Food Traceability Systems: Performance Evaluation and Optimization

Published version:

DOI:10.1016/j.compag.2010.10.009

Food Traceability Systems: Performance Evaluation and Optimization

Fabrizio Dabbene[a], Paolo Gay[b]

[a] IEIIT-CNR - Politecnico di Torino, 24 Corso Duca degli Abruzzi, 10129 Torino – Italy,
fabrizio.dabbene@polito.it;

[b] D.E.I.A.F.A. – Università degli Studi di Torino, 44 Via Leonardo da Vinci, 10095 Grugliasco (TO) – Italy,
paolo.gay@unito.it;

Corresponding author: Paolo Gay
Email: paolo.gay@unito.it Tel: +39 011 6708620 Fax: 011 6708591

Abstract

The aim of a traceability system is to collect in a rigorous way all the information related to the displacement of the different products along the supply chain. This information proves essential when facing food safety crisis, and allows efficiently managing the consequent product recall action. Although a recall action could be absolutely critical for a company, both in terms of incurred costs and of media impact, at present most companies do not posses reliable methods to precisely estimate the amount of product that would be discarded in the case of recall.

The skill of limiting the quantity of recalled products to the minimum can be assumed as a measure of the performance and of the efficiency of the traceability system adopted by the company. Motivated by this consideration, this paper introduces novel criteria and methodologies for measuring and optimizing the performance of a traceability system. As opposed to previous introduced methods, which optimize indirect measures, the proposed approach takes into direct account the worst-case (or the average) quantity of product that should be recalled in the case of a crisis. Numerical examples concerning the mixing of batches in a sausage production process are reported to show the effectiveness of the proposed approach.

Keywords: traceability, optimization, supply chain management, batch dispersion, MILP

1. Introduction

Traceability in the agricultural/food chain is nowadays a fundamental requirement, which is becoming mandatory in almost all developed countries. As discussed in Ràbade & Alfaro (2006), traceability represents a mechanism for reinforcing the level of coordination between producers and firms, and between firms and retailers

Primal goal of a traceability system it to precisely log the history and the location of the different products along the supply chain. Recently, the technological advances in this direction have led to the design of ICT instruments, such as e.g. bar codes and RF-ID devices, aimed at facilitating data acquisition and reducing the traceability management costs (Gandino et al., 2009; Regattieri et al., 2007; and Sahin et al. 2002), and to the development of data bases and web-based systems for data processing (Alfaro & Ràbade, 2009; Ruitz-Garcia et al., 2010). The information collected by the traceability systems becomes strategic in the unfortunate case when a batch of product has to be recalled (Bechini et al., 2008). Indeed, besides the media impact of this action, the firm has to incur costs related to the recall and the destruction of all the products that are, in some way, connected with the incriminated batch (Jacobs, 1996). Since this occurrence could be absolutely critical for a company, some studies have been carried on for modelling and forecasting the effects of recall actions (e.g., see Kumar & Budin, 2006 and Randrup et al., 2008). However, at present most companies do not have reliable methods to precisely estimate the amount of product that has to be discarded in the case of a recall. Indeed, this quantity, to which we associate a *recall cost* (RC), depends on many factors:

- the size of the batches that have been individually tracked and managed by the traceability system;
- the way the batches of different components have been mixed to obtain the final product;
- the skill of the firm to manage and maintain segregated different batches of product, especially in the case of continuous processes (e.g. milk processing in a dairy, grain or soya, see for instance Thakur & Hurburg, 2009, Thakur et al., 2010, Thakur & Donnelly, 2010, and Skoglund & Dejmek, 2007).

From the analysis of these factors, it is clear that a simple reduction in the size of the batches, and their consequent increase in number, leading to a finer granularity of the traceability system (Bertolini et al, 2006), may be not sufficient to minimize the amount of product to be discarded. For a discussion on the role of different levels of granularity the interested reader is referred to Karlsen (2011) and references therein, where an example related to farmed salmon is also presented.

The previous considerations suggest selecting the recall cost as a natural measure of the performance for a traceability system. This gives raise to two fundamental problems: i) the evaluation of the performance of a given traceability system, ii) the optimization of the supply-chain design in order to minimize its performance.

To better formalize these problems, some nomenclature has to be introduced. Moe (1998) has introduced the concept of *traceable resource unit* (TRU) for batch processes as "unique unit, meaning that no other unit can have exactly the same, or comparable, characteristics from the point of view of traceability". In modern agricultural supply system, units must be uniquely identifiable within each system in which they are processed. To this extent, Bollen et al. (2007) introduced the *identifiable unit* (IU), whose size reflects the granularity of the traceability system. In many supply chains granularity is the consequence of a combination of tradition, short-term convenience and use of available facilities. In very few cases granularity depends on the results of a formal analysis and optimization in the supply chain. The simple implementation of a finer granularity by itself has no value unless it provides more precise traceability. The precision of a traceability system can be evaluated, as discussed in Bollen et al. (2007), as the ratio between IUs at two points in the supply chain and it is the consequence of the number and the nature of the transformations of IUs and of the extent, nature and accuracy of data recorded. If a IU is split up, the separated parts keep the identification of the parent IU, while if some IUs are put together, the identification of the IU is different from the identification of the parent IUs.
One possible solution to maintain the same level of traceability precision consists of breaking the processing into segments of relative homogeneity, both for processing conditions and product origin, and recording all relevant information.

Finally, one has to take into account if the product is processed in completely separated runs, or if some mixing can occur between products of two succeeding batches. In the latter case, it is necessary to specify if tolerances can be accepted. This problem has been addressed by Skoglund & Dejmek (2007) for the case of continuous processing where it has been referred as *fuzzy traceability*, while Riden & Bollen (2007) considered the case of discrete products, with an application to packhouse processing transformations.

