol{\beta}^1(\boldsymbol{\pi}) \ + \ \boldsymbol{\rho}^{*T} \cdot \boldsymbol{\beta}^2 \quad \text{from Eq. (38)} \\ &= \ -\boldsymbol{\rho}^{*T} \cdot \boldsymbol{g}(\boldsymbol{q}^*) \quad \text{from Eqs (55), (76) and the LP strong duality} \end{aligned} \tag{78}$$

□

Proposition 4.4. *The algorithm presented in this section for the solution of the MP formulation of Eq. (15) will terminate in a finite number of iterations returning an optimal solution of this formulation.*

Proof: Proposition 4.2 has established that the vector $-\boldsymbol{\rho}^*$ employed by the algorithm at any suboptimal point $\boldsymbol{\pi}$ of the dual solution space, is indeed a feasible ascending direction at $\boldsymbol{\pi}$. Furthermore, the result of Eq. (77) in Lemma 4.3 implies that the inner product $-\boldsymbol{\rho}^{*T} \cdot \boldsymbol{g}(\boldsymbol{q}^*)$ defines the directional derivative in the direction of $-\boldsymbol{\rho}^*$. Hence, $\boldsymbol{g}(\boldsymbol{q}^*)$ is the gradient of a facet of the polyhedral function $\tilde{\theta}(\cdot)$ that contains the considered point $\boldsymbol{\pi}$ and increases in the direction of $-\boldsymbol{\rho}^*$. On the other hand, Eq. (75) of Lemma 4.3 implies that the selected direction $-\boldsymbol{\rho}^*$ moves the initial point $\boldsymbol{\pi}$ along the coordinates of $\boldsymbol{g}(\boldsymbol{q}^*)$ that define feasible improving dimensions, in an isotropic manner.

In other words, at every visited suboptimal point $\boldsymbol{\pi}$, the algorithm will select an improving facet of the underlying polyhedral function passing through $\boldsymbol{\pi}$, and it will move $\boldsymbol{\pi}$ isotropically

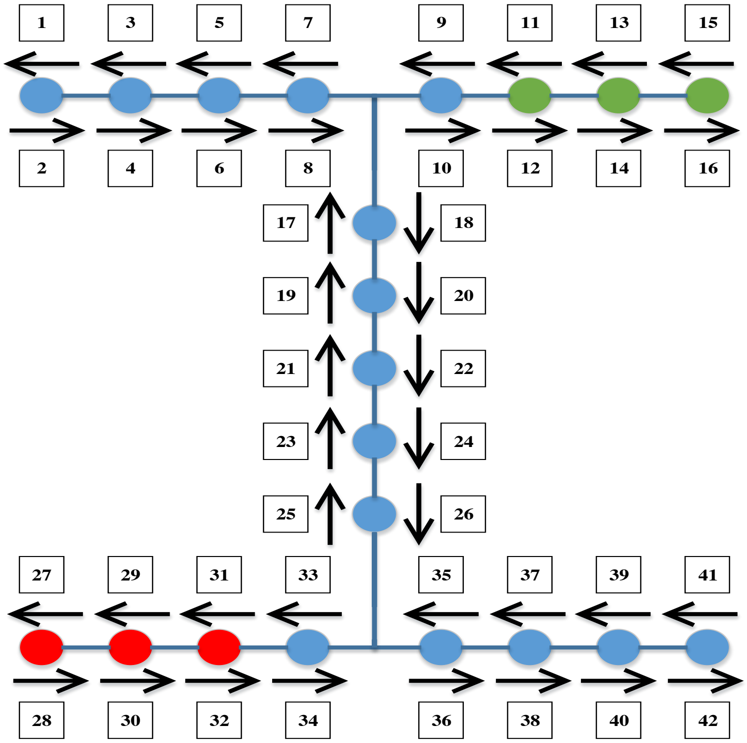

Figure 2: The physical layout for the scheduling problem instance of the example of Section 5

