

Feedback control and optimization for the production of commercial fuels by blending

M. Chèbre, Yann Creff, Nicolas Petit

▶ **To cite this version:**

M. Chèbre, Yann Creff, Nicolas Petit. Feedback control and optimization for the production of commercial fuels by blending. Journal of Process Control, 2010, 20 (4), pp.441-451. 10.1016/j.jprocont.2010.01.008 . hal-00465535

HAL Id: hal-00465535

https://minesparis-psl.hal.science/hal-00465535

Submitted on 22 Mar 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Feedback control and optimization for the production of commercial fuels by blending

M. Chèbre, Y. Creff, and N. Petit [a,b,c]

[a] *Advanced Process Control Department, Technical Direction TOTAL Refining & Marketing, Le Havre, France*

[b] *Technology Division, Control Department, IFP Lyon, France*

[c] *Centre Automatique et Systèmes, Unité Mathématiques et Systèmes, MINES ParisTech, France*

Abstract

This paper presents a control algorithm for blending systems. Such systems are used in refining to produce mixtures having specified properties from several components. The underlying control problem is multi-variable, with constraints on the inputs and outputs, and involves large uncertainties. To address this complexity, a constrained optimization problem is formulated, while the uncertainties are treated in closed-loop by an estimator of the components properties. Besides a theoretical study of the main algorithm and a study of convergence, the paper presents numerous technical details that are needed to solve blending problems as they appear in refining operations. Among them are a general lack of measurements, variable delays, measurement synchronization, infeasibilities management, hydraulic constraints, and pre-blends. Industrial case-studies are provided and stress the relevance of the approach.

Introduction

The main contribution of this paper is a new method to control blending systems. These systems are used to produce a mixture having some desired properties. This control problem is very common in refineries. The method we propose can be used in various situations where non-reactive components are blended and linearly impact on the properties of the blend, provided nonlinear properties transformations are used (e.g. log scaling).

The proposed method has the advantage of dealing with uncertainties of the components properties. This robustness is of particular interest when time-

varying upstream plants are considered. These upstream units produce components with time-varying, unmeasured, unknown or poorly known properties.

From an application point of view, the blending objectives are to produce a mixture with some prescribed properties. For gasoline, diesel or fuel production, these can be the octane number, sulfur concentration, Reid Vapor Pressure among others. Minimization of production costs is also an important topic.

Over the past 40 years, blending control systems have attracted much attention (see [18,5,10,17,22]). There has been significant research effort to propose closed-loop strategies using signals from on-line analysers located downstream the blender. It is worth mentioning that, usually, only downstream measurements are considered. The main reason for this choice is to minimize the number, and thus the cost, of required analysers. This point will be illustrated in Section 1. Basic strategies use single-variable controllers (mostly integral effect) in a single-input single-output modeling approach. A priori estimates of the components properties are used to assign the feedback loops. Following a common practice [23,9], we design a multi-variable strategy to improve performance. As will appear, we propose a genuine approach in which measurements are used to update knowledge on the components properties.

Another point of interest, in real applications, is the constraints. Constraints on the blend properties arise from production requirements. Others include flows limitations and pumping constraints. Simple rules can be used to take these into account one by one. More generally, in our approach, it is also possible to take them into account: we solve a constrained optimization problem.

The starting point of our study originates in the late 70's when, although commercial blending control packages were already available, the problem of robustness against uncertainties on the properties of components had not received any satisfactory answer. Various companies and teams of researchers have been involved since then. TOTAL has inherited a long experience in blending control, which can be traced back to the early 80's and the first on-line applications of the Anamel software. In the late 90's, it was decided to address the robustness problem: the result of this work is the development of the latest version of Anamel [6], which is now being used on 17 blenders located in 6 refineries within the TOTAL group and is presented here.

The main algorithm of this latest version of Anamel (in service since 2001) consists of two distinct, though connected, layers: an optimization problem and a feedback loop with an observer. The optimization problem permits to account for various constraints and production cost minimization. The observer is used to partially estimate the components properties in a spirit of adaptive control methods. Both layers are required to provide convergence

and guarantee a successful blend. Other approaches based on estimates of the components properties perform a preliminary or in-line full identification of these properties (with laboratory sampling). In particular, one can refer to [18] for an estimation procedure performed before the recipe can be updated. On the other hand, one can refer to [10] for a strategy using direct in-line measurements of the components properties. Interestingly, to provide convergence of the blend properties to a prescribed target, Anamel's observer needs not to converge to the actual unknown values of the components properties. This might be the case though, but may not be so common in practice. Most of the time, the blend target is reached before accurate estimates of the components properties are obtained. This is not a concern, because blend properties are definitely the primary target, and should not be surprising to anyone familiar with adaptive control [15,2]. Interestingly, this behavior can be analyzed using LaSalle's invariance principle (see e.g. [16]) for the underlying dynamical system.

The paper is organized as follows. In Section 1, we present the blending process, the actuators and the available sensors. In Section 2, we expose the control problem. We introduce the notations to be used throughout the paper. In Section 3, we detail the solution and prove its theoretical convergence. Numerous important practical issues are discussed. We expose solutions for them. Finally, industrial results are presented and discussed in Section 4, while we draw some conclusions in Section 5.

1 Process description

In this section, we present the blending process which is used to obtain finished or semi-finished products from refined components or upstream units flows. All the key elements in the flow-sheet are detailed. We describe the available sensors and actuators. As will appear, the main operational problem is that the components properties are poorly known and usually not measured on line. In fact, only the blend properties are analyzed on line.

1.1 Obtaining semi-finished or finished products from refined components

In refineries, semi-finished or finished products (in particular commercial fuels) are not directly extracted as parts of crude oils, but are produced by blending several *components*. These components are transformed petroleum cuts, resulting from preliminary separations in atmospheric and vacuum columns. Mixing the various components with the right proportions provides the final *blend*, with properties required by the m specifications of interest.

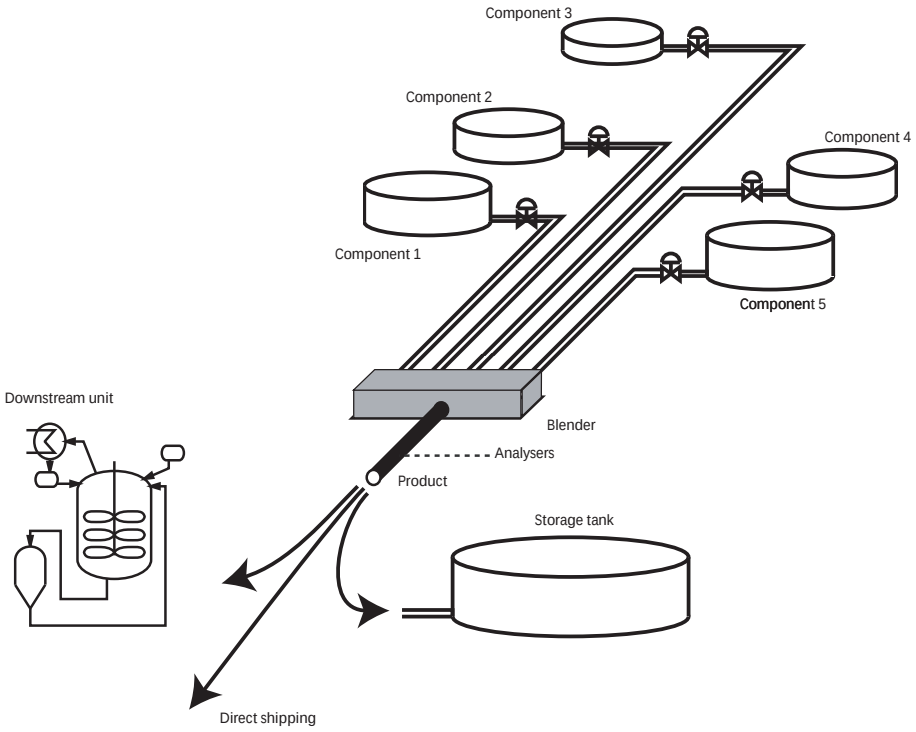


Fig. 1. Typical blending process. Components from storage tanks are pumped to a blender to be stored in a final product tank.

