Computational Management Science manuscript No.
(will be inserted by the editor)

Optimization and Sustainable Development

Leo Liberti

August 1, 2014

Abstract In this opinion paper, I argue that "optimization and sustainable development" indicates a set of specific engineering techniques rather than a unified discipline stemming from a unique scientific principle. On the other hand, I also propose a mathematical principle underlying at least some of the concepts defining sustainability when optimizing a supply chain. The principle is based on the fact that since demand constraints are usually expressed as inequalities, those which are not active at the optimal solution imply the existence of some wasted activity, which may lead to an unsustainable solution. I propose using flow-type equation constraints instead, which help detect unsustainability through infeasibility.

1 Introduction

This paper investigates the relationship between Optimization and Sustainable Development (OSD).

By *optimization*, I mean a set of mathematical results and algorithmic techniques for solving problems of the following very general form:

$$\mathsf{opt}_{x \in X} f(x), \tag{1}$$

where X is a suitably defined set (often a subset of some Euclidean space), f is a map from X to $\mathbb{R}^d$ (in the case of a single objective problem, which is the main focus of the present paper, $d = 1$), and opt indicates either minimization or maximization. The concision and generality of the "optimality principle" given by Eq. (1) are mathematically remarkable; its syntax and semantics can be expressed by means of formal languages. By further specifying conditions on f, X, d, mathematical formulations can be obtained to describe most optimization problems. Many such problems are very useful in the solution of more concrete problems arising in technical, scientific and social fields of human knowledge.

Defining *sustainable development* (SD) in conjunction with optimization is harder. Although the term was introduced during the 1960s and 1970s, it was popularised in the Brundtland Report, released by the United Nations in 1987: "sustainable development is

L. Liberti
LIX, École Polytechnique, F-91128 Palaiseau, France E-mail: `liberti@lix.polytechnique.fr`

development that meets the needs of the present without compromising the ability of future generations to meet their own needs" [21]. The Brundtland definition was criticized for being too vague [4]. The main reason why this definition is problematic with respect to mathematical optimization is the explicit mention of time: formulating the Brundtland definition in terms of Eq. (1) would require a precise planning of all human activity for all future times, clearly an impossible task. On the other hand, one of the main goals of quantitative economics is to predict the future evolution of socio-economic systems in time, and this is often done by means of mathematical optimization tools, which obviously take time into account. Moreover, mathematical optimization is routinely employed to decide the optimal scheduling of activities over a limited time period, or under the assumption that the decisions are periodic. I shall discuss time-dependency quantitatively in Sect. 5.

Many special interests groups adapted the terms to their own agenda. An internet search shows that SD has been related to the following terms (among others):

- a set of laws and regulations for manufacturing firms
- a moral obligation for all humans
- production with low CO_2 emissions
- reliance on "clean energy"
- recycling waste
- using public/clean transportation instead of private cars
- making sure hazardous material transportation is equitable
- development that can last forever
- making sure the earth can survive
- preservation of biodiversity
- helping third-world countries.

Although the above terms are more specific than "sustainable development", and some of them at least seem to be able to withstand some degree of formalization, none of them exactly catches the essence of all problems arising in SD.

Part of my duties, as the scientific director of the "Optimization and Sustainable Development" (OSD) Chair sponsored by Microsoft at the Laboratoire d'Informatique at École Polytechnique, is to apply optimization techniques to problems arising in SD. Whilst in this role, I observed a wild diversity of optimization problems related to SD. A very natural question occurs: is there any mathematical principle underlying these problems? Insofar as the vast majority of optimization problems can be formulated in a form given by Eq. (1), it would be useful to identify a similarly general principle underlying SD problems. The intersection of the two problem classes (optimization and SD) could then refer to the conjunction of the two corresponding principles. According to my own experience (see Sect. 2 below), a possible *a posteriori* definition of SD is a set of engineering practices whose environmental impact is controlled: but I see no way to generalize this definition to a formal mathematical principle.

This paper attempts to shed some light on the question: *is there a mathematical principle underlying problems related to SD?* Although I believe the answer to be negative in general, I propose that a good starting point is that SD implies a development that can last forever. In Sect. 3, I discuss a treatment of this principle from the point of view of avoiding waste in supply chain networks. More precisely, when demand constraints are formulated as inequalities of the type "production $\geq$ demand", some of them may be inactive at some optimal solution, which implies the existence of some waste; I propose using flow-type equations instead. Simply adding a surplus variable to yield "production $+$ waste $=$ demand" is not sufficient to generalize this no-waste principle to supply-chain network problems, however, as I shall argue in Sect. 4.3. To this purpose, I introduce the concept of a *transformation flow*. As demonstrated in Sect. 4, transformation flows provide a much more effective tool.

The rest of this paper is organized as follows. In Sect. 2, I argue that optimization problems arising in SD are so diverse that it is difficult to relate them all formally by a unique underlying mathematical principle. In Sect. 3-4, I discuss a set of transformation flow equations which can be used to model sustainable production processes. In Sect. 5, I address the issue of time-dependent transflows.

2 Diverse problems

This section lists some of the optimization problems I encountered whilst working for the OSD Chair. I hope the reader will be convinced that they differ in almost everything, aside from the optimization principle. I thus hope to provide sufficient evidence in favour of my negative claim, i.e. that there is no unifying principle underlying all problems in SD. Since this statement is expressed in natural language, it is neither true nor false: it is simply my reasoned opinion.

2.1 Scheduling nuclear plant outages

This problem was the subject of the popular yearly ROADEF/EURO Computational Challenge (`http://challenge.roadef.org`), whose 2010 edition, sponsored by Electricité de France (EDF) was won by a team belonging to the OSD Chair, including V. Jost and D. Savourey. From the document detailing the 2010 Challenge, EDF power generation facilities in France stand for a total of 98.8GW of installed capacity. The varied range of EDF facilities mixes all forms of energy: thermal (nuclear, coal, fuel oil and gas), hydraulic and other renewable energies. Most of the electricity EDF generates in France is produced by thermal power plants: 90% in 2008, 86% of which by nuclear power plants. The outages of thermal power plants, specially the nuclear ones, which have to be repeatedly shut down for refuelling and maintenance, must be scheduled in such a way as to satisfy safety, maintenance, logistics and plant operation constraints, as well as to lead to production programs of minimum average cost. The time horizon of decisions is 5 years, and the solution must fulfil a stochastic demand robustly. The solutions adopted by national energy suppliers such as EDF have a huge overall impact on the environment: in this sense, this problem is concerned with optimization and SD.

Power plants are of two types: those that can be continuously supplied with fuel (Type 1), and those that can only be supplied with fuel at regular intervals (Type 2). Type 2 plants cannot continue to produce whilst being supplied. The customer demand is given as a set of uncertain scenarios. Production cost for Type 1 plants is proportional to the power output and depends on the scenario and time step. Refuelling Type 2 plants leads to a cost which is proportional to the amount of available fuel and the time step, but not the demand scenario. The total amount of fuel for Type 2 plants is given at the outset of the planning period. The following decision variables are considered:

- dates for outages to refuel Type 2 power plants (for each time step)
- the amount of fuel used in supplying a Type 2 power plant (for each time step)
- production levels for Type 1 and Type 2 power plants (for each time step and demand scenario).

The exact functional forms of objective function and constraints are detailed in [25]. This problem is a combination of resource allocation and scheduling. The solution approach proposed by the winning OSD Chair team is based on solving an Integer Linear Program (ILP) that relaxes some of the technical constraints, and then heuristically trying to achieve feasibility with respect to all constraints.

2.2 The concentrator placement problem

Many large-scale electricity distribution networks are in the process of being transformed into "smart grids". This means that the network must be: tolerant to occasional demand spikes and hardware failures, able to accommodate renewable sources of energy at any scale (large wind farms as well as customer-installed solar panels), easily monitored in its every part, and user-transparent as regards costs, which must be billed at the smallest possible monetary unit. In order for the grid to be "smart", then, very detailed consumption data are required: specifically, each end consumer must be provided with a "smart meter" that can collect all sorts of data, transmit it to central servers, and receive information on the current state of the network, as well as on the user account.