along all its dimensions that constitute feasible directions of improvement, and with the direction of the motion for each dimension being defined by the sign of the corresponding component in the gradient of the selected facet. This movement will persist until either (i) the function $\tilde{\theta}(\cdot)$ starts decreasing in the direction of $-\boldsymbol{\rho}^*$, or (ii) (at least) one of the components of $\boldsymbol{\pi}$ reaches its boundary value of 0. At that point, a new ascending direction and a new facet will be determined in the same manner. Furthermore, the concave polyhedral structure of $\tilde{\theta}(\cdot)$ implies that once a facet is abandoned by the process described above, it cannot be revisited. But since the number of facets in the considered polyhedral function are finite, the algorithm will terminate returning an optimal solution after a finite number of iterations. □

5 Example

In this section, we apply the theoretical developments of the earlier parts of this paper on the instance of the QMP scheduling problem that is depicted in Figure 2. The circles depicted in Figure 2 represent the ion traps that implement physically the corresponding guidepath network, while the the guidepath graph G that is employed in the eventual formulation of the considered scheduling problem is the digraph that is defined by the directed edges that are drawn in black color in Figure 2. As discussed in Section 2.1, each pair of the depicted edges abstracts one of the original ion traps. The considered routing problem involves three qubits, Q_1, Q_2 and Q_3. These three qubits are initially located on edges 28, 30 and 32, and they must be transported to edges 16, 14 and 12, respectively. Furthermore, in the considered operational setting, it is assumed that qubits can reverse the direction of their motion on any given edge; for instance, a qubit located on edge 5 in the depicted digraph can move at the next instant on edge 6. This feature can be easily accommodated in the mathematical formulation of Section 2.2 through the corresponding adjustment of the sets $e^{\bullet}$ for the various edges $e \in E$.

On the other hand, it is also interesting to notice that the routing problem that is defined in Figure 2 has some quite interesting and challenging features, to the point that it has been used as a "benchmark" problem among the researches who work on this class of problems in the QMP operational context. In order to help the reader understand the conceptual and computational intricacies that characterize this problem instance, we make the following remarks: First, it is easy to see that in the considered guidepath structure, each qubit has a single shortest path taking it from its current location to its destination edge. Furthermore, the shortest paths for the three qubits present a nested structure, with the shortest path of qubit Q_2 subsuming the shortest path of qubit Q_3, and the shortest path of qubit Q_1 subsuming the shortest path of qubit Q_2 (and, therefore, the shortest path of qubit Q_3, as well). Finally, in the final configuration for the three qubits that is stipulated by the considered scheduling problem, the qubits must have reversed their order w.r.t. their arrangement in the initial configuration. The above three remarks further imply the following: (i) Qubit Q_2 will not be able to reach its final destination until qubit Q_1 has cleared through this edge. And similarly, qubit Q_3 will be able to reach its destination edge only after qubits Q_1 and Q_2 have cleared through it. Hence, there is an (implicit) set of precedence constraints that must be observed by any routing solution of the considered scheduling problem. (ii) Furthermore, these precedence constraints, when combined with the topology of the underlying guidepath network, also imply that qubits Q_2 and Q_3 will need to allow the qubits that are behind them in the initial arrangement, to "overpass" them; and this effect can only be achieved by having these two qubits move into one of the two "sidetracks" of the depicted digraph, that are defined, respectively, by the edges 1–8 and 35–42. Hence, the construction of even a feasible solution for the considered scheduling problem requires the demonstration of significant coordination and "intelligence" by the participating agents.

With the above understandings, and assuming the traffic dynamics that were defined in Section 2, it can be easily checked that any optimal routing schedule for this problem will have all three qubits reaching their final destinations in 16 time units, where the time unit is the time required by these qubits to go through a single ion trap.