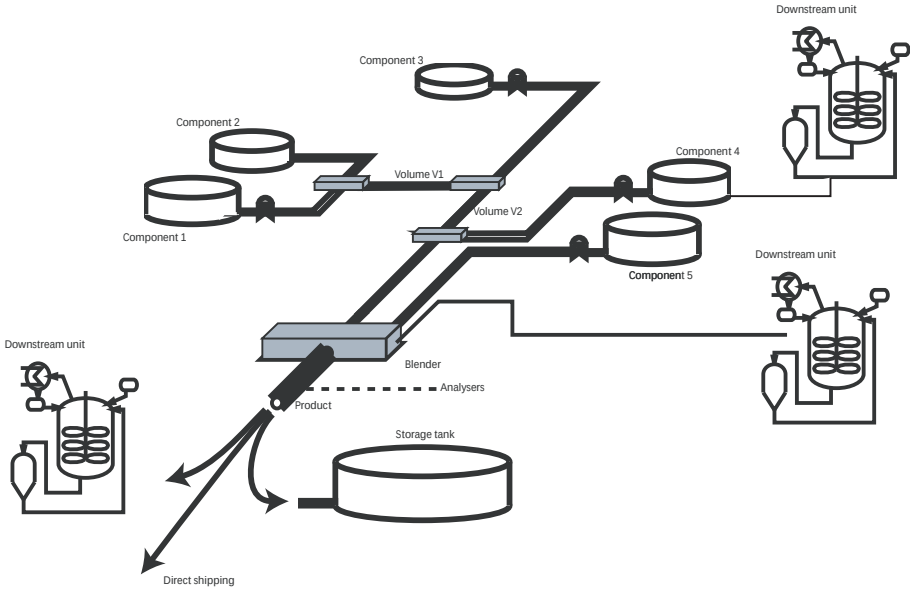


Fig. 2. Typical blending process with pre-blends.

A typical blending process is pictured in Figure 1. Several components (n) are pumped to a blender (also referred to as static mixer) from intermediate storage tanks or pipes (run down). Downstream the blender, the product is either stored in a final product tank, or routed to another refining unit, or shipped off the refinery.

Several elements appear in the flow-sheet. These are set during the line-up procedure. Pipes connect tanks outlets to the blender, with the possibility of "pre-blends". This possibility, pictured in Figure 2, is used to reduce the total length of pipes. Additionally, a flushing tank and several product tanks can be found at the blender outlet. They enable alternative production modes such as sequenced batch (also called "rundown blending" which is frequent when very large productions are considered).

1.2 Actuators and sensors

The blend properties can be controlled with the n blender motorized inlet valves. Given a blender outlet total volume flow rate F, the valves openings define a control vector consisting of n volume ratios $u = (u_1, \ldots, u_n)^T$, referred to as *the recipe*. Low-level flow controllers (named *ratio control system*) guarantee that this vector tracks any reference signal.

Let us now detail the sensors system. For cost and reliability reasons (sensors drift over time), the components properties are generally not measured on line. Yet, there exist large uncertainties on their values. Among the sources of uncertainty are the possible drifts in the operation of upstream units that may be inaccurately controlled during some periods of time, and slicing phenomena in storage tanks. From time to time, laboratory data provide samples measurements. These measurements have a very low time resolution, and can also be inaccurate and delayed. The main culprits are the semi-manual definition of the time stamp, the location of the sampling point, and faulty data input in the laboratory information system. The most interesting information is actually found downstream the blender. A limited number of analyzers are installed at the blender outlet. With these, the m properties of the blend are measured on-line. Usually, m is larger than n. Yet, depending on the type of sampling loop and the involved measurement technology, these measurements can suffer from large and variable delays.

2 Control problem

The primary goal of any blending system is to produce a mixture having some specified properties. In other words, the blending system has to find a n-dimensional recipe u such that the m properties of the mixture satisfy some objective.

The instantaneous blend properties are considered as the output of the system. They are denoted by the $m-$dimensional vector y. Alternatively, if a product

tank is considered, it may be desired to control the average (m-dimensional) properties z of this tank.

The components properties are grouped in a $m \times n$ matrix B.

The two ends of the flow-sheet are the components inlets, and the blender. Flows through the pipes of the flow-sheet are assumed to satisfy a one dimensional plug-flow model. When they are considered, "pre-blends" consist of several pipes merging at the same location (see again Figure 2).

In this paper, the blending models are assumed to be linear. This assumption is not restrictive, because, up to some change of variables, numerous properties actually satisfy this linearity assumption. Therefore, preliminary vector coordinate wise non linear mappings can be used to validate this assumption. In particular, at steady-state, the following relation holds

$$y = Bu \tag{1}$$

Balance equations can be considered for non steady cases.

Several constraints on the recipe u need to be considered. For mathematical consistency, the recipe vector u coordinates must all lie in $[0, 1]$ and satisfy $\sum_{i=1}^{n} u_i = 1$. From an operational and economical point of view, u should remain close to a recipe of interest u^{opt}. Further, hydraulic constraints (physical limitations of the pumps and pipes) and components availability impose upper and lower bounds on the coordinates u_i, $i = 1, ..., n$.

Various constraints on the blend properties y need to be considered as well. A reference y^{r} and/or upper and lower bounds are associated to each coordinate of y. From a practical point of view, these bounds can be considered as *hard bounds* (related to commercial specifications) or *soft bounds*, which can be violated at the expense of profit losses (also referred to as "give-away").

The matrix of the components properties B is poorly known. Yet, $\hat{B}$ an initial estimate for it (most frequently given by laboratory samples) and some on-line blend properties measurements are available.

Before we present our control solution and discuss implementation issues, let us summarize our goal.

Problem 1 *Given $\hat{B}$ an initial estimate of B, given real time measurements of the output blend properties y, find a closed-loop control scheme, acting on u, such that y converges to y^{r} and remains between pre-specified bounds. At all times, u must satisfy the operational constraints, and preferably be close to a recipe of interest u^{opt}.*

3 Proposed solution

To solve the control problem discussed in Section 2, we propose a twofold approach. The constraints and the various control objectives are formulated in an optimization problem. Simultaneously, an observer reconstructs an estimate of the components properties. These two parts of the control law closely interact. Under some mild simplifying assumptions, theoretical convergence of this strategy is studied. Interestingly, the reconstructed components properties need not to converge to their actual values to guarantee a successful blend.

3.1 Optimization problem

It is considered here, that $\hat{B}$ an estimate of the components properties matrix B is given. Later, in § 3.2 this matrix will be updated.

As discussed in Section 2, not every blend property needs to match a specified reference. Some of them are simply asked to remain within some given bounds. Values of blend properties associated to specified references y^{r} ($\dim y^{\mathrm{r}} = r \leq m$) can be estimated, using (1), through an $(r \times n)$ sub-matrix $\hat{B}^{\mathrm{r}}$ of $\hat{B}$. Typically, r ranges from 2 to 5, while n ranges from 5 to 10. Similarly, the blend properties associated to hard and soft bounds can be computed using the sub-matrices $\hat{B}^{\mathrm{h}}$ and $\hat{B}^{\mathrm{s}}$. Lower and upper bounds vector on the hard and soft constraints are noted $y^{\mathrm{h,lb}}$, $y^{\mathrm{h,ub}}$, $y^{\mathrm{s,lb}}$, and $y^{\mathrm{s,ub}}$, respectively. Vector lower and upper bounds on the control vector are noted u^{lb} and u^{ub}. Taking into account the consistency equation $\sum_{i=1}^{n} u_i = 1$, the recipe of interest u^{opt}, and, most importantly, the blending objectives, we formulate the control problem under the form of the following optimization problem

$$\begin{aligned} &\min_{u} \|u - u^{\mathrm{opt}}\|_Q^2 \\ &\begin{cases} 0 \leq u^{\mathrm{lb}} \leq u \leq u^{\mathrm{ub}} \leq 1 \\ \sum_{i=1}^{n} u_i = 1 \\ \hat{B}^{\mathrm{r}} u = y^{\mathrm{r}} \\ y^{\mathrm{h,lb}} \leq \hat{B}^{\mathrm{h}} u \leq y^{\mathrm{h,ub}} \\ y^{\mathrm{s,lb}} \leq \hat{B}^{\mathrm{s}} u \leq y^{\mathrm{s,ub}} \end{cases} \end{aligned} \tag{2}$$

where a symmetric definite matrix Q is used to weight the Euclidian norm, i.e. $\|u\|_Q^2 = u^T Q u$. This matrix is chosen to promote or to penalize the use of some components (further details are given in §3.7.2). The optimization problem (2) is a quadratic programming problem. It can be handled with various software

packages such as IMSL [14]. Its solution gives an open-loop control u.

3.2 Feedback

As mentioned earlier, on-line blend properties measurements can be used to update the open-loop control law. These measurements are usually used to monitor the blend, and we wish to take them into account under the form of a feedback into the optimization problem (2).

In the presented context of linear multi-variable control, a classic way to proceed (e.g. in Model Predictive Control [11,13] or Internal Model Control [12]), is to introduce an integral term fed by the difference between the measurement and the setpoint, or a filter of the difference between the measurement and the prediction of a model. Such classic procedures fail here. A main reason is the equality constraint on the control vector, which reduces the number of free control parameters by one. The actual gains of the $(n-1)$ remaining control variables are in fact unknown, because they consist of differences between entries of the B matrix which are also unknown. For sake of illustration, a very simple test case is given in Appendix A. Therefore, another approach must be considered.

