
APC: A Status Report (The Patient Is Still Breathing!)

White Paper

Dr. James R. Ford, P.E.



Introduction	3
Early Developments	4
The Need for Process Control.....	4
Unmeasured Disturbances.....	5
Automatic Control Valves	6
Measured Disturbances	8
The Emergence of MPC	10
Where We Are Today	15

Introduction

Advanced process control (APC) is a fairly mature body of engineering technology. Its evolution closely mirrors that of the digital computer and its close cousin, the modern microprocessor. APC was born in the '60s, evolved slowly and somewhat painfully through its adolescence in the '70s, flourished in the '80s (with the remarkable advances in computers and DCSs), and reached maturity in the '90s, when model predictive control (MPC) ascended to the throne of supremacy as the preferred approach for implementing APC solutions.

As Zak Friedman¹ dared to point out in a recent article, the current decade has witnessed tremendous APC industry discontent, self-examination and retrenchment. He lists several reasons for the malaise, among them: “cutting corners” on the implementation phase of the projects, poor inferred property models (these are explained later), tying APC projects to “optimization” projects, and too-cozy relationships between APC software vendors / implementers and their customers. In a more recent article² in the same magazine, Friedman interviewed me because I offer a different explanation for the APC industry problems.

The purpose of this paper is to trace the history of the development of process control, advanced process control and related applied engineering technologies, and to discuss the reasons why I think the industry has encountered difficulties. I will present some recommendations to improve the likelihood of successful APC project implementation, and make some predictions about the future direction of the technology.

Early Developments

The discovery of oil in Pennsylvania in 1859 was followed immediately by the development of processes for separating and recovering the main distillable products, primarily kerosene, heating oil and lubricants. These processes were initially batch in nature. A pot of oil was heated to boiling, and the resulting vapor was condensed and recovered in smaller batches. The first batch was extremely light (virgin naphtha), and the last batch was heavy (fuel oil or lubricating oil). Eventually, this process was transformed from batch to continuous, providing a means of continuously feeding fresh oil and recovering all distillate products simultaneously. The heart of this process was a unit operation referred to as countercurrent, multi-component, two-phase fractionation.

Whereas the batch process was manual in nature and required very few adjustments (other than varying the heat applied to the pot), the continuous process required a means of making adjustments to several important variables, such as the feed rate, the feed temperature, the reflux rate, etc., in order to maintain stable operation and to keep products within specifications. Manually operated valves were initially utilized to allow an operator to adjust the important independent variables. A relatively simple process could be operated in a fairly stable fashion with this early “process control” system.

Over the next generation of process technology development, process control advanced from purely manual, open-loop control to automatic, closed-loop control. To truly understand this evolution, we should examine the reasons why this evolution was necessary and how those reasons impact the application of modern process control technology to the operation of process units today.

The Need for Process Control

Why do we need process control at all? The single most important reason is to respond to process disturbances. If process disturbances did not occur, the manual valves mentioned above would suffice for satisfactory, stable operation of process plants. What, then, do we mean by a process disturbance? A process disturbance is a change in any variable that affects the flow of heat and/or material in the process. We can further categorize disturbances in two ways: by time horizon and by measurability.

Some disturbances occur slowly over a period of weeks, months or years. Examples of this type of disturbance are:

- Heat exchanger fouling — slowly alters the rate of heat transfer from one fluid to another in the process.
- Catalyst deactivation — slowly affects the rate and selectivity, etc., of the reactions occurring in the reactor.

Automatic process control was NOT developed to address long time-horizon disturbances. Manual adjustment for these types of disturbances would work almost as well. So, automatic process control is used to rectify disturbances that occur over a much shorter time period of seconds, minutes or hours. Within this short time horizon, there are really two main types of disturbances: measured and unmeasured.