On the other hand, the solution of the corresponding dual problem of Eq. (15) through the algorithm of Section 4.6, with an initial dual solution of $\boldsymbol{\lambda} = \mathbf{0}$, $\boldsymbol{\mu} = \mathbf{0}$, $\nu_a = 1/|\mathcal{A}|$, $\forall a \in \mathcal{A}$, and a time horizon parameter T set at the value of 30 time units, led to a maximal value for the dual function θ of 11.9172.[13] In view of the integrality of the optimal value of the primal MIP formulation, w^*, this result defines a lower bound of 12 time units for w^*. Hence, the observed duality gap for this problem instance is equal to 4 units.

Figure 3 depicts the evolution of the values $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ that are generated by the considered algorithm, throughout its entire execution. We can see that (i) these values constitute a monotonically increasing sequence, and (ii) eventually the algorithm terminated with an optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$, after 68 iterations. It is also interesting to notice the presence of some "plateaus" in the plot of Figure 3, indicating that the algorithm had to go through some regions of the underlying solution space where the stepwise progress w.r.t. the dual function θ was very minuscule, but still the algorithm managed to not get entrapped in these regions and reach a global optimum. Finally, Figure 3 also reveals that, for the considered problem instance, the lower bound of 12 time units for w^* was obtained very early in the overall computation.

Figure 4 depicts the graph $\mathcal{G}_{a_3}$ that encodes the set of all optimal routes for Qubit Q_3, for the

[13] We remind the reader that the dual function θ relates to the objective function $\tilde{\theta}$ of the dual problem of Eq. (15), through the equation $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) = \tilde{\theta}(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) + T + 1$.

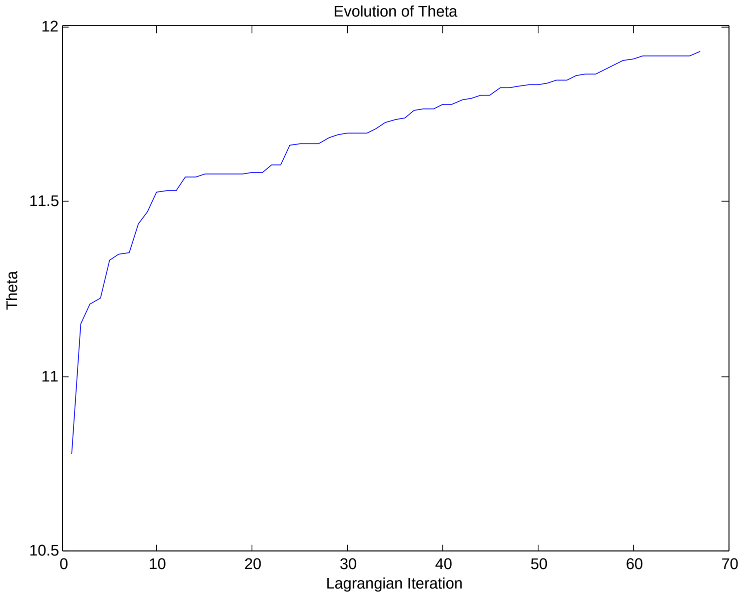

Figure 3: Evolution of the values $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu})$ that are generated during the iterations of the ascent algorithm that is presented in Section 4.6.

“shortest path” formulation of Eq. (16) that is specified by the derived optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ of the corresponding dual problem of Eq. (15).[14] We remind the reader that these graphs $\mathcal{G}_a$ provide a spatio-temporal representation of all the optimal routes of the corresponding “shortest path” formulation of Eq (16), where the edge labels (e, t) specify the location of agent a in the guidepath graph $\mathcal{G}$ at time t. It can be seen that all of the proposed routing schedules place the qubit Q_3 at its destination edge in 12 time units, which is consistent with the bound that is specified by the optimal dual value $\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$. Furthermore, some of the routes that are specified by the graph of Figure 4 for the considered qubit *do* make use of the lower right branch of the guidepath graph $\mathcal{G}$ of Figure 2 (those including edges 35, 36 and 38) or of the upper left branch of $\mathcal{G}$ (those including edges 5, 7 and 8). Hence, in the case of Qubit Q_3, the set of routes that are generated by $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$, convey some of the required “intelligence” for the construction of a feasible routing schedule for the considered problem instance.