First, we relate the measurements, which are assumed to be done continuously, to the current values of the control variable by $y = Bu$. Then, the estimation $\hat{B}$ of B, which is assumed to be constant, is updated as follows. Considering its j^{th} row $\hat{B}_j$, the continuous-time update law is

$$\frac{d\hat{B}_j^{\mathrm{T}}}{dt} = -\beta_j H u \left(\hat{B}_j u - y_j\right), \tag{3}$$

where H is the following diagonal scaling matrix ($\bar{u}$ being a reference recipe, e.g. a constant value close to u^{opt})

$$H = \frac{1}{\|\bar{u}\|} \begin{pmatrix} \frac{1}{\bar{u}_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\bar{u}_n} \end{pmatrix}$$

and β_j is a strictly positive parameter. This update law is analogous to those found in adaptive control [15,2]. Considering the output relation (1), the essential idea is the comparison of the observed system response Bu with the model output $\hat{B}u$.

Finally, our feedback control law is designed as follows.

Solution to problem 1 (Simple version) *Sequentially, solve the optimization problem* (2) *and update the estimate of the components properties* $\hat{B}$ *when new measurements are available. The discrete-time update is the sampled version of the continuous-time update law* (3).

We now prove convergence of this strategy under the assumption that the optimization problem (2) is always feasible, i.e. admits a (necessarily unique because of strict convexity) solution.

3.3 Convergence

Solution 1 combines an on-line parameter estimator (3) and a control law which is defined as the solution of the optimization problem (2). From this description, it can be viewed as an (indirect) adaptive controller [15]. As will now appear, $\hat{B}$ is continuously adjusted so that $\hat{B}u(t)$ approaches $Bu(t)$ as $t \to +\infty$. Yet, no particular effort is made to design the input $u(t)$ so that $\hat{B}$ converges toward B, as would normally be desired in an on-line parameter estimation technique. This is not one of the objectives, as it could cause large variations of the input signal $u(t)$ (e.g. to satisfy some persistency of excitation property [15,3,16]).

Consider, for property $j = 1, \ldots, m$, the scalar function (Lyapunov function candidate)

$$\Psi(\hat{B}_j) = \frac{1}{2}(\hat{B}_j - B_j)H^{-1}(\hat{B}_j - B_j)^{\mathrm{T}}$$

This function is strictly positive away from B_j, where it equals 0. Its time-derivative along the trajectories of (3) is

$$\frac{d\Psi}{dt}(\hat{B}_j) = -\beta_j \left(\hat{B}_j u - y_j\right)^2 \leq 0$$

Therefore, $\Psi(\hat{B}_j)$ is a Lyapunov function for system (3) (see [16]). From LaSalle's invariance principle, for any initial condition, the solution of system (3), $\hat{B}_j(t)$, converges when $t \to +\infty$ towards the largest invariant set of (3), included in the subset $\{\hat{B}_j \text{ s. t. } d\Psi/dt(\hat{B}_j) = 0\}$. Therefore, $\hat{B}_j(t)$ converges in a way such that $B_j u = \hat{B}_j u$. Yet, by definition of the optimal control problem (2), which is assumed to possess a solution (which is necessarily unique), $\hat{B}_j u$ satisfies the blend objectives. Therefore, so does $y_j(t) = B_j u(t)$, in the limit as $t \to +\infty$. The same reasoning applies to all the blend properties. In summary, the blend is successful, even though $\hat{B}_j$ does not converge

to B_j. In details, we have

$$\begin{aligned} \lim_{t\to+\infty} \hat{B}^{\mathrm{r}}(t)u(t) &= y^{\mathrm{r}}, \\ y^{\mathrm{h,lb}} \leq \lim_{t\to+\infty} \hat{B}^{\mathrm{h}}(t)u(t) &\leq y^{\mathrm{h,ub}}, \\ y^{\mathrm{s,lb}} \leq \lim_{t\to+\infty} \hat{B}^{\mathrm{s}}(t)u(t) &\leq y^{\mathrm{s,ub}} \end{aligned}$$

while the equality $\lim_{t\to+\infty} \hat{B} = B$ *might not hold.*

3.4 Handling infeasibility with cascaded optimization problems

The (preliminary) solution to problem 1 we propose converges provided that the optimization problem (2) is always feasible. Certainly, it is a reasonable assumption to require that the blend is feasible using the components at our disposal. Yet, one should notice that the optimization problem is not written in terms of the components properties matrix B, but in terms of its estimate $\hat{B}$. Thus, the feasibility assumption bears on an estimate, which, as it changes over time, might reveal troublesome.

For real-time applications, it is required that the control algorithm always provides an answer, even a partly disappointing one when it seems impossible to satisfy every control objective. When the optimization problem (2) is not feasible, it remains possible to solve easier problems, with a smaller number of constraints. As already discussed, the output constraints are sorted with respect to their relative importance. This ranking is a common formulation of blending operators. When the convex optimization problem (2) is not feasible, the constraints can be relaxed according to the following procedure (see Appendix B for details)

- *Step 1: Hard constraints management.* Lower and upper bounds on u are considered, along with the consistency equation $\sum_{i=1}^{n} u_i = 1$. The equality constraints for the references and the inequality constraints corresponding to the soft constraints are totally forgotten. Hard constraints are substituted with (positive) slack variables s appearing in the modified objective function. These slack variables are used as penalty functions to be minimized. The target recipe u^{opt} does not appear in this new objective function. This problem is always feasible. Its solution (u^*, s^*) gives achievable values for the hard constraints.
  The resolution is achieved sequentially. The typical problem solved at iter-

ation $k+1$ is given by

$$\min_{u,s} \|\hat{B}^{\mathrm{h}}_{k+1} u + s - y^{\mathrm{h,ub}}_{k+1}\|^2$$
$$\begin{cases} 0 \leq u^{\mathrm{lb}} \leq u \leq u^{\mathrm{ub}} \leq 1 \\ s \geq 0 \\ \sum_{i=1}^{n} u_i = 1 \\ \hat{B}^{\mathrm{h}}_{1k} u \leq y^{\mathrm{hach,ub}}_{1k} \end{cases}$$

where $\hat{B}^{\mathrm{h}}_{k+1}$ and $y^{\mathrm{h,ub}}_{k+1}$ correspond to the hard constraints treated at iteration $k+1$; $\hat{B}^{\mathrm{h}}_{1k}$ correspond to the hard constraints relaxed at iterations 1 to k, the values of which being given by $y^{\mathrm{hach,ub}}_{1k}$. Only maximum bounds have been considered here to simplify the formulation.

- *Step 2: References management.* Lower and upper bounds on u are considered, along with the consistency equation $\sum_{i=1}^{n} u_i = 1$. The previously updated hard constraints, now achievable, are also considered, but the soft constraints are totally forgotten. References simply appear in the objective function, while the target recipe u^{opt} does not appear in the formulation. This problem is also always feasible. Its solution u^{**} gives achievable values for the references.
  The typical problem solved in this step is given by

$$\min_{u} \|\hat{B}^{\mathrm{r}} u - y^{\mathrm{r}}\|^2$$
$$\begin{cases} 0 \leq u^{\mathrm{lb}} \leq u \leq u^{\mathrm{ub}} \leq 1 \\ \sum_{i=1}^{n} u_i = 1 \\ \hat{B}^{\mathrm{h}} u \leq y^{\mathrm{hach,ub}} \end{cases}$$

  where the hard constraints updated in Step 1 are given by $y^{\mathrm{hach,ub}}$. Only maximum bounds have been considered here to simplify the formulation.
- *Step 3: Soft constraints management.* Lower and upper bounds on u are considered, along with the consistency equation $\sum_{i=1}^{n} u_i = 1$. The previously updated hard constraints are still considered, as well as the updated references that are now achievable. Soft constraints are substituted with (positive) slack variables s appearing in the modified objective function. The target recipe u^{opt} does not appear in the formulation. This problem is always feasible. Its solution u^{***} is applied to the blending process.
  The resolution is achieved sequentially. The typical problem solved at iter-

ation $k+1$ is given by

$$\min_{u,s} \|\hat{B}^{\mathrm{s}}_{k+1} u + s - y^{\mathrm{s,ub}}_{k+1}\|^2$$
$$\begin{cases} 0 \leq u^{\mathrm{lb}} \leq u \leq u^{\mathrm{ub}} \leq 1 \\ s \geq 0 \\ \sum_{i=1}^{n} u_i = 1 \\ \hat{B}^{\mathrm{h}} u \leq y^{\mathrm{hach,ub}} \\ \hat{B}^{\mathrm{r}} u = y^{\mathrm{rach}} \\ \hat{B}^{\mathrm{s}}_{1k} u \leq y^{\mathrm{sach,ub}}_{1k} \end{cases}$$

where $\hat{B}^{\mathrm{s}}_{k+1}$ and $y^{\mathrm{s,ub}}_{k+1}$ correspond to the soft constraints treated at iteration $k+1$; $\hat{B}^{\mathrm{s}}_{1k}$ correspond to the soft constraints relaxed at iterations 1 to k, the values of which being given by $y^{\mathrm{sach,ub}}_{1k}$; y^{rach} correspond to the achievable references determined in Step 2. Only maximum bounds have been considered here to simplify the formulation.