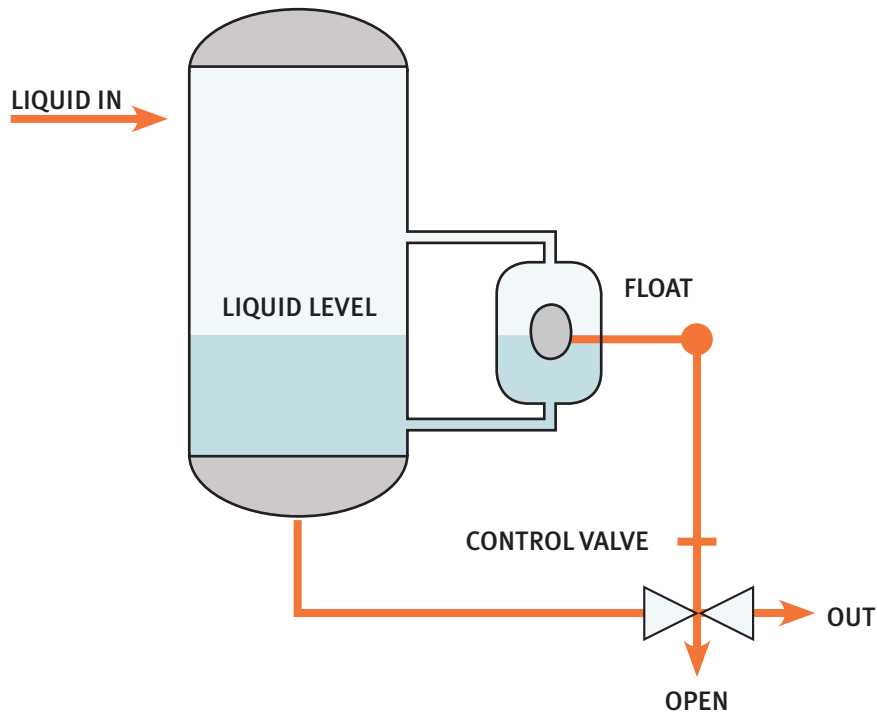
Unmeasured Disturbances

Automatic process control was initially developed to respond to unmeasured disturbances. For example, consider the first automatic devices used to control the level of a liquid in a vessel. Refer to Figure 1. The liquid level in the vessel is sensed by a float. The float is attached to a lever. A change in liquid level moves the float up or down, which mechanically or pneumatically moves the lever, which is connected to a valve. When the level goes up, the valve opens and vice versa. The control loop is responding to an unmeasured disturbance, namely, the flow rate of material into the vessel.

The first automatic controllers were not very sophisticated. The float span, lever length, connection to the valve and control valve opening had to be designed to handle the full range of operation. Otherwise, the vessel could overflow or drain out completely. This type of control had no specific target or “set point”. At constant inlet flow, the level in the vessel would reach whatever resting position resulted in the proper valve opening to make the outflow equal to the inflow.

Level controllers were probably the first type of automatic process controller developed because of the mechanical simplicity of the entire loop. Later on, it became obvious that more sophisticated control valves were needed to further automate other types of loops. The pneumatically driven, linear-position control valve evolved over the early 20th century in all its various combinations of valve body and plug design in order to handle just about any type of fluid condition, pressure drop, etc. The development of the automatic control valve ushered in the era of modern process control.

FIGURE 1
FLOAT-ACTUATED LEVEL CONTROL



Automatic Control Valves

The first truly automatic control valves were developed to replace manual valves in order to control flow. This is the easiest type of variable to control, for two reasons. First, there is essentially no dead time and very little measurement lag between a change in valve opening and a change in the flow measurement. Second, a flow control loop is not typically subjected to a great deal of disturbance. The only significant disturbance is a change in upstream or downstream pressure, such as might occur in a fuel gas header supplying fuel gas through a flow controller for firing a heater or boiler. Other less-significant disturbances include changes in the temperature and density of the flowing fluid. The flow control loop has become the foundation of all automatic process control, for several reasons.

Unlike pressure and temperature, which are intensive variables, flow is an extensive variable. Intensive variables are key control variables for stable operation of process plants because they relate directly to composition. Intensive variables are usually controlled by adjusting flows, the extensive variables. In this sense, intensive variables are higher in the control hierarchy. This explains why a simple cascade almost always involves an intensive variable as the master, or primary, in the cascade and flow as the slave, or secondary, in the cascade. When a pressure or temperature controller adjusts a control valve directly (rather than the flow in a cascade), the controller is actually adjusting the flow of material through the valve.

What this means, practically, is that, unlike the intensive variables, a flow controller has no predetermined or “best” target for any given desired plant operation. The flow will be wherever it needs to be in order to maintain the higher level intensive variable at its “best” value. This explains why “optimum” unit operation does not require accurate flow measurement. Even with significant error in flow measurement, the target for the measured flow will be adjusted (in open or closed loop) to maintain the intensive variable at its desired target. This also explains why orifices are perfectly acceptable as flow controller measurement devices, even though they are known to be rather inaccurate.

The above comments apply to almost all flow controllers, even important ones like the main unit charge rate controller. The target for this control will be adjusted to achieve an overall production rate, to push a constraint, to control the inventory of feed (in a feed drum), etc. What about additional feed flows, like the flow of solvent in an absorption or extraction process? In this case, there is a more important, higher level control variable, an intensive variable, namely, the ratio of the solvent to the unit charge rate. The flow of solvent will be adjusted to maintain a “best” solvent / feed ratio. Again, measurement accuracy is not critical; the ratio target will be adjusted to achieve the desired higher level objective (absorption efficiency, etc.), regardless of how much measurement inaccuracy is present.

Almost all basic control loops, either single-loop or simple cascades, are designed to react, on feedback, to unmeasured disturbances. Reacting to unmeasured disturbances is called “servo,” or feedback control. Feedback control is based on reacting to a change in the process variable (the PV) in relation to the loop target, or set point (SP). The PV can change in relation to the SP for two reasons: either because a disturbance has resulted in an unexpected change in the PV, or because the operator or a higher level control has changed the SP. Let’s ignore SP changes for now.

Types of Feedback Control

So, feedback control was initially designed to react to unmeasured disturbances. The problem in designing these early feedback controllers was figuring out how much and how fast to adjust the valve when the PV changed. The first design was the float-type level controller described earlier. The control action is referred to as “proportional” because the valve opening is linearly proportional to the level. The higher the level, the more open the valve.

This type of control may have been marginally acceptable for basic levels in vessels, but it was entirely deficient for other variables. The main problem is that proportional-only control cannot control to a specific target or set point. There will always be offset between the PV and the desired target. To correct this deficiency, the type of control known as “integral,” or “reset,” was developed. This terminology is based on the fact that, mathematically, the control action to correct the offset between SP and PV is based on the calculus operation known as integration. (The control action is based on the area under the SP-PV offset curve, integrated over a period of time). The addition of this type of control action represented a major improvement in feedback control. Almost all flow and pressure loops can be controlled very well with a combination of proportional and integral control action.