On the other hand, it is also true that the graph $\mathcal{G}_{a_2}$, i.e., the counterpart of the graph of Figure 4 for Qubit 2, fails to specify similar intelligent behavior for that qubit. Hence, for this rather hard problem instance,[15] any attempt to generate a feasible routing schedule by restricting the agent routes to the routing sequences that are encoded in the three graphs $\mathcal{G}_{a_i}$, $i = 1, 2, 3$, that are specified by the dual optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$, but possibly allowing for more general timing patterns for the execution of these routes, would lead to an infeasible problem. Such a construction of a feasible schedule from the graphs $\mathcal{G}_a$ that are specified by the dual optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ can be possible for simpler (less constrained) problem instances. But the last finding that is reported above motivates the need for the development of additional methodology that will provide good heuristic (sub-optimal) solutions to the considered scheduling problem, along lines similar to those pursued in [15, 12]. In such an endeavor, the results presented in this paper can still play a significant role by providing lower bounds to the optimal makespan, and thus, assessing

[14] Similar graphs can be generated for the other two qubits Q_1 and Q_2, but we omit them due to space considerations.

[15] for the reasons that were discussed at the introductory part of this example

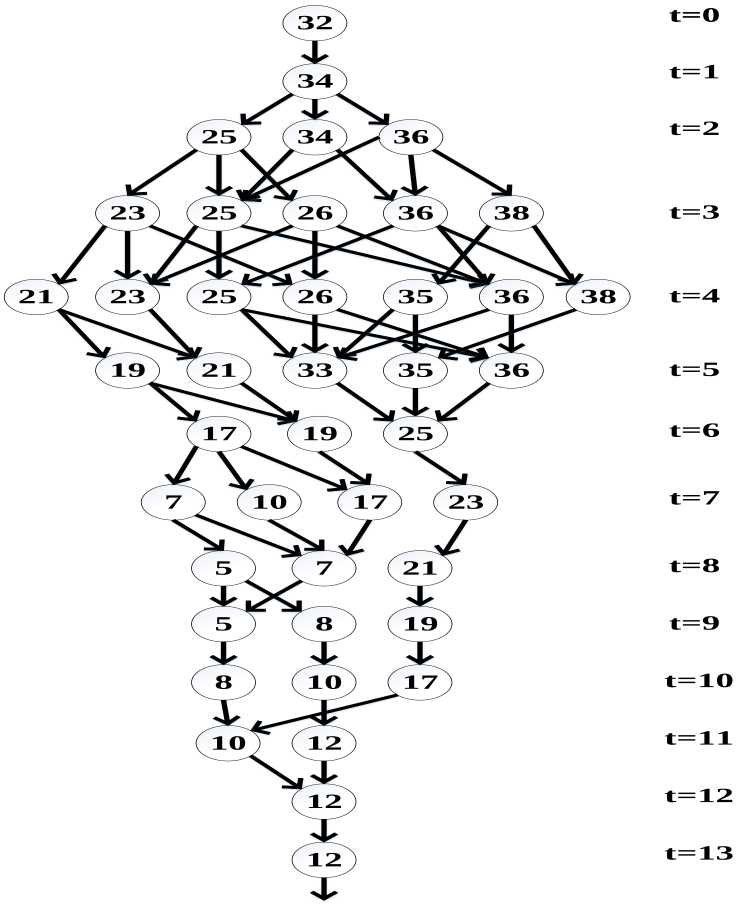

Figure 4: The set of all optimal routes for the "shortest path" formulation of Eq. (16) for Qubit Q_3, as specified by the derived optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ of the corresponding dual problem of Eq. (15).

and characterizing the extent of the sub-optimality in those further methods.