In practice, hard and soft constraints are subdivided into sets of decreasing priorities or ranks. Steps (1) and (3) are treated iteratively, each iteration involving a set of constraints with equal priorities. In summary, these steps always produce a recipe u which satisfies as many of the constraints appearing in the original optimization problem as possible. This is interesting because it is consistent with the requirements of real-time control.

3.5 Accounting for delays

As had appeared earlier (in §3.3), the feedback law uses an error between predicted measurements and actual ones (see Equation (3)). The delays, that are the main sources of mis-synchronization, must be carefully accounted for, for the error to be correctly computed. These delays either come from the measurement process, or from dead volumes (pre-blends) that induce transportation delays. Interestingly, the delays are either known or can be estimated with a good accuracy. In practice, it appears that we do not have to use specific control methods robust to delay uncertainty (one can refer to [21] for an overview of such methods).

3.5.1 Measurement delays

The measurement delay may induce important mismatches and inconsistencies in the above error calculation. Fortunately, measurements come along with

sampling dates (time stamps) and it is possible to compare them against the right (delayed) predictions. The formula (3) is modified accordingly.

3.5.2 Variable transportation delays

In practice, transport delays must be accounted for in the preceding methodology. Interestingly, these transport delays are not constant. They actually depend on the control variables, which are themselves updated by the control algorithm. Under a plug-flow assumption, this dependance can be fully determined.

We now expose means to calculate the transport delays in a very straightforward manner which is eventually implemented.

First, let us consider a simple example, pictured in Figure 3. Consider two connected pipe sections. Their volumes are noted V_1 and V_2, respectively. We note b_1, b_2 and b_3 the composition of the fluid flowing through this pipe. These variables have distinct values when the fluid contained in the two pipe sections is not homogeneous (e.g. due to upstream transients). We note F the time varying flow rate. Under the assumption of incompressibility, the volumic flow rate is spatially uniform. The compositions are constrained by the following

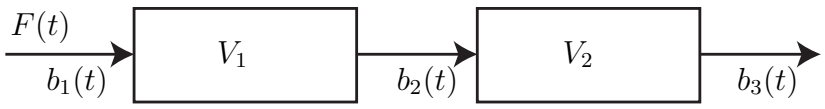


Fig. 3. Transport phenomenon through two connected pipe sections.

delayed equalities

$$b_2(t) = b_1(t - \delta_1(t)), \quad \text{with } V_1 = \int_{t-\delta_1(t)}^{t} F(\tau)d\tau$$

$$b_3(t) = b_2(t - \delta_2(t)), \quad \text{with } V_2 = \int_{t-\delta_2(t)}^{t} F(\tau)d\tau$$

The preceding integral relations (see also [20,19]) implicitly define the varying delays δ_1 and δ_2. Therefore, the upstream and downstream properties satisfy the following equality

$$b_3(t) = b_1(t - \delta_2(t) - \delta_1(t - \delta_2(t)))$$

These relations can be generalized to cases of interest in the context of blending processes such as the one depicted in Figure 4.

In the setup presented in Figure 4, five storage tanks are connected to a single blender through a network of pipes using three pre-blends. Three hold-ups

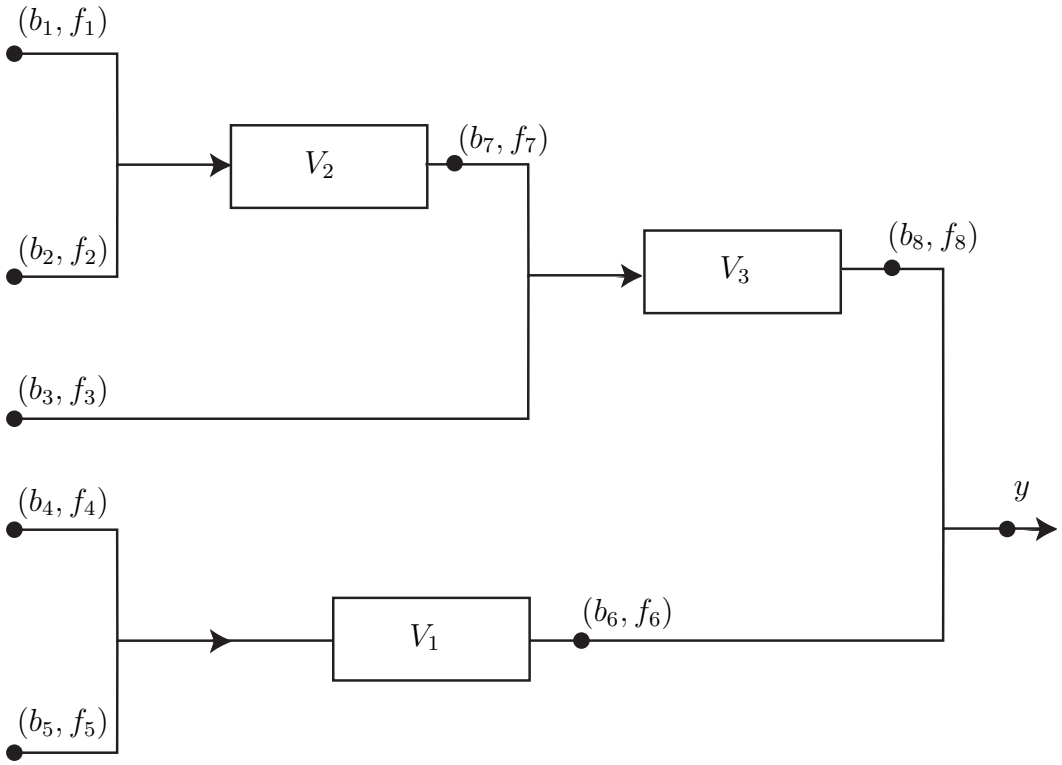


Fig. 4. Transport phenomenon with pre-blends.

play a role in the corresponding transport phenomena. These are noted V_1, V_2, and V_3. Besides the storage tanks compositions $b_{i=1,\ldots,5}$, one has to consider intermediate blends properties. These are noted b_6, b_7, and b_8. Flow rates from the storage tanks are noted $f_{i=1,\ldots,5}$. Intermediate flow rates are $f_6 = f_4 + f_5$, $f_7 = f_1 + f_2$, and $f_8 = f_1 + f_2 + f_3$.

Following the calculations presented in the introductory case, transport delays can be derived from the following equations

$$V_1 = \int_{t-\delta_1(t)}^{t} f_6(\tau)d\tau, \quad V_2 = \int_{t-\delta_2(t)}^{t} f_7(\tau)d\tau, \quad V_3 = \int_{t-\delta_3(t)}^{t} f_8(\tau)d\tau$$

Then, intermediate blends properties are easily determined by

$$\begin{aligned}
b_6(t) &= \frac{f_4(t-\delta_1(t))}{f_6(t-\delta_1(t))}b_4 + \frac{f_5(t-\delta_1(t))}{f_6(t-\delta_1(t))}b_5 \\
b_7(t) &= \frac{f_1(t-\delta_2(t))}{f_7(t-\delta_2(t))}b_1 + \frac{f_2(t-\delta_2(t))}{f_7(t-\delta_2(t))}b_2 \\
b_8(t) &= \frac{f_7(t-\delta_3(t))}{f_8(t-\delta_3(t))}b_7(t-\delta_3(t)) + \frac{f_3(t-\delta_3(t))}{f_8(t-\delta_3(t))}b_3
\end{aligned}$$