In France, the electricity provider, EDF, and the network handler, Electricité Réseau Distribution France (ERDF) are two separate entities. The latter is responsible for routing the electricity to the end consumers. Within the framework of the nationally sponsored project Chip2Grid, the OSD Chair is currently working on the problem of routing the masses of data collected by the smart meters to the central servers. The overall solution consists in placing some intermedate data aggregators and retransmitters called *concentrators*; the communication between smart meters and concentrators occurs on the same cables used to transmit power, along a different frequency. Concentrators are costly piece of equipment, and can serve at most a given number of meters: the problem is then to choose locations on the distribution network where fewest concentrators can be placed so as to cover the whole set of meters.

Depending on whether the network is considered as a subset of points in $\mathbb{R}^2$ or as an abstract graph, this problem can be seen as a geographical location problem on a network, as well as a vertex covering problem. There are several technical constraints, such as voltage drops along the distribution lines, that must be taken into account. Also, although the French network topology is close to a tree, other countries have other types of topologies, such as meshes.

This problem is not about sustainability *per se*, but it serves the purpose of improving the efficiency and accountability of smart grids, which in turn is crucial for a power network to be sustainable.

2.3 Cost of equitability

This problem arose as part of the Ph.D. work of F. Roda [26] within the OSD Chair. It concerns the difference between risk and damage in a problem concerning the transportation of hazardous materials (hazmat) on a geographical network. Hazmat transportation is usually regulated so that the risk is equitably split among the administrative partitions composing the geographical region of interest. "Equitably" here means that no region is at a disadvantage with respect to the risk of a catastrophic accident. We showed that equitable solutions do not necessarily minimize the damage, and that, to put it bluntly, being equitable might cost lives. This is not surprising, as risk is an evaluation of the costs due to a future, uncertain disaster, whereas the damage is the cost due to a past and certain disaster (see Fig. 1).

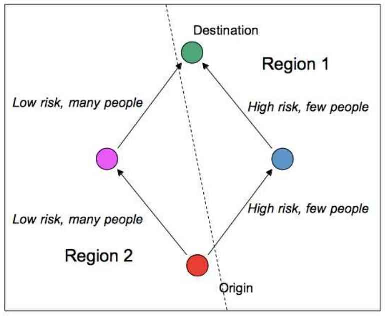

Fig. 1 Minimize damage: route through region 1. Risk equitability: route through both regions (this may cost lives).

Consider a geographical road network modelled by a directed graph $G = (V,A)$, together with a set $C \subseteq V \times V$ where each $(s,t) \in C$ represents the need for transporting hazmat from node s to node t — in other words, there is a pair (s,t) for each batch of hazmat to be transported. For each arc $(u,v) \in A$ and each $(s,t) \in C$ we are given: (a) the risk p^{st}_{uv} of an accident occurring on the given arc and for the given hazmat; (b) the damage Δ^{st}_{uv} (expressed in monetary units) caused by an accident; (c) the cost c^{st}_{uv} of transporting a unit of hazmat (s,t) on arc (u,v). The traditional risk of an accident occurring when transporting hazmat (s,t) over arc (u,v) is $r^{st}_{uv} = p^{st}_{uv}\Delta^{st}_{uv}$. Minimum cost routes with and without risk equitability constraints are shown for a simple case in Fig. 2.

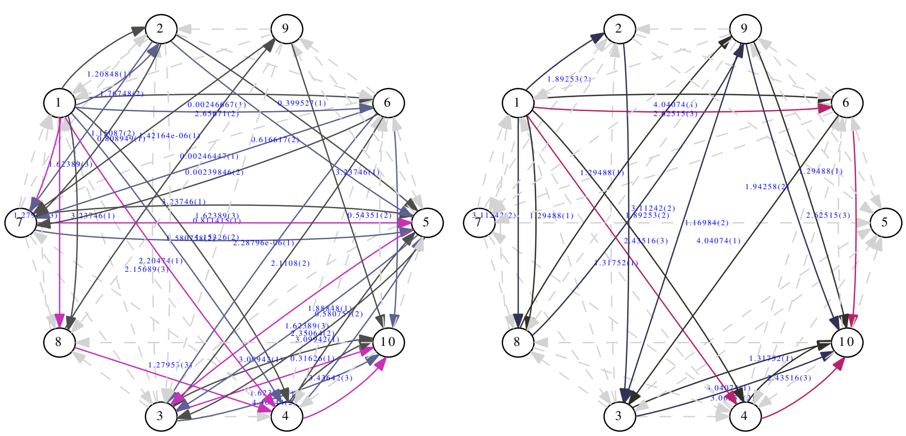

Fig. 2 On the left, solution with risk equitability (total damage: 855.045 monetary units). On the right, without risk equitability (total damage: 677.997 monetary units).

2.4 Multi-feature shortest paths

Traffic has a huge impact on energy consumption and CO_2 emissions, which adversely impact the environment. GPS-enabled devices allow drivers to compute shortest paths, but at the time this project was carried out, these devices did not use to take traffic data into account. Even now, most of the low-cost GPS devices do not consider traffic data. At the OSD Chair we have been working (through two Ph.D. students, G. Nannicini and D. Kirchler) with a traffic data handling firm, in order to develop shortest path algorithms which consider continuously varying traffic data [20, 19]. Our algorithms are able to efficiently compute shortest paths on large-scale road networks (such as Europe), with respect to different objectives (such as minimization of time-of-travel, CO_2 emissions, and fuel costs), whilst keeping track of multi-modality constraints, which can encode anything from changing mode of transportation to stopping at certain sites. At the core of our algorithms are bi-directional implementations of traditional multi-criteria shortest paths algorithms, be they label setting or label-correcting, and a pre-processing of traffic data that minimizes the impact of its continuously changing nature. Multi-modality is achieved by pairing a finite-state automaton to the main path search algorithm. The state transitions of this automaton are labeled with symbols, which are employed by the main algorithm to distinguish between those arcs that are allowed whilst exploring a node star, and those that are not [15].

2.5 Diversity in sustainability

The four problems above represent a small part of the problems we worked on in the OSD Chair [12], but they show that SD can appear in many different forms. Problems focusing on improving the production and/or use of energy are categorized as SD because the current mainstream energy production ways are clearly unsustainable in the long term. Any problem involving the reduction of CO_2 emissions, or any method that can be potentially applied to such problems, is categorized as SD because if our industrial society keeps emitting CO_2 as it does now, the environmental changes might be major, and even catastrophic — clearly an unsustainable situation. If there is an accident involving hazardous materials, the environmental damage might be extensive, and in the long run, a situation where the risks associated to this type of logistics are unchecked is unsustainable. Thus, analysing the trade-off between risk and damage minimization in this setting is also classified as SD.

Again, these problems differ wildly. A common point is that all of the involve pairwise relations, which are modelled as graphs. In particular, the concentrator placement problem, the equitability problem, and the shortest path problems are defined on physical networks; the scheduling problem only involves the formalization of a distance relation on outages. This common feature reflects in part the very real issue that sustainability often has to do with the interaction of human technology with the planet, which often involves the geographical transportation of commodities. However, it is also in part due to my own disposition to employ graphs and their language to model problems. Although, due to my own bias, I will not emphasize this feature as a main driver behind OSD, it does in part motivate the employment of network flows in the following.

3 Development that can last forever

I shall give a unifying principle for an important class of optimization problems in SD, i.e. those problems that concern certain sustainable supply chains that can be modelled by flows of matter, energy, resources or information. In order to do so, I shall consider the lexical meaning of the words "sustainable development": it describes a development that can exist over time. To take this concept to its abstraction, SD *can last forever*. I shall restrict the meaning of "development" to the development of a product (artifact or service) into its own environmental, social, technological and commercial environment. In other words, I shall consider the supply chain for a product or system thereof. Although I am still writing in natural language, this concept is precise enough so that I can provide a formalization.

In what follows, I shall represent flows of matter as a mathematically defined quantity travelling along a network; matter transforms when passing through the network nodes, in a way that is formally described by equations, which impose a generalized form of mass/energy balance: what flows into a node can be transformed, but what flows out of the node must correspond exactly to that transformation. This idea is not new in SD — it is mentioned e.g. in [4], but in informal language, and limited to a study of qualitative type, where ideas are expressed in the form general guidelines for policymakers. In this paper, I give a quantitative and formal version of the idea, and express it in a form that can be understood and put in practice by mathematically oriented engineers and decision makers.