The problem of the performance evaluation and optimization of traceability systems was first introduced by Dupuy et al. (2005), and successively applied in different endeavours (see for instance Donnelly et al., 2009). Tamayo et al. (2009) employ genetic algorithms to solve the optimization problem proposed by Dupuy et al. (2005). Finally, Wang et al. (2010) propose the joined optimization of traceability and manufacturing performances, acting both on batch sizes and batch dispersion, by introducing risk functions. In all these works, the performance of a traceability system is associated to the number of active paths between raw-materials and finished products, as formally detailed in Section 3. This measure is indeed related to the final quantity to be recalled, since it aims at reducing the mixing of different batches, and was proven effective in the above-mentioned works. However, it should be remarked that, in general, the minimization of this index does not necessarily result in the minimization of the recall cost, when intended as the quantity of products to be recalled in the worst-case.

In this paper, we introduce a modelling framework and optimization strategy to cope with this problem, directly adopting the recall cost as performance criterion. Similarly to Dupuy et al. (2005), the optimization problem is expressed in the form of mixed-integer linear programming (MILP), for which efficient numerical solvers are available. To show its effectiveness, the proposed approach has been first applied to the numerical example presented in Dupuy et al. (2005) and in Tamayo et al. (2009), and finally to a larger test case.

2. Modelling

A complete food production process can be seen as a sequence of storage/carrying actions and of unit operations. Bulk products are stored and carried in containers (as for instance tanks, vats, bins etc.) depending on the nature of the products. Unit operations can be conducted on a batch of product at a time (e.g. concentration in a bull, cooking in a oven) or continuously, as the processes of milk pasteurization/sterilization, or of concentration in a continuous evaporator.

From the point of view of traceability, this second instance (continuous unit operations) can also be interpreted as a batch process situation, by either guaranteeing proper cleaning cycles between two subsequent lots, or by allowing (and then neglecting) small percentages of contamination (Skoglund & Dejmek, 2007). Each container/processing-unit that individually stores/ processes a batch of product, at a certain time, can be modeled as a node in a graph.

Formally, at each node k one associates a variable Q_k, which accounts for the quantity of product contained in the node. This variable can be bounded by the capacity of the container, or by the amount of product that can be processed at a time. This corresponds to imposing the constraint $Q_k \le \overline{Q}_k$. In some cases, it is also possible to introduce the equality constraint $Q_k = \overline{Q}_k$ to reflect the cases when one wants to fix the quantity of material in node k precisely to the value $\overline{Q}_k$. This is the case, for instance, of final products that are sold in fixed-weight packages, or of middle nodes where a fixed amount of product is processed at a time.

The flow of the batches inside the supply chain is modeled via a number of oriented arrows (links of the graph). These links are formally described introducing a $m \times 2$ matrix $L \in N_+^{m,2}$ of positive numbers, where m is the number of links. The entries of L have the following structure: $L_{i,1}$ indicates the starting node of the link i, while $L_{i,2}$ represents its destination node. The amount of material transferred through the i-th link is expressed by the variable $\alpha_i \in R$, $\alpha_i \ge 0$, $i = 1,2,\ldots,m$. Associated to the variable α_i, one can define the *binary* variable $\overline{\alpha}_i$, which is true whenever the i-th link is active, that is

$$\overline{\alpha}_i = \begin{cases} 1 & \text{if } \alpha_i > 0 \\ 0 & \text{if } \alpha_i = 0. \end{cases} \tag{1}$$

Nodes can be schematically grouped into three sets: input, processing and output nodes. Letting $I = \{1,2,\ldots,n_{nodes}\}$ be the set of the indexes of all nodes, one can define $I_{in} \subset I$ and $I_{out} \subset I$ as the sets of indexes of the input and output nodes, respectively. The cardinality of these sets (i.e. the number of elements belonging to the set) is denoted as n_{in} and n_{out}, respectively. Then, to each input node $k \in I_{in}$ is associated the initial quantity of available product (raw material) Q_k^0 that has to be transferred and/or processed by the network.

For any node $k \in I$, one defines the sets of links entering and leaving the node as

$$in_k = \left\{ i \in I : L_{i,2} = k \right\} \quad \text{and} \quad out_k = \left\{ i \in I : L_{i,1} = k \right\}. \tag{2}$$

These indices allow expressing the mass balances for each node k as follows

$$\sum_{i \in in_k} \alpha_i = \sum_{i \in out_k} \alpha_i = Q_k. \tag{3}$$

The modeling framework proposed in this paper relies on the definition of specific "state variables" $S_k^l \in \{0,1\}$, $k,l \in I$. The binary variable S_k^l is true whenever the node k contains a product arising from node l, that is whenever there exists a *path* in the graph connecting node l to node k. Notice that the state S_k^l can be recursively calculated using the following relation

$$S_k^l = \bigvee_{i \in in_k} \left(S_{L_{i,1}}^l \wedge \overline{\alpha}_i \right), \tag{4}$$

where $\vee$ and $\wedge$ represent the logical OR and AND operators respectively. The initial conditions for recursion (4) are given by

$$S_k^k = 1 \text{ for } k \in I_{in} \quad \text{and} \quad S_k^l = 0 \text{ for } k \in I_{in} \setminus \{l\}. \tag{5}$$

We recall that both OR and AND operators may be rewritten as linear operations on binary variables, see for instance Achterberg et al. (2007). More precisely, for binary variables $a,\ b,\ r \in \{0,1\}$, one can write

$$r = a \wedge b \quad \Leftrightarrow \quad \begin{cases} r \le a \\ r \le b \\ r \ge a + b - 1 \end{cases} \tag{6}$$

$$r = a \vee b \quad \Leftrightarrow \quad \begin{cases} r \ge a \\ r \ge b \\ r \le a + b \end{cases} \tag{7}$$

A simple example of the modeling framework introduced in (1)-(5) is presented in Figure 1. To illustrate the meaning of the introduced variables, in this figure the explicit construction of the states introduced in (4) is provided for node 6 and source 1 as an example. The recursive nature of the states is clearly evidenced: the state S_6^1 depends on S_4^1 and S_5^1, which in turn are functions of the state of the sources, which are given by the initial conditions introduced in (5). Notice that, after the recursion is resolved, one obtains a relationship whose interpretation is clear: node 6 contains material from node 1 whenever both links α_1 and α_6, or links α_2 and α_8, are simultaneously active.