6 Discussion

The analytical results of Section 4 have formally established that the algorithm introduced in this paper for the solution of the dual problem of Eq. (15) has the attractive properties of (i) finite convergence and (ii) monotone improvement of the quality of the derived solutions, where the latter is defined by the value of the dual function $\tilde{\theta}$. At the same time, the computational results that are reported in Section 5 have demonstrated these algorithmic properties and highlighted their significance in terms of generating lower bounds for the original scheduling formulation.

As discussed in [5], for polyhedral dual functions, one can obtain finitely convergent algorithms for the solution of the corresponding dual problem through another class of algorithms that are known as "cutting plane" algorithms. But the general implementations of these algorithms tend to converge very slowly to the optimal solution, and they do not possess the property of the monotonic improvement of the derived solutions that is possessed by our algorithm. On the contrary, as remarked in [5], the performance of the cutting plane algorithms w.r.t. this last attribute can be quite erratic, especially in the earlier stages of their computations. In fact, the development of strictly ascending algorithms for the class of the (dual) optimization problems that are defined in the context of Lagrangian Duality theory, as is the case with the problem considered in this work, is a very challenging task, and the corresponding results are very limited and computationally cumbersome (c.f. [5], Section 6.3.4). Our ability to establish this ascending property for the algorithm that is presented in this work stems from our ability (i) to provide an efficient representation of the subdifferential $\partial\tilde{\theta}(\boldsymbol{\pi})$, and (ii) to pertinently employ this representation in the results of Propositions 4.1 and 4.2.

The presented algorithm is also quite efficient in terms of the involved computations, with the most difficult task in these computations being the iterative formulation and solution of the LP of Eqs (55), (56), (59) and (60) – please, c.f., Step (2a) in the algorithm description that is provided in Section 4.6. These LPs can grow pretty large for larger problem instances, especially as the problem parameter T gets significantly large values. But, at the same time, the defining elements of these LP formulations possess extensive levels of sparsity that, when properly exploited, can streamline substantially the involved computations.[16]

Finally, we notice, for completeness, that the idea of using a Lagrangian relaxation of the MIP formulations of some scheduling problems for the generation of (i) lower bounds, and even (ii) suboptimal solutions to the original scheduling formulations, has been considered in the past literature; some relevant results are reported, for instance, in [6, 18, 9]. Also, reference [7] is an excellent tutorial survey on the methodological developments concerning the application of the Lagrangian relaxation method on the MIP formulations of various important combinatorial optimization problems; this paper also provides some very keen insights on the major findings at the time of its appearance, and on the "spirit" that drove the corresponding research endeavors. When it comes to the particular area of combinatorial scheduling, almost all the corresponding past developments concern some scheduling problems that constitute variations of the "job shop" scheduling problem [15] and do not involve many of the resource allocation dynamics and constraints that are important defining elements for the scheduling formulations addressed in this paper. Furthermore, while they seek to take advantage of some underlying special structure in the evaluation of the dual function $\theta(\boldsymbol{\pi})$ at any given point $\boldsymbol{\pi}$ (the, so called, "Lagrangian problem" in the terminology of [7]), most of these past results employ some "vanilla" and/or heuristic methods for the solution of the dual problem.[17] Hence, when viewed from this standpoint, it is possible that the methodological and the algorithmic developments that are presented in this work can also complement and strengthen the developments of those past endeavors.

7 Conclusions

This paper has introduced a class of novel and hard scheduling problems that concern the routing of a set of mobile agents over the edges of an underlying guidepath network, in a way that (i) guarantees the safety / integrity and the liveness of the agent motion, and (ii) minimizes the schedule makespan. In the context of these problems, our results have primarily addressed the

[16] It can also be argued that the large size of the considered LPs results actually from the representation for the original scheduling problem that is adopted in the MIP formulation of Section 2.2, and therefore, it is not an inherent handicap of the considered algorithm. On the other hand, at this point it is not clear whether the presented algorithmic developments would be easily adaptable to some other (MIP) formulation of the original scheduling problem.