By characterizing SD as "development that can last forever", the *forever* can be taken literally, in which case time is not explicitly part of the modelling paradigm, or just as an extremely long but finite horizon — and engineering literature reports time-indexed models with very long horizons, stretching many years, see e.g. [23]). The optimization techniques used to solve such models often involve the integration of the time component within the flow network itself, by means of the addition of "event nodes" which represent a plant at a particular time [22]. Most of the paper will be devoted to the *steady-state* case, where time is not explicitly part of the model. Sect. 5 will propose a generalization to the time-dependent case.

3.1 The supply chain network

By *development* I mean an interconnected network of processes that transform input flow into output flow (see Fig. 3). The network is modelled by means of a digraph $G = (V, A)$

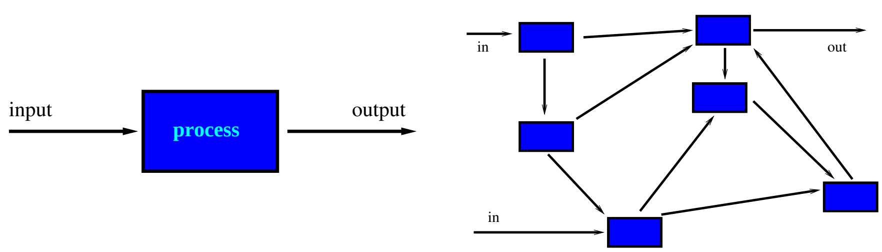

Fig. 3 A process (left) and a process network (right).

where V is a set of processing nodes, and A is the set of arcs (or links). G processes a set K of commodities. Each node $v \in V$ has two functions (see Fig. 4):

1. v transforms an input commodity $h \in K$ into an output commodity $k \in K$, so that one unit of h yields π_v^{hk} units of k;
2. v may have a surplus of ω_v^k units of commodity k.

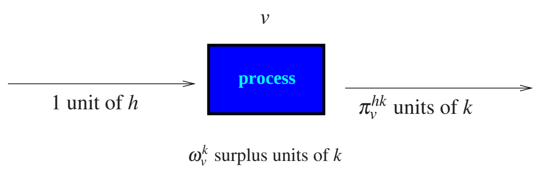

Fig. 4 Basic node transformation.

Standard graph notation is used to express the graph topology: for each $v \in V$, the *outward star* $N^+(v)$ of v consists of the nodes $u \in V$ such that $(v,u) \in A$; and the *inward star* $N^-(v)$ of v consists of the nodes $u \in V$ such that $(u,v) \in A$.

3.2 Transformation flows

I am going to approximate the movement of commodities over the arcs of the process network by means of a continuous flow: for each commodity $k \in K$ and arc $(u,v) \in A$ let f_{uv}^k be the flow (quantity of commodity per time unit) of k over (u,v). The concept of flows is well known in the study of networks. If no commodity transformation occurs at nodes, then the *flow conservation principle* states that flow is conserved across nodes, compatibly with surplus capacity:

$$\forall v \in V \qquad \sum_{u \in N^-(v)} f_{uv} - \sum_{u \in N^+(v)} f_{vu} = \omega_v. \tag{2}$$

In multi-commodity flow networks, the flow conservation principle holds for each separate commodity:

$$\forall v \in V, k \in K \qquad \sum_{u \in N^-(v)} f_{uv}^k - \sum_{u \in N^+(v)} f_{vu}^k = \omega_v^k. \tag{3}$$

In the present case, the natural extension of Eq. (2)-(3) is the following:

$$\forall v \in V, k \in K \qquad \sum_{h \in K} \pi_v^{hk} \left(\omega_v^h + \sum_{u \in N^-(v)} f_{uv}^h \right) - \sum_{u \in N^+(v)} f_{vu}^k = \omega_v^k. \tag{4}$$

Eq. (4) says that at each process node v, the amount of commodity k consists of whatever is transformed into k from every commodity h (either as a surplus at v or arriving on each arc incident to v), plus whatever quantity of k is a surplus at v.

I shall thus formally define a *process network* as a quadruplet (G,K,π,ω) where $G = (V,A)$ is a digraph, K is a set, $\pi : V \times K^2 \to [0,1]$ and $\omega : V \times K \to \mathbb{R}_+$ are two scalar functions. In physical processes, the transformation factor π_v^{hk} can neither be negative, nor be greater than one. Moreover, the function π must be such that:

$$\begin{aligned} &\forall m \in \mathbb{N}, \mathbf{k} = (k_i \mid i \le m) \in K^m, \mathbf{v} = (v_i \mid i \le m) \in V^m \\ &\qquad (v_1 = v_m \wedge \{(v_i, v_{i+1}) \mid i < m\} \subseteq A) \to \pi_{v_1}^{k_1 k_2} \cdots \pi_{v_m}^{k_m k_1} \le 1. \end{aligned} \tag{5}$$

Eq. (5) asserts that no iterated transformation over a cycle in G can result a cumulative transformation factor that is greater than one.

Definition 1 (Transflow) A *transformation flow* (also called *transflow*) over a process network $\mathcal{N} = (G, K, \pi, \omega)$ is a function $A :\to \mathbb{R}$ satisfying the *transflow equations* (4).

3.3 Resource limits

The activity of a supply chain is obviously limited by available resources. These constraints can be enforced rather naturally by means of generalized budget constraints, either on arcs or on nodes of the network. For arcs (u, v) and each commodity k, consider an operator $\gamma_{uv}^k : A \to \mathbb{R}$ and a scalar $C_{uv}^k \in \mathbb{R}$. The arc budget constraints are:

$$\forall (u,v) \in A \quad \gamma_{uv}^k (f_{uv}^k) \leq C_{uv}^k. \tag{6}$$

For nodes v, let $\beta_v^k : V \to \mathbb{R}$ and $B_v^k \in \mathbb{R}$. The node budget constraints are:

$$\forall v \in V, k \in K \quad \beta_v^k \left(\omega_v^k + \sum_{u \in N^-(v)} f_{uv}^k \right) \leq B_v^k. \tag{7}$$

In their simplest form, both γ and β can simply be multiplicative constants — this is appropriate when the resource measures are proportional to the quantity of commodity. Budget constraints (7) may be aggregated over sets of nodes, as required by the exact nature of the problem. If arcs represent the movement of commodity over a route network, the arc measure often involves transportation capacity (for example number of lorries required to transport each commodity unit); capacity constraints (6) can be aggregated over different commodities sharing the same link. Node measures often involve the cost of transforming a commodity unit.

3.4 Quality requirements

Quality measures are defined in function of the quantities of processed commodities may give rise to generalized quality constraints over arcs and nodes. These are formally very similar to Eq. (6)-(7), with the $\leq$ relation operator replaced by a $\geq$. Quality measures may include minimum amount of output commodities that the market needs, desirability, and so on.

3.5 Objective functions

Generalized budget and quality constraints may also be cast in the form of objective functions: one may wish to minimize the operations costs, instead of limit them from above; or to maximize the quality of the output products, instead of bounding them from below.

4 Optimizing the chain

In Sect. 3, I introduced transflows and their corresponding constraints. These are mathematical expressions involving the symbols $f, \pi, \omega, \beta, \gamma, B, C$. In order to define optimization problems over transflows, one must clearly identify which symbols are decision variables, and which are parameters. The latter encode the input of the problem, whilst the former will be assigned values by the optimization algorithms tasked with solving the problem (in other words, they represent the problem output).

The transflow framework is general enough to accommodate several practical optimization problems arising in the literature. For example, if f are decision variables and the other symbols are parameters, one obtains a transflow routing problem: only the routing of commodities is to be determined optimally. If f, ω are decision variables, one also decides on the allowable surplus at each node. Deciding π controls the efficiency level of the transformations. Deciding B,C is related to sensitivity analysis: what is the tightest possible resource limit (or quality requirement)? Deciding β, γ might be useful in planning situations, during the conception of the infrastructure linked to the supply chain.

In the rest of the section, I shall give examples of *transflow problems*, i.e. optimization problems arising from transflows. I will employ the language of Mathematical Programming (MP) in order to describe the optimization problems formally: MP problems consist of a set of parameter symbols encoding the input data; a set of decision variable symbols encoding the solution after the optimization process has taken place; one or possibly more objective functions to be optimized, and zero or more constraints to be satisfied. Objective and constraints are expressed by means of mathematical expressions, having parameters and decision variables as arguments. I shall also make use a very simple, illustrative process network example, shown in Fig. 5. Node 1 represents a plant, where some kind of material transformation occurs, nodes 2-5 represent input stock nodes, and nodes 6-9 represent output stock nodes. The material flow follows the arrows from nodes 2-5 into node 1, and from node 1 out to nodes 6-9.