Eventually, the end-product property is easily deduced, through a cascade of mixing laws, by combining the time-varying transport delays. After some easy

algebra, the sought-after relation is

$$\left\{\begin{array}{l}(f_1(t)+f_2(t)+f_3(t)+f_4(t)+f_5(t))y(t)= \\ \qquad f_8(t)\dfrac{f_7(t-\delta_3(t))}{f_8(t-\delta_3(t))}\dfrac{f_1(t-\delta_2(t-\delta_3(t)))}{f_7(t-\delta_2(t-\delta_3(t)))}b_1 \\ \qquad +f_8(t)\dfrac{f_7(t-\delta_3(t))}{f_8(t-\delta_3(t))}\dfrac{f_2(t-\delta_2(t-\delta_3(t)))}{f_7(t-\delta_2(t-\delta_3(t)))}b_2 \\ \qquad +f_8(t)\dfrac{f_3(t-\delta_3(t))}{f_8(t-\delta_3(t))}b_3+f_6(t)\dfrac{f_4(t-\delta_1(t))}{f_6(t-\delta_1(t))}b_4+f_6(t)\dfrac{f_5(t-\delta_1(t))}{f_6(t-\delta_1(t))}b_5\end{array}\right. \tag{4}$$

Interestingly, in the case of constant flow rates from the storage tanks, this last formula can be simplified down to the following usual relation

$$(f_1+f_2+f_3+f_4+f_5)y(t)=f_1b_1+f_2b_2+f_3b_3+f_4b_4+f_5b_5$$

More generally (see Appendix C), Equation (4) takes the form

$$y(t)=\sum_{i=1,\ldots,n}U_i(t)b_i, \tag{5}$$

where $U_i(t)$ can be expressed in terms of (ratios of) values of present and past values of components of the vector recipe. Vectors b_i are the columns of the properties matrix B. Denoting $U(t)$ the $n-$dimensional column vector with entries $U_i(t)$, we simply have $y(t)=BU(t)$. This expression is a substitute to Equation (1) for systems with pre-blends. It is used in estimator (3), which is transformed into:

$$\frac{d\hat{B}_j^{\mathrm{T}}}{dt}=-\beta_j HU\left(\hat{B}_jU-y_j\right), \tag{6}$$

It can be easily shown that this modified estimator still converges to the same invariant set [4].

3.6 Actual control algorithm

The proposed solutions to the issues raised in § 3.4 and in § 3.5 are incorporated in the control algorithm. This yields the following solution.

Solution to problem 2 (Actual version) *Sequentially, solve the optimization problem* (2)*, using whenever required the procedure described in § 3.4 to handle infeasibilities, and then update the estimate of the components properties $\hat{B}$ when new measurements are available. The discrete-time update is the sampled version of the continuous-time update law* (6)*.*

Further enhancements could be considered. It can be noted that, in the optimization problem (2), the expressions $y=\hat{B}u$ are still used in all cases,

with or without pre-blends. This can be viewed as solving a predictive control problem with a control horizon equal to 1 ("single move"), and a single prediction point at infinity, as for a given τ, $U_i(t+\tau) = u_i(t), \forall i \in \{1, \ldots, n\}$.

It is possible to extend problem (2), by using (5) to express the output constraints at various times from current time t to $t+\tau$. This can be done at the expense of introducing strong nonlinearities in the optimization problem. Such an approach has been tested through simulations [4], but has not been implemented, as it did not provide any significant improvement for problems of practical interest.

3.7 Further functionalities

A blending controller has to propose more functionalities than those described so far in this paper. For instance, the total flowrate $F(t)$ is generally handled, for sake of output maximization. Additives have to be managed for diesel fuels. Two other important features are the quality control of storage tanks, and the recipe optimization (the way to choose u^{opt} is equation (2)). These two features are briefly presented in the next sections. All the functionalities stated above are included in Anamel.

3.7.1 Reference trajectories

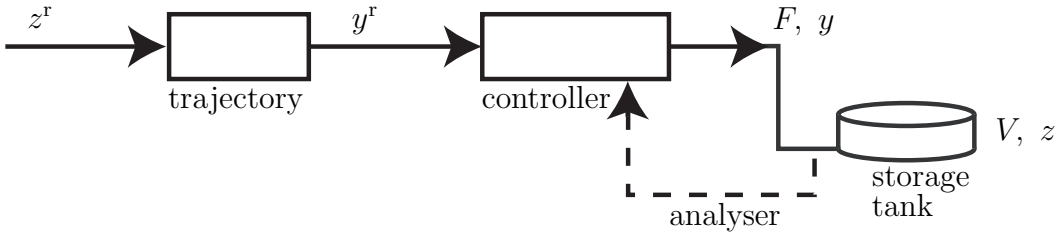


Fig. 5. Reference trajectories management in the case of storage tank composition control.

The control strategy presented in the previous sections focuses on tracking a given (instantaneous) blend properties reference. This reference need not be constant, and can be updated over time. It is possible to cascade the main control algorithm with a trajectory tracking strategy. This can be done to manage blends in storage tanks. A simple (generic) scenario with one storage tank is reported in Figure 5. Note V the current volume of fluid contained in the storage tank, z its composition. Consider H a finite time horizon over which the total blender flow rate F and the blend properties y are assumed constant. The storage tank properties values satisfy the following balance equation $(V + FH)z(t+H) = Vz(t) + FHy$. It is desired that $z(t)$ tracks a reference

signal $z^{\mathrm{r}}(t)$. Then, we set $z(t+H) = z^{\mathrm{r}}(t)$ to define the reference $y^{\mathrm{r}}(t)$ as

$$y^{\mathrm{r}}(t) = z^{\mathrm{r}}(t) + \frac{V}{FH}\left(z^{\mathrm{r}}(t) - z(t)\right)$$

In practice, $z(t)$ is not measured, but frequently estimated through the balance equation given above.

In the case when the storage tank is not empty at the beginning of the blend, a heel management strategy is used. Based on the same balance equation, it consists of defining a reference trajectory originating at this known value.

3.7.2 Recipe optimization

When problem (2) is feasible, the objective function can be used to optimize the recipe, i.e. to promote or penalize the usage of some specific components, which also impacts on the cost of the blend. Using several techniques detailed in [6], it is possible to push the blend properties toward their specified hard constraints. We briefly sketch an overview of such techniques.

When no particular recipe optimization is desired, u^{opt} is either set equal to the initial value provided by the scheduling department of the considered refinery, or equal to the filtered current recipe u^f, to avoid oscillations. When recipe optimization is an objective, u^{opt} equals $u^f + \zeta$, where the entries of ζ are positive (resp. negative) for components to promote (resp. to limit) in the blend. When the cost of the blend must be minimized, a price is associated to each component. Then, the prices of the components with respect to the cost of the current blend, are used to set ζ entries. Details can be found in [6].

4 Industrial results

The algorithm detailed in the previous section, has been implemented in C, and installed in numerous refineries of TOTAL under the name "Anamel". Thanks to its relatively low complexity, the whole control scheme can be run on standard control systems at a rate consistent with the process dynamics (e.g. every 1 to 5 minutes).

4.1 Statistical study

After an initial installation phase of this Anamel software in TOTAL refineries from 2001 to 2004, a statistical study of the performance of the blends

was conducted at the TOTAL Leuna (Germany) refinery, which was then the latest of this installation program. The study was done so as to compare the results of the new on-line blending optimizer that was installed in 2004 in Germany: all the blends of the year 2003 was thus compared to the blends of year 2005 (with 405 gasoline blends, 425 diesel blends, and 173 fuel blends in 2005). Enhancements are detected on gasoline blends with a higher proportion (42% of improvement for Reid Vapor Pressure RVP and 30% for octane number MON) of the blends closer to the hard contraint (lower give-away for monitored qualities). On diesel blenders, a better convergence with less oscillations was obtained. In particular, the cloud point regulation has been improved significantly. Anamel users in this refinery are convinced that the new release of the software has contributed to improve the performance of the overall blend operations.

Besides these general comments, we now briefly present two representative test cases. The first one is a continuous blend upstream a process unit for diesel production, the second one is a batch production of gasoline. Today (2008), Anamel is installed on 17 distincts blending systems (5 for gasoline production, 7 for diesel production, and 5 for fuel production).

4.2 Industrial case 1: continuous blending upstream an HDS unit

In the industrial case presented here, the product is diesel fuel. The blender outlet is directly connected to an hydro-desulphurization unit (HDS). In this context, the instantaneous properties of the blend are the variables that must be controlled.

In details, 6 components are used out of which 3 (representing more than 20% of the total volume flow rate) are set to constant flow rates and cannot be manipulated. There remains a $n = 3$ dimensional recipe vector. One control objective is to maximize the incorporation of component 1.

As before, Anamel is run (according to Solution 2) every 5 minutes. The recipe is subjected to several contraints including hydraulic contraints, min. and max. bounds. Also, its rate of change is limited.

Three properties are of particular interest. These are the flash point, and the cloud point which must track their setpoints and the sulphur content which must remain below some max. value. Further, three other properties must be monitored. These are the density (D15), the cetane index, and the ASTM360.