[17] However, it is also true that the work presented in [6], one of the earliest works to use Lagrangian Duality on a complex scheduling problem, has developed a customized algorithm for the solution of the dual problem that is addressed in it, with properties similar to the properties of the corresponding algorithm that we present in this paper. But the two scheduling formulations are quite different, and the established properties are based on different insights and techniques in each case.

effective and efficient solution of some Lagrangian relaxations of these problems through a novel, customized algorithm. Some important attributes of this algorithm are (a) its finite convergence to an optimal solution of the corresponding dual problem, and (b) its "ascending" nature, i.e., its ability to improve monotonically the quality of the derived solutions w.r.t. the corresponding dual function.

Our future work will seek to further assess and exploit the potential of the presented algorithm to define pertinent "seeds" for the construction of efficient schedules for the original scheduling formulation. We shall also seek to develop additional methodology for computing such efficient solutions for the considered scheduling problems, by adapting and extending the "decomposition" methods that are discussed in [15].[18] Finally, a third future task is the integration of the results obtained from the aforementioned other two tasks into an appropriate MPC framework that will solve the broader scheduling problem concerning the multi-leg trips of the system agents.

References

[1] J. Adams, E. Balas, and D. Zawack. The Shifting Bottleneck Procedure for Job Shop Scheduling. *Management Science*, 34:391–401, 1988.

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms and Applications.* Prentice Hall, Englewood Cliffs, NJ, 1993.

[3] D. Bertsekas. Dynamic Programming and Suboptimal Control: A Survey from ADP to MPC. *European Journal of Control*, 11:310–334, 2005.

[4] D. P. Bertsekas. *Dynamic Programming and Optimal Control, Vol. 1.* Athena Scientific, Belmont, MA, 1995.

[5] D. P. Bertsekas. *Nonlinear Programming (2nd ed.).* Athena Scientific, Belmont, MA, 1999.

[6] M. L. Fisher. Optimal solutions of scheduling problems using Lgrange multipliers: Part I. *Operations Research*, 21:1114–1127, 1973.

[7] M. L. Fisher. The Lagrangian relaxation method for solving integer programming problems. *Management Science*, 27:1–18, 1981.

[8] S. S. Heragu. *Facilities Design (3rd ed.).* CRC Press, 2008.

[9] D. J. Hoitomt, P. B. Luh, and K. R. Pattipati. A practical approach to job-shop scheduling problems. *IEEE Trans. on Robotics & Automation*, 9:1–13, 1993.

[10] B. Kouvaritakis and M. Cannon. *Model Predictive Control: Classical, Robust and Stochastic.* Springer, London, UK, 2015.

[11] D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming (4th ed.).* Springer, NY, NY, 2016.

[12] T. E. Morton and D. Pentico. *Heuristic Scheduling Systems.* John Wiley, NY, NY, 1993.

[13] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization.* Wiley, NY, NY, 1988.

[18] Currently, the most celebrated result coming from these decomposition approaches is the "shifting bottleneck" heuristic [1] for the job shop scheduling problem.

[14] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, UK, 2010.

[15] M. Pinedo. *Scheduling*. Prentice Hall, Upper Saddle River, NJ, 2002.

[16] S. Reveliotis and E. Roszkowska. On the complexity of maximally permissive deadlock avoidance in multi-vehicle traffic systems. *IEEE Trans. on Automatic Control*, 55:1646–1651, 2010.

[17] S. A. Reveliotis. Conflict resolution in AGV systems. *IIE Trans.*, 32(7):647–659, 2000.

[18] H. D. Sherali, S. C. Sarin, and R. Desai. Models and algorithms for job selection, routing and scheduling in a flexible manufacturing system. *Annals of Operations Research*, 26:433–453, 1990.

[19] M. Weiss. Semiconductor factory automation. *Solid State Technology*, pages 89–96, 1996.