Knowledge Based Adaptive Blood Pressure Control: A Simplexys Expert System Application

by

J.O. Lammers

EUT Report 90-236
ISBN 90-6144-236-2

June 1990
KNOWLEDGE BASED ADAPTIVE BLOOD PRESSURE CONTROL: A Simplexys expert system application

by

J.O. Lammers

EUT Report 90-E-236
ISBN 90-6144-236-2

Eindhoven
June 1990
Final report of the post-graduate course "Information and Communication Engineering" of the Institute for Continuing Education (IVO) of the Eindhoven University of Technology, followed in the period May 1988 till May 1990.

Supervisors: Prof. dr. ir. J.E.W. Benekens and Dr. ir. J.A. Blom,
Division of Medical Electrical Engineering,
Faculty of Electrical Engineering,
Eindhoven University of Technology,
P.O. Box 513,
5600 MB Eindhoven,
The Netherlands

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG
Lammers, J.O.
Knowledge based adaptive blood pressure control: a Simplexys expert system application / by J.O. Lammers. - Eindhoven: Eindhoven University of Technology, Faculty of Electrical Engineering. - Fig., tab. - (EUT report, ISSN 0167-9708; 90-E-236)
ISBN 90-6144-236-2
SISO 608.1 UDC 616-089.5 NUGI 742
Trefw.: anesthesie; regelsystemen / patiëntbewaking; expertsystemen.
Abstract

During and following heart surgery patients often have their blood pressure artificially lowered by an infusion of a continuous flow of the drug Sodium Nitroprusside (SNP). In order to regulate the blood pressure of the patient, the anaesthetist frequently adjusts the flow rate of the pump; each adjustment results in a change of the pressure some minutes later. Automatic control can also be applied; a controller can change the flow rate continuously, so that the pressure reaches, and remains at, a desired level.

Control is difficult because the sensitivity to the drug is unknown and varies between patients over a wide range. Moreover during the case the pressure changes due to many other causes than SNP. The anaesthetist, however, is able to regulate the pressure safely and therefore control is based on an expert system in which specific medical and control engineering knowledge is incorporated. The controller must be properly tuned, and its performance is guarded by the expert system. When the controller does not perform as desired or when the patient does not react as expected, the expert system changes or overrides the controller.

Lammers, J.O.
KNOWLEDGE BASED ADAPTIVE BLOOD PRESSURE CONTROL: A Simplexys expert system application.
Faculty of Electrical Engineering, Eindhoven University of Technology, The Netherlands, 1990.
EUT Report 90-E-236
I should like to thank the Department of Anesthesiology of the Eindhoven Catharina Hospital, and in particular Dr. Korsten for the collaboration in clinical evaluation.

This project would not have been possible without the Simplexs toolbox. Hans Blom, thanks for your critical thoughts and assistance.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Summary</strong></td>
<td>8</td>
</tr>
<tr>
<td><strong>Glossary</strong></td>
<td>11</td>
</tr>
<tr>
<td><strong>Introduction</strong></td>
<td>12</td>
</tr>
<tr>
<td><strong>1 Global description of the knowledge base</strong></td>
<td>14</td>
</tr>
<tr>
<td>1.1 Patient's response to the applied drug</td>
<td>14</td>
</tr>
<tr>
<td>1.2 Requirements for an automatic blood pressure controller</td>
<td>17</td>
</tr>
<tr>
<td>1.3 A Simplexys expert system for blood pressure control</td>
<td>18</td>
</tr>
<tr>
<td>1.4 The functional units of the complete system</td>
<td>19</td>
</tr>
<tr>
<td>1.5 Knowledge about gain adaptation</td>
<td>21</td>
</tr>
<tr>
<td>1.6 Knowledge about detection of fast pressure changes</td>
<td>22</td>
</tr>
<tr>
<td>1.7 Setup of clinical testing</td>
<td>23</td>
</tr>
<tr>
<td><strong>2 Control strategy</strong></td>
<td>24</td>
</tr>
<tr>
<td>2.1 PID control for regulation and stabilization</td>
<td>24</td>
</tr>
<tr>
<td>2.2 Adaptation of the controller to the sensitivity</td>
<td>28</td>
</tr>
<tr>
<td>2.3 Regions for gain adaptation</td>
<td>30</td>
</tr>
<tr>
<td>2.4 Final remarks about PID control</td>
<td>31</td>
</tr>
<tr>
<td><strong>3 Gain up adaptation</strong></td>
<td>32</td>
</tr>
<tr>
<td>3.1 The need for gain up adaptation</td>
<td>32</td>
</tr>
<tr>
<td>3.2 Conditions for gain up adaptation</td>
<td>32</td>
</tr>
<tr>
<td>3.3 Resulting actions</td>
<td>34</td>
</tr>
<tr>
<td>3.4 Final remarks about gain up adaptation</td>
<td>34</td>
</tr>
</tbody>
</table>
4 Gain down adaptation 1: large setpoint change up
   4.1 The need for gain down adaptation
   4.2 Conditions for gain down adaptation due to setpoint change
   4.3 Final remarks about large setpoint changes

5 Gain down adaptation 2: large change of flow rate
   5.1 High control gain results in high flow rate change
   5.2 A relative measure of change of control
   5.3 Conditions for gain down adaptation due to large change of flow
   5.4 Final remarks about flow rate changes

6 Gain down adaptation 3: pressure changes fast
   6.1 Detection of fast pressure change
   6.2 Implementation of the mechanism
   6.3 Final remarks about regions for gain adaptation

7 Gain down adaptation 4: oscillation detection
   7.1 Detection of oscillation
   7.2 Counting the number of border crossings
   7.3 Final remarks about oscillation detection

8 Actions in case of large pressure fluctuations
   8.1 The need for transient detection
   8.2 Some common techniques for transient detection
   8.3 Expected pressure change due to a change of the flow rate
   8.4 Small pressure changes
   8.5 Detection and treatment of up transients
   8.6 Detection and treatment of down transients
   8.7 Consequences of incorrect detection of transients
   8.8 Impact on the adaptation mechanism
   8.9 Final remarks about transients

9 Ineffective control
   9.1 Causes of ineffective control
   9.2 Detection of ineffective control
   9.3 Gain change request versus gain change adaptation
   9.4 Final remarks about ineffective control
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 Invalid measurements</td>
<td>60</td>
</tr>
<tr>
<td>10.1 Validation of the arterial pressure</td>
<td>60</td>
</tr>
<tr>
<td>10.2 Actions taken in case of invalid measurements</td>
<td>60</td>
</tr>
<tr>
<td>10.3 Final remarks about validation</td>
<td>61</td>
</tr>
<tr>
<td>11 The human interface</td>
<td>62</td>
</tr>
<tr>
<td>11.1 Contents of the display</td>
<td>62</td>
</tr>
<tr>
<td>11.2 Manual mode of operation</td>
<td>63</td>
</tr>
<tr>
<td>11.3 Automatic mode of operation</td>
<td>64</td>
</tr>
<tr>
<td>12 Noise reduction</td>
<td>66</td>
</tr>
<tr>
<td>12.1 Moving average filters</td>
<td>66</td>
</tr>
<tr>
<td>12.2 Filters used in the knowledge base</td>
<td>67</td>
</tr>
<tr>
<td>13 Infusion pump failure</td>
<td>68</td>
</tr>
<tr>
<td>14 Open loop control</td>
<td>69</td>
</tr>
<tr>
<td>Conclusions</td>
<td>70</td>
</tr>
<tr>
<td>References</td>
<td>73</td>
</tr>
</tbody>
</table>
To lower the blood pressure of patients undergoing heart surgery, frequently the drug Sodium Nitroprusside (SNP) is infused. An automatic controller changes the flow rate continuously in order to maintain a certain target pressure.

Models are common in control engineering. A first order model of the patient's reaction to the applied drug is used to design the controller. The model parameters represent the sensitivity and the dynamic response. The higher the flow rate, the lower the pressure, but this relation is neither linear nor instantaneous. The sensitivity symbolizes the final pressure decrease due to a certain change of flow, and can vary between patients over a wide range. Since classical automatic control cannot cope with this large sensitivity range, adaptive control is applied: the controller adapts to the sensitivity of the patient. The dynamic model parameters vary less than the sensitivity, and the controller is designed so that it can cope with the complete range of the dynamic characteristics. A robust controller that is adapted by an expert system will guarantee safety for all patients.

The characteristics of the individual patients make control difficult, and so does the environment. Blood pressure measurement is often unreliable or not available due to taking blood samples, calibrating pulses, irregular heart function and so on. In those situations the measurement does not represent the real pressure, thus control based on that measurement must be avoided. In order to validate the measurements, in each heart cycle the shape of the pressure curve is analyzed. If the shape of the curve is acceptable, the measurement is used for control, otherwise the controller continues without a new measurement.

While the controller must of course keep the pressure close to the target, the main requirement is safe control. Therefore besides control engineering knowledge about adaptation, the expert system contains medical knowledge that has to do with the patient's safety. The advantage of the expert system approach is that the knowledge base is self documenting and can be checked for completeness, correctness and consistency before the final program is built.

To adapt the controller to the sensitivity of the patient, a measure of progress of the pressure toward the target level is necessary. If the controller is adapted correctly, the progress is within certain limits. If the progress is significantly faster or significantly slower, the controller is not adapted correctly and needs to be changed by the expert system.
To obtain a measure of progress, several regions are defined around the target; the controller will attempt to bring the pressure to the middle region (target level). The actual pressure will take some time to move through each region, until it reaches the target. If the pressure stays in a region for too long, progress is too slow and the control gain is enlarged. The control gain is decreased if the pressure moves through a region too quickly.

For safety reasons the flow rate change per unit of time is limited. When the controller is adapted correctly, it will not change the flow so quickly that it needs to be limited. Therefore if the flow is limited repeatedly, the expert systems needs to adapt the controller.

The pressure of the patient will often show spontaneous variations, for instance caused by the supply of other drugs, by pain reactions or by manipulation of the heart by the surgeon. Slow pressure fluctuations are compensated by the controller, fast fluctuations are not. A transient is a temporary large increase or decrease of the pressure; a transient is faster than a physiological response to the applied drug can be. Transients are temporary; when such a large pressure change occurs, the pressure is expected to return to the original level some minutes later. The controller must not compensate for transients since the pressure changes that occur are too fast and too large to compensate for. Any attempt to compensate for transients will cause a large change of flow, without results. Moreover the controller should no try to suppress transients since a fast changing pressure is a diagnostic aid for the anaesthetist.

Therefore when a quickly rising pressure occurs, the flow is kept constant until the pressure is back at the normal level. On the other hand when the pressure drops quickly, there might be danger for the patient, hence the flow is shut off immediately. When the transient is over, the flow is resumed at the level it had just before the transient, and normal control continues.