In a generic supply chain, the lots of products are displaced and/or processed according to some rules that govern each mixing occurrence. In the proposed setup, these rules are generically referred to as *recipe rules*, and are defined on sets of nodes containing homogeneous products. In this way, a (possibly large) number of nodes that are devoted to contain the same type of product can be grouped into a single set. Recipes, which are related to sets, are valid for each node belonging the involved set. More formally, one can define n_{type} disjoint sets $T_p \subseteq I$, $p = 1,2,\ldots,n_{type}$, of indices, where the set T_p is formed by the indices of the nodes that contain a product of type *p*. Then, recipe constraints can be generally expressed as linear relationships between product types. In particular, one can define assembling and disassembling recipes as in Dupuy et al. (2005). This is done by introducing the matrices of coefficients $D \in R^{n_{type},n_{type}}$ and $A \in R^{n_{type},n_{type}}$. Then, a disassembling constraint allows describing the situation when each product belonging to a given type p has to be destined to nodes belonging to the set T_j according to the percentage expressed by $D_{p,j}$, i.e.

$$\sum_{\substack{i \in out_k \\ L_{i,2} \in T_j}} \alpha_i = D_{p,j}\, Q_k \quad \text{for} \quad \forall k \in T_p \ \text{ and } \ p = 1,\ldots,n_{type}. \tag{8}$$

Analogously, assembly constraints impose that each product of type p has to be composed by product of type j, according to the percentage $A_{p,j}$.

$$\sum_{\substack{i \in in_k \\ L_{i,1} \in T_j}} \alpha_i = A_{p,j}\, Q_k \quad \text{for} \quad \forall k \in T_p \ \text{ and } \ p = 1,\ldots,n_{type} \tag{9}$$

It should be remarked that the modelling framework proposed in this section improves upon the one in Dupuy et al. (2005) in the following points: i) the original approach of Dupuy et al. (2005) considers only three-stages production systems, with a raw-materials stage, a components stage and a finished products one; ii) only fully-interconnected networks (each node at one stage is connected to each node in the successive stage) are considered in Dupuy et al. (2005). On the contrary, the introduction of the link matrix L allows considering arbitrary networks, that is graph configuration consisting of multiple stages and arbitrary link configurations. In particular, this second feature allows excluding a-priori undesired or logistically unfeasible links, thus leading to a significant reduction in the complexity of the ensuing optimization problem. Also, the introduction of the state variables S allows a clear formalization of the desired performance measures related to the recall cost, as shown in the next section.

3. Performance evaluation

As discussed in the Introduction, different measures can be defined for assessing the performance of a traceability system. In particular, Dupuy et al. (2005) defined three performance indices: the downward dispersion, the upward dispersion and the batch dispersion. The downward dispersion of a raw material batch is the number of final batches that contain parts of a specific raw material batch. The upward dispersion of a finished product batch is the number of different raw material batches used to produce this batch, while the batch dispersion is defined in Dupuy et al. (2005) as the sum of links between the raw material batches and the finished product batches. The analytical expression of these indices is formally defined at the end of this section.

As previously discussed, the approach in Dupuy et al. (2005) does not take into account quantities, but only the active paths that are upward/downward involved. However, this setup has the great advantage of allowing a formalization of the optimization problem in terms of mixed-integer linear programming (MILP). Motivated by the fact that the number of variables and constraints in the ensuing MILP problem may increase exponentially, Tamayo et al. (2009) proposed to solve this problem by means of genetic algorithms.
Proceeding along the same lines that in Dupuy et al. (2005), in this paper novel performance indices are introduced, which better quantify the cost of product recall as perceived by the industry. To this extent, first introduce the recall cost of product l, $RC(l)$, as the total amount of (final) product that has to be recalled in the case when the batch of raw material contained in node l is recognized as lacking the requirements. This corresponds to the mass of final product that contains – owing to mixing operations – part of the material originally stored in the incriminated node l.

On the basis of the formalism presented in the previous section, the recall cost relative to node l can then be directly defined as

$$RC(l) = \sum_{k \in I_{out}} S_k^l Q_k \tag{10}$$

The typical interest of a company is to know – and possibly to reduce – the worst-possible amount of product that could be necessary to recall. This corresponds to defining the *worst-case recall cost* ($WCRC$)

$$WCRC = \max_{l \in I_{in}} RC(l) = \max_{l \in I_{in}} \sum_{k \in I_{out}} S_k^l Q_k \,. \tag{11}$$

as the largest amount of product that has to be recalled when a batch of raw material is found unsafe. Analogously, it is possible to define the *average recall cost* (ARC) index as

$$ARC = \frac{1}{n_{in}} \sum_{l \in I_{in}} RC(l) = \frac{1}{n_{in}} \sum_{l \in I_{in}} \sum_{k \in I_{out}} S_k^l Q_k \,, \tag{12}$$

which represents the average mass of product to be recalled when one of the entering material is found inappropriate.

It should be remarked that the ARC cost defined in (12) can be readily adapted to the case when suppliers of the input batches have a different level of reliability and/or one can associate to different input batches different probabilities of lacking the requirements. This can be modelled introducing appropriate weights w_i, $i = 1,\ldots,n_{in}$, (that can be interpreted also as probabilities). This leads to the following *weighted recall cost* (WRC) index

$$WRC = \sum_{l \in I_{in}} w_i RC(l) = \sum_{l \in I_{in}} w_i \sum_{k \in I_{out}} S_k^l Q_k \,, \tag{13}$$

Finally, it should be remarked that the introduced setup can easily handle the *batch dispersion cost (BDC)* introduced in Dupuy et al. (2005), by introducing the downward dispersion from node l as

$$D\_DISP(l) = \sum_{k \in I_{out}} y(l,k) \tag{14}$$

where

$$y(l,k) = \begin{cases} 1 & \text{if } S_k^l Q_k > 0 \\ 0 & \text{if } S_k^l Q_k = 0 \end{cases} \quad (15)$$

Then, the BDC index can be written as

$$BDC = \sum_{l \in I_{in}} D\_DISP(l) = \sum_{l \in I_{in}} \sum_{k \in I_{out}} y(l,k), \quad (16)$$

and it represents the sum of links between the raw material batches and the finished product batches. Notice that (16) is exactly the index minimized in Dupuy et al. (2005).