A less important type of control action was developed to handle situations where the loop includes significant measurement lag, such as is often seen in temperature loops involving a thermocouple, inside a thermowell, which is stuck in the side of a vessel or pipe. Control engineers noted that these loops were particularly difficult to control because the measurement lag introduced instability whenever the loops were tuned to minimize SP-PV error. For these situations, a control action was developed that reacts to a change in the “rate of change” of the PV. In other words, as the PV begins to change its “trajectory” with regard to the SP, the control action is “reversed,” or “puts on the brakes,” to head off the change that is coming, as indicated by the change in trajectory. The control action was based on comparing the rate of change of the PV over time, or the derivative of the PV. Hence the name of the control action: derivative. In practice, this type of control action is utilized very little in the tuning of basic control loops.

The other type of change that produces an offset between SP and PV is an SP change. For flow control loops, which are typically adjusted to maintain a higher level intensive variable at its target, a quick response to the SP change is desirable; otherwise, additional response lag is introduced. A flow control loop can and should be tuned to react quickly and equally effectively to both PV disturbances and SP changes.

Unfortunately, for intensive variables, a different closed loop response for PV changes due to disturbances vs. SP changes is called for. For temperature, and especially pressure, these loops are tuned as tightly as possible to react to disturbances. This is because intensive variables are directly related to composition, and good control of composition is essential for product quality and yield. However, if the operator makes an SP change, a much less “aggressive” control action is preferred. This is because the resulting composition change will induce disturbances in other parts of the process, and the goal is to propagate this disturbance as smoothly as possible, so as to allow other loops to react without significant upset.

Modern digital control systems (DCSs) can provide some help in this area. For example, the control loop can be configured to take proportional action on PV changes only, ignoring the effect of SP changes. Then, following an SP change, integral action will grind away on correcting the offset between SP and PV. However, these features do not fully correct the deficiency discussed above. This dilemma plagues control systems to this very day, and is a major justification for implementation of advanced controls that directly address this and other shortcomings of basic process control.

Measured Disturbances

The other major type of disturbance is measured. Common examples are changes in charge rate, cooling water temperature, steam header pressure, fuel gas header pressure, heating medium temperature and ambient air temperature, etc., where instruments are installed to measure those variables. The first 20 years of the development of APC technology focused primarily on using measured disturbance information for improving the quality of control. Why? Modern process units are complex and highly interactive. The basic control system, even a modern DCS, is incapable of maintaining fully stable operation when disturbances occur. APC was developed to mitigate the destabilizing effects of disturbances, and thereby to reduce process instability. This is still the primary goal of APC. Any other claimed objective or direct benefit is secondary.

Why is it so important to reduce process instability? Process instability leads to extremely conservative operation so as to avoid the costly penalties associated with instability, namely, production of off-spec product and violation of important constraints related to equipment life and human safety. Conservative operation means staying well away from constraint limits. Staying away from these limits leaves a lot of money on the table in terms of reduced yields, lower throughput and greater energy consumption. APC reduces instability, allowing for operation much closer to constraints, thereby capturing the benefits that would otherwise be lost.

As stated above, early APC development work focused on improving the control system's response to measured disturbances. The main techniques were called "feedforward," "compensating" and "decoupling". In the example of a fired heater mentioned earlier, adjusting the fuel flow for changes in heater charge rate and inlet temperature is feedforward. The objective is to head off upsets in the heater outlet temperature that are going to occur because of these feed changes. In similar fashion, the fuel flow can be "compensated" for changes in fuel gas header pressure, temperature, density and heating value, if these measurements are available. Finally, if this is a dual-fuel heater (fuel gas and fuel oil), the fuel gas flow can be adjusted when the fuel oil flow changes, so as to "decouple" the heater from the firing upset that would otherwise occur. This decoupling is often implemented as a heater fired duty controller.

A second area of initial APC development effort focused on controlling process variables that are not directly measured by an instrument. An example is reactor conversion or severity. Hydrocracking severity is often measured by how much of the fresh feed is converted to heating oil and lighter products. If the appropriate product flow measurements are available, the conversion can be calculated and the reactor severity can then be adjusted to maintain a target conversion. Work in this area of APC led to the development of a related body of engineering technology referred to as "inferred properties" or "soft sensors." More on this later.

A third area of APC development work focused on pushing constraints. After all, if the goal of APC is to reduce process instability so as to operate closer to constraints, then why not implement APCs that accomplish that goal? (Note that this type of APC strategy creates a measured process

disturbance — we are going to move a major independent variable to push constraints, so there had better be APCs in place to handle those disturbances.) Especially when the goal was to increase the average unit charge rate by pushing known, measured constraints, huge benefits could often be claimed for these types of strategies. In practice, these types of strategies were difficult to implement and were not particularly successful. More on this later.

While all this development work was focused on reacting to changes in measured disturbances, the problems created by unmeasured disturbances continued to hamper stable unit operation (and still do today). Some early effort also focused on improving the only tool available at the time, the proportional-integral-derivative (PID) control algorithm, to react better to unmeasured disturbances.