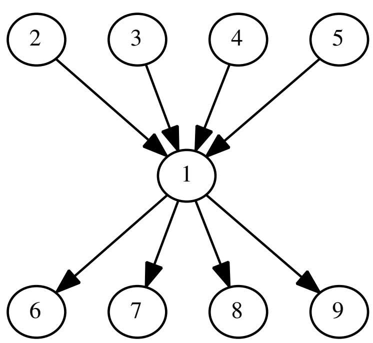

Fig. 5 A simple process network with one transformation plant.

4.1 Relations to other known problems

Transflow problems are generalizations of several other types of problems found in the MP literature. They can be seen as multicommodity flow problems [14] with a variable set of origin/destination nodes. The variant with fixed percentages, and limited to a single plant, can be seen as a mixing/blending problem, one of the classic applications of LP [10]. The variants which decide percentages (Sect. 4.4) are close to pooling problems [1,3,18,9]. The time-dependent variants (Sect. 5) are close to discrete-time dynamic network flows [2]. In the time-expanded setting, the node surplus variables ω can also be used to partly address delays at nodes (in which case they may be considered as denoting "storage at a node"), which may arise in dynamic network flows [16].

4.2 Routing optimization

The first MP I will present decides the optimal routing of different transflows through the process network $\mathscr{N} = (G, K, \pi, \omega)$, where $G = (V, A)$ is a digraph. Let $\phi : A \times K \to \mathbb{R}$ be a function such that ϕ^k_{uv} is the cost routing one unit of commodity $k \in K$ over the arc $(u, v) \in A$. For each $k \in K$, let us also consider two subsets S^k, T^k of nodes in V, called respectively *sources* and *targets* or *destinations* of the commodity k. Source nodes are also sometimes called *production nodes*, and target nodes are also sometimes called *demand nodes*. Source nodes have no incoming flow of commodity k, and target nodes have no outgoing flow of commodity k. In this example, I assume for simplicity that process nodes (i.e. all nodes that are not sources or targets) have no surplus capacity. Given source levels s^k_v for commodity k and source vertex $v \in S^k$, and demand levels d^k_v for commodity k and target vertex $v \in T^k$, the MP is as follows.

$$\left.\begin{array}{rrcl} \min\limits_{f,\omega} & \sum\limits_{\substack{(u,v)\in A\\ k\in K}} \phi^k_{uv} f^k_{uv} & & \\ \forall k \in K, v \in S^k & \omega^k_v & = & s^k_v \\ \forall k \in K, v \in T^k & \omega^k_v & = & d^k_v \\ \forall k \in K, v \in V \smallsetminus (S^k \cup T^k) & \omega^k_v & = & 0 \\ \forall k \in K, v \in S^k, u \in N^-(v) & f^k_{uv} & = & 0 \\ \forall k \in K, v \in T^k, u \in N^+(v) & f^k_{vu} & = & 0 \\ & f & \multicolumn{2}{l}{\text{is a transflow.}} \end{array}\right\} \tag{8}$$

The MP formulation in Eq. (8) is a Linear Program (LP), for which efficient solutions methods exist, capable of determining, in polynomial time: (a) whether the given instance is bounded or not; (b) whether the given instance is feasible or not; (c) in the case of a bounded feasible instance, its optimal solution f^* [24]. This implies that one can efficiently determine whether a given supply chain problem is sustainable or not: if the transformation occurring in the process nodes does not satisfy the conservation of transflow, then running the plant indefinitely will accumulate an infinite amount of surplus commodity in those process nodes where

$$\sum_{h\in K} \pi^{hk}_v \sum_{u\in N^-(v)} f^{h*}_{uv} < \sum_{u\in N^+(v)} f^{k*}_{vu}$$

(recall that we assumed $\omega^k_v = 0$ at all process nodes).

Example 2 Consider the network in Fig. 5 with four commodities: $K = \{1,2,3,4\}$. Let nodes $2,3$ be sources for commodity 1 at source level 1, let nodes $4,5$ be sources for commodity 2 with source level 1, let nodes $6,7$ be targets for commodity 3 with demand level 1, and nodes $8,9$ be targets for commodity 4 with demand level 1. The process node 1 transforms one unit of commodity 1 into one unit of commodity 3 *and* one unit of commodity 4, and the same holds for commodity 2. Formally:

1. $S^1 = \{2,3\}$, $S^2 = \{4,5\}$, $T^3 = \{6,7\}$, $T^4 = \{8,9\}$ (all other S^k, T^k are set to $\varnothing$);
2. $\pi_1^{13} = \pi_1^{14} = \pi_1^{23} = \pi_1^{24} = 1$ (all other π_v^{hk} are set to 0);
3. $s_2^1 = s_3^1 = s_4^2 = s_5^2 = d_6^3 = d_7^3 = d_8^4 = d_9^4 = 1$ (all other s_v^k and d_v^k are set to 0).

Does there exist a transflow f satisfying Eq. (8)? Since $s_2^1 = s_3^1 = 1$, there are two units of commodity 1 entering node 1, and by $\pi_1^{13} = \pi_1^{23} = 1$ these two units of commodity 1 are transformed into two units of commodity 3 at node 6, which are too many since there is a unit demand for commodity 3 at node 6. Although the small size of this example makes it is easy to establish infeasibility, this is much more complicated if the network size increases. The same conclusion can be reached for any unsustainable instance by solving Eq. (8) and observing its infeasibility. In this case, I used AMPL's pre-solving procedure [11], to get:

```
presolve, variable f[2,1,2]:
impossible deduced bounds: lower = 0, upper = -2
presolve, variable f[4,1,1]:
impossible deduced bounds: lower = 0, upper = -2
presolve, variable f[4,1,1]:
impossible deduced bounds: lower = 0, upper = -2
presolve, variable f[4,1,1]:
impossible deduced bounds: lower = 0, upper = -2
Infeasible constraints determined by presolve.
```

In order to obtain a sustainable process network, the availability at the source nodes can be limited to smaller values, or the process at node 1 can be changed so that $\pi_1^{13} = \pi_1^{14} = \pi_1^{23} = \pi_1^{24} = 0.5$. The corresponding solution is shown in Fig. 6. □

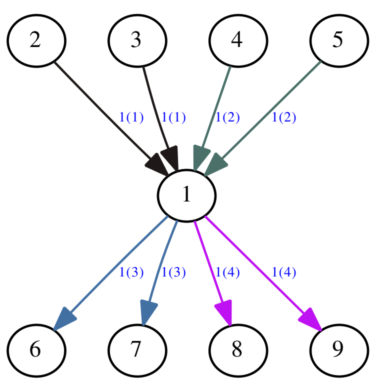

Fig. 6 A sustainable process network. Arc labels are $f^*(k)$, where f^* is the value of the optimal flow, and k is the commodity index.

Example 3 Modelling the network in Example 2 without transflows can be done as follows: for all $k \in K, v \in V$ let x_v^k be the quantity of commodity k at node v. Then the constraints

$$\forall k \in K, v \in S^k \quad x_v^k \leq s_v^k \tag{9}$$
$$\forall k \in K, v \in T^k \quad x_v^k \geq d_v^k \tag{10}$$
$$\forall k \in K, v \in V \smallsetminus (S^k \cup T^k) \quad x_v^k \leq \sum_{h \in K} \pi_v^{hk} \sum_{u \in N^-(v)} x_u^h \tag{11}$$

are the (unsustainable) standard way to model the process. The solution x given by:

- $x_2^1 = x_3^1 = x_4^2 = x_5^2 = 1$
- $x_1^3 = x_1^4 = 2$
- $x_6^3 = x_7^3 = x_8^4 = x_9^4 = 2$
- all other x_v^k are set to 0

is evidently feasible for Eq. (9)-(11), but it is not sustainable, since the constraints of Eq. (10) have a nonzero slack at x, which implies the existence of unaccounted waste. □

4.3 Necessity or usefulness?

Example 3 shows that modelling by transflows links the question of sustainability to the feasibility of the problem: a SD-aware solver would reject outright any set of data for which the supply chain network produces waste. By having decision support software refuse to provide a solution for such instances, it is hoped that decision-makers will be made more aware about SD issues.