Earlier prototypes of the system have been tested with simulations and animal experiments. The prototype that is described in this document was tested during heart surgery. When the system was introduced for clinical testing, first it was used with manual adjustment of the flow. Control was not activated, but the expert system's adaptation and safety mechanisms were active. The system's decisions were analyzed afterwards, and the analysis resulted in refinements of the implemented knowledge about transients.

After 35 cases with manual control, another 30 cases were performed with automatic control. It turned out that the system was safe; especially actions taken on transients were correct. Automatic control was slightly better then manual control; because one of the adaptation mechanisms turned out to be inappropriate, the controller was sometimes slower than expected.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNP</td>
<td>Sodium Nitroprusside</td>
</tr>
<tr>
<td>MAP</td>
<td>Mean Arterial Pressure</td>
</tr>
<tr>
<td>flow</td>
<td>rate of SNP that is infused</td>
</tr>
<tr>
<td>sensitivity</td>
<td>final MAP decrease per unit of flow</td>
</tr>
<tr>
<td>setpoint</td>
<td>desired (target) pressure level</td>
</tr>
<tr>
<td>PID-control</td>
<td>a type of control that is widely used in practice</td>
</tr>
<tr>
<td>adaptation</td>
<td>modification of the controller</td>
</tr>
<tr>
<td>gain adaptation</td>
<td>adjustment of the control gain to the sensitivity of the patient</td>
</tr>
<tr>
<td>model</td>
<td>mathematical description of the relevant characteristics of the patient</td>
</tr>
<tr>
<td>simulation</td>
<td>analysis of the performance of the system using a number of patient models</td>
</tr>
<tr>
<td>playback</td>
<td>analysis of the controller's performance using the stored data of a previous case</td>
</tr>
</tbody>
</table>
The main task of an anaesthetist is to bring the patient into an optimal condition for surgery. This includes tasks like making the patient unconscious, suppression of pain and muscle reactions, as well as regulation of the pressure. Many drugs used in anaesthesia also lower the pressure, but during and following heart surgery patients frequently have their pressure additionally lowered by an infusion of a continuous flow of the drug Sodium Nitroprusside (SNP). Adjustment of the rate of the infusion pump changes the flow of SNP that is infused, normally resulting in a pressure change some minutes later. The patient's response is rather fast, so that regulation and stabilization of the pressure is possible by varying the SNP flow rate.

Automatic control can also be applied; a controller can change the flow rate continuously, so that the pressure reaches, and remains at, the target level. There are some problems in regulating the pressure with SNP. Especially the sensitivity of the patient to the drug is unknown; its variation is assumed to be over a range from 1/9 of normal to 9 times normal.

Classical automatic control cannot cope with this sensitivity range: for a very sensitive patient the controller will oscillate, for a very insensitive one the target will never be reached. Therefore control must be adaptive. Principally not all methods of adaptation can be applied: there are severe restrictions on the control signal. Also system identification is not possible because during the case pressure changes are due to many other causes than just SNP.

Although some automatic control methods fail, the anaesthetist is able to regulate the pressure safely. Thus it seems that much specific medical knowledge must be incorporated into a successful controller. In an expert system, that medical knowledge is separately specified in a knowledge base, which is easy to understand, to change and to update.

An accurately tuned PID-controller is used for pressure control. Its performance is monitored by the expert system. When the controller does not perform as desired or when the patient does not react as expected, the expert system overrules or adapts the controller.

The new knowledge based pressure controller is implemented with the Simplexys toolbox; Simplexys supports the design of real time expert systems. It was designed by the Medical Electrical Engineering division of Eindhoven University of Technology [Blom, 1990].
The controller and the expert system were initially designed and evaluated through a variety of simulations. The system was introduced and clinically evaluated during heart surgery by Zwart [1990], who has attended the system's performance and noted the remarks and opinions of the anesthetists during 30 cases. It turned out that control was safe and slightly better than manual control, but sometimes slower than expected.

The next chapter illustrates the context of this research and describes the total system briefly. More detailed information about the control strategy is given in chapter 2. Adaptation of the controller to the sensitivity of an individual patient is discussed in chapters 3 - 7. The safety mechanisms and some other related topics are described in the remaining chapters.

The main results are included at the end of each chapter. Recommendations to update the knowledge base were studied through a playback of the cases, using the stored data but with new versions of the knowledge base. These recommendations are not yet implemented.
This chapter is an introduction to the knowledge base and the context of this research. The control problem is formulated in sections 1.1 and 1.2. The expert system methodology through which the problem is solved is motivated in section 1.3. Section 1.4 describes the functional units of the system, and sections 1.5 and 1.6 introduce the knowledge base, one of these units. Finally section 1.7 reports on the method of clinical testing. The remainder of this document describes the knowledge based pressure controller in more detail.

1.1 Patient's response to the applied drug

Figure 1.1 shows the response of the pressure to a unit dose of SNP.

![Figure 1.1](image)

*Figure 1.1 Impulse response*

Models are commonly used in control engineering; the patient's reaction to the applied drug Sodium Nitroprusside (SNP) can also be modelled. The model structure is a simplification of reality; this is allowed because a simplified model describes the system accurately enough for the design of a controller.
A first order system with time delay is used as the model:

\[ Y(s) = \frac{-sT - K}{s\tau + 1} U(s) \]  \[ Y(s) = f(y(t) - y(0)) \]  \[ U(s) = f(u(t)) \]

The symbol \( f \) denotes the Laplace operator, \( y \) the pressure and \( u \) the infusion flow rate.

The three parameters of the model portray the step response (figure 1.2). The \textit{delay time} \( T \) is the time before the pressure shows any change after the flow is changed; the \textit{sensitivity} \( K \) is the final change of pressure when it has stabilized again. The \textit{time constant} \( \tau \) is the time it takes, after the delay, for the pressure to move 63% toward its final value.

![Step response of the model](image)

\textit{Figure 1.2}  
\textit{Step response of the model}

The parameter \( K \) describes the \textit{static} behavior: what is the final pressure change after a unit flow change. The parameters \( T \) and \( \tau \) describe the \textit{dynamic} behavior: \( T \) shows after which time the pressure shows a change for the first time, \( \tau \) the time after which 63% of the complete effect of a flow change is realized.

All parameters of the model are unknown, vary between patients and change (slowly) in time. The sensitivity \( K \) is assumed to vary from \(1/9\) to \(9\) times normal; the normal \( K \) is assumed to be \(5\) [mmHg/(mg/h)] or \(25\) [mmHg/(\(\mu\)g/kg/min)] for a 80 kg human. The sensitivity will change in time, and will decrease at higher flow rates due to a non linearity of the response [Blom, 1990].
During the introduction of the system in clinical testing, manual control was employed first (section 1.7). In twelve cases it was possible to estimate the model parameters. During these cases, the flow rate was several times adjusted by a large amount, so that the step response could be analyzed graphically and the model parameters could be derived (table 1.1). Patients that are less sensitive do not show a distinct step response, so that dynamic characteristics can only be derived from sufficiently sensitive patients.

Due to a variety of effects that influence the pressure, in reality the step response is not as smooth as the one shown in figure 1.2. It is therefore difficult to derive the dynamic model parameters accurately. Table 1.1 shows the estimated model parameters $T$ and $\tau$ that were derived from 62 setpoint changes that occurred during these 12 cases. The $T$ table contains only 54 entries because some of the 62 step responses were useless for estimation of the delay time.

<table>
<thead>
<tr>
<th>Parameter estimate of one stepwise flow adjustment</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
</tr>
<tr>
<td>0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100</td>
</tr>
<tr>
<td>$\tau$</td>
</tr>
<tr>
<td>0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100</td>
</tr>
</tbody>
</table>

Table 1.1
Estimate of the dynamic model parameters

Table 1.1 shows that the time constant $\tau$ is mostly between 40 and 90 seconds; 60 seconds is chosen as the nominal time constant. The delay time $T$ is mostly between 35 and 60 seconds; 50 seconds is chosen as the nominal time delay. These results agree with Slate [1980].

Table 1.2 shows the estimated model parameters for the 12 patients, which are derived by averaging the parameter estimates of the step responses for each patient. Table 1.2 shows that in none of the cases both dynamic model parameters are far from nominal simultaneously. Stated differently, the large and the small time delays of table 1.1 are found for different patients then the large and small time constants.

<table>
<thead>
<tr>
<th>Parameter estimates of 12 patients</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
</tr>
<tr>
<td>&lt;30 30 40 50 60 70 &gt;70</td>
</tr>
<tr>
<td>&lt;20 20 40 50 60 70 &gt;70</td>
</tr>
<tr>
<td>&lt;10 20 30 40 50 60 70 &gt;70</td>
</tr>
</tbody>
</table>

Table 1.2
1.2 Requirements for an automatic blood pressure controller

The variability of the characteristics of individual patients makes control difficult, and so does the environment. The pressure measurement is sometimes unreliable or not available. Therefore for each heart cycle the shape of the pressure curve is analyzed. If the shape of the curve is acceptable, that period is used to computing the Mean Arterial Pressure (MAP). Each 5 seconds a new MAP is computed; when there is no acceptable period in that 5 seconds, the controller must continue without a new MAP measurement.

Even through averaging takes place over a 5 second period, the MAP still shows random fluctuations with an amplitude of about 2 mmHg. This is not too much of a problem, because the MAP is allowed to vary in a band of 5 mmHg around the target. The administration of SNP is sometimes disturbed as well. When the infusion line is manipulated or when other drugs are delivered through the same infusion line, the SNP flow is temporary disturbed.

The MAP is not only influenced by SNP, but also by many other influences. Some of these are temporary, others are permanent; for example the application of other drugs, manipulation of the heart by the surgeon or the reaction to pain stimuli will influence the pressure.

Under these circumstances, an automatic controller must supply the correct SNP flow to keep the MAP at the desired level. Moreover control must be safe for all patients, under all circumstances. This means that the controller never supplies a flow that is too high, never forces the MAP lower than the target level, and never allows the MAP to be too high for a long period. To achieve this, the controller adapts to the estimated sensitivity of the patient, and some additional safety mechanisms are included.

Besides safety, requirements for the control performance are:

1. When a new setpoint (desired MAP level) is chosen, within 10 minutes the flow must have been adjusted in such a way that the setpoint is reached; during that period the MAP may not overshoot the setpoint by more than 10 mmHg.

2. In steady state conditions the controller must keep the MAP within a 5 mmHg band around the setpoint.

3. The controller must only attempt to stabilize the MAP if it is capable to do so.

4. The infusion flow rate is limited to a certain maximum, and should change smoothly.
1.3 A Simplexys expert system for blood pressure control

Adaptive control of some kind is necessary, but due to random fluctuations (noise) and the requirements for the control signal, not all adaptive methods can be applied. Moreover, in order to be able to handle special cases that have to do with the safety of the patient, medical knowledge must be incorporated.