One of the main weaknesses of PID is its inability to maintain stable operation when there is significant dead time and / or lag between the valve movement and the effect on the control variable. For example, in a distillation column, the reflux flow is often adjusted to maintain a stable column tray temperature. The problem arises when the tray is well down the tower. When an unmeasured feed composition change occurs, upsetting the tray temperature, the controller responds by adjusting the reflux flow. But there may be dead time of several minutes before the change in reflux flow begins to change the tray temperature. In the meantime, the controller will have continued to take more and more integral action in an effort to return the PV to SP. These types of loops are difficult (or impossible) to tune. They are typically detuned (small gain and integral), but with a good bit of derivative action left in as a means of “putting on the brakes” when the PV starts returning toward SP.

Some successes were noted with algorithms such as the Smith Predictor, which relies on a model to predict the response of the PV to changes in controller output. This algorithm attempts to control the predicted PV (the PV with both dead time and disturbances included) rather than the actual measured PV. Unfortunately, even the slightest model mismatch can cause the controller using the Smith Predictor to become unstable.

We have been particularly successful in this area with development of our “smart” PID control algorithm. In its simplest form, it addresses the biggest weakness of PID, namely, the overshoot that occurs because the algorithm continues to take integral action to reduce the offset between SP and PV, even when the PV is returning to SP. Our algorithm turns the integral action on and off according to a proven decision process made at each controller execution. This algorithm excels in loops with significant dead time and lag. We use this algorithm on virtually all APC projects.

The Need for Models

By the mid-’80s, many consulting companies and in-house technical staffs were involved in the design and implementation of the types of APC strategies described in the last few paragraphs. A word that began to appear more and more associated with APC was “model.” For example, to implement the constraint-pushing APC strategies discussed above, a “dynamic” model was needed to relate a change in the independent variable (the charge rate) to the effect on each of the dependent, or constraint, variables. With this model, the adjustments that were needed to keep the most constraining of the constraint variables close to their limits could be determined mathematically.

Why was it necessary to resort to development of models? As mentioned earlier, many of the early constraint-pushing efforts were not particularly successful. Why not? It’s the same problem

that plagues feedback control loops with significant dead time and lag. There are usually constraints that should be honored in a constraint-pushing strategy that may be far removed (in time) from where the constraint-pushing move is made. Traditional feedback techniques (PID controllers acting through a signal selector) do not work well for the constraints with long dead time. We addressed this by developing special versions of our smart PID algorithm to deal with the long dead times, and were fairly successful in doing so.

The Emergence of MPC

In his Ph.D. dissertation work, Dr. Charles Cutler developed a technique that incorporated normalized step-response models for the constraint (or control) variables, or CVs, as a function of the manipulated variables, or MVs. This allowed the control problem to be “linearized,” which then permitted the application of standard matrix algebra to estimate the MV moves to be made to keep the CVs within their limits. He called the matrix of model coefficients the “dynamic matrix” and developed the “dynamic matrix control” (DMC) control technique. He also incorporated an objective function into the DMC algorithm, turning it into an “optimizer.” If the objective function is the sum of the variances between the predicted and desired values of the CVs, then DMC becomes a minimum variance controller that minimizes the output error over the controller time horizon.

Thus was ushered in the control technology known in general as multi-variable, model-predictive control (MVC or MPC). Dr. Cutler’s work led eventually to formation of his company, DMC Corporation, which was eventually acquired by AspenTech. The current version of this control software is known as DMCPlus™. There are many competing MPC products, including Honeywell RMPCT™, Invensys Connoisseur™ and others.

MPC has become the preferred technology for solving not only multivariable control problems, but just about any control problem more complicated than simple cascades and ratios. Note that this technology no longer relies on traditional servo control techniques, which were first designed to handle the effect of unmeasured disturbances and which have done a fairly good job of doing so for about 100 years. MPC assumes that our knowledge of the process is perfect and that all disturbances have been accounted for. There is no way for an MPC to handle unmeasured disturbances, other than to readjust at each controller execution the bias between the predicted and measured value of each control variable. This can be likened to a form of integral-only control action. This partially explains MPC’s poor behavior when challenged by disturbances unaccounted for in the controller.

While DMCPlus uses linear, step-response models, other MPC developers have incorporated other types of models. For example, Pavilion Technologies has developed a whole body of modeling and control software based on neural networks. Since these models are nonlinear, they allow the user to develop nonlinear models for processes that display this behavior. Polymer production processes (e.g., polypropylene) are highly nonlinear, and neural net-based controllers are said to perform well for control of these processes.

GE (MVC®) uses algebraic models and solves the control execution prediction problem with numerical techniques.

MPC vs. ARC

There are some similarities between the older APC techniques (feedforward, etc.) and MPC, but some important differences. Let's call the older technique advanced regulatory control (ARC). To illustrate, let's take a simplified control problem, such as a distillation column where we are controlling a tray temperature by adjusting the reflux flow, and we want feedforward action for feed rate changes. The MPC will have two models, one for the response of the temperature to feed rate changes and one for the response of the temperature to reflux flow changes. For a feed rate change, the controller knows that the temperature is going to change over time, so it estimates a series of changes in reflux flow required to keep the temperature near its desired target.

The action of the ARC is different. In this case, we want to feedforward the feed rate change directly to the reflux flow. We do so by delaying and lagging the feed rate change (using a simple dead time and lag algorithm customized to adjust the reflux with the appropriate dynamics), then adjusting the reflux with the appropriate steady-state gain or sensitivity (e.g., three barrels of reflux per barrel of feed). The ultimate sensitivity of the change in reflux flow to a change in feed rate varies from day to day; hence, this type of feedforward control is adaptive and, therefore, superior to MPC (the MPC models are static). Note: Dr. Cutler recently formed a new corporation, and he is now offering adaptive DMC, which includes real-time adjustment of the response models.