Are transflows really necessary, however? Mathematically, the fundamental difference between the models in Examples 2 and 3 is between equations and inequalities. Whereas the former accounts for all types of waste (which are seen as commodities), the latter does not. Although this is a very basic principle, and, for experienced mathematical programmers, not a very deep one, in my experience both academics and practitioners often model problems using inequalities rather than equations whenever possible. This partly justifies transflows as a useful modelling tool.

Transflows, however, achieve considerably more than just replace inequality with equations. I mentioned in the introduction that rewriting demand constraints such as "production ≥ demand" as "production + waste = demand" does not suffice to settle the SD issue in supply chain network modelling. Consider a constraint $x_v^k + w_v^k = d_v^k$, where x is the quantity of commodity k at node v, d_v^k is the demand for commodity k at v, and w_v^k is the corresponding waste. Usually, the cost for waste is positive, so that w_v^k will be minimized: if the cost is large, chances are that $w_v^k = 0$ at the solution, which yields a sustainable process: but there is no guarantee that this will be the case. Quite on the contrary, because often the costs associated to waste are borne by society as a whole and thus difficult to quantify, it is possible that waste variables will take nonzero values at the optimum. This is why I believe that modelling by transflows is not only useful, but actually *necessary* for SD purposes.

4.4 Deciding percentages

The above routing model is appropriate for certain types of transformation processes, but not for others. When exploiting the mass balance of the transformation of coal, for example, the following holds:

1 coal $\longrightarrow$ 0.05 tar + 0.015 benzol + 500 methane

This means that one unit of coal is transformed into 0.05 units of tar *and* 0.015 units of benzol, *and* 500 units of methane. Transflows with fixed π's can be used for exactly this purpose. Consider another type of transformation process, occurring in the dairy industry: some fraction of the input milk is pasteurised, another is transformed into cheese, yet another into butter, and the rest is sold to other industries. Instead of only deciding how much milk should flow into the transformation plant, the values for these fractions should also be decided optimally. To this purpose, I introduce the fraction p_v^{hk} of commodity h to be transformed into commodity k at process node v, as an additional decision variable, and rewrite the transflow conservation Eq. (4) as:

$$\forall v \in V, k \in K \quad \sum_{h \in K} p_v^{hk} \left(\omega_v^h + \sum_{u \in N^-(v)} f_{uv}^h \right) - \sum_{u \in N^+(v)} f_{vu}^k = \omega_v^k \tag{12}$$

$$\forall v \in V, h \in K \qquad \sum_{k \in K} p_v^{hk} = 1 \tag{13}$$

This is essentially the same as considering π as decision variables in Eq. (4). The resulting MP, under similar assumptions as that of Eq. (8), is:

$$\left. \begin{array}{rrcl} \min\limits_{f,p,\omega} & \sum\limits_{\substack{(u,v)\in A \\ k \in K}} \phi_{uv}^k f_{uv}^k & & \\ \forall k \in K, v \in S^k & \omega_v^k & = & s_v^k \\ \forall k \in K, v \in T^k & \omega_v^k & = & d_v^k \\ \forall k \in K, v \in V \smallsetminus (S^k \cup T^k) & \omega_v^k & = & 0 \\ \forall k \in K, v \in S^k, u \in N^-(v) & f_{uv}^k & = & 0 \\ \forall k \in K, v \in T^k, u \in N^+(v) & f_{vu}^k & = & 0 \\ & f, p, \omega \text{ satisfy Eq. (12)-(13).} & & \end{array} \right\} \tag{14}$$

Because of the products between p and f in Eq. (12), The MP formulation (14) is a nonconvex Nonlinear Program (NLP), which can only be solved using appropriate algorithms, such as spatial Branch-and-Bound (sBB) [5].

Example 4 The sustainable process network of Example 2 can be tested using the COUENNE solver [5]. The solution is exactly the same as shown in Fig. 6, with $p_1^{13} = p_1^{14} = p_2^{23} = p_2^{24} = 0.5$. Example 3 can be adapted to this setting to show how modelling with inequalities yields an unsustainable solution. □

A limitation of the sBB algorithm is that it cannot certify optimality and infeasibility exactly, but only approximately to within a given ε tolerance. Luckily, however, the mathematical form of Eq. (14) is special, in that it can be reformulated exactly [17] to an LP. Define two new sets of variables $x : V \times X \to \mathbb{R}$ and $z : V \times K \times K \to \mathbb{R}$ defined as follows.

$$\forall v \in V, k \in K \quad x_v^k = \omega_v^k + \sum_{u \in N^-(v)} f_{uv}^k \tag{15}$$

$$\forall v \in V, h, k \in K \quad z_v^{hk} = p_v^{hk} x_v^h. \tag{16}$$

Now multiply Eq. (13) by x_v^h, for all $v \in V, h \in K$, and use Eq. (16) to replace the resulting bilinear terms $p_v^{hk} x_v^h$ by z_v^{hk}. This yields:

$$\forall v \in V, h \in H \quad \sum_{k \in K} z_v^{hk} = x_v^h. \tag{17}$$

Moreover, using the same replacement of bilinear terms, the nonlinear transflow conservation equations Eq. (12) become:

$$\forall v \in V, k \in K \quad \sum_{h \in K} z_v^{hk} = \omega_v^k + \sum_{v \in N^+(v)} f_{uv}^k. \tag{18}$$

Consider now the MP given by:

$$\left.\begin{array}{rrcl}
\min\limits_{f,p,x,z,\omega} & \sum\limits_{\substack{(u,v)\in A \\ k\in K}} \phi_{uv}^k f_{uv}^k & & \\
\forall k \in K, v \in S^k & \omega_v^k & = & s_v^k \\
\forall k \in K, v \in T^k & \omega_v^k & = & d_v^k \\
\forall k \in K, v \in V \smallsetminus (S^k \cup T^k) & \omega_v^k & = & 0 \\
\forall k \in K, v \in S^k, u \in N^-(v) & f_{uv}^k & = & 0 \\
\forall k \in K, v \in T^k, u \in N^+(v) & f_{vu}^k & = & 0 \\
 & \multicolumn{3}{l}{f,p,x,z,\omega \text{ satisfy Eq. (13),(15)-(18).}}
\end{array}\right\} \tag{19}$$

A MP formulation Q is an *exact reformulation* of another MP formulation P if there exists a mapping ϕ from the set $\mathscr{G}(Q)$ of global optima of Q onto the set $\mathscr{G}(P)$ of those of P. An exact reformulation Q is useful if it is easier to solve than P is.

Theorem 5 *If $f,\omega \geq 0$, Eq.* (19) *is a useful exact reformulation of Eq.* (14)*.*

Proof Let (f,ω,x,z) be an optimal solution of Eq. (19). First, remark that by $f,\omega \geq 0$ and Eq. (15), we have $x \geq 0$, which by Eq. (16) also implies $z \geq 0$. For all $v \in V$ and $h,k \in K$ such that $x_v^h > 0$, we define $p_v^{hk} = \frac{z_v^{hk}}{x_v^h}$; whenever $x_v^h = 0$, we define $p_v^{hk} = 0$. This satisfies Eq. (16). Since Eq. (19) was obtained from Eq. (14) by replacing the bilinear terms in p,f with the variables z, Eq. (12) holds, which implies that (f,ω,p) is feasible for Eq. (14). Since the objective function only depends on f, (f,ω,p) is also optimal for Eq. (14). Usefulness follows because Eq. (14) is a nonconvex NLP, whereas Eq. (19) is an LP.

Notice that Thm. 5 can be generalized to hold independently of the constraints fixing the values of ω in source, target and process nodes in Eq. (14) and Eq. (19). Moreover, the restriction on f,ω to be nonnegative simply reflects physical reality: negative flows really correspond to positive flows on an inverse arc, and negative surplus (deficit) really corresponds to a demand.

A partial converse of Thm. 5 also holds.

Proposition 6 *Eq.* (14) *is an exact reformulation of Eq.* (19)*.*

Proof Given an optimal solution (f,ω,p) of Eq. (14), use Eq. (15)-(16) to define a unique solution (f,ω,x,z) that is by definition both feasible and optimal in Eq. (19).

Example 7 Reformulating Example 4 according to Thm. 5 and solving the resulting MP formulation using the CPLEX [13] solver yields the same solution as in Example 4. □

4.5 Multiple inputs

Some transformations require more types of input that transform as a whole. For example, the mass balance of the transformation of methane is based on the following:

$1\ CH_4 + 2\ O_2 \longrightarrow 1\ CO_2 + 2\ H_2O.\quad (*)$

In this setting, the parameters π_v^{hk} as defined above make no sense, because h,k range over the set K of commodities, whereas in $(*)$ sets of commodities transform together. To address the issue, h,k are re-defined to take values in the set $\mathscr{P}(K)$ of all subsets of commodities (i.e. the power set of K). Not all subsets of K need be taken into account, according to the definition below.