4. Optimization

Different approaches can be adopted to optimize the performance of the traceability system. The first possibility is to *compare different scenarios*. Even if this technique cannot properly be referred to as optimization, it permits to compare via simulation some selected configurations of the production process and/or the supply chain (Gay et al., 2009). It is a helpful approach when a decision among few possible alternatives has to be taken. However, clearly this methodology does not bring out optimal solutions that have not been already a-priori selected.

A more rigorous approach is *direct optimization,* as in Dupuy et al. (2005) and Tamayo et al. (2009). This methodology is to be preferred whenever one or more parameters of the supply chain have to be designed according to an optimality criterion. In our case, as in Dupuy et al. (2005), the parameters to be designed are the product flows α_i, $i = 1,\ldots,m$ and the considered optimality criteria are the batch dispersion recalled in (16), and the worst-case/average recall costs defined in (11) – (12).

It should be noticed that the framework introduced in Section 2 allows formulating the problem of minimizing both the original batch dispersion measure (Dupuy et al., 2005) and newly introduced more realistic performance measures $WCRC$ and ARC in terms of mixed-integer linear programs. To this end, first notice that both $WCRC$ and ARC objective functions contain the product of terms $S_k^l \in \{0,1\}$ and $Q_k \in R$. These quantities depend both on the optimization variables α_i. However, this nonlinearity can be converted in a set of linear inequalities. To see this, remark that an optimization problem of the type $\min S_k^l Q_k$ can be reformulated by introducing an additional real variable $r = S_k^l Q_k$, as follows

$$\min\ S_k^l Q_k \quad \Leftrightarrow \quad \begin{aligned} &\min\ r \\ &\text{subject to}: \quad r \geq 0 \\ &\qquad\qquad\qquad r \geq Q_k - M\left(1 - S_k^l\right) \\ &\qquad\qquad\qquad r \leq Q_k \end{aligned} \tag{17}$$

where M is a sufficiently large number. Notice also that the minimization of the cost function $WCRC$ in (11), which presents a maximization term, can be reduced to a linear problem by simply writing

$$\min \max_{l \in I_{in}} RC(l) \quad \Leftrightarrow \quad \begin{aligned} &\min\ \gamma \\ &\text{subject to}: \ RC(l) \leq \gamma,\ l \in I_{in} \end{aligned} \tag{18}$$

Finally, also the binary version $\overline{\alpha}_i$ of α_i introduced in (1) can be reformulated in the integer programming paradigm by introducing the following two linear constraints

$$\overline{\alpha}_i \leq M\alpha_i, \qquad \overline{\alpha}_i \geq \frac{\alpha_i}{M} \tag{19}$$

where again M is a sufficiently large number. The same operation can be made for introducing equation (14).

5. Numerical examples

In order to demonstrate the effectiveness of the proposed approach, the same example proposed in Dupuy et al. (2005), and subsequently elaborated in Tamayo et al. (2009), is here considered. The problem concerns a sausages fabrication chain modelled as a three-level network, consisting of four batches of input (raw) material divided into two types of product (RM1 and RM2), six processing batch units ("components", according to the notation introduced in Dupuy et al., 2005) divided into two types (SP1 and SP2), two batches of bought components (additional inputs), one of each type (i.e., SP1 and SP2 again), and four batches of finished product, also divided into two types (FP1 and FP2). Since the network is fully interconnected, all α_i, $i = 1,\ldots,56$, coefficients have to be determined.

To solve the problem, the batch dispersion minimization and the newly introduced *WCRC* and *ARC* minimization problems have been written as MILPs using the YALMIP software, that allows parsing of optimization problems under Matlab, see Löefberg (2004) for additional details. The resulting MILP programs were then solved using the commercial solver CPLEX (ILOG-IBM), on a 2.53GHz Macbook Pro.

In the first numerical example, the exact same numerical setup used in Tamayo et al. (2009) was adopted, with initial values for the first nodes equal to $Q_1^0 = Q_3^0 = 1{,}000$ and $Q_2^0 = Q_4^0 = 1{,}200$, and with final desired quantities in the last four nodes set to the values $\overline{Q}_{13} = \overline{Q}_{14} = \overline{Q}_{15} = \overline{Q}_{16} = 2{,}000$. The solution minimizing the average cost criterion ARC was sought. The CPLEX solver returned, after 2.7 sec of elaboration, the solution reported in Figure 2, which is guaranteed to be optimal. The average recall cost of this configuration is $ARC = 3{,}333$. The graph contains a total of 23 links, and ten direct source-destination paths, hence providing a batch dispersion measure $BDC = 10$. These figures can be compared with the numerical solution obtained in Tamayo et al. (2009), which has a recall cost $ARC = 3{,}667$. This corresponds to a 10% improvement of our solution. However, it should be noted that the solution in Tamayo et al. (2009) presents 13 direct source-destination paths (*BDC*=13), and hence it is not optimal also for the batch dispersion cost introduced by Dupuy et al. (2005). This fact is not surprising, since the genetic algorithm approach in Tamayo et al. (2009) does not provide any guarantee of returning an optimal solution. Hence, we computed the optimal solution using the BDC cost, which provided the same performance as our ($ARC = 3{,}333$ and $BDC = 10$), with a computation time of 5.8 sec. A similar behavior was observed when comparing with the worst-case optimality criterion $WCRC$.