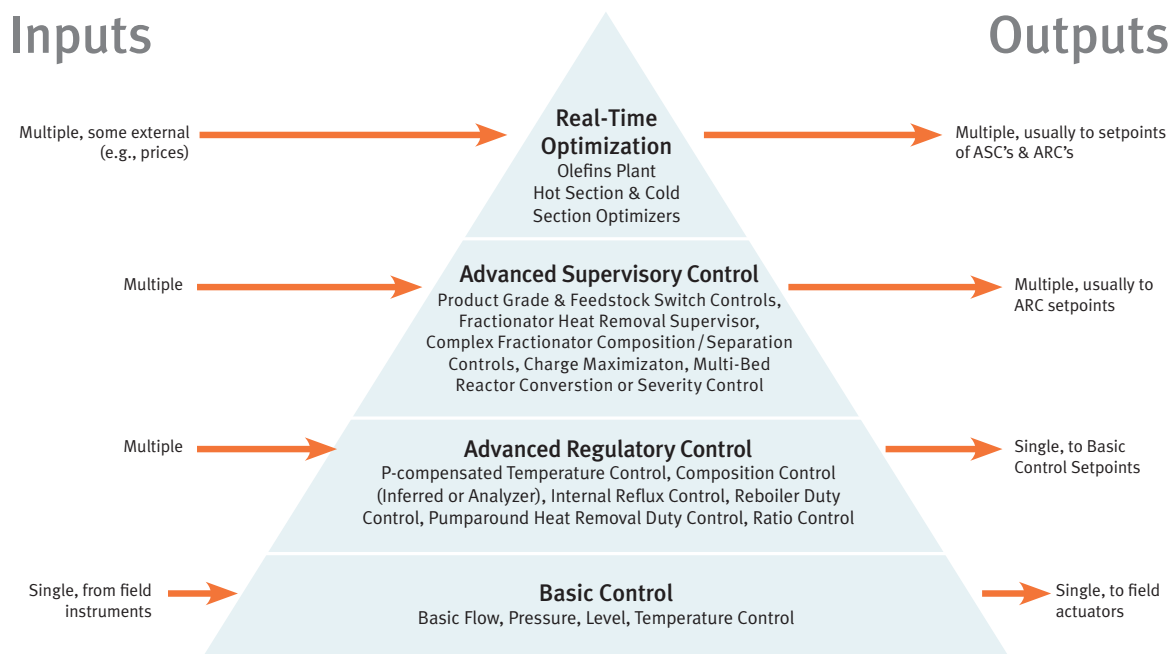
How does the MPC handle an unmeasured disturbance, such as a feed composition change? As mentioned earlier, it can only do so when it notices that the temperature is not where it's supposed to be according to the model prediction from the series of recent moves of the feed rate and reflux. It resets the predicted vs. the actual bias and then calculates a reflux flow move that will get the temperature back where it's supposed to be, a form of integral-only feedback control. On the other hand, the ARC acts in the traditional feedback (or servo) manner with either plain or "smart" PID action.

Hierarchy

Prior to MPC, most successful APC engineers used a process engineering-based, hierarchical approach to developing APC solutions. Such an approach is best described in Figure 2. The bottom of the control hierarchy, its foundation, is what we referred to earlier as "basic" process control, the single loops and simple cascades that appear on P&IDs and provide the operator with the first level of regulatory control.

Simple processes that are not subject to significant disturbances can operate in a fairly stable fashion with basic process control alone. Unfortunately, most process units in refineries and chemical plants are very complex, highly interactive and subject to frequent disturbances. The basic control system is incapable of maintaining fully stable operation when challenged by these disturbances — thus, the emergence of APC to mitigate the destabilizing effects of disturbances. The hierarchical approach to APC design identifies the causes of the disturbances in each part of the process, then layers the solutions that deal with the disturbances on top of the basic control system, from the bottom up. Each layer adds complexity and its design depends upon the disturbance(s) being dealt with.

As an example of the first layer of the APC hierarchy (ARC in Figure 2), consider the classic problem of how to control the composition of distillation tower product streams, such as the overhead product stream. The hierarchical approach is based upon identifying and dealing with

FIGURE 2

the disturbances. The first type of disturbance that typically occurs is caused by changes in ambient conditions (air temperature, rainstorms, etc.), which lead to a change in the temperature of the condensed overhead vapor stream, namely the reflux (and overhead product). This will cause the condensation rate in the tower to change, leading to a disturbance that will upset column separation and product qualities. Hierarchical APC design deals with this disturbance by specifying, as the first level above basic control, internal reflux control (IRC).

The IRC first calculates the net internal reflux with an equation that includes the heat of vaporization, heat capacity and flow of the external reflux, the overhead vapor temperature and the reflux temperature. The control next back-calculates the external reflux flow required to maintain constant IR, and then adjusts the setpoint of the external reflux flow controller accordingly. This type of control provides a fast-responding, first-level improvement in stability by isolating the column from disturbances caused by changes in ambient conditions. There are multiple inputs to the control (the flow and temperatures which contribute to calculation of the internal reflux), but typically only one output – to the setpoint of the reflux flow controller.