Definition 8 A *multiple input transflow process* is a quadruplet $H = (H^-, H^+, p^-, p^+)$ such that $H^-, H^+ \subseteq K$, $p^- : H^- \to \mathbb{R}_+$ and $p^+ : H^+ \to \mathbb{R}_+$.

By the above definition I mean to encode with H^- the set of commodities entering the process with proportions given by p^- and transforming as a whole into the output set H^+ of commodities proportioned according to p^+.

Example 9 In the methane transformation $(*)$ above, the quadruplet (H^-, H^+, p^-, p^+) given by $H^- = \{CH_4, O_2\}$, $p^-(CH_4) = 1$, $p^-(O_2) = 2$, $H^+ = \{CO_2, H_2O\}$, $p^+(CO_2) = 1$ and $p^+(H_2O) = 2$ is a multiple input transflow process. □

At each process node $v \in V$, many different multiple input transflow processes $H = (H^-, H^+, p^-, p^+)$ may be active at the same time: denote the set of all such processes by $\mathscr{H}_v$. Given a multiple input transflow process H, the multiple input transflow conservation equations are as follows:

$$\forall v \in V, k \in K \quad x_v^k = \omega_v^k + \sum_{u \in N^-(v)} f_{uv}^k \tag{20}$$

$$\forall v \in V, k \in K \quad y_v^k = \omega_v^k + \sum_{u \in N^+(v)} f_{vu}^k \tag{21}$$

$$\forall v \in V, H \in \mathscr{H}_v \quad \sum_{k \in H^-} p_k^- x_v^k = \sum_{k \in H^+} p_k^+ y_v^k. \tag{22}$$

Using Eq. (20)-(22) as constraints, define a MP formulation P similar to Eq. (8) but with the transflow conservation equations replaced by the multiple input transflow conservation equations. Since these are linear, the resulting MP is a LP.

Example 10 Consider the methane transformation $(*)$ at node 1 of the process network of Fig. 5. Commodities are encoded as follows: $CH_4 \leftrightarrow 1$, $O_2 \leftrightarrow 2$, $CO_2 \leftrightarrow 3$, $H_2O \leftrightarrow 4$; $\mathscr{H}_1$ simply consists of the multiple input transflow process given in Example 9. The source level s_v^k of commodity k at node v is defined as follows: $s_2^1 = 10$, $s_3^1 = 15$, $s_4^2 = 40$, $s_5^2 = 10$. Solving the formulation using CPLEX yields the solution shown in Fig. 7. It is easy to see that the output level of H_2O is twice that of CO_2. Again, Example 3 can be adapted to this setting to show how a different modelling may bring about an unsustainable solution. □

4.6 Chemical balances

Chemical reactions also produce energy. For example, the methane transformation $(*)$ also produces around 891 kJ. We cannot, however, integrate this energy production into a transflow equation $CH_4 + 2O_2 = CO_2 + 2H_2O + 891\,\text{kJ}$. From a chemical point of view, mass and energy balances must give rise to two separate equations: the interpretation of the transformation

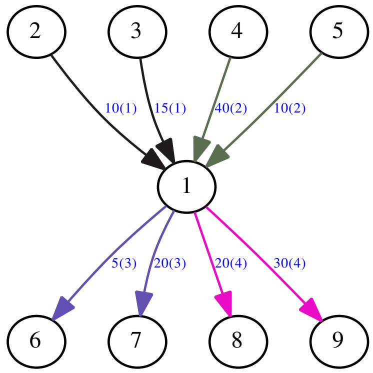

Fig. 7 The methane transformation process.

$1\ CH_4 + 2\ O_2 \longrightarrow 1\ CO_2 + 2\ H_2O + 891$ kJ (†)

is *not* that a part of the mass of methane and oxygen is used to produce energy, but rather that a purely mass-related transformation occurs, which gives rise to a new commodity, namely energy. Mathematically, the surplus variable corresponding to the energy source depends on a nonlinear function of other decision variables, as shown in Eq. (24)-(25) below.

Of course, the energy yielded by chemical transformations really does correspond to mass according to the theory of relativity. However, I believe that using multiple input transflow processes using the equivalence between mass and energy would result in extremely badly scaled constants in the resulting LP (recall that $E = mc^2$).

I shall therefore think of the energy unit measure, kJ, as a new commodity having as source node the process node of the chemical transformation.

Definition 11 A *chemical transflow process* is a 8-tuple $H = (H^-, H^+, p^-, p^+, J^-, J^+, q^-, q^+)$, such that $H^-, H^+, J^-, J^+ \subseteq K$, $p^- : H^- \to \mathbb{R}_+$, $p^+ : H^+ \to \mathbb{R}_+$, $q^- : J^- \to \mathbb{R}_+$, $q^+ : J^+ \to \mathbb{R}_+$, and $J^- \cap J^+ = \varnothing$.

The head 4-tuple (H^-, H^+, p^-, p^+) in a chemical transflow process H (call it H') is a multiple input transflow process (see Defn. 8). The interpretation of tail 4-tuple (J^-, J^+, q^-, q^+) (call it H'') is as follows:

- J^- is the set of commodities that are needed to engender (or are absorbed by) the multiple input transflow process H';
- J^+ is the set of commodities that are produced as by-products of (or originate from) the multiple input transflow process H'';
- q^-, similar to p^-, encodes the proportions of commodities in J^-;
- q^+, similar to p^+, encodes the proportions of commodities in J^+.

Example 12 Consider Example 9, and call H' its multiple input transflow process; let H'' be the 4-tuple given by $J^- = \varnothing$, $J^+ = \{\text{kJ}\}$, q^- be the empty function, and $q^+(\text{kJ}) = 891$. Then the 8-tuplet H formed by the components of H' followed by those of H'' is a chemical transflow process describing (†). □

As in the previous section, I subscript chemical transflow process symbols with the index of the network process node they correspond to. A commodity $k \in J_v^-$ is therefore absorbed by a transformation at the process node v. Accordingly, define a new target:

$$\omega_v^k = \sum_{u \in N^-(v)} f_{uv}^k. \tag{23}$$

Similarly, a commodity $k \in J_v^+$ originates from a transformation at the process node v. Accordingly, define a new source:

$$\omega_v^k = \sum_{u \in N^+(v)} f_{vu}^k.$$

Lastly, I shall enforce the fact that the quantities of commodities in J^-, J^+ are proportional to the coefficients in the given chemical transformation recipe (such as (†)). Thus, given any $h \in H^-$,

$$\forall v \in V, H \in \mathscr{H}_v, k \in J^- \quad \frac{\omega_v^k}{q_k^-} = \frac{x_v^h}{p_h^-} \tag{24}$$

$$\forall v \in V, H \in \mathscr{H}_v, k \in J^+ \quad \frac{\omega_v^k}{q_k^+} = \frac{x_v^h}{p_h^-}. \tag{25}$$

A MP (of the LP type) for chemical transflow process can be obtained by replacing the transflow conservation equations in Eq. (8) by the constraints given by Eq. (20)-(22) and Eq. (23)-(25).

Example 13 Consider the chemical transformation (†) at node 1 of the process network of Fig. 5. Use the same encoding for commodities and the same multiple input transflow process and data defined in Example 10. Add an energy commodity kJ (indexed as 5), with an additional target node 10 with $d_{10}^5 = 22275$. The chemical transflow process based MP is solved using CPLEX, which yields the solution shown in Fig. 8. Again, Example 3 can be

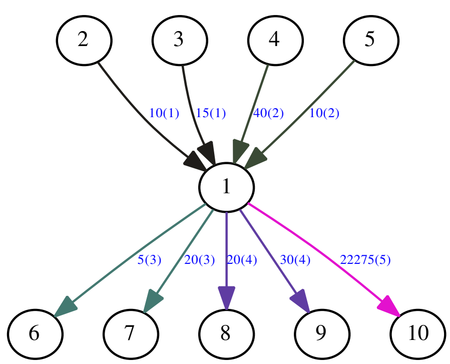

Fig. 8 The chemical transformation process for methane.

adapted to this setting to show how a different modelling may bring about an unsustainable solution. □

4.7 Real-life application of transflows

The above ideas are generalizations of application work I carried out some years ago with M. Bruglieri. We were commissioned the task of computing the optimal cost of planning and then running a biomass-based energy production process, in view to argue in favour of new investor funding for our clients. More precisely, we had to define a minimum cost transportation network connecting several transformation plants with biomass production sites, in such a way as to optimize the energy yield, as well as to satisfy several technical and financial constraints. Initially, we tried to conceive a formulation based on classical multi-commodity network flows; none of these attempts gave rise to an accurate model. We later realized that the network transformed flows according to the mass balance equations called "transflow equations" above, and successfully implemented a MP formulation based on these concepts. The details can be found in [6,7].