Programs that contain a mix of medical and control engineering knowledge tend to be difficult to understand, to change and to maintain. In standard software, the knowledge is mostly spread out all through the program, and it is difficult to find out exactly what knowledge is incorporated, and where it can be found. In an expert system the knowledge is separately described in a knowledge base in a more or less self-documenting way. Moreover the Simplexys toolbox contains programs that can check the knowledge base for completeness, correctness and consistency before the final program is built.

The Medical Electrical Engineering division has studied the use of SNP for pressure control for several years. A first prototype of an automatic controller was clinically tested on Yorkshire pigs [Blom and de Bruyn, 1982]. More recently pressure measurements of human patients were analyzed, and an algorithm was built that analyzes and validates the pressure measurements [Melissen, 1989].

Last year a new prototype was developed and tested with simulations. Moreover the human interface, such as the graphics output and keyboard input were developed [Hoogendoorn, 1989]. Through all these earlier studies the (medical) knowledge of SNP blood pressure control was collected; knowledge acquisition, which often requires a large period of time when building an expert system, has been done during these years. The third prototype, that is described in this document, was recently tested in the Eindhoven Catharina Hospital during heart surgery [Zwart, 1990].
1.4 The functional units of the complete system

An accurately tuned PID controller, that is adapted by the expert system, is used for pressure control. The dynamic response (the time before a change of dose shows results) as well as the static response (the final pressure change) can vary intra- and inter-individually. The PID controller copes with the varying dynamic patient characteristics and partly with varying sensitivity. The expert system performs gain adaptation to cope with the remainder of the sensitivity range.

The controller is rather robust: if only one of the three model parameters deviates from nominal while the other two are nominal, the controller remains safe until the deviation is more than a factor of 3 larger or smaller; if all three parameters deviate from nominal simultaneously, it is safe until all parameters deviate by a factor of two. The worst case for the controller is the combination of a large delay and a large sensitivity (mismatch relative to the control gain). Control robustness is discussed further in section 2.1.

For the dynamic characteristics such large deviations are not expected. The expected range of the patient sensitivity is much larger, thus ordinary PID control is not suitable for all patients. Therefore the expert system will monitor the patient's response continuously, and adapt the controller if necessary. Due to gain adaptation the sensitivity of the patient should deviate at most a factor of 2 from the control gain. The combination of a robust PID controller that is monitored by an expert system, is expected to make the complete controller fast enough and safe for all patients.

Note that the controlled system responds quite slowly compared with the dynamical response of the patient. The reaction to a change of flow occurs within 3 minutes (figure 1.1), while the controller must bring the MAP to the target in 10 minutes (figure 2.1). Because of this relatively slow control, robustness can be enlarged so that control is also safe for patients whose characteristics are far from nominal.

Each 5 seconds the controller computes an increment or decrement of the flow rate. The increment or decrement is determined by the parameters of the controller (that are fixed) and the overall control gain (that is adapted). When the control gain is increased, the increment or decrement will be by a factor of 3, but the control signal itself changes smoothly (see figure 3.1)
Figure 1.3 shows the system's configuration schematically. Besides knowledge about gain adaptation, the knowledge base contains additional safety mechanisms. The Simplexys inference engine uses the knowledge base to draw conclusions and to take decisions. Keyboard input facilities are to choose the operating mode (manual or automatic), adjust the SNP flow in manual mode, and change the setpoint in automatic mode. Besides the current pressure and flow rate in numbers, the screen contains a window for system messages and trend displays that shows the pressure, the setpoint and the SNP flow as well as their history over the last 30 minutes graphically (figure 11.1).

The controller uses the Mean Arterial Pressure (MAP) that is the average pressure over the last 5 seconds. As noted in section 1.2, the pressure measurement is validated before it is used for control purposes. When the pressure measurement is flagged as disturbed, it usually does not represent the real pressure; thus control based on that measurement should be avoided. When no valid MAP can be computed, the expert system has to take appropriate actions.
1.5 Knowledge about gain adaptation

Requirements for the PID controller are: a smooth change of the flow after a setpoint change, a smooth change of the MAP to that target, and finally safe behavior for patients whose characteristics are far from nominal.

Initially the controller assumes that the patient is very sensitive; in many cases the patient will be less sensitive, and thus the control gain is multiplied by a factor of 3. For gain adaptation a measure of progress of the MAP toward the setpoint is necessary. If the control gain is correct, the progress is within certain limits; if the control gain is too low, the progress is significantly slower, and if the gain is too high the progress is too fast.

To obtain a measure of progress, several regions are defined around the target level (setpoint). After a setpoint change the MAP settles in one of these regions; the controller must then bring the MAP to the middle (target) region. The width of the different regions is chosen in such a way that when the control gain is correct, the MAP stays in each region for about 2 minutes. Thus when the MAP stays in a region for longer than 4 minutes, the progress is considered too slow, and the control gain enlarged. On the other hand, when the MAP moves through a region within one minute, the control gain is reduced with one step (see table 2.2).

The progress is determined by the time the MAP stays in a region: this measure is a slope (MAP change per unit of time). Direct computation of the slope in a noisy environment does not result in a reliable measure for gain adaptation. In order to reduce the noise influence, the slope should be computed over a certain time or over a certain MAP change.

In the first case the averaged slope is computed after a certain period. This is a reliable method for the detection of a slowly changing MAP, but special care needs to be taken to prevent a fast changing MAP from reaching dangerous values while the slope computation is pending.

In the second case, the average slope is computed when the MAP has changed by a certain amount. This is an accurate method when the MAP is changing quickly, but it will take a long time before a measure of slope is available when the MAP is changing slowly.

Regions implement a mixture of these two methods of slope computation. If the MAP changes slowly, after some minutes it will turn out that the MAP is still in a region and the gain is increased; if the MAP changes quickly, after a certain change of the pressure a region border is crossed, and the gain is decreased.
From simulation studies it turned out that decreasing the gain when the MAP moves quickly through a region, is sometimes too late. It takes about 2 minutes before the MAP shows any progress after a setpoint change and then another minute before it has travelled through a region. Then the control gain is decreased, and it takes another 2 minutes before that shows results. The MAP has steadily moved toward the target during those 5 minutes, and may already have overshot it. Therefore a faster gain down adaptation mechanisms is added that acts on relatively large changes of the flow rate.

For safety reasons the relative increment of the flow rate is limited to plus or minus 7% each 5 seconds. This roughly means that the control signal can grow two times larger in about one minute. If the controller is adapted correctly, it will not change the flow rate that quickly. The gain down mechanism is activated by large relative flow changes. Thus when the flow change is repetitiously larger than 7%, the control gain is too high. The most effective method to limit the change of the flow rate is to decrease the control gain.

If the control gain is too high, under certain circumstances oscillation can result; if oscillation is detected, the control gain is decreased. This mechanism works quite slowly: it takes about 10 minutes before oscillation is detected. Moreover this mechanism by definition acts too late: oscillation should be prevented rather then detected. The mechanism is added for safety reasons, however.

1.6 Knowledge about detection of fast pressure changes

A transient is a temporary, large, fast increase or decrease of the pressure; a transient is significantly faster than the progress to a new setpoint can be. Transients are temporary; when such a large MAP change occurs, the MAP is expected to return to the original level after some minutes by itself. When a transient is caused by pain, drugs are administered and as a result the pressure returns to the original level. Also manipulation of the heart by the surgeon can cause a temporarily large change of the pressure.

A transient needs to be detected for several reasons. First the controller cannot compensate for a transient: the pressure change that occurs is too large and too fast to compensate for. Any attempt to compensate for a transient will cause a large change of the flow rate, without the desired results. When the transient is over, the controller must recover and bring the flow back to the pre-transient level. It is better to detect a transient, and keep the flow at a constant rate until the transient is over.
Second, it is undesirable to try to suppress a transient: a fast changing pressure is a diagnostic aid for the anaesthetist. Compensating with SNP for a fast rising pressure that is caused by pain should be avoided. Finally gain adaptation must be disabled during a transient, because the pressure change is not caused by the controller.

If a fast rising pressure is detected, the SNP flow is kept constant until the MAP is back at the previous level. A down transient is treated in a different way. Because a low pressure is dangerous for the patient, the flow is shut off immediately. When the transient is over, the flow is resumed but kept constant at the level it had just before the transient. The MAP is expected to show an overshoot as a reaction to the previous zero flow period. When the overshoot is over, or if it does not occur, normal control is resumed.

1.7 Setup of clinical testing

The system was introduced into the operating rooms in three stages. During the first stage the system was used in manual mode only. The SNP flow rate was adjusted with the keyboard of our system, rather than with switches on the pump. The aim of this stage was to get the anaesthetists used to the system, to analyze the reliability of the hardware under operating room circumstances, and to analyze the behavior of the MAP validation.

In the second stage the system was also used with manual control, but in addition the anaesthetists were asked to define a target level. In that way the controller could compute its control signal, that later could be compared to the SNP flow which the anaesthetist had given. During this stage the anaesthetists got familiar with the concept of targets in addition to flow rates.

More than thirty cases were involved in these two stages, in which the model parameters that are shown in table 1.1 and 1.2 were derived from the patients' step responses to the manual flow rate changes.

In the third stage automatic control was applied. A detailed analysis of the 30 cases of that stage is given in [Zwart, 1990], but the most important conclusions are also reported in this document, since they form the basis for a new prototype system.
The controller is an adaptive PID controller. PID controllers are widely used because they are easy to describe, easy to implement and easy to tune. Moreover an accurately tuned PID controller is stable for a wide range of system characteristics; it is robust against rather large variations in the dynamic patient characteristics and smaller variations in the sensitivity (section 2.1). However even a robust PID controller cannot cope with the complete sensitivity range that is expected for patients; the sections 2.2 and 2.3 describe the method through which the controller is adapted to the patient’s sensitivity.

2.1 PID control for regulation and stabilization

The first control task is the administration of a certain infusion flow rate of Sodium Nitroprusside (SNP), in order to affect the Mean Arterial Pressure (MAP) in such a way that it eventually reaches the desired target pressure level or setpoint. Adjustment of the SNP flow rate in order to bring the MAP to the setpoint is called regulation.

The second control task is stabilization of the MAP at the target level. Due to several causes, for instance the administration of other drugs, the MAP will drift away from the target level. Stabilization performs continuous adjustment of the flow rate so that the MAP stays as close as possible to the target level.

These two control tasks are generally conflicting: a controller that is tuned primarily for regulation is likely to be inaccurate for stabilization. Therefore for pressure control, two PID controllers are designed: one that is tuned for regulation and a slightly different one that is tuned for stabilization. Since it is generally not clear whether the most suitable control method is regulating or stabilizing, as a compromise the actual controller is formed by a weighted average of the two control methods. The distance between the pressure and the target level determines whether control is mainly stabilizing (at small distances) or regulating (at large distances).