Based on this observation, a second numerical example was run. This second example considers the same node configuration of the first one, but with the quantities in the first nodes, and the final desired quantities in the last four nodes, now unbalanced. That is, $Q_1^0 = 450$, $Q_2^0 = 2{,}350$, $Q_3^0 = 150$, $Q_4^0 = 1{,}450$, and $\overline{Q}_{13} = 1{,}750$, $\overline{Q}_{14} = 3{,}150$, $\overline{Q}_{15} = 2{,}250$, $\overline{Q}_{16} = 850$. The solution of the three different optimality indices BDC, ARC and $WCRC$ were again computed using the CPLEX solver. Optimal solutions were returned respectively in 92.5, 47.8 and 0.6 seconds. The respective optimal configurations are shown in Figures 3, 4 and 5, and the relative quantities of interest are reported in Table 1.

A few comments are at hand for the figures reported in Table 1. First, it can be observed that the BDC solution, although presenting fewer direct paths between input and final nodes, provides an average recall cost that is around 23% worse than the optimal ARC one, and a worst-case recall cost that is more than 55% worse that the optimal $WCRC$ one. Second, this improvement in performance is obtained together with an even more significative improvement in terms of computational cost: the $WCRC$ optimization was about 156 times faster than BDC, and about 81 times faster than ARC.

Finally, the BDC and $WCRC$ optimization criteria were compared in a larger example, consisting of on four layers, with 8 batches of input (raw) material, 7 nodes in the second layer, 16 nodes in the third layer and 13 batches of finished product was considered. This network is only partially interconnected, according to the diagram reported in Fig. 6. As it can be seen, the initial configuration contains 78 feasible links and 44 nodes (a fully-interconnected configuration would involve 376 links). The maximum capacity of each node, the amount of raw material of the input batches and the desired quantity in the output nodes are also reported in Fig. 6. Remark that the particular features in this example, i.e. a number of layers and the definition *a-priori* of the set of feasible links, can not be managed by the formulation in Dupuy et al. (2005).

The solutions of the two minimizations are reported in Figures 7 and 8, respectively for the BDC and the $WCRC$ criteria. The relative quantities of interest are reported in Table 2. To comments these results, it can be observed that the direct minimization of the worst-recall cost allows a 25% improvement of the $WCRC$ performance. This can be interpreted as follows: by adopting the configuration in Figure 8 one is guaranteed that, no matter what is the initial product found inadequate, the quantity of material to be recalled will be less than 24,600. Contrary, adopting the BDC solution of Dupuy et al. (2005) reported in Figure 7, one can incur a recall cost as high as 30,750. Also, it can be noted that, as a side product, the $WCRC$ minimization allows to exclude from the supply chain four nodes (nodes 23, 26, 27, 31), while the optimal BDC one excludes two nodes only (22, 31). In a typical industrial situation, this would easily correspond to a save in the production costs. Moreover, also in this case, the computation time for the BDC criterion was dramatically higher than the one of $WCRC$ (7 days compared to half an hour). This behavior, which was observed in all our simulations, can be explained by the fact that the BDC cost function formulation (15) requires the introduction of $n_{in} \times n_{out}$ binary variables, which are not present in the $WCRC$ one. This seems to explain the large increase in the computational time.

4. Conclusions and future research directions

In this paper novel criteria and methodologies for measuring and optimizing the performance of a traceability system have been introduced. As opposed to the methods previously adopted, which optimize indirect measures, the proposed approach takes in direct account the worst-case or the average quantity of product that should be recalled in the case of a crisis. Numerical examples testify to the effectiveness of the proposed methodology, both in terms of performance and of computational cost.

5. Acknowledgements

The authors would like to thank Constantino M. Lagoa for the fruitful and inspiring discussions on the topics of this paper that arose while he was visiting the first author. This work was partially supported by the grants of the project Namatech – Converging Technologies (CIPE2007), Regione Piemonte, Italy.

References

Achterberg, T., Brinkmann R., Wedler M., 2007. Property checking with constraint integer programming, ZIB-Report 07-37, Konrad-Zuse-Zentrum fur Informationstechnik Berlin.

Alfaro, J. A., Ràbade, L. A., 2009. Traceability as a strategic tool to improve inventory management: A case study in the food industry. International Journal of Production Economics 118, 104-110.

Bechini, A., Cimino, A., Marcelloni, F., Tomasi, A., 2008. Patterns and technologies for enabling supply chain traceability through collaborative e-business. Information and software technology 50, 342-359.

Bertolini, M., Bevilacqua, M., Massini, R., 2006. FMECA approach to product traceability in the food industry. Food Control 17(2), 137-145.

Bollen, A. F., Riden, C. P., Cox, N. R., 2007. Agricultural supply system traceability, Part I: Role of packing procedures and effects of fruit mixing. Biosystems Engineering 98, 391-400.

Donnelly, K. A. M., Karlsen, K., Olsen, P., 2009. The importance of transformations for traceability – A case study of lamb and lamb products. Meat Science 83, 68-73.

Dupuy, C., Botta-Genoulaz, V., Guinet, A., 2005. Batch dispersion model to optimize traceability in food industry. Journal of Food Engineering 70, 333-339.

Gay, P., Piccarolo, P., Tortia, C., 2009. Food traceability systems: performance evaluation and optimization. Proc. of XXXIII CIOSTA, Reggio Calabria, Italy, 17-19 June, 465-469.

Gandino, F., Montrucchio, B., Rebaudengo, M., Sanchez, E.R., 2009. On Improving Automation by Integrating RFID in the Traceability Management of the Agri-Food Sector. IEEE Transactions on Industrial Electronics 56 (7), 2357-2365.

Löfberg, J., 2004. YALMIP : A Toolbox for Modeling and Optimization in MATLAB. In Proc. of the CACSD Conference, Taipei, Taiwan.

Jacobs, R. M., 1996. Product recall – a vendor/vendee nightmare. Microelectronics Reliability 36 (1), 101-103.