A further interesting feature of this problem is that certain transformation process sites explicitly modelled for waste, because certain types of waste can be recycled, and thus appear with a negative cost (i.e. a positive revenue) in the objective function. This further reinforces the case for necessity of transflows in SD: going back to the discussion of Sect. 4.3, the slack-based equality constraints $x_v^k + w_v^k = d_v^k$ (for a process node v and a commodity k) would impose no bound to the waste w_v^k. Whenever waste has a positive revenue coefficient, the network process might try to maximize the production of waste at a certain site. If the modeller has not explicitly catered for this, or has overestimated the waste revenue, the optimal solution might simply state to maximize the production of waste (which is very likely *not* what is being sought). Transflows provide an appropriate tool to deal with these situations, since waste is modelled as any other commodity.

5 Time-dependent transflows

As mentioned in Sect. 1, time appears in some MP formulations arising in economics, scheduling, transportation, and other industrial settings. All the transflow optimization formulations above deal with a steady state. Assuming the production process (e.g. a chemical reaction) does not depend on time, this is a reasonable assumption, and transflows will guarantee sustainability to the extent that unaccounted waste is mathematically infeasible. In practice, however, input data such as e.g. demand *does* depend on time. Moreover, for most industrial problems, data will become available at certain timesteps, rather than being supplied as a continuous function of time.

5.1 Limitations of steady-state transflows

The simplest solution to the time dependence of data is to solve a sequence of steady state transflow problems, one at each timestep. This has two main drawbacks. The first, and in my opinion the least, is that if the timesteps are too close, the time available to a computational system for finding a real-time solution to the problem may be insufficient. The second is far more serious: steady-state solutions do not account for the fact that a limited amount of waste may be acceptable, provided it is eventually disposed of. In other words, a sequence of steady-state problems would result in infeasibility across the whole sequence, because the only feasible way to dispose of the waste is to produce it at a certain time and dispose of it at another (e.g. when an appropriate facility becomes available). Examples 2 and 14 provide an illustration of this fact.

Time-dependent transflow optimization also addresses a further layer of decisions in transformation processes: the scheduling of activities. By contrast, in steady state formulations we decide the activities but not their scheduling. From a purely formal point of view, a sequence of activities can be turned into a set of activities by simply projecting time away, but not every set of activities corresponds to a feasible sequence, given the scheduling constraints. The upshot is that an optimal solution of a steady-state formulation might not correspond to any feasible schedule.

5.2 Time-expanded networks

There is a graph product operation which can be applied to a transflow network $G = (V,A)$ in order to turn it into a time-dependent one $\mathscr{G} = (\mathscr{V},\mathscr{A})$. Let $T = \{1,\ldots,t_{\max}\}$ be a finite timestep index set. Then $\mathscr{V}$ is the set of couples (v,t), where $v \in V$ and $t \in T$, and $((u,t_1),(v,t_2)) \in \mathscr{A}$ if and only if either $t_1 = t_2$ and $(u,v) \in A$, or if $u = v$ and $t_2 = t_1 + 1$. More succinctly, if $\mathbf{T} = (T, \{(t-1,t) \mid t \in T \smallsetminus \{1\}\})$, then $\mathscr{G} = G \times \mathbf{T}$. Note that this models the simplest situation where all activities take one timestep to complete. Longer activities can be modelled by requiring the presence of arcs $((u,t-k),(u,t))$ in $\mathscr{A}$ for some $k > 1$, and appropriate $u \in V, t \in T$.

This operation can be generalized to a non-complete extended p-sum (NEPS) [8, Sect. 2.5] so that $\mathscr{A}$ can be specified more precisely. When a time-dependent network is obtained by a product-like operation on a steady-state network, it is usually called a *time-expanded* network (see Fig. 9 for an example).

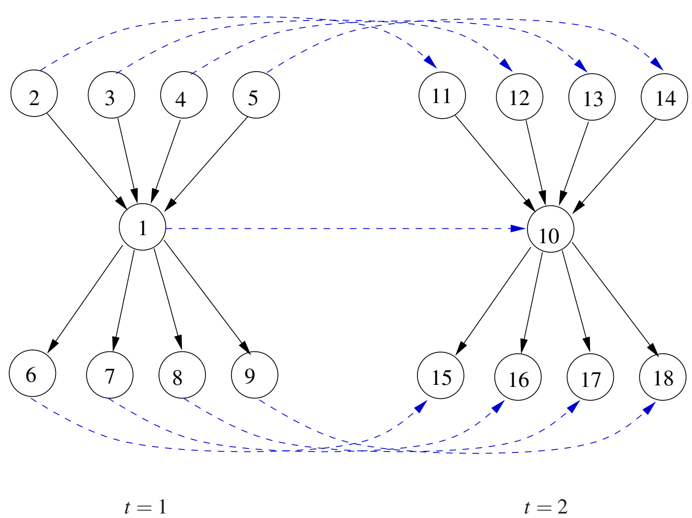

Fig. 9 The network G from example of Fig. 5 in a graph product with $\mathbf{T} = (\{1,2\},\{(1,2)\})$.

A transflow $f : \mathscr{A} \to \mathbb{R}$ defined on a time-expanded network of the simplest type (i.e. a graph product) models a commodity which travels through the network in either of the two components $G, \mathbf{T}$. An arc in $\mathscr{A}$ has either the form $((u,t),(v,t))$ (the *transportation arcs*) or the form $((v,t-1),(v,t))$ (the *time arcs*) for u,v with $(u,v) \in A$ and some $t > 1$. Accordingly, f_{uv}^{kt} describes a flow of commodity k through the arc $((u,t),(v,t))$ if $u \neq v$ (a flow in the steady-state component G), or a commodity flow through the arc $((v,t-1),(v,t))$ if $u = v$ (a flow in the time component $\mathbf{T}$). Informally, this allows a commodity to move in space on G and in time on $\mathbf{T}$.

As briefly mentioned above, the assumption underlying the simplest graph product model is that all time movement takes a unit timestep to occur. If this assumption is unrealistic, a carefully constructed NEPS can address the issue, for example by requiring arcs in $\mathscr{A}$ between t and $t+1,\ldots,t+r$ for a given time offset r in the time component, and in a given disc in the space component. Notice that in this generalization, flow variables should be indexed differently (e.g. $f_{uv}^{kt_1t_2}$), to account for the fact that not all time-indexed arcs join immediately successive timesteps t and $t+1$.

5.3 Sustainability for time-dependent transflows

I shall only illustrate how to generalize the basic transflow equations Eq. (4) to time dependence. The corresponding generalizations for Sect. 4.4-4.6 are similar.

First, we index relevant parameter and decision variable symbols by the timestep t, i.e. ω_v^k, f_{uv}^k become arrays of symbols having components $\omega_v^{kt}, f_{uv}^{kt}$ respectively, for each $t \in T_0$, where $T_0 = T \cup \{0\}$. The zero timestep is used to relate quantities at timestep t to timestep $t-1$ for each $t \in T$. All decision variable symbols indexed at $t = 0$ (or, more in general, at timesteps where the network status is known) can optionally be fixed to certain given parameters called *boundary conditions*. Infinite time horizons can also be modelled as long as they are cyclic, by requiring that the network status at the final timestep is equal to the network status at some intermediate timestep (most usually the initial timestep, but this need not be necessary).