Three parameters govern the behavior of a PID controller; these parameters are chosen in such a way that the controlled system has the desired characteristics. For the design of the PID controller for pressure control, the first order model described in section 1.1 is used.
The controller does not perform optimally with respect to the desired behavior, when one or more system parameters differ from the assumed (nominal) value. However a robust PID controller does not suffer from small deviations; the larger the permitted deviations are before control becomes unsafe or unstable, the larger control robustness.

From section 1.1, the model parameters are recalled. When the SNP flow rate is increased by one unit, the pressure responds as shown in figure 1.2. The delay time is the time during which the MAP does not yet react to the flow rate increase (about \( \frac{1}{2} \) to \( 1\frac{1}{2} \) minute). After the delay, the MAP will slowly decrease, and will finally reach a new stable value. The rise time is defined as the time it takes, after the delay, to reach the new stable value within a deviation of 5 percent (about \( 1\frac{1}{2} \) to 4 minutes). The rise time equals three times the time constant (see table 1.2). In total, therefore, it takes about 2 to 5 minutes (delay time plus rise time) before the full effect of a flow rate change is realized. The sensitivity determines the final MAP decrease, that is the difference between the original MAP and the new MAP (per unit of flow rate). The sensitivity varies more than the delay and the time constant, both inter- and intra-individually. The rise time and the delay are dynamic model parameters; the sensitivity is a static parameter.

If both static and dynamic parameters were known, a controller could be designed that brings the MAP to a new target in about 2 to 4 minutes. This control method would demand accurate estimates of the individual patient characteristics; in practice, these are not available, however.

Instead of for maximum speed, the controller is tuned for maximum robustness. A robust controller is stable for a wide range of patient characteristics; it is fast enough when the individual patient's characteristics agree with the expected characteristics, and safe enough even if the characteristics do not agree.

A PID controller provides a proportional term, an integration term and a derivative term. The integration term (I-term) of the controller increases the flow rate when the pressure is above the setpoint, and decreases the flow rate when the pressure is below its setpoint. Over a longer time the I-term is the time integral of the distance to the setpoint. The I-term is used for adjusting the flow rate as long as the pressure is not at the target level.

The proportional term (P-term) of the controller increases the flow rate when the pressure is increasing and decreases the flow rate when the pressure is decreasing. The P-term works against any change of pressure and is used for anticipating a changing pressure before it has crossed the setpoint yet. The pressure controller does not contain a derivative term (D-term). If there were less measurement noise, the D-term could be applied in order to limit an eventually overshoot.
In order to prevent unlimited growth of the integration term (I-term) when the control signal is limited for a certain period, the *dog lead* principle is applied for the implementation of the PID controller.

\[ u_k = u_{k-1} + G \left[ I \left( y_k - r_k \right) + P \left( y_k - y_{k-1} \right) \right] \]  \[4\]

In this formula, \( u_k \) and \( y_k \) represent the current flow rate and pressure, and \( u_{k-1} \) and \( y_{k-1} \) the previous flow and pressure. The complete list of parameters is:

<table>
<thead>
<tr>
<th>Parameter ( y )</th>
<th>Description</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blood pressure</td>
<td></td>
<td>[mmHg]</td>
</tr>
<tr>
<td>Target or setpoint</td>
<td></td>
<td>[mmHg]</td>
</tr>
<tr>
<td>Flow rate (( u ))</td>
<td></td>
<td>[ml/h] or equally [mg/h]</td>
</tr>
<tr>
<td>Sample time (( T ))</td>
<td></td>
<td>[s]</td>
</tr>
</tbody>
</table>

**G** overall control gain (1/9 or 1/3 or 1 or 3 or 9)

**I** I-parameter

<table>
<thead>
<tr>
<th>I-parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.072</td>
<td>([\text{ml/h}] / \text{mmHg})</td>
</tr>
<tr>
<td>2</td>
<td>0.096</td>
<td>([\text{ml/h}] / \text{mmHg})</td>
</tr>
</tbody>
</table>

**P** P-parameter

<table>
<thead>
<tr>
<th>P-parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0036</td>
<td>([\text{ml/h}] / \text{mmHg})</td>
</tr>
<tr>
<td>2</td>
<td>0.0056</td>
<td>([\text{ml/h}] / \text{mmHg})</td>
</tr>
</tbody>
</table>

\( f = \frac{y - r}{100} \)  \[5\]

\( P = f \times P_1 + (1 - f) \times P_2 \)  \[6\]

\( I = f \times I_1 + (1 - f) \times I_2 \)  \[7\]

The parameter \( f \) represents the distance to the setpoint; if \( |y - r| \) equals 100 mmHg, parameter set 1 is used, if it equals 0 mmHg, set 2 is used instead. At other distances the actual parameters are a weighted average of the two sets. For instance when the distance to the setpoint equals 35 mmHg, the actual parameters equal 35\% of set 1 plus 65\% of set 2. Since the distance to the setpoint is generally less than 35 mmHg, in practice the P and I-parameters are almost constant.

The requirements for automatic control have been formulated in section 1.2. In order to analyze the robustness of the controller, simulations with a model were done. From table 1.1 the *nominal* dynamic model parameters are derived. The nominal delay \( T \) equals 50 seconds; the nominal time constant \( \tau \) equals 60 seconds. Table 2.1 is derived from simulations and shows the control performance when the model parameters are nominal (1), two and three times nominal (2, 3), and one half and one third of nominal (1/2, 1/3) respectively. The parameter \( G \) equals the control gain times the sensitivity (table 1.2); if the controller is adapted correctly, \( G \) equals 1, if the sensitivity is two times larger, \( G \) equals 2 etc. The ideal control case is if all parameters \( T, \tau \) and \( G \) have their nominal values.
In table 2.1 an entry is marked with an x if unstable control results. Unstable control is a large overshoot or oscillation (or both); in the cases with a large time constant, a large overshoot occurs, in the cases with a small time constant oscillation results as well. An entry is marked with an o if some small overshoot occurs, but control is stable because the overshoot is less than 10 mmHg (70 mmHg setpoint change).

The design requirements for the stability of the controller are that all parameters are allowed to deviate simultaneously a factor of two from nominal; thus the outer entries of the matrices in table 2.1 are not in the design range. Table 2.1 shows that the controller is stable for any arbitrary combination of the three model parameters from $\frac{1}{2}$ to 2 times nominal. The worst cases, marked with an o, are the combinations with a large delay (2), a large gain (G = 2 chart) and a time constant that is not nominal (1/2 and 2). Given the nominal dynamic parameters, this concludes that control is stable for any arbitrary combination of a delay between 20 and 100 seconds, a time constant between 30 and 180 seconds and a sensitivity estimate up to a factor of 2 too low or too high. When these ranges are compared with those of table 1.2, it turns out that the controller is stable, provided that adaptation guarantees that the actual control gain is not larger than two times nominal. Gain adaptation is the topic of the next section.
2.2 Adaptation of the controller to the sensitivity

The PID controller can cope with the expected range of dynamic patient characteristics, but it cannot cope with the complete range of the sensitivity; therefore gain adaptation is necessary. When the control gain is too low relative to the sensitivity, control is inaccurate: a new setpoint will be reached slowly and the compensation for drift will be insufficient. When the control gain is too high, the MAP will move quickly to a new setpoint but may overshoot it, while compensation for drift can be so strong that oscillation results.

The sensitivity of the patient is initially unknown; it also changes in time. Moreover the response to the drug is non-linear: the sensitivity will decrease if the SNP flow rate is high (a flow rate increment from 1 to 2 [ml/h] will often have less result than a previous change from 0 to 1). Finally the sensitivity is expected to decrease slowly with the total SNP dose that has been administrated; several physiological control mechanisms fight against an (artificially) lowered pressure.

The patients are divided into 5 sensitivity groups as shown in table 2.2 below. The sensitivity factor (1/9, 1/3, 1, 3 or 9) is the approximate final MAP decrease when the SNP flow rate is increased by 0.2 [mg/h] (or 0.04 [µg/kg/min]). The control gain should ideally be made equal to the inverse of the sensitivity: in that case the control is adapted correctly.

<table>
<thead>
<tr>
<th>Patient sensitivity</th>
<th>1/9</th>
<th>1/3</th>
<th>1</th>
<th>3</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>very insensitive</td>
<td>insensitive</td>
<td>normal</td>
<td>sensitive</td>
<td>very sensitive</td>
</tr>
<tr>
<td>Controller sensitivity</td>
<td>very strong</td>
<td>strong</td>
<td>normal</td>
<td>weak</td>
<td>very weak</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>1</td>
<td>1/3</td>
<td>1/9</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.2
The control gain is adapted until it equals the inverse of the sensitivity

For safety reasons initially the maximum sensitivity is assumed; in most cases the patient will be less sensitive, and the MAP will move to the setpoint slowly, which will activate the gain up adaptation mechanism.
From table 2.1 it was concluded that the PID controller is stable unless all the model parameters simultaneously deviate by more than a factor of 2 from nominal (note that G equals the sensitivity times control gain). Gain adaptation does make robustness better nor worse. Since the gain is changed with a factor of 3, ideally the control gain is enlarged when G is less than $1/\sqrt{3}$, and reduced when G is greater than $\sqrt{3}$. Hence the classes $G = 1/3$ and $G = 3$ in table 2.1 give no problems since adaptation transforms these classes to the $G = 1$ class. For the $G = 1/3$ class the gain up adaptation mechanism that is described in chapter 3 will act, for the $G = 3$ class the gain down adaptation mechanism that is described in chapter 6 will act.

For the $G = 2$ class ($2 > \sqrt{3}$) gain down adaptation to the $G = 2/3$ class is not certain for all entries under all circumstances; control is unstable if $T$ is three times nominal and no gain down adaptation occurs. In that case oscillation will occur, and from simulations it turns out that the safety mechanism that is described in chapter 7 will eventually decrease the gain, which results in $G = 2/3$. Since a patient with such a large delay and with $G = 2/3$ responds too slowly, after some time a gain up adaptation will occur, and oscillation will result again.

Finally for the $G = 1/2$ class ($1/2 < 1/\sqrt{3}$) a gain up adaptation might occur for some entries under some circumstances, which gives problems in some cases with a large time delay. Similarly to the $G = 2$ class, in such cases oscillations results, which activates the gain down adaptation mechanism. However after a while again a gain up adaptation might follow.

The final conclusion is that control is stable unless the model parameters deviate by more than a factor 2 from nominal. Within the range of a factor of 2, the gain adaptation mechanisms perform as desired. Gain adaptation does not make the control robustness better, but is just necessary to keep a stable control for all sensitivity groups.
2.3 Regions for gain adaptation

For gain adaptation, a measure of progress of the pressure to the setpoint needed. When the gain is correct, the progress is within certain limits; when the gain is too low, the progress is significantly slower, and when the gain is too high, it is significantly faster. To obtain a measure of progress, 11 regions are defined around the target level (figure 2.1). After a setpoint change, the MAP settles in one of these regions; the controller will try to bring the MAP to the setpoint (region 5) again. The width of the different regions is chosen so that if the control gain is correct, the MAP stays in each region for about 2 minutes. If the MAP stays in a region for longer than 4 minutes, the progress is too slow and the gain is made 3 times larger. On the other hand if the MAP moves through a region within one minute, the gain is divided by a factor 3 (figure 2.2).