The results are presented in Figure 6. The recipe histories, the flash point histories, the cloud point histories, and the sulphur rate are presented in (a), (b), (c) and (d), respectively. The D15, cetane index and ASTM360 are easily

kept inside their bounds. They are not reported in the plots. The time horizon reported here is approximately 24 hours (one day).

The results stress that the flash point and the cloud point are kindly controlled. Over time, the controller tends to maximize the rate of component 1, while keeping the sulphur rate below its maximum. At the beginning of the trends, the flash point is below its minimum. This induces a strong control action which vanishes once the problem is solved.

The case reported here is representative of the behavior of the proposed control scheme used on continuous blends. In the presented case of blending upstream an HDS unit, several beneficial aspects have been noted. Gradually, it has been possible to increase the overall feed rate of the unit by 60% over 4 years. This is mainly due to a revamping and a debottlenecking of the unit. But according to end-users in the refinery, it can also be considered as an indirect effect of the controller which keeps the properties of interest within the prescribed bounds and thus minimizes the input disturbances of the HDS unit. It also allowed to maximize the throughput without causing any give-away, and to reduce the variability of the feed rate. Together with a better cloud point control, the blending optimization and the increase of the feed rate have yielded substantial profits which are approximately of 4.3 million Euros/year for a single HDS unit [1].

4.3 Industrial case 2: batch gasoline blend

The second case presented here is a gasoline blend. In the considered setup, the blender outlet is connected to a product tank, the properties of which are of interest. Before the blend is started, the product tank already contains about 20000 cubic meters of gasoline. A remaining amount of 15000 cubic meters has to be added during the blend. In this situation, the heel management technique discussed in §3.7.1 is of importance. A total of 7 components are used, out of which 2 (representing 25% of the total volume flow rate) are set to some given ratios (in percentage) that are to remain unchanged during this operation. The recipe dimension is $n = 5$. To maximize profits, the overall flow rate must be optimized, the target being 1000 cubic meters per hour. No particular recipe optimization is desired. As discussed in § 3.7.2, the target is the initial recipe. A total of $m = 9$ properties are considered. Two properties need to be controlled to their references. These are the motor octane number (MON) and the vapor pressure (VP). During transients, their values (in the storage tank) must remain between some min. and max. bounds. Besides, other properties such as the research octane number (RON), the gravity, the sulphur content, the benzene content, the olefins contents, the aromatics contents, and the 70% distillation point must be monitored. They correspond to hard and

soft bounds.

Anamel is run (according to Solution 2) every 5 minutes. Further, as discussed in Section 2, the recipe is required to satisfy several contraints such as: hydraulic constraints (which can be formulated using incorporation ratios depending upon the current total volume flow rate), min. and max. values, and bounded rate of change (from one run to the next).

The results are presented in Figure 7. The recipe u histories, the MON histories, and the VP histories are presented in (a), (b), and (c), respectively. The total blend time is approximately 15 hours. Early, both MON and VP converge to their setpoints. To achieve these very good results, it is necessary to use varying instantaneous MON and VP setpoints. The recipe varies significantly during the batch, while remaining strictly within its bounds. Noises on the measurements do not impact on the convergence of the algorithm.

The control method is quite robust. This is particularly visible when, after 2.75 h, one of the analyser fails and provides unusable information. Simply, Anamel "freezes" the value of the recipe until the analyser recovers, which happens at t = 3.25 h.

During the first 6 hours (approximately), the system is rather difficult to control because the tank has been filled with off-specifications products and the MON and VP are rather far from their objectives. In particular, the VP in the product tank violates the allowed maximum at the start of the blend. This infeasibility is treated using the technique presented in § 3.4. After about 6 hours, the properties have almost reached their targets. Then, the recipe optimization methods starts, and smoothly brings the current recipe close to the initial recipe.

At the end of the batch, the end-user is left with a tank filled with the desired amount of product with desired specifications and a recipe close to the target one. During the batch all the constraints have been satisfied, and despite the sensor failure, the controller has provided a smooth and efficient transient.

5 Conclusion and perspectives

The presented algorithm addresses the problem of blending control in a way which is new in this field. The algorithm consists of two connected layers forming an adaptive controller: one is an estimator which partially estimates the plant parameters, and one is an optimization problem solver. Available measurements are used in a feedback loop passing through the estimator, which does not need to converge to the actual components properties to guaran-

tee the success of the blend. This fact is of great importance in applications. In the presented context of industrial blending, measurement synchronization has appeared as an important issue, and the algorithm incorporates a solution to compute input-dependent delays in an effective and accurate way. Handling the infeasibilities of the optimal control problem formulated to solve the multi-variable control problem has also revealed an important topic, especially for real-time implementation. The software package resulting from this work, "Anamel" is now used in numerous refineries. It is constantly improved further and is the subject of active research efforts [7,8].

Acknowledgements

The authors wish to thank the following persons: Claude Jablon, Philippe Zaccagnino, Erik Michelsen, Patrick Caillon, Jean-Claude Bonneau, Jean-Louis Frin, Edith di-Crescenzo, Guillaume Pitollat, Bruno Romeyer, Jean-Claude De Wit, Fari Djenab, Jean-Marie Gorszczyk, David Vissière, Julien Barraud, Vincent Bachelot, Michel Bernier, Valérie Guyard, Tarik Idaoubella, Christophe Legros, Stéphane Bourgeon, Cyrille Mirgain, Roland Forterre, Pierre Hoornaert, Bertrand Ecale, Elena Braceras, Lionel Armand, and the TOTAL Grandpuits refinery west-side operators.

References

[1] Anamel HDS2. Post benefit study. Real-time blending optimization. Technical report, TOTAL, Dec. 2008.

[2] K. J. Åström and B. Wittenmark. A survey of adaptive control applications. In *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, volume 1, pages 649–654 vol.1, Dec 1995.

[3] K. J. Åström and B. Wittenmark. *Adaptive control*. Dover, second edition, 2008.

[4] J. Barraud. *Commande de procédés à paramètres variables*. PhD thesis, École des Mines de Paris, 2006.

[5] T. Bay, T.-S. Chen, and J. W. Schwartzenberg. System for feed blending control. Patent, Oct. 1969. Patent number 3,473,008.

[6] M. Bernier, N. Petit, Y. Creff, and M. Chèbre. Method and device for controlling production of a mixture of components, in particular a mixture with premix dead volumes. Patent, May. 2006. Dpt 06.04868, extended may 2007 PCT/FR2007/000886, GCCP/2007/8408, Thailand 0701002593.

[7] M. Chèbre. On line blend monitoring tools in TOTAL refineries. In *ERTC MaxAsset & Computing Conference*, 2006.

[8] M. Chèbre and G. Pitollat. Feed control for an hydrodesulphurization unit using Anamel blend optimizer. In *ERTC MaxAsset & Computing Conference*, 2008.

[9] Blend Property Control. *Release 55.0 Revision 2.* Honeywell Hi-Spec Solutions, http://hpsweb.honeywell.com/NR/rdonlyres/03E1522F-092C-4845-86FE-32ADA378C462/36666/BPCBp5500pdR2.pdf, 2003.

[10] R. T. Feld, N. J. Pitman, T. C. Jr. Cattrall, N. Y. Huntington, H. F. Hoffmann, C. N. Smith, W. Smith, and N. J. Woodbury. Fluid blending system. Patent, May 1968. Patent number 3,385,680.

[11] J. B. Froisy. Model predictive control-building a bridge between theory and practice. *Computers and Chemical Engineering*, (30):1426–1435, 2006.

[12] C. E. Garcia and M. Morari. Internal model control. 2. Design procedure for multivariable systems. *Ind. Eng. Chem. Process Des. Dev.*, 24:472–484, 1985.

[13] C. E. Garcia, D. M. Prett, and M. Morari. Model predictive control: Theory and practice- a survey. *Automatica*, 25(3), 1989.

[14] User's guide. *IMSL C Numerical Library.* Visual Numerics, San Ramon, CA, USA, 2006.

[15] P. A. Ioannou and J. Sun. *Robust adaptive control.* Prentice-Hall, Englewood Cliffs, 1996.

[16] H. K. Khalil. *Nonlinear Systems.* MacMillan, 1992.

[17] D. A. Le Febre and L. M. Lane. Process for controlling blending. Patent, Jul. 1995. Patent number 5,430,295.

[18] J. H. Perkins. Method for predictive intrinsic properties of a mixture. Patent, Dec. 2000. Patent number 6,159,255.

[19] N. Petit. *Systèmes à retards, platitude en génie des procédés et contrôle de certaines équations des ondes.* PhD thesis, École des Mines de Paris, 2000.

[20] N. Petit, Y. Creff, and P. Rouchon. Motion planning for two classes of nonlinear systems with delays depending on the control. In *Proc. of the* 37*th IEEE Conf. on Decision and Control*, pages 1007– 1011, 1998.