The time-dependent version of Eq. (4) simply adds the appropriate flows from the new arcs $((v,t-1),(v,t))$ in the expanded network:

$$\forall v \in V, k \in K, t \in T \quad \sum_{h \in K} \pi_v^{hk} \left(\omega_v^{ht} + \omega_v^{h,t-1} + \sum_{u \in N^-(v)} f_{uv}^{ht} \right) = \\ = \quad \omega_v^{kt} + \omega_v^{k,t-1} + \sum_{u \in N^+(v)} f_{vu}^{kt}. \tag{26}$$

Example 14 The generalization of Example 2 (which is infeasible in a steady state setting) to a time-dependent setting, when the sources are only active at $t = 1$ but the target demands must be satisfied at both $t = 1$ and $t = 2$ yields a feasible solution, shown in Fig. 10. Dotted arcs indicate unused transportation arcs $((u,t),(v,t))$, light-colored dashed arcs indicated unused time arcs $((v,t-1),(v,t))$. Dashed arcs indicate a flow (denoted as units[commodity]) along time arcs. Line arcs indicate a flow (denoted as units(commodity)) along transportation arcs.

The solution uses the source nodes $4,5$ at time $t = 1$ and transforms the two units of commodity 2 into two units of commodity 3 and two of commodity 4, which arrive at the four target nodes $6,7,8,9$. The surplus units of commodity 1 are stored at source nodes $1,2$ and used at time $t = 2$ in a similar way. □

Notice that time-dependency also allows us to define power on, ramp up and power off costs for plants [27] by means of suitable binary variables indexed by node and timestep.

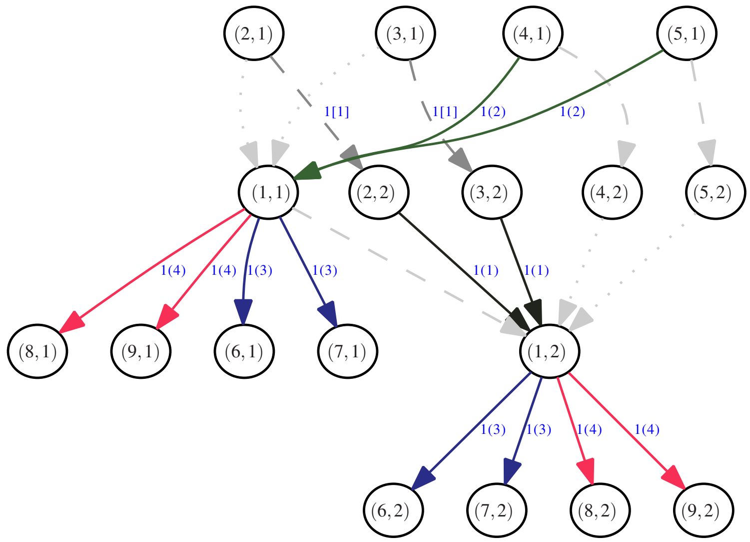

Fig. 10 Feasible solution of the time-dependent generalization of Example 2.

6 Conclusion

In this paper I argue that the term "optimization and sustainable development" really defines a class of very diverse problems with no discernible underlying mathematical principle. Using a more specific definition of the term, i.e. optimization in connection with a form of development that can last forever, I propose a set of mathematical programming related modelling techniques based on the concept of *transformation flows*, an extension of multi-commodity flows which involves the transformation of incoming flow into outgoing flow of different type at selected process nodes. Such equations in principle allow solvers to detect (by infeasibility) when a supply chain fails to account for all input commodities in a transformation process. Feasible transflow processes are formally defined to be sustainable, in the sense of the process lasting indefinitely under unchanging conditions.

Acknowledgments

I am extremely grateful to Microsoft France and Microsoft Research for the "Optimization and Sustainable Development" Chair endowment, which helped sponsor more than four years of research in optimization and sustainable development topics at LIX, École Polytechnique. Specifically, I would like to acknowledge the scientific contributions of Philippe Baptiste, Claudia D'Ambrosio, Youssef Hamadi, Vincent Jost, Dominik Kirchler, Fabio Roda, David Savourey and Roberto Wolfler Calvo, as well as the invaluable help of Sylvie Tonda-Goldstein, Valérie Masson-Patrimonio, Evelyne Rayssac and Pierre-Louis Xech. I am also extremely grateful to a referee for very precious suggestions about this paper.

References

1. N. Adhya, M. Tawarmalani, and N.V. Sahinidis. A Lagrangian approach to the pooling problem. *Industrial and Engineering Chemistry Research*, 38:1956–1972, 1999.
2. J. Aronson. A survey of dynamic network flows. *Annals of Operations Research*, 20:1–66, 1989.
3. C. Audet, J. Brimberg, P. Hansen, S. Le Digabel, and N. Mladenović. Pooling problem: Alternate formulations and solution methods. *Management Science*, 50(6):761–776, 2004.
4. P. Bartelmus. *Environment, Growth and Development*. Routledge, New York, 1994.
5. P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods and Software*, 24(4):597–634, 2009.
6. M. Bruglieri and L. Liberti. Optimal running and planning of a biomass-based energy production process. *Energy Policy*, 36:2430–2438, 2008.
7. M. Bruglieri and L. Liberti. Optimally running a biomass-based energy production process. In J. Kallrath, P. Pardalos, S. Rebennack, and M. Scheidt, editors, *Optimization in the Energy Industry*, Energy Systems, pages 221–232. Springer, New York, 2009.
8. D. Cvetković, P. Rowlinson, and S. Simić. *An introduction to the theory of graph spectra*. Cambridge University Press, Cambridge, 2010.
9. C. D'Ambrosio, J. Linderoth, and J. Luedtke. Valid inequalities for the pooling problem with binary variables. In O. Günlük and G. Woeginger, editors, *IPCO*, volume 6655 of *LNCS*, pages 117–129, Heidelberg, 2011. Springer.
10. G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ, 1963.
11. R. Fourer and D. Gay. *The AMPL Book*. Duxbury Press, Pacific Grove, 2002.
12. Y. Hamadi and L. Liberti. Microsoft-CNRS Chair "Optimization for Sustainable Development", Activity Report 2009-2011. Technical Report MSR-TR-2011-42, Microsoft Research, 2011.
13. IBM. *ILOG CPLEX 12.2 User's Manual*. IBM, 2010.
14. J. Kennington. A survey of linear cost multicommodity network flows. *Operations Research*, 26(2):209–236, 1978.
15. D. Kirchler, L. Liberti, and R. Wolfler Calvo. Efficient computation of shortest paths in time-dependent multi-modal networks. *ACM Journal of Experimental Algorithmics*, to appear.
16. B. Kotnyek. An annotated overview of dynamic network flows. Technical Report 4936, INRIA, Sept. 2003.
17. L. Liberti. Reformulations in mathematical programming: Definitions and systematics. *RAIRO-RO*, 43(1):55–86, 2009.
18. R. Misener and C. Floudas. Global optimization of large-scale generalized pooling problems: quadratically constrained MINLP models. *Industrial Engineering and Chemical Research*, 49:5424–5438, 2010.
19. G. Nannicini, Ph. Baptiste, G. Barbier, D. Krob, and L. Liberti. Fast paths in large-scale dynamic road networks. *Computational Optimization and Applications*, 45:143–158, 2010.
20. G. Nannicini and L. Liberti. Shortest paths in dynamic graphs. *International Transactions in Operations Research*, 15:551–563, 2008.
21. World Commission on Environment and Development. *Our Common Future*. Oxford University Press, Oxford, 1987.
22. S. Pallottino and M.G. Scutellà. Shortest path algorithms in transportation models: classical and innovative aspects. In P. Marcotte and S. Nguyen, editors, *Equilibrium and Advanced Transportation Modelling*. Kluwer, Dordrecht, 245-281.
23. S. Pallottino, G. Sechi, and P. Zuddas. A DSS for water resources management under uncertainty by scenario analysis. *Environmental Modelling and Software*, 20(8):1031–1042, 2005.
24. C.H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover, New York, 1998.
25. M. Porcheron, A. George, O. Juan, T. Simovic, and G. Dereu. Challenge ROADEF/EURO 2010: A large-scale energy management problem with varied constraints. Technical Report `http://challenge.roadef.org/2010/files/sujetEDFv22.pdf`, Electricité De France, Recherche et Développement, February 2010.
26. F. Roda, P. Hansen, and L. Liberti. The price of equity in the hazmat transportation problem. In G. Nicosia and A. Pacifici, editors, *Proceedings of Cologne/Twente Workshop*, pages 235–238, Rome, 2011. Università di Roma 2 — Tor Vergata.
27. M. Tahanan, W. van Ackooij, A. Frangioni, and F. Lacalandra. Large-scale unit commitment under uncertainty: a literature survey. *4OR*, submitted.