Several factors influence the time the MAP stays in a region. First, it depends on the dynamic characteristics of the individual patient. Second, it depends on how well the control gain reflects the sensitivity. Third, after a setpoint change the MAP settles in a region; it is possible that the MAP starts in the middle of a region, or near the region border.

From simulations it turns out that the second factor is the most important: it turns out that the time spent in a region is a reliable way to determine whether the control gain is too high, too low or correct. Regions offer two main gain adaptation mechanisms; in chapter 3 and 6 the gain up and gain down adaptation mechanisms are described in more detail.

To prevent the MAP from traveling from one region to another and back repetitiously due to small spontaneous pressure variations, the MAP is filtered and the region borders have hysteresis. Filtering in order to reduce the noise influences is used all through the knowledge base; chapter 12 provides an overview. Hysteresis means that a region border is considered crossed only if the MAP is already 2 mmHg in the new region. For example, since the region 5/4 border is at 5 mmHg, the MAP must increase to 7 before it enters region 4; after that it must decrease to 3 again before it re-enters region 5.
2.4 Final remarks about PID control

From the tests it turned out that control robustness is sufficient. For the dynamic parameters $\tau$ and $T$ the variations that were found between the patients were less than the worst case the controller was designed for.

Originally the PID controller had three constant parameters. From the first introduction of the system in which control was not yet active, it turned out that measurement noise was larger than expected, and moreover that the noise was not white but that sometimes some frequencies dominated. In order to guarantee stable control, before automatic control was applied, the D-term of the controller was deleted. In order to keep the robustness and the dynamical response the same as originally, the P- and I-terms were re-tuned, and made dependent of the distance to the setpoint. These changes resulted in the controller that was described in this chapter.
In the previous chapter the concept of regions, which are used for gain adaptation, was introduced; this chapter describes the gain up adaptation mechanism that acts when the MAP stays in a region for too long. The gain down adaptation mechanism, that acts when the MAP moves too fast through a region, is described in chapter 6.

3.1 The need for gain up adaptation

For safety reasons, the controller starts with the lowest control gain (1/9); initially a very sensitive patient (sensitivity 9) is assumed. In many cases, however, the patient is less sensitive, and control is inaccurate. It will take much time before the SNP flow rate is sufficiently high to let the MAP reach the setpoint. In order to make control faster, the control gain must be increased.

3.2 Conditions for gain up adaptation

For each region (except region 5) there is a Simplexsys rule that requests a gain up adaptation if certain conditions hold. For region 0 this rule is as follows.

GainUpRegion0: 'Gain up adaptation requested in region 0'
Region0 > (260) and GainConst > (180) and SetpConst > (120) then tr: GainUpRequest

Rules for other regions are similar; the numeric values of the histories of the first term all are between 210 and 295.

There are three conditions to be satisfied. First the MAP must be in this region for at least 260 seconds (about 4 minutes). This value is chosen because from simulations it turned out that the MAP leaves region 0 within 260 seconds if the control gain is sufficiently high.

The second condition is that the gain has not been changed during the last 180 seconds (3 minutes). The state rule GainConst is always true; when the gain is changed, the GainConst timer (that tells how long the rule has been true) is reset to zero by making the rule false for once, and then true again.
The second condition thus guarantees that at least 180 seconds pass between successive gain up adaptations (see figure 3.1); three minutes is chosen because it takes about that time before the effects of a previous gain change are realized (this time equals the nominal model delay time plus two times the time constant, see section 2.1, where the dynamic characteristics were discussed).

The third condition requires that the target was not changed during the last 120 seconds (2 minutes). The timer of SetpConst is kept up to date in the same way as GainConst; when the setpoint is changed by at least 10 mmHg, the history timer is reset to zero. The setpoint can be changed with smaller steps then 10 mmHg; small steps accumulate and a setpoint change is detected after a 10 mmHg total change.

The third condition states that the target must be constant within a 10 mmHg margin, and is added for the following reason. Suppose the user changes the setpoint repeatedly: when the MAP decreases 5 mmHg or so, the user decreases the setpoint with the same amount (which could be a reasonable action). Then the MAP may stay in one region for a long time due to repeated setpoint changes. Without the third condition, it would be assumed that the MAP does not make sufficient progress, and a gain up adaptation would follow. To prevent this, gain up adaptation is delayed until 2 minutes after the last setpoint change.

Figure 3.1
Control starts at 12:38 with the smallest control gain. The flow shows the effect of the gain up adaptations at 12:43 and 12:46.
The first gain change occurs when the MAP is in a region for about 4 minutes. When it stays in that region for a longer time, a second gain up adaptation is performed 3 minutes (second condition) later. If the MAP stays in that region for another 2 minutes, it is likely that the MAP cannot be further affected by SNP: even a large change of the flow rate will not bring the MAP closer to its target level. In such cases of ineffective control (chapter 9) further gain up adaptation is disabled. Therefore no third gain up adaptation occurs in figure 3.1 at 12:49.

3.3 Resulting actions

When the three conditions are satisfied, a gain up adaptation is requested (then tr: GainUpRequest). A gain up request is usually acknowledged, and then results in a one step increase of the control gain (table 2.2). However, there are circumstances in which the gain up request is ignored; this happens when a transient is detected (chapter 8), when the gain is already at the maximum level, or in case of ineffective control (chapter 9).

If the gain up request is acknowledged, the GainConst timer is reset; this has an impact on the other gain adaptation mechanisms, because all these mechanisms require the gain to be constant for some period. Generally, gain up mechanisms will require GainConst to be true for a longer time than gain down mechanisms.

3.4 Final remarks about gain up adaptation

A relative long stay in a region is necessary to reach a sufficient control gain for less sensitive patients; it turned out from the clinical cases that only a few patients were more sensitive than nominal. Indeed the MAP was stable enough to be in a region for 4 minutes or longer if the control gain is too low. The applied filtering of the MAP and hysteresis between regions (section 2.3) are necessary to eliminate the influence of noise and small pressure variations, and turned out to be sufficient.

Development of other measures of progress rather then the time spent in a region may be interesting. Another measure of progress that may be successful is the integral of the error minus a certain allowed border offset. As long as the error is less than the border, the integrand is limited to zero, and the integral measure does not grow. When the difference is larger than the border, the integral grows, and the larger the difference, the more it grows. When the integral has exceeded a certain limit, the control gain could be increased.
The knowledge base contains four gain down adaptation mechanisms. In this chapter the fastest one is discussed; the other mechanisms are discussed in the next three chapters.

### 4.1 The need for gain down adaptation

Increasing the control gain is necessary when the actual sensitivity of the patient is low compared to the control gain. In some cases, gain down adaptation is necessary. The sensitivity can temporarily be at a lower level, for instance at low target levels, because of the possibly non-linear response to high SNP flow rates [Blom, 1990]. In such cases a gain up adaptation can be necessary. When a gain up adaptation is performed and the setpoint increases after some time, the flow rate will decrease, and the sensitivity is expected to return to the original level. In that case the gain down adaptation mechanism that is described in this chapter will act.

Gain down adaptation due to a large increase of the setpoint is the fastest mechanism of the gain down family. It acts before the flow or the pressure show any effect of the setpoint change. In some cases the gain down action will be not necessary, and will be followed by a gain up adaptation some time later.

### 4.2 Conditions for gain down adaptation due to setpoint change

When the setpoint is increased by a large amount and if the SNP flow rate is also larger than half its maximum level, the following rule requests a gain down adaptation:

\[
\text{GainDownSetpointChange}: \text{'Gain down request due to setpoint change up'} \\
\text{LargeSetpChange and FlowHalfMax} \\
\text{then tr: GainDownRequest}
\]

Rule LargeSetpChange is true if the target is increased by at least 40 mmHg. as for SetpConst (section 3.2), small changes are accumulated. The second condition is added to prevent unnecessary gain down adaptation. For low flow rates it is unlikely that the sensitivity has decreased by such an amount that a gain up adaptation did occur.
If both conditions hold, this rule requests a gain down adaptation. Usually the gain down request is acknowledged, and then results in a one step decrease of the control gain (table 2.2). However, there are circumstances in which the gain down request is ignored; this happens when a transient is detected (chapter 8) or when the gain is already at its minimum level. The exact conditions for acknowledgement of gain change requests are given in section 9.4. When the gain up request is acknowledged, the GainConst timer is reset; this has an impact on the other gain adaptation mechanisms.

4.3 Final remarks about large setpoint changes

This mechanism has never acted, because such large setpoint changes did not occur.
In chapter 4 the fastest gain down adaptation mechanism was described, which acts when the setpoint is increased by a large amount. In this chapter a slower mechanism is described: the control gain is decreased when the flow rate changes quickly. This mechanism too is quite fast, since it acts before the change of flow has affected the MAP.

Two more gain down adaptation mechanism exist; chapter 6 describes a mechanism that acts on fast changes of the MAP, and chapter 7 describes oscillation detection.

5.1 High control gain results in high flow rate change

For safety reasons and to obtain quiet control, the PID controller changes the flow rate slowly. Due to the delay time, the patient does not react instantaneously to the change of the flow rate. When the flow rate changes quickly after a setpoint change, during the delay time the flow will change by a large amount, so that the MAP might eventually overshoot the target level.

If the MAP is at the target level, fluctuations of the pressure will result in similar fluctuation of the flow rate; theoretically this can cause unstable control or oscillation. In such cases the relative change of control is large and an appropriate action to make the control more quiet is to decrease the gain.

Therefore for safety reasons the permitted relative change of the flow rate is limited to plus or minus 7%. Since the sampling time equals 5 seconds, this roughly means that the flow can change by a factor of two in about one minute. The PID controller is tuned so that if the control gain is correct, the flow rate will not change that quickly. Thus when the flow rate change is repetitiously larger than 7%, it is likely that the control gain is too high; moreover, the most effective method to limit the change of flow is to decrease the gain.

5.2 A relative measure of change of control

Each 5 seconds, the SNP flow rate is incremented (or decremented) by an amount which is determined by the control gain, the P- and I-terms, the distance between the MAP and the target level and the slope of the MAP (equation [4] of section 2.1).
This increment is added to the previous flow rate to obtain the new flow. Hence this increment is a measure for the change of the flow rate.

At high flow rates the increment is generally larger than at low flow rates; a relative measure is the increment divided by the control signal. To prevent very large values when the flow rate is almost zero, the increment is divided by 0.5 if the flow rate is less than 0.5. This relative measure should be suitable both for sensitive as well as for insensitive patients.