Karlsen, K.M., Donnelly K.A., Olsen P., 2011. Granularity and its importance for traceability in a farmed salmon supply chain. Journal of Food Engineering102,1-8

Kumar, S., Budin, E., 2006. Prevention and management of product recalls in the processed food industry: a case study based on an exporter's perspective. Technovation 26, 739-750

Moe, T., 1998. Perspective on traceability in food manufacture. Trends in Food Science e Technology 9, 211-214.

Ràbade, L. A., Alfaro, J. A., 2006. Buyer–supplier relationship's influence on traceability implementation in the vegetable industry. Journal of Purchasing & Supply Management 12, 39-50.

Randrup, M., Storøy, J., Lievonen, S., Margeirsson, S., Arnason, S. V., Olavsstovu, D., Møller, S. F., Frederiksen, M. T., 2008. Simulated recalls of fish products in five Nordic countries. Food Control 19, 1064-1069.

Riden, C. P., Bollen, A. F., 2007. Agricultural supply system traceability, Part II: Implications of packhouse processing transformations. Biosystems Engineering 98, 401-410.

Regattieri, A., Gamberi, M., Manzini, R., 2007. Traceability of food products: General framework and experimental evidence. Journal of Food Engineering 81, 347-356.

Ruiz-Garcia, L., Steinberger, G., Rothmund, M., 2010. A model and prototype implementaion for tracking and tracing agricultural batch products along the food chain. Food Control 21, 112-121.

Sahin, E., Dallery, Y., Gershwin, S., 2002. Performance evaluation of a traceability system: an application to the Radio Frequency Identification Technology. Proc. of the IEEE Int. Conf. on System, Man and Cybernetics 3, 210–218.

Skoglund, T., Dejmek, P., 2007. Fuzzy traceability: a process simulation derived extension of the traceability concept in continuous food processing. IChem 85, 354-359.

Tamayo, S., Monteiro, T., Sauer, N., 2009. Deliveries optimization by exploiting production traceability information. Engineering Applications of Artificial Intelligence 22, 557 – 568.

Thakur, M., Donnelly, K.A.M., 2010. Modeling traceability information in soybean value chains. Journal of Food Engineering 99, 98-105

Thakur, M., Hurburg, C. R., 2009. Framework for implementing traceability system in the bulk grain supply chain. Journal of Food Engineering 95, 617-626

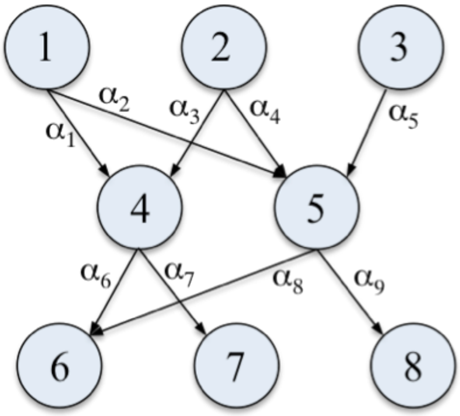
Thakur, M., Wang, L., Hurburgh, C.R., 2010. A multi-objective optimization approach to balancing cost and traceability in bulk grain handling. Journal of Food Engineering 101, 193-200

Wang, X., Li, D., O'brien, C., Li Y., 2010. A production planning model to reduce risk and improve operations management. International Journal of Production Economics 124, 2, 463-474.

Notation table

k, i, l	Indexes
Q_k	Quantity of product contained in the node k
$\overline{Q}_k$	Maximum capacity of the node k
Q_k^0	Initial amount of product in the node k
N_+	Set of natural positive numbers
R	Set of real numbers
m	Number of links
L	Link matrix
$L_{i,1}$	Starting node of the link i
$L_{i,2}$	Destination node of the link i
α_i	Amount of material transferred through the link i
$\overline{\alpha}_i$	Binary variable equal to one when $\alpha_i > 0$
I	Set of indexes of all nodes
I_{in}	Set of indexes of input nodes
I_{out}	Set of indexes of output nodes
n_{nodes}	Number of nodes
n_{in}	Number of input nodes
n_{out}	Number of output nodes
in_k	Set of links entering the node k
out_k	Set of links leaving the node k
S_k^l	Binary variable equal to one when a product from batch node l is present at node k
$\vee$	OR - operator
$\wedge$	AND - operator
a, b, r	Binary variables
M	Sufficiently large number
n_{type}	Number of product types
p	Product type
T_p	Set of nodes containing the type of product p
D	Matrix of recipe disassembling coefficients
A	Matrix of recipe assembling coefficients
$RC(l)$	Recall cost of the product l
w_i	Weights of the weighted recall cost
$WCRC$	Worst-case recall cost
ARC	Average recall cost
WRC	Weighted recall cost
BCD	Batch dispersion cost
$D\_DISP(l)$	Downward dispersion from node l

Figures



$$L = \begin{bmatrix} 1 & 4 \\ 1 & 5 \\ 2 & 4 \\ 2 & 5 \\ 3 & 5 \\ 4 & 6 \\ 4 & 7 \\ 5 & 6 \\ 5 & 8 \end{bmatrix}$$

$I_{in} = \{1,2,3\}, I_{out} = \{6,7,8\}$

For node 4 we have:

$in_4 = \{1,2\},\ out_4 = \{6,7\}$

$\alpha_1 + \alpha_3 = Q_4 = \alpha_6 + \alpha_7$

$Q_4 \leq \overline{Q}_4$

State of node 6 with respect to source node 1:

$$S_6^1 = \bigvee_{i=6,8} \left(S_{L_{i,1}}^1 \wedge \overline{\alpha}_i \right) = \left(S_4^1 \wedge \overline{\alpha}_6 \right) \vee \left(S_5^1 \wedge \overline{\alpha}_8 \right)$$
$$S_4^1 = \bigvee_{i=1,3} \left(S_{L_{i,1}}^1 \wedge \overline{\alpha}_i \right) = \left(S_1^1 \wedge \overline{\alpha}_1 \right) \vee \left(S_2^1 \wedge \overline{\alpha}_3 \right) = \left(1 \wedge \overline{\alpha}_1 \right) \vee \left(0 \wedge \overline{\alpha}_3 \right) = \overline{\alpha}_1$$
$$S_5^1 = \bigvee_{i=4,5} \left(S_{L_{i,1}}^1 \wedge \overline{\alpha}_i \right) = \left(S_1^1 \wedge \overline{\alpha}_2 \right) \vee \left(S_2^1 \wedge \overline{\alpha}_4 \right) \vee \left(S_3^1 \wedge \overline{\alpha}_5 \right) = \left(1 \wedge \overline{\alpha}_2 \right) \vee \left(0 \wedge \overline{\alpha}_4 \right) \vee \left(0 \wedge \overline{\alpha}_5 \right) = \overline{\alpha}_2$$
$$\Rightarrow S_6^1 = \left(\overline{\alpha}_1 \wedge \overline{\alpha}_6 \right) \vee \left(\overline{\alpha}_2 \wedge \overline{\alpha}_8 \right)$$

Figure 1: An illustrating example showing the formalism introduced in the paper. An example of equations (2), (3) is reported for node 4. Also, explicit construction of the states introduced in (4) is given for node 6 and source 1.

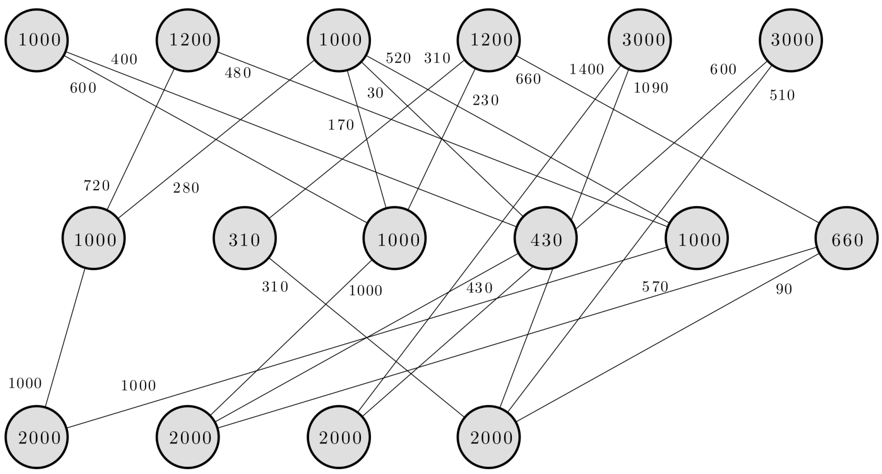


Figure 2: Optimal solution of the example in Tamayo et al. (2009) using the ARC index. The average recall cost of this solution is ARC=3,333.

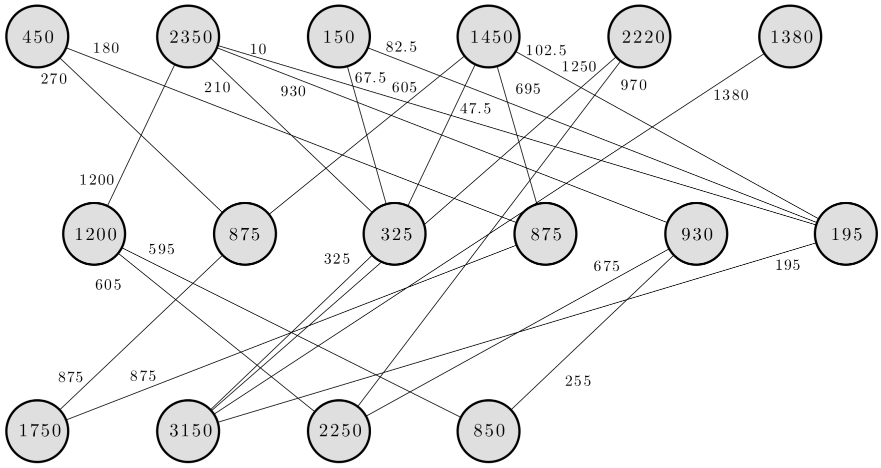


Figure 3: Second numerical example. Optimal solution obtained minimizing the BDC cost.

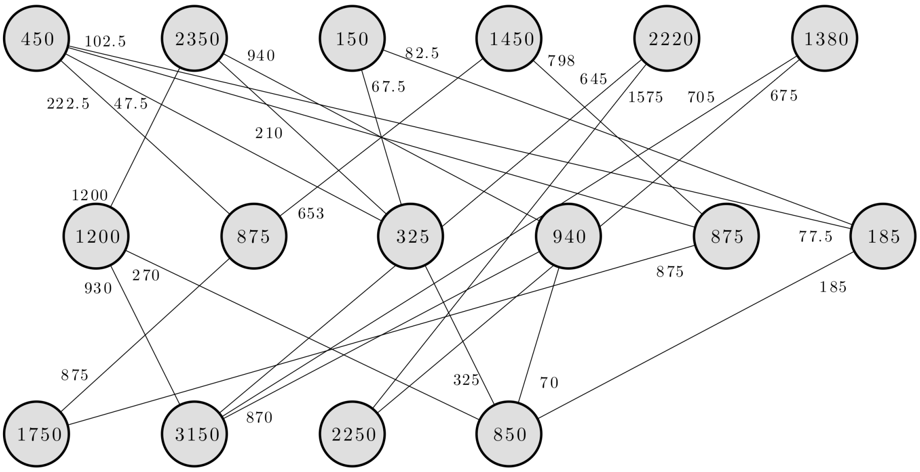


Figure 4: Second numerical example. Optimal solution obtained minimizing the ARC cost.