Moving up the hierarchy to the advanced supervisory control level (ASC in Figure 2), the overhead product composition can be further stabilized by controlling a key temperature in the upper part of the tower, since temperature (at constant pressure) is directly related to composition. And, since the tower pressure could vary (especially if another application is attempting to minimize pressure), then the temperature that is being controlled should be corrected for pressure variations. This control adjusts the setpoint of the IRC to maintain a constant pressure-corrected temperature (PCT). Feedforward action (for feed rate changes) and decoupling action (for changes in reboiler heat) can be added at this level.

Further improvement in composition control can be achieved by developing a correlation for the product composition, using real-time process measurements (the PCT plus other variables such as reflux ratio, etc.), then using this correlation as the PV in an additional APC that adjusts the setpoint of the PCT. This type of correlation is known as an inferred property or soft sensor.

Thus, the hierarchical approach results in a multiple-cascade design, an inferred property control adjusting a PCT, adjusting an IRC, adjusting the reflux flow.

Figure 2 is somewhat of an over-simplification, but it gets the point across. In general, APCs designed using a hierarchical approach consist of layers of increasingly complex strategies. Some of the important advantages of this approach are:

- Operators can understand the strategies – they appeal to human logic because they use a “systems” approach to problem-solving, breaking a big problem down into smaller problems to be solved.
- The control structure is more suitable for solutions at a lower level in the control system –such solutions can often be implemented without the requirement for additional hardware and software.
- The controls “degrade gracefully” – when a problem prohibits a higher level APC from being used, the lower level controls can still be used and can capture much of the associated benefit.

How would we solve the above control problem using MPC? There are two approaches. The standard approach is to use the inferred property as a CV and the external reflux flow controller as the MV. In this case, then, how does the MPC deal with the other disturbances such as the reflux temperature, reboiler heat and feed rate? These variables must be included in the model matrix as additional independent variables. A step-response model must then be developed for the CV (the inferred property) as a function of each of the additional independent variables. This is the way most of the APC industry designs an MPC.

The hierarchical approach would suggest something radically different. The CV is the same, because the product composition is the variable that directly relates to profitability. However, in the hierarchical design, the MV is the PCT. The lower level controls (the PCTC and IRC) are implemented at a lower level in the control hierarchy, typically in the DCS.

Unfortunately, the industry-accepted approach to MPC design violates the principles of hierarchy. Rarely, if ever, are intermediate levels of APC employed below MPC. There is no hierarchy – just one huge, flat MPC controller on top of the basic controllers, moving all of them at the same time in seemingly magical fashion. Operators rarely understand them or what they are doing. And, they do not degrade gracefully. Consequently, many fall into disuse.

Other Problems with MPC

A non-hierarchical design is only one of the many problems with MPC. The limitations of MPC have been thoroughly exposed, though probably widely ignored. (The emperor is surely wearing clothes! See the excellent article by Alan Hugo, “Limitations of Model-Predictive Controllers,” that appeared in the January 2000 issue of *Hydrocarbon Processing*, PAGE 83.) Here are a few other problems.

In many real control situations, even mild model mismatch introduces seriously inappropriate controller moves, leading to inherent instability as the controller tries at each execution to deal with the error between where it thinks it is going and where it really is.

MPC is not well suited for processes with non-continuous phases of operation, such as delayed cokers. The problem here is how to “model” the transitory behavior of key control variables during coke drum pre-warming and switching. Unfortunately, every drum-switching operation is

different. This means that both the “time to steady state” and ultimate CV gain are at best only approximations, leading to model mismatch during every single drum operation. No wonder they are so difficult to “tune” during these transitions.

Correcting model mismatch requires retesting and remodeling, a form of very expensive maintenance. MPC controllers, particularly complex ones implemented on complex units with lots of interactions, require a lot of “babysitting” – constant attention from highly trained (and highly paid) control engineers. This is a luxury that few operating companies can afford.

The licensors of MPC software will tell you that their algorithms “optimize” operation by operating at constraints using the least costly combination of manipulated variable assets. That is certainly correct, mathematically – that is the way the LP or QP works. In practice, however, any actual “optimizing” is marginal at best. This is due to a couple of reasons.

The first is the fact that, in most cases, one MV dominates the relationships of other MV’s to a particular CV. For example, in a crude oil distillation tower, the sensitivity between the composition of a side-draw distillate product and its draw rate will be much larger than the sensitivity with any other MV. The controller will almost always move the distillate draw rate to control composition. Only if the draw rate becomes constrained will the controller adjust a pumparound flow or the draw rate of another distillate product to control composition, regardless of the relative “cost” of these MVs.

The second reason is the fact that, in most cases, once the constraint “corners” of the CV / MV space are found, they tend not to change. The location of the corners is most often determined either by “discretionary” operator-entered limits (for example, the operator wants to limit the operating range of an MV) or by valve positions. In both situations, these are constraints that would have been pushed by any kind of APC, not just a model-predictive controller. So, the MPC has not “optimized” any more than would a simpler ARC or ASC.

When analyzing the MPC controller models that result from plant testing, control engineers often encounter CV / MV relationships that appear inappropriate and which are usually dropped because the engineer does not want that particular MV moved to control that particular CV. Thus, the decoupling benefit that could have been achieved with simpler ASC is lost.

If every single combination of variables and constraint limits has not been tested during controller commissioning (as is often the case), then the controller behavior under these untested conditions is unknown and unpredictable. For even moderately sized controllers, the number of possible combinations becomes unreasonably large such that all combinations cannot be tested, even in simulation mode.