5.3 Conditions for gain down adaptation due to large change of flow

When the flow rate changes more than 7% repeatedly, it is likely that the control gain is too high. To prevent once-only large flow rate changes from resulting in gain down adaptation, some hysteresis is provided. State rule HighFlowSlope becomes true when the flow rate changes more than 8.5%, and stays true until it changes less than 5.5%. The following rule performs gain down adaptation when that state rule is active for more than 15 seconds.

GainDownUSlope: 'Gain down because flow rate changes too fast'
HighFlowSlope > (15) and GainConst > (30) and SetpConst > (15)
then tr: GainDownRequest

The first condition is clear from the preceding. The second condition is added to allow the flow rate to change quickly for 30 seconds after a previous gain (up) adaptation; the third condition is added to allow a temporary high flow rate change for 15 seconds following a setpoint change.

Rule HighFlowSlope is a state rule which is the reverse of LowFlowSlope. One of them is active in automatic mode of operation, neither is active in manual mode. When switching to automatic, state LowFlowSlope is activated. If the flow rate change exceeds 8.5%, there is a transition from state LowFlowSlope to HighFlowSlope. When the flow rate changes less than 5.5%, a transition from state HighFlowSlope to LowFlowSlope occurs.

Note that the flow rate change is limited to 7%; the measure of flow rate change used in this mechanisms is computed before limiting.

Usually the gain down request is acknowledged, but as stated in section 4.2, there are some exceptions.
5.4 Final remarks about flow rate changes

Measurement noise affects the flow rate via the P-term of the controller. Because the measure of flow rate change is a relative one, the noise influence is larger at lower flow rates. When the gain has been increased one or two times, and the flow rate decreases to a low level, the border of 8.5% is easily exceeded, which results in an unnecessary gain down adaptation. During the tests these unnecessary gain down adaptations happened in 50% of the cases, resulting in a slower than indicated control.

To reduce the influence of noise, filtering could be applied. Furthermore flow increments should be distinguished from flow decrements: three states are necessary (LowFlowSlope, HighFlowSlopeUp and HighFlowSlopeDown) instead of two. Hysteresis should also be larger: state HighFlowSlopeUp is entered if the flow rate increment exceeds 10%, and state LowFlowSlope is entered again if the slope is less than 4%. State HighFlowSlopeDown is similarly entered when the flow rate decreases more than 10%. Finally a gain down should be requested after 30 seconds of high flow slope, rather than after 15 seconds. It turns out (from a playback of the cases) that the number of gain down adaptations, which were unnecessary because a gain up follows some time later or which result in a gain that is far too low compared to the (estimated) patient sensitivity, is reduced when the knowledge base would be changed in this way.
The gain up adaptation mechanism that acts when the MAP stays in a region for too long was discussed in chapter 3. This chapter describes the gain down adaptation mechanism that acts when the MAP moves too fast through a region. This mechanism acts later than the gain down mechanisms that were described in the previous sections.

6.1 Detection of fast pressure change

The time that the MAP stays in a region is recorded. It can either be long, resulting in a gain up adaptation, correct or short. A short stay in a region does not necessarily mean that the control gain is too high. For instance it is possible that, after a setpoint change, the MAP enters a new region close to a region border, and thus leaves that region after a short period. Therefore gain down adaptation should result only when the MAP moves fast through two successive regions in the same direction.

6.2 Implementation of the mechanism

When the MAP leaves region number 1, at most one of the following rules becomes true:

ShortDown1: 'Too fast from region 1 to 2'
Region1 < (45) and RegionCross12 and SetpConst > (30)
then tr: ShortDown

ShortUp1: 'Too fast from region 1 to 0'
Region1 < (45) and RegionCross10 and SetpConst > (30)
then tr: ShortUp

Long1: 'Long enough in region 1'
Region1 > (45)
then tr: Long

RegionCross12 means that the MAP moves from region 1 to region 2. All other regions have similar rules, which make ShortDown, ShortUp or Long true via the then tr part. Thus rule Long is true when the MAP leaves an arbitrary region and was in it for at least 45 seconds; rule ShortUp or rule ShortDown can become true instead when the MAP was in a region for less than 45 seconds.
Whether rule ShortUp or ShortDown becomes true depends on the MAP's direction, either increasing or decreasing. All regions have the same history time of 45 seconds.

The condition of rule Long1 is clear from the preceding. Rule ShortDown1 has three conditions; the first and second conditions state the MAP is decreasing and leaves region 1 while the MAP was less than 45 seconds in that region. The value of 45 seconds is chosen because from simulations it turned out that if the control gain is correct, the MAP stays longer than 45 seconds in a region, but if the gain is too large, it is likely that the MAP moves through a region within 45 seconds.

The third condition states that the setpoint must have been constant for at least 30 seconds. After a setpoint change, the location of the regions is changed, and it takes some time before the MAP reaches the new region; this is because there is at most one region border crossing in each 5 seconds period. While settling in a new region, the MAP moves fast through other regions, and within 30 seconds the final region is reached. This condition prevents detection of ShortUp or ShortDown due to a setpoint change. Note that the MAP will not move to the new setpoint during the first 30 seconds because the model delay time is larger.

When ShortDown is twice true without Long or ShortUp being true in between, state rule ShortDownTwice becomes true, and requests a gain down adaptation. Similarly when ShortUp is twice true without Long or ShortDown in between, state rule ShortUpTwice becomes true. The following on-statements perform these transitions.

- `on ShortUp from LongState to ShortUpOnce`
- `on ShortUp from ShortUpOnce to ShortUpTwice`
- `on Long from ShortUpOnce to LongState`
- `on Long from ShortUpTwice to ShortUpOnce`
- `on GainDownShort from ShortUpTwice to LongState`
- `on ShortDown from ShortUpOnce to LongState`
- `on ShortDown from LongState to ShortDownOnce`
- `on ShortDown from ShortDownOnce to ShortDownTwice`
- `on Long from ShortDownOnce to LongState`
- `on Long from ShortDownTwice to ShortDownOnce`
- `on GainDownShort from ShortDownTwice to LongState`
- `on ShortUp from ShortDownOnce to LongState`
The following rule requests the gain down adaptation:

GainDownShort: 'Gain down requested: too fast through regions' (ShortDownTwice or ShortUpTwice) and GainConst > (75) then tr: GainDownRequest

The first condition (ShortDownTwice or ShortUpTwice) is clear from the preceding. The second condition requires that the gain has not been changed for 75 seconds. Similar to the gain up rules discussed in section 3.2, this history time could have been 180 seconds, because that is the time that elapses before a previous gain change shows its results. For safety reasons however, gain down adaptation is permitted after a shorter period.

Rule GainDownShort is a trigger rule for the transition from ShortUp Twice or from ShortDownTwice to LongState; when a gain down request is performed, the protocol returns to the state LongState and a second gain down request can occur only if rule ShortUp (or ShortDown) is true twice again.

If no gain down request occurs, because the second condition of rule GainDownShort does not hold, the state ShortUpTwice or ShortDownTwice stays true until transition Long or GainDownShort occurs.

Usually the gain down request is acknowledged, but not during a transient (chapter 8) because this also causes fast movement through regions.

6.3 Final remarks about regions for gain adaptation

From the tests it turned out that the gain down adaptation mechanism that acts on a large change of control performed some unnecessary gain down adaptations (section 5.4). Therefore the control gain was rarely too high, so that this mechanism did not act as often as expected.

Although (as a consequence of the previous) this mechanism was not fully tested during the tests, it turned out that regions 4 and 6 are quite narrow. This results for instance in the fact that when the MAP is in region 3, that a small drift of 20 mmHg will move the MAP to region 7, which can cause a gain down adaptation if the drift occurs within 90 seconds.
This problem transpired only twice during the tests, but when the gain down adaptation mechanism that acts on large change of control is changed as suggested in section 5.4, from playback of cases it turns out that this problem develops more often.

To solve these problems, the regions could be made wider, for instance by merging the regions 3 and 4, and 6 and 7. However, the gain up adaptation rules, that are based on the time spent in a region must be given other history times when the regions are made wider.

Another solution is to make the history times of the ShortUp and ShortDown rules of region 3, 5 and 7 equal to 30, and for region 4 and 6 equal to 15, instead of the original values that equal 45 seconds for all regions. The history times of the Long rules should stay the same. Playback shows that when the knowledge base is changed that way, small drifts do not cause unnecessary gain down adaptations.
7.44 Gain down adaptation 4: oscillation detection

If the control gain is too high for too long, oscillation might develop. The last mechanism that can perform gain down adaptation acts when oscillation is detected. Other gain down adaptation mechanisms will hopefully prevent oscillations from developing so that this mechanism will hardly need to come into action. It is added for safety reasons, in case all other mechanisms fail.

There are patients whose pressure is inherently unstable: the MAP shows spontaneous fluctuations which are comparable to oscillations. The best action in such cases is to decrease the gain, in order to stabilize the control.

7.1 Detection of oscillation

In case of oscillation, the MAP is alternately higher and lower than the target level; thus to detect oscillation, the number of setpoint crossings per unit of time could be computed. However due to measurement noise the setpoint will be crossed often, but fluctuations around the setpoint of less then 5 mmHg (region 5) are no oscillations. Therefore instead the number of times the region 5/4 or 5/6 border is crossed is counted; oscillation is detected when 4 border crossings have been counted in a short time. Four border crossings over a long period should not result in oscillation detection, and therefore the counter is decremented every 250 seconds (limited to 0). Since other causes such as setpoint changes also result in region borders crossing, counting for oscillation detection is somewhat more complex, and is discussed in more detail in the next section.

The numeric values are motivated as follows. Because of the dynamic model parameters, it is unlikely for oscillation to occur with a cycle period of less than 4 minutes. Hence oscillation is only detected when the cycle period is less than 250 seconds. During one cycle period 4 border crossings occur; after 1½ cycle the counter is increased due to 6 border crossings, and decreased by at most 2 because 1½ periods take less then 500 seconds. The resulting value is larger then or equal to 4 and thus oscillation is detected.
Oscillation is therefore detected within 2 cycle periods, when the cycle period is less than 4 minutes and the amplitude larger than 5 mmHg, and results in gain down adaptation as the following rules show.

DetectOscillation: 'Oscillation detected'
\[\text{btest } \text{NumberBorderCrossings } \geq 4\]
\[\text{then do } \text{NumberBorderCrossings} := 0\]

GainDownOscillation: 'Gain down because oscillation is detected'
DetectOscillation \text{ and } \text{GainConst } > (100) 
\[\text{then tr: GainDownRequest}\]

The rule DetectOscillation resets the counter to zero when the counter is larger than or equal to 4 (and also if no gain down adaptation is requested). The second condition of rule GainDownOscillation prevents a gain down request from following a previous gain down adaptation too quickly. Suppose an other mechanism performs a gain down adaptation when the counter equals 3. It will take some time before the MAP does not show oscillatory behavior anymore; during that time it is likely that another border is crossed, so that DetectOscillation results in true. In that case no gain down is requested, but the counter is reset to zero; otherwise rule DetectOscillation stays true, and after 100 seconds the condition GainConst > (100) holds, which would result in another gain down adaptation. Oscillation that results from a previous gain up adaptation takes more than 100 seconds to develop; condition GainConst > (100) does not delay the action that should follow in such a case.