[21] J.-P. Richard. Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39(10):1667 – 1694, 2003.

[22] A. Singh, J. F. Forbes, P. J. Vermeer, and S. S. Wood. Model-based real-time optimization of automotive gasoline blending operations. *Journal of Process Control*, 10:43–58, 2000.

[23] P. J. Walton and C. J. Swart. The Boss of blending. *Hydrocarbon Engineering*, July 2003.

A A test case for simple feedback loops

This section proposes an example to discard the classic methods discussed in §3.2. As indicated in this section, a classic control technique is to introduce an integral term fed by the difference between the measurement and the setpoint (*Strategy 1*), or a filter of the difference between the measurement and the prediction of a model (*Strategy 2*). Due to the uncertainties in the components properties, the gains of the systems are poorly know. As will appear, it is difficult to distinguish situations in which closed-loop stability can be guaranteed. To illustrate this point, let us consider the following simple example. Assume that 3 components are mixed to produce a blend with a single property of interest. Note b_1, b_2, b_3 the components properties. It is desired to track a setpoint y^r. In the optimization problem (2), no hard nor soft bounds are considered. Finally, $u^{\text{lb}} = 0$ and $u^{\text{ub}} = 1$.

Let $y^r = 5.7$ and $u^{\text{opt}} = [0.3, 0.4, 0.3]$. By going through the iterations of the control system, various behaviors can be observed, depending on the value of the above mentioned parameters. With $[b_1, b_2, b_3] = [4, 5, 6]$ and an initial estimate $[\hat{b}_1, \hat{b}_2, \hat{b}_3] = [4, 6, 7]$, both *Strategy 1* and *Strategy 2* applied repeatedly provide convergence.

With the exact same setup except $[b_1, b_2, b_3] = [6.5, 5, 6]$, both *Strategy 1* and *Strategy 2* fail. It is not a feasibility issue. To get some insight into this phenomenon, consider Strategy 1. As already discussed, the equality constraint reduces the number of free control parameter to 2. Consider that u_1, and u_2 are used. Their actuals gains are $[b_1 - b_3, b_2 - b_3]$ while they are estimated to $[\hat{b}_1 - \hat{b}_3, \hat{b}_2 - \hat{b}_3]$. In the first case, these quantities evaluate to

$$[b_1 - b_3, b_2 - b_3] = [-2, -1], \quad [\hat{b}_1 - \hat{b}_3, \hat{b}_2 - \hat{b}_3] = [-3, -1]$$

while, in the second case

$$[b_1 - b_3, b_2 - b_3] = [0.5, -1], \quad [\hat{b}_1 - \hat{b}_3, \hat{b}_2 - \hat{b}_3] = [-3, -1]$$

It can be shown, through a simple analysis of the optimization problem, that the solution $[u_1, u_2, u_3]$ is such that $[u_1 - u_3, u_2 - u_3] = \alpha[\hat{b}_1 - \hat{b}_3, \hat{b}_2 - \hat{b}_3]$, where $\alpha \in \mathbb{R}$. The induced improvement in the tracking error is $[u_1 - u_3, u_2 - u_3].[b_1 - b_3, b_2 - b_3]^t$ instead of the expected $[u_1 - u_3, u_2 - u_3].[\hat{b}_1 - \hat{b}_3, \hat{b}_2 - \hat{b}_3]^t$. If $[b_1 - b_3, b_2 - b_3].[\hat{b}_1 - \hat{b}_3, \hat{b}_2 - \hat{b}_3]^t < 0$, the closed loop strategy diverges instead of converging.

B Cascaded optimization

Consider the following simple optimization problem

$$\min_u J(u)$$
$$\begin{cases} u^{\text{lb}} \leq u \leq u^{\text{ub}} \\ b_1^T u \leq y^1 \\ b_2^T u \leq y^2 \end{cases}$$

where y^1 and y^2 are scalar constraints. Infeasibility means that the two inequality constraints can not be both enforced for $u^{\text{lb}} \leq u \leq u^{\text{ub}}$. It can be important however to provide a solution which satisfies them "at best". Assume that the first inequality constraint is more important than the second one. Introducing the slack variable s, we can consider the problem

$$\min_{u,s} \|b_1^T u + s - y^1\|^2$$
$$\begin{cases} u^{\text{lb}} \leq u \leq u^{\text{ub}} \\ s \geq 0 \end{cases}$$

It is always feasible, with solution (u^*, s^*). $s^* > 0$ means that satisfying $b_1^T u \leq y^1$ is possible with $u^{\text{lb}} \leq u \leq u^{\text{ub}}$. But $s^* = 0$ means that $b_1^T u^*$, equal to or greater than y^1, is the nearest achievable value from y^1 for $u^{\text{lb}} \leq u \leq u^{\text{ub}}$. Accordingly, we set the achievable value $y^{1,\text{ach}}$ as $y^{1,\text{ach}} = y^1$ if $s^* > 0$ and $y^{1,\text{ach}} = b_1^T u^*$ if $s^* = 0$. Then we consider

$$\min_{u,s} \|b_2^T u + s - y^2\|^2$$
$$\begin{cases} u^{\text{lb}} \leq u \leq u^{\text{ub}} \\ s \geq 0 \\ b_1^T u \leq y^{1,\text{ach}} \end{cases}$$

This problem is always feasible, with solution (u^{**}, s^{**}). u^{**} can be seen as an acceptable relaxed solution of the initial problem. The cascaded optimization problems described in section 3.4 are solved in a similar way, from highest priority hard constraints to lowest priority soft constraints. Notice that the slack variable s is not required to deal with the equality constraints used for the references.

C Varying delays

Consider a blend network with p pre-blends. Note

- $f_i(t)$ the volume flow rate from the storage tank i at time t, with $i \in \{1, \ldots, n\}$;
- $f_{n+i}(t)$ the total volume flow rate through the dead volumes i at time t, with $i \in \{1, \ldots, p\}$;
- $F(t)$ the total volume rate of the blender at time t, $F(t) = \sum_{i=1,n} f_i(t)$;
- V_i the dead volume associated to pre-blend i;
- b_i the properties of component i.

To each component i, one can associate a path Π_i defined by the sequence of the p_i dead volumes that are passed by from the storage tank to the blender. This sequence consists of p_i distinct integers corresponding to the numbering of pre-blends, $\Pi_i = \{\pi_i^1, \pi_i^2, \ldots \pi_i^{p_i}\}$, with $\pi_i^j \in \{1, \ldots, p\}$ for all $j \in \{1, \ldots, p_i\}$. $\Pi_i = \emptyset$, i.e. $p_i = 0$ means that component i directly feeds the blender.

For each pre-blend i, one can define the set Γ_i of its q_i input flow rates. This set consists of q_i distinct integers corresponding to the numbering of the volume flow rates, $\Gamma_i = \{\gamma_i^1, \gamma_i^2, \ldots \gamma_i^{q_i}\}$, with $\gamma_i^j \in \{1, \ldots, n+p\}$ for all $j \in \{1, \ldots, q_i\}$.

When $\Pi_i = \emptyset$, the properties b_i appear under the form $f_i(t)/F(t)b_i$ in the blend equation. The case $\Pi_i \neq \emptyset$ is different. Consider pre-blend π_i^j, its total flow rate is $f_{\pi_i^j}(t) = \sum_{k \in \Gamma_{\pi_i^j}} f_k(t)$. The transportation delay in this pre-blend is given by $\delta_{\pi_i^j}(t)$ with

$$V_{\pi_i^j} = \int_{t-\delta_{\pi_i^j}(t)}^{t} f_{\pi_i^j}(\tau) d\tau$$

Let us define $\Delta_i^j(t) : t \mapsto t - \delta_{\pi_i^j}(t)$, for all π_i^j in Π_i. For a given i, the composition of these functions is