Control engineers often drop models from the model matrix in order to ensure a reasonably stable solution (avoiding approach to a singular matrix in the LP solution). This is most common where controllers are implemented on fractionation columns with high-purity products, and where the controller is expected to meet product purity specifications on both top and bottom products. The model matrix then reduces to one that is almost completely decoupled. In this case, single-loop controllers, rather than an MPC, would be a clearly superior solution.

What about some of the other MPC “selling points?” One that is often mentioned is that, unlike traditional APC, MPC eliminates the need for custom programming. This is just not true. Except for very simple MPC solutions, custom code is almost always required – for example, to calculate a variable used by the controller, to estimate a flow from a valve position, etc.

The benefits of “standardization” are often touted. Using the same solution tool across the whole organization for all control problems will reduce training and maintenance costs. But this is like saying that I can use my new convertible to haul dirt even though I know that using my used pick-up would be much more appropriate. Such an approach ignores the nature of real control problems in refineries and chemical plants, and relegates control engineers to pointers and clickers.

Where We Are Today

Perhaps the previous discussion paints an overly pessimistic picture of MPC as it exists today. Certainly many companies, particularly large refining companies, have recognized the potential return of MPC, have invested heavily in MPC, and maintain large, highly trained staffs to ensure that the MPCs function properly and provide the performance that justified their investment. But, on the other hand, the management of many other companies have been seduced by the popular myth that MPC is easy to implement and maintain – a misconception fostered at the highest management levels by those most likely to benefit from the proliferation of various MPC software packages.

So, what can companies do today to improve the utilization and effectiveness of APC in general and MPC in particular? They can do several things, all focused primarily on the way we analyze operating problems and design APC solutions to solve the problems and thereby improve productivity and profitability. Here are some ideas:

As mentioned earlier, the main goal of APC is to isolate operating units from process disturbances. What does this mean when designing an APC solution? First, identify the disturbance and determine its breadth of influence. Does the disturbance affect the whole unit? If so, how? For example, the most common unit-encompassing disturbance is a feed rate change. But how is the disturbance propagated? In many process units, this disturbance merely propagates in one direction only (upstream to downstream) with no other complicating effects (such as those caused by recycle or interaction between upstream and downstream unit operations). In this case, a fairly straightforward APC solution involves relatively simple ARCs for inventory and load control – adjusting inter-unit flows with feedforward for inventory control, and adjusting load-related variables such as distillation column reflux with ratio controls or with feedforward action in the case of cascades (for example, controlling a PCT by adjusting an IRC). No MVC is required here to achieve significantly improved unit stability.

If the effect of the disturbance can be isolated, then design the APC for that isolated part of the unit and ignore potential second-order effects on other parts of the unit. A good example is the IR control discussed earlier. The tower can be easily isolated from the ambient conditions disturbance by implementing internal reflux control. The internal reflux can then be utilized as an MV in a higher level control strategy (for example, to control a pressure-compensated tray temperature as an indicator of product composition) or an inferred property.

What about unmeasured disturbances, like feed composition, where the control strategy must react on feedback alone? As mentioned earlier, we have had a great deal of success with “intelligent” feedback-control algorithms that are much more effective than simple PID. For example, our “Smart PID” algorithm includes special logic that first determines, at each control algorithm execution, whether or not integral action is currently advised based upon the PV’s recent trajectory toward or away from setpoint. Secondly, it determines the magnitude of the integral move based on similar logic. This greatly improves the transient controller response, especially in loops with significant dead time and / or lag.

Another successful approach is to use model-based, adaptive control algorithms, such as those incorporated in products like Brainwave™.

Here are some additional suggestions:

- Using a hierarchical approach suggests implementing the lower level APC strategies as low in the control system as possible. Most modern DCSs will support a good bit of ARC and ASC at the DCS level. Some DCSs such as Honeywell Experion (and earlier TDC systems) have dedicated, DCS-level devices designed specifically for implementation of robust APC applications. We are currently implementing very effective first level APC in such diverse DCSs as Foxboro, Yokogawa, Delta-V and Honeywell.
- When available, and assuming the analyses are reliable, use lab data as much as possible in the design of APCs. Develop inferred property correlations for control of key product properties and update the model correlations with the lab data.
- Track APC performance by calculating, historizing and displaying a variable which indicates “quality of control.” We often use the variance (standard deviation), in engineering units, of the error between SP and PV of the key APC (for example, an inferred property). Degradation in the long-term trend of this variable suggests that a careful process analysis is needed to identify the cause of the degraded performance and the corrective action needed to return the control to its original performance level.

Recommendations for Using MPC

If MPC is being considered for solution of a control problem, apply it intelligently and hierarchically.

Intelligent application of MPC involves asking important questions, such as:

- Is the control problem to be solved truly multi-variable? Can (and should) several MVs be adjusted to control one CV? If not, don't use MPC.
- Are significant dead time and lag present, such that the model-based, predictive power of MPC would provide superior control performance when compared to simple feedforward action?
- Look at the model matrix. Are there significant interactions between MANY dependent variables and MANY independent variables, or are there just isolated “islands” of interactions where each island represents a fairly simple control problem that could just as easily be solved with simpler technology?
- How big does the controller really have to be? Could the problem be effectively solved with a number of small controllers without sacrificing the benefits provided by a unit-wide controller?