7.2 Counting the number of border crossings

The counter NumberBorderCrossings is incremented when the MAP enters or leaves region 5. After a setpoint change the MAP first leaves region 5, and re-enters it some time later. The number of border crossings will thus be incremented: by one when the target is changed, and by one when the new target is almost reached. Setpoint changes should not cause oscillation to be detected; thus when a setpoint change larger than 5 mmHg occurs, the counter is decreased by 2 (limited to -2 because the region border crossings follow some time later: the first because it takes some time before the MAP has settled in a new region, the second due to the response of the patient).

A transient (chapter 8) also results in a certain (unknown) number of border crossings; therefore during a transient the counter is set to zero.
Since the number of border crossings is updated at several places all through the knowledge base, an overview follows:

The counter is incremented by one if one of the borders 5/4, 5/6, 4/5 or 6/5 is crossed.

Each 250 seconds (about 4 minutes) it is decremented by one, if it was larger than 0 before.

After a setpoint change larger than 5 mmHg it is decremented by 2, if it was larger than or equal to zero before.

If a transient is detected the counter is set to zero, and stays zero until the transient is over.

Finally if the counter reaches the value 4, oscillation is detected and the counter is set to zero; when also the gain is constant for some time, a gain down adaptation results.

7.3 Final remarks about oscillation detection

True oscillations did not occur in any of the cases, but it was detected incorrectly in 5 of the 30 cases. Small MAP changes from region 5 to 4 and back to region 5 also increment the counter, but such fluctuations are not oscillations. Therefore after a region 5/4 border cross, the counter should not be incremented again until a region 5/6 borders crossing happens. This is easily implemented by a Petri net, with states AboveSetpoint and BelowSetpoint. The number of transitions

on RegionCross56 from UnderSetpoint to AboveSetpoint  and  
on RegionCross54 from AboveSetpoint to UnderSetpoint

that occur, is counted; a complete oscillation cycle increments the counter by 2 and oscillation should be detected after 1½ cycle if 3 such border crossings are counted.
8.1 The need for transient detection

The word *transient* refers to a truly physiological pressure to which the controller must react in a special way (section 1.6). In particular, a transient is defined as an *unexpected, temporary, large* pressure increase or decrease; if the MAP is increasing it is called an *up transient*, otherwise a *down transient*.

Because transients are by definition temporary, the controller must not try to bring the MAP toward the target level during a transient; any attempt to do so would require a large change of the flow rate, without results, that the controller must recover from again when the transient is over. A better reaction to a transient is to keep the flow rate at a constant level until the transient is over. If the pressure change turns out to be permanent after all (thus it is *not* a transient) normal control should resume.

For *up transients* this is the correct strategy. When a transient is detected, the flow rate is kept constant. When the pressure change turns out to be permanent, a temporarily high MAP is acceptable and could be suppressed by other drugs if necessary.

However to keep the flow at a constant rate when a *down transient* occurs, is not the best strategy to ensure the patient's safety. If the transient is temporary, it might be the correct action to freeze the flow rate, but if the decrease is permanent, the flow must be shut off because a low pressure is dangerous. The problem is of course that when a large pressure change down is detected, it is not known whether that change will turn out to be permanent or temporary. The strategy is to assume the worst: shut off the flow when the pressure drops quickly, and resume the flow rate at its pre-transient value when it turns out to be a temporary decrease after all.

For several reasons transient detection is a quite complicated mechanism. A transient is an *unexpected* change of the MAP, thus MAP changes that are caused by a change of the *flow rate* (either manual adjustment or automatic control) must not be detected as transients (section 8.3).
Second, a decrease of the MAP that is preceded by a somewhat smaller increase should be treated as the end of the prior small increase, rather than as the start of a down transient (section 8.4).

Finally the end of an up transient, which can show a pressure decrease like the start of a down transient, should not be detected as such (section 8.5 and 8.6).

**8.2 Some common techniques for transient detection**

In order to detect transients, each sample time the new MAP (if valid) is compared with previous measurements. When the difference exceeds a certain limit, a transient is detected. For instance an up transient is detected if the MAP rises more than 20 [mmHg] in less than 15 seconds, or 25 [mmHg] in 30 seconds or 40 [mmHg] in 150 seconds. The slower the MAP changes, the more it must rise before a transient is detected.

Special care is necessary in case of invalid measurements. A measurement is invalid if the characteristics that are derived in each heart cycle differ from the characteristics of previous cycles. When a transient develops, the pressure changes quickly, and thus it is likely that invalid measurements occur. The display (figure 8.1) shows a valid measurement by a solid vertical line and an invalid measurement by a dot instead; the last valid MAP is maintained until a new valid measurement is available. After a period of invalid measurements the first valid MAP can differ considerably from the previous valid MAP, that is in worst case several minutes old. To prevent transient detection in such cases, only valid MAPs are compared with previous valid MAP measurements.

The end of a transient is detected when the MAP is almost back to its pre-transient value. The pre-transient pressure is the pressure at the beginning of the transient, and it equals the average value of the lower and the higher MAPs that are involved in transient detection. For instance in figure 8.1 the start of the transient is detected when the MAP is 110 [mmHg], because it was 70 [mmHg] 45 seconds earlier; the pre-transient pressure therefore equals 90 [mmHg] and the end of the transient is detected if the MAP is lower than 90 [mmHg] again. The pre-transient pressure should not be too high (in case of an up transient) because then it is likely that the end of the transient is detected too quickly. On the other hand if the pre-transient pressure is too low, there is a risk that the end of the transient is not detected as such. As a compromise the pre-transient pressure is computed as described above.

Besides the pre-transient pressure, at the start of the transient the pre-transient flow is computed. It equals 95% of the filtered flow; the flow is filtered and limited to 95% in order to disregard the small increase of the flow during the very beginning of the transient.
8.49 Actions in case of large pressure fluctuations

During an up transient the flow is kept constant at this pre-transient flow (figure 8.1). The general filtering method is discussed in section 12.1; the parameters and the characteristics of each individual filter that is used in the knowledge base, are described in section 12.2.

After detection of the end of the transient, the MAP will generally decrease further (in figure 8.1 finally to 70 [mmHg]). Then a down transient could be detected because the pressure is 40 [mmHg] lower than the previous value of 110 [mmHg]. This should not happen: the end of an up transient is not the start of a down transient. Therefore at the end of the transient (crossing of the pre-transient pressure) all old MAPs that are used for transient detection are reset to the pre-transient pressure. As a consequence a down transient is detected only if the pressure drops below the pre-transient pressure (for instance to 55 [mmHg]).

The end of a transient cannot always be detected; therefore several time-out mechanisms are included. If no transient end is detected within 5 minutes after the start of the transient, it is assumed that the pressure change was not temporary but is permanent; the transient is decided to be over and normal control continues.

The common techniques described in this section are used for up transients (section 8.5) as well as for down transients (section 8.6).

8.3 Expected pressure change due to a change of the flow rate

The controller is designed in order to let the MAP move smoothly to the target level. Even if the control gain is too high, no transient-like MAP changes will occur. Manual adjustment of the flow rate is more abrupt than the control action, but this will generally not result in a MAP change that is so fast that a transient is detected either. However, this is not certain: if the zero flow key is pressed, the flow is shut off immediately and generally after about a minute the MAP will rise quickly and an up transient could be detected.

During automatic control it is also possible that the pressure changes so fast that a transient is detected. Suppose for instance that the setpoint is decreased, that the flow is then manually set from zero to 1, and that then operation is switched to automatic mode (this can be an acceptable way of use). Some time later, the pressure will decrease (maybe in a transient-like fashion), and the controller will change the flow rate in order to restore the pressure to the setpoint. No transient should be detected if the change of the MAP is caused by a changing flow rate.
If the flow rate is quickly rising, either in manual or in automatic mode of operation, the MAP is expected to decrease, and if the flow is quickly decreasing the MAP is expected to increase. If the flow rate is stable, the MAP is expected to remain almost constant. When a transient-like pressure decrease/increase occurs, but the MAP is expected to decrease/increase, no down/up transient is detected unless the pressure is above/below the setpoint.

A fast changing flow is detected with some filtering techniques (chapter 12). The flow rate is compared with the filtered flow rate and if the relative difference exceeds 45%, the flow rate is considered to be quickly changing and the MAP is expected to change as well; if the difference is less than 25% the flow is stable (again) and the MAP is also expected to be stable:

\[
\text{FlowIncrease} : \text{'Flow rate has increased strongly'}
\]
\[
\text{btest } \text{SlowFlowSlope} > 45
\]

\[
\text{FlowDecrease} : \text{'Flow rate has decrease strongly'}
\]
\[
\text{btest } \text{SlowFlowSlope} < -45
\]

\[
\text{FlowConstant} : \text{'Flow rate is constant'}
\]
\[
\text{btest } -25 < \text{SlowFlowSlope} < 25
\]

\[
\text{on } \text{FlowIncrease from } \text{ExpectMAPEqual to } \text{ExpectMAPDecrease}
\]
\[
\text{on } \text{FlowDecrease from } \text{ExpectMAPEqual to } \text{ExpectMAPIncrease}
\]
\[
\text{on } \text{FlowConstant from } \text{ExpectMAPDecrease to } \text{ExpectMAPEqual}
\]
\[
\text{on } \text{FlowConstant from } \text{ExpectMAPIncrease to } \text{ExpectMAPEqual}
\]

The next section explains the impact of the state rules ExpectMAPIncrease, ExpectMAPEqual and ExpectMAPDecrease on transient detection.

8.4 Small pressure changes

A change of the MAP that is preceded by a somewhat smaller change in the opposite direction needs special care. For instance in figure 8.4 there is at 09:59 a small pressure decrease from 55 to 40 [mmHg] (too small to detect a down transient). After 20 seconds the pressure increases from 40 to 70 [mmHg], which is large enough to detect an up transient. No up transient should be detected at this point, however, because the MAP just recovers (to 55 [mmHg]) from its previous small decrease, and rises 15 [mmHg] more. Only when the MAP increases to 100 [mmHg] or more within a short period, an up transient should be detected (the pressure in figure 8.4 changes too slowly).
A *pre-condition* for an up transient is a small increase of the MAP just before, and a condition that *disables* detection of an up transient is a small MAP decrease just before a large pressure increase is detected.