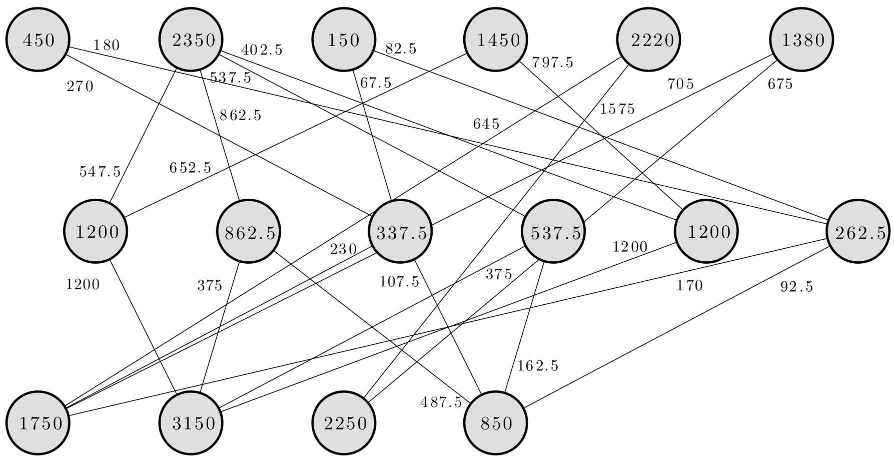


Figure 5: Second numerical example. Optimal solution obtained minimizing the WCRC cost.

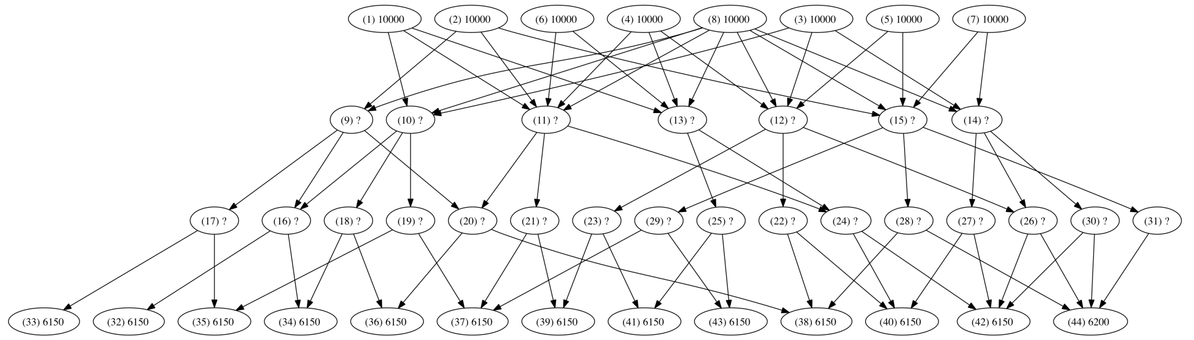


Figure 6: Initial configuration for the four-stage example.

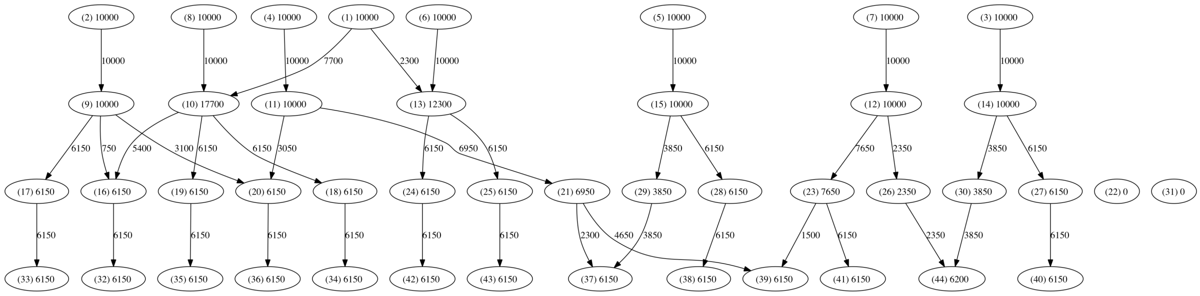


Figure 7: Optimal configuration for the four-stage example obtained minimizing the BDC index.

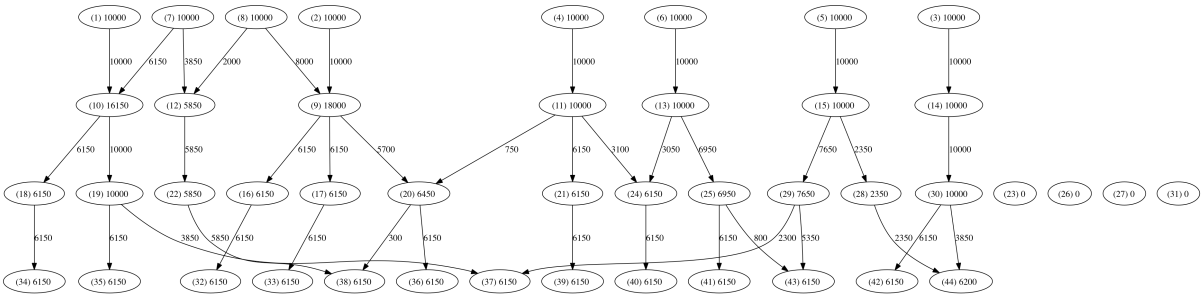


Figure 8: Optimal configuration for the four-stage example obtained minimizing the WCRC index.

Tables

Optimization index	*Solution time*	*# of links*	*BDC*	*ARC*	*WCRC*
BDC mimimization	*92.54*	*24*	***10***	*4,100*	*6,250*
ARC mimimization	*47.81*	*23*	***10***	***3,333***	*5,400*
WCRC mimimization	***0.59***	*24*	*13*	*3,392*	***4,000***

Table 1. Numerical results of the second example.

Optimization index	*Solution time*	*# of links*	*BDC*	*ARC*	*WCRC*
BDC mimimization	*640,780*	*41*	***23***	*17,694*	*30,750*
WCRC mimimization	***2,031***	*43*	*31*	*20,769*	***24,600***

Table 2. Results of the optimization of the four levels problem considering BDC and WCRC optimization criteria