Applying MPC hierarchically means:

- Handle isolated disturbance variables with lower level ARC. For example, stabilize and control a fired heater outlet temperature with ARC, not MPC. The ARC can easily handle disturbance variables such as feed rate, inlet temperature, fuel gas pressure, etc.
- Use the lower level ARCs, such as the fired heater outlet temperature mentioned above, as MVs in the MPC controller.
- Use intensive variables, when possible, as MVs in the controller. For example, use a distillate product YIELD, rather than flow rate, as the MV in a controller designed to control the quality of that product – this eliminates unit charge rate as a disturbance variable.

What's in Store for the Next 40 Years?

A cover story that appeared in *Hydrocarbon Processing* about 20 years ago included an interview with the top automation and APC guru of a major US refining company. The main point being proffered by the interviewee was that the relentless advances in automation technology, both hardware- and software-wise (including APC), would lead one day to the “virtual” control room in which operating processes would no longer require constant human monitoring – no process operators required. I'm not sure how this article was received elsewhere, but it got a lot of chuckles in our office.

Process control, whether continuous or discrete, involves events that are related by cause and effect. For example, a change in temperature (the cause event) needs a change in a valve position (the effect event) to maintain control of the temperature. Automation elevates the relationships of the cause and effect events by eliminating the requirement for a human to connect the two. With the huge advances in automation technology accomplished in the period of the '70s and '80s, one might imagine a situation in which incredibly sophisticated automation solutions would eventually eliminate the need for direct human involvement.

That's not what this author sees for the next 40 years. Why not? Three reasons. First, the more sophisticated automation solutions require process models. For example, MPC requires accurate MV / CV or DV / CV response models. Real-time rigorous optimization (RETRO) requires accurate steady state process models. Despite the complexity of the underlying mathematics and the depth of process data analysis involved in development of these models, the fact is that these will always remain just “models.” When an optimizer runs an optimization solution for a process unit, guess what results? The optimizer spits out an optimal solution for the model, not the process unit! Anybody who believes that the optimizer is really optimizing the process unit is either quite naïve, or has been seduced by the same type of hype that accompanied the advent of MPC and now seems to pervade the world of RETRO.

Don't misinterpret my comments. When well-designed and -engineered (and -monitored), RETRO can move operations in a more profitable direction and thereby improve profitability in a major way. But, the point is that models will never be accurate enough to eliminate the need for human monitoring and intervention.

The second important reason involves disturbances. Remember that there are two types of process disturbances, measured and unmeasured. Despite huge advances in process measurement technology (analyzers, etc.), we will never reach the point (at least in the next 40 years) when ALL disturbances can be measured. Therefore, there will always be target offset (deviation from setpoint) due to unmeasured disturbances and model mismatch. Furthermore, it is an absolute certainty that some unmeasured disturbances will be of such severity so as to lead to process “upsets,” conditions that require human analysis and intervention in order to prevent shutdowns, unsafe operation, equipment damage, etc. Operators may get bored in the future, but they'll still be around for the next 40 years to handle these upset conditions.

The third important reason is the “weakest link” argument. Control system hardware (primarily field elements) is notoriously prone to failure. No automation system can continue to provide full functionality, in real time, when hardware fails – human intervention will be required to maintain stable plant operation until the problem is remedied.

Indeed, the virtual control room is still more than 40 years away. Process automation will continue to improve with advances in measurement technology, equipment reliability and

modeling sophistication, but process operators, and capable process and process-control engineers, will still maintain and oversee its application. APC will remain an “imperfect” technology that requires for its successful application experienced and quality process engineering analysis.

The most successful APC programs of the next 40 years will have the following characteristics:

- The APC solution will be designed to solve the operating problem being dealt with and will utilize the appropriate control technology – MPC will be employed only when required.
- The total APC solution will utilize the principle of “hierarchy” – lower level, simpler solutions will solve lower level problems, with ascending, higher level, more complex solutions achieving higher level control and optimization objectives.
- An intermediate level of model-based APC technology, somewhere between simple PID and MPC (Brainwave™, as an example), will receive greater utilization at a lower cost (in terms of both initial cost and maintenance cost), but will provide an effective means of dealing with measured and unmeasured disturbances.
- The whole sub-field of APC known as “transition management” (e.g., handling crude switches) will become much more dominant and important as an adjunct to continuous process control.
- APC solutions will make greater use of inferred properties, and these will be based on the physics and chemistry of the process (not “artificial intelligence,” like neural nets).
- The role of operating company, in-house control engineers will evolve – from a purely technical position to more of a program manager. Operating companies will rely more on APC companies to do the “grunt work,” allowing the in-house engineers to manage the projects and to monitor the performance of the control system.

APC technology has endured and survived its growing pains, it has reached a relatively stable state of maturity, and it has shaken out its weaker components and proponents. Those of us who have survived this process are wiser, humbler and more realistic about how to move forward. Our customers and employers will be best served by a pragmatic, process engineering-based approach to APC design that applies the specific control technology most appropriate to achieve the desired control and operating objectives at lowest cost and with the greatest probability of long-term durability and maintainability.

MAVERICK Technologies, LLC

265 Admiral Trost Road | P.O. Box 470 | Columbia, IL 62236 USA

+1.618.281.9100 | Fax +1.618.281.9191

www.mavtechglobal.com



engineering | systems integration | consulting