This is implemented as follows. The state rules TransUpEnabled and TransDownEnabled enable up and down transient detection; these states are made active from TransDisabled, as the following on-statements show:

```
on HalfUp     from TransDisabled      to TransUpEnabled
on EndHalfUp  from TransUpEnabled     to TransDisabled
on HalfDown   from TransDisabled      to TransDownEnabled
on EndHalfDown from TransDownEnabled  to TransDisabled
```

HalfUp: 'there is a small unexpected slope increase'
SlopeSmUp and not (ExpectMAPIncrease and UnderSetpoint)

EndHalfUp: 'pressure is not increasing'
SlopeSmDown or TransUpEnabled > (120)
then do ResetTransientDetection (Current Pressure)

The rules HalfDown and EndHalfDown are similar.

The rule SlopeSmUp marks a *small* pressure increase of 10 [mmHg] in 15 seconds or 17 [mmHg] in 75 seconds; the rule SlopeUp marks a *large* pressure increase (see next section). The second condition of rule HalfUp requires that the MAP is not expected to increase (section 8.3).

Normally state TransDisabled is active; if the rule HalfUp is true, the corresponding on-statement is executed, and the state TransUpEnabled is made active. In that state an up transients is expected to be detected (see next section). If no up transient is detected within 120 second (2 minutes) or when a small slope down is detected instead, the rule EndHalfUp is true, and the state TransDisabled is made active again. Transient detection is also reset; as described in section 8.2, all old MAP values that are used for the detection of transients are reset to the current pressure. As a consequence rule HalfDown is not made true unless the pressure becomes lower than the current level.

### 8.5 Detection and treatment of up transients

An up transient is detected when the MAP rises more than 20 [mmHg] in less than 15 seconds, or 25 [mmHg] in 30 seconds, or 40 [mmHg] in 150 seconds (rule SlopeUp of the on-statement below). As described in the previous section, rule SlopeUp can only be made true when a smaller, unexpected MAP increase has enabled transient detection.
Transient detection is not different in manual mode of operation, although in manual mode the flow rate is not affected and can still be changed manually. However when switching to automatic during an up transient, the flow is frozen at the level that was last adjusted, and control does not start until the transient is over.

The end of the transient is detected if the MAP is below its pre-transient level (trigger rule CloseUp in the on-statement below); transient detection is reset as described in section 8.2. If no transient end is detected within 5 minutes, it is assumed that the pressure increase is permanent, and normal control continues. However transient detection is not reset: if the MAP returns to its pre-transient pressure after all, no down transient should be detected. Figures 8.1 and 8.2 show transients that occurred during the tests.

Figure 8.1
Upper part: transient from 10:52:30 - 10:53:30
Lower part: the flow is kept constant

Figure 8.2
Up transient from 09:12:40 - 09:15:20

Two on-statements and two state rules are used for up transients. Normally state No (Transient) is active. When state TransUpEnabled (section 8.4) is also active, trigger rule SlopeUp is evaluated, and if it returns true, state Up (Transient) is made active. State Up enables the second on-statement, and if trigger rule CloseUp returns true, the states No and TransDisabled are re-entered.

\text{on SlopeUp from No TransUpEnabled to Up}
\text{on CloseUp from Up to No TransDisabled}
The detection of the start of a down transient is similar to the detection of an up transient. A down transient is detected if the MAP decreases more than 23 [mmHg] in less than 15 seconds, or 30 [mmHg] in less than 30 seconds. The pre-transient pressure and the pre-transient flow that are used in a subsequently stage of the transient, are similarly computed.

However, the procedure in case of a down transient is quite different: the flow is shut off when the transient is detected, and stays zero until the pressure is above the pre-transient pressure again. The flow is then kept constant at the pre-transient flow, and a further increase of the MAP is expected as a reaction to the previous zero flow period (figure 8.5). When that increase is also over, normal control continues.

![Petri net that is used for transients](image)

**Figure 8.3**
The Petri net that is used for transients

The complete cycle of a down transient is illustrated in figure 8.5, and implemented by the Petri net shown in figure 8.3 (the upper part of the Petri net implements the detection of up transients).

(0) SlopeSmDown is true due to a small pressure decrease and if such a decrease is not expected, the state TransDownEnabled is true, and enables down transient detection (in figure 8.5 at 11:00). The connection of the state rule TransDownEnabled with the Petri net shown in figure 8.3 is not indicated (section 8.4).
(1) SlopeDown marks the start of the down transient. The pre-transient pressure as well as the pre-transient flow is computed (marked by "define PrePress" in figure 8.3), state Down1 is activated and the flow is shut off (in figure 8.5 at 11:00:30).

(2) Rule CloseDown is true if the MAP is above the pre-transient pressure; state Down2 is made active and the flow is restored (at 11:01:20) but kept constant at the pre-transient level.

In section 8.2 the reasons were discussed for defining the pre-transient pressure below the pressure preceding the transient. As a consequence the MAP is expected to increase above the pre-transient pressure. The flow is resumed when the MAP reaches the pre-transient pressure rather than when the MAP is back to the original level, in order to prevent the zero flow period from resulting in a large MAP increase.

(3) The MAP is now expected to rise 15 [mmHg] above the pre-transient pressure, which occurs in figure 8.5 at 11:02. State Down3 is entered, and the flow is still kept constant. Transient detection is reset at the current pressure in order to enable detection of an up transient that may result from the zero flow period.

(4) An up transient (SlopeUp) is detected at 11:03; a new pre-transient pressure is computed and state Down4 is entered.

(5) Finally the end of pressure increase is detected at 11:05. Normal control is continued and the initial states No and TransDisabled are re-entered.

Figure 8.4
Down transient at 10:04

Figure 8.5
Down transient at 11:00
When the MAP is expected to increase in state Down2 or state Down3, a transient-like pressure decrease might occur instead. In that case state Down1 is re-entered.

Several timeout mechanisms are added. When the MAP does not rise in state Down1, after 4 minutes the initial states are made active again and normal control continues. In many cases the complete sequence will not be passed through (for instance in figure 8.4 no pressure increase follows in step 2). When state Down2 is active for more than 90 seconds, no increase is expected anymore and the initial states are re-entered. When state Down3 is active for more than 90 seconds, a transition to state Down4 occurs and a new pre-transient pressure is computed in order to have a normal transition in step 5. The timeout periods of 90 seconds for these states equal the sum of the model delay time and time constant. When state Down4 is active for more than 150 seconds (time delay plus two times the time constant) the initial states are also made active again.

8.7 Consequences of incorrect detection of transients

From the clinical tests it turned out that transient detection is performed accurately. Nevertheless a discussion of the consequences of incorrect decisions is necessary.

A change of the flow rate will generally not result in transient-like MAP changes, and if so, a mechanism exists that prevents transient detection (section 8.3). In order to deter this mechanism from preventing the detection of a (real) transient, a change of MAP that overshoots the target level is treated as a transient, even when it is expected.

There are cases in which it is undecidable whether the MAP change should be treated as a transient or not. It is possible that a transient is detected while the anaesthetist would not have taken special actions. When an up transient is detected, the consequence is that the flow is kept constant for at most 5 minutes (time-out period). If the constant level is too high, the MAP will decrease and as a result the end of the transient will be detected; normal control continues and the flow will decrease. However if the constant level is too low, the MAP will be too high, and a dangerous situation might occur.

If a down transient is detected incorrectly, the flow is shut off. After some period the MAP will rise, and the end of the transient is detected; the flow is resumed and hopefully the pressure will not increase too much in the mean time.
Similarly it is possible that a transient should be detected, but is not. In the case of a rising MAP the controller will increase the flow rate. Because gain adaptation is still active, it is likely that the control gain is decreased (section 8.8). As a result the flow will increase slowly, and when the transient is over, hopefully not too high flow rate results. On the other hand when the pressure is falling, the flow rate decreases (which is the correct action), but then too a gain down adaptation is likely to occur, which will slow down the control action.

8.8 Impact on the adaptation mechanism

When a transient is detected, a gain down adaptation is likely to be requested because the MAP moves fast through regions (chapter 6). However, this request must not be honored. Other gain adaptation mechanisms can also act, for instance a gain down due to large changes of the flow rate when the transient first occurs (chapter 5), or a gain up due to a long stay in a region (chapter 3). Therefore gain adaptation is disabled during a transient, and it stays disabled until 3 minutes after the end of the transient. The complete set of rules for gain adaptation is given in section 9.4.

Detection of oscillation is also affected. During a transient, the counter NumberBorderCrossings is set to zero because the MAP can cross the region 5 borders several times. Since it is not known beforehand how many times that border will be crossed, the counter is kept zero during a transient.

8.9 Final remarks about transients

The transient detection mechanism is quite complicated; it turned out in practice that transient detection was accurate and that the corresponding actions were correct. Much knowledge about transients was obtained during the first introduction of the system, and this knowledge was incorporated into the system before closed loop testing started (section 1.7).
9.1 Causes of ineffective control

Control is *ineffective* if a further change of the SNP flow rate will not bring the MAP closer to the desired level; this happens at high as well as at low setpoints. If the setpoint is above the MAP, the flow rate will decrease; control is ineffective if the SNP flow decreases to zero, while the MAP is still lower than the target level. Generally this is no serious problem; the patient just does not need artificial pressure lowering and a low pressure is generally not dangerous. Similarly control is *ineffective* if the flow rate is maximal (the flow rate is limited because SNP is toxic) while the MAP is still higher than its target level. This could give problems (a too high pressure for a certain period can be dangerous), and other drugs could be administrated.

9.2 Detection of ineffective control

Several situations give evidence for ineffective control. First, a *very long stay* in a region means that the pressure does not move toward the setpoint; after a certain period in a region it is unlikely that the MAP will eventually reach the setpoint. Second, control is ineffective when the flow rate is at *maximum or minimum level* for some time. Third, a *gain up request* while the gain is already maximal is a symptom for ineffective control.

In the first situation the MAP proceeds to the setpoint very slowly, and maybe the target level is not even reachable. As described in chapter 3, when the MAP is in a region for about 4 minutes, the control gain is enlarged; when it stays another 3 minutes in that region, the gain is enlarged again. A third gain up adaptation would follow after another 3 minutes. However in that case it is likely that further change of the flow rate will not have results: even less sensitive patients (1/3 and 1/9 class) are expected to leave a region after 2 gain up adaptations. Before a third gain up adaptation is requested, one of the following rules is true:

NoProgress1: 'Very long above target'
Region0 > (510) or Region1 > (510) ... Region4 > (505)

NoProgress2: 'Very long under target'
Region10 > (510) or Region9 > (510) ... Region6 > (505)
Ineffective control

The second case of ineffective control is when the flow rate is more than 95% or less than 2% of the maximum level for more than 60 seconds. But after a setpoint change the flow rate is allowed to be at the maximum or minimum level for some time:

- **FlowLimit1**: 'Flow is at Maximum rate