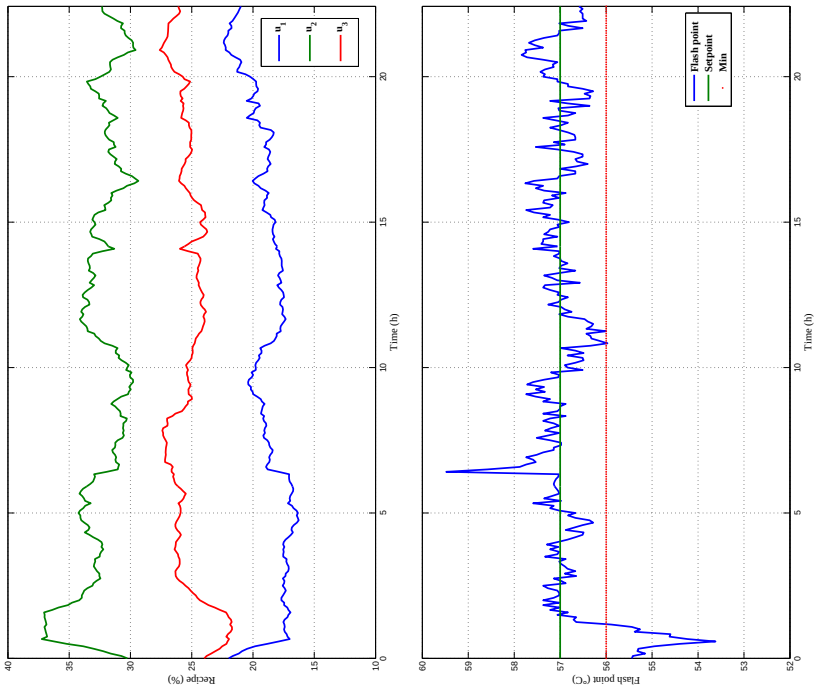
$$\Delta_i^{k,j}(t) \triangleq \Delta_i^k(\Delta_i^j(t)) : t \mapsto t - \delta_{\pi_i^j}(t) - \delta_{\pi_i^k}(t - \delta_{\pi_i^j}(t))$$

and $\Delta_i^{l,k,j}(t) \triangleq \Delta_i^l(\Delta_i^{k,j}(t))$. Note

$$U_i(t) = \frac{f_{\pi_i^{p_i}}(t)}{F(t)} \frac{f_{\pi_i^{p_i-1}}(\Delta_i^{p_i}(t))}{f_{\pi_i^{p_i}}(\Delta_i^{p_i}(t))} \frac{f_{\pi_i^{p_i-2}}(\Delta_i^{p_i-1,p_i}(t))}{f_{\pi_i^{p_i-1}}(\Delta_i^{p_i-1,p_i}(t))} \cdots \frac{f_{\pi_i^1}(\Delta_i^{2,\ldots,p_i}(t))}{f_{\pi_i^2}(\Delta_i^{2,\ldots,p_i}(t))} \frac{f_i(\Delta_i^{1,2,\ldots,p_i}(t))}{f_{\pi_i^1}(\Delta_i^{1,2,\ldots,p_i}(t))}$$

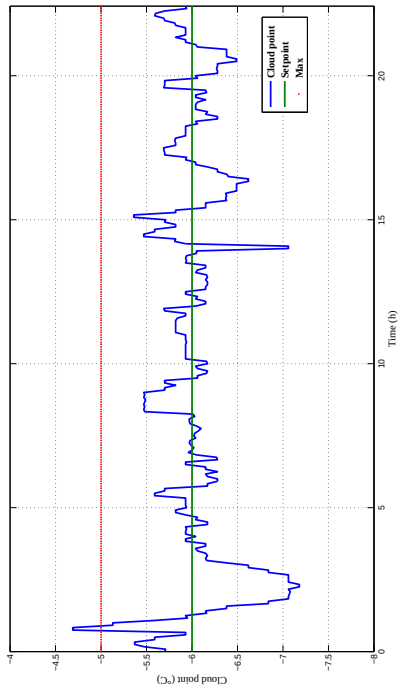
We have $U_i(t) = u_i(t) = f_i(t)/F(t)$ for $\Pi_i = \emptyset$ and

$$y(t) = \sum_{i=1}^{n} U_i(t) b_i.$$

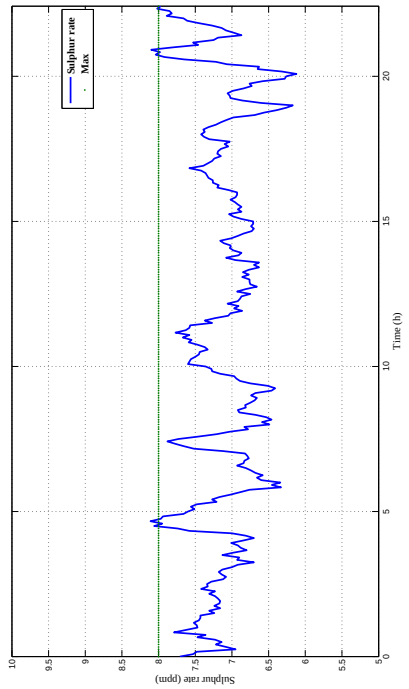


(a)

(b)

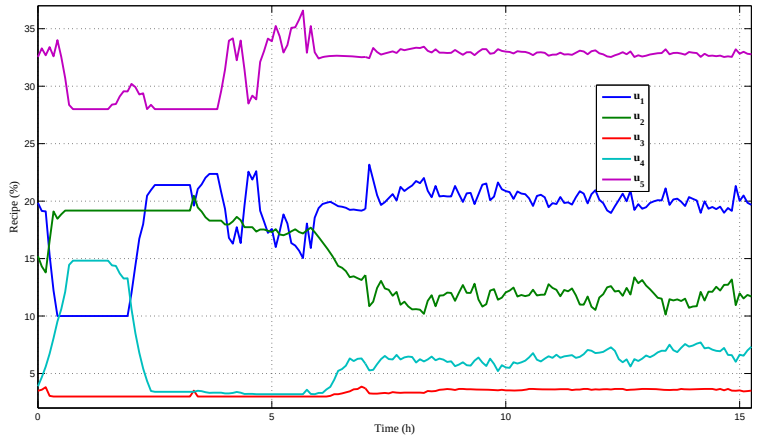


(c)

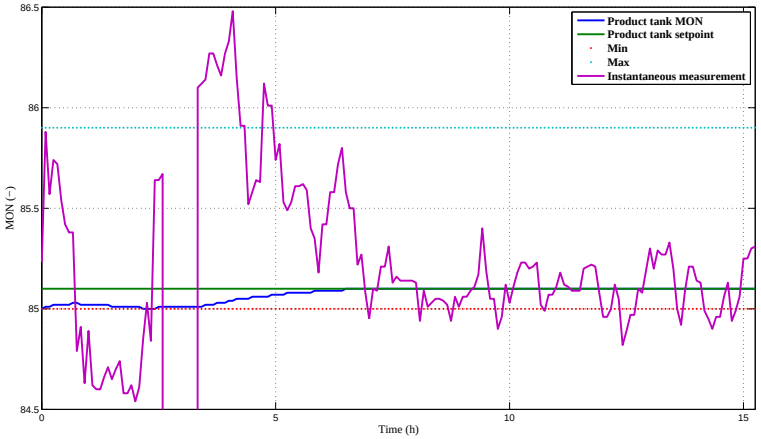


(d)

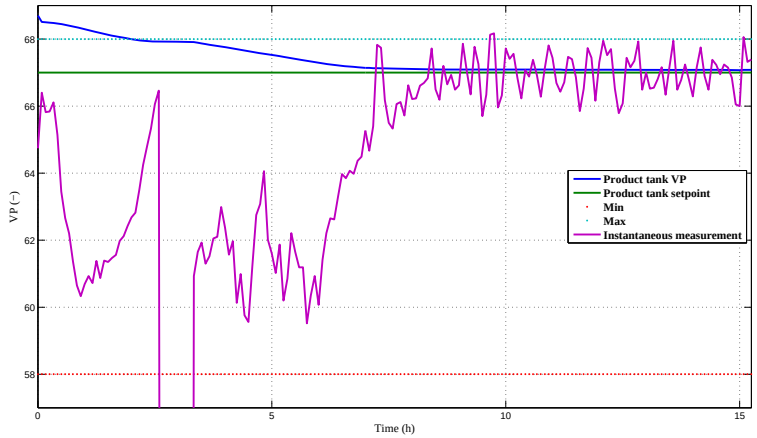
Fig. 6. Diesel fuel blend industrial results. A total of 3 components (a) are controlled to produce a blend in a continuous mode. Three properties are of interest. These are the flash point (b) and the cloud point (c) which must track their setpoints, and the sulphur content (d) which must remain below a specified maximum. Thanks to the proposed control method, these goals are reached. At the beginning of the trends, a strong control action (a) can be observed which is computed to bring the flash point above its minimum.



(a)



(b)



(c)

Fig. 7. Gasoline blend industrial results. A total of 5 components (a) are controlled to produce a blend in a storage tank. Two properties are of interest (MON (b) and VP (c)). Initially, the storage tank is more than half-filled with an off-specifications product. This situation is handled by the heel-management strategy. The total blend time is 15 hours. Convergence of both the MON and the VP to their setpoints is achieved early. Despite large noises, the proposed algorithm is quite robust and computes control values (a) that remain within their bounds. At the end of the blend, the storage tank is filled with the desired amount of product with the desired properties, while the recipe is close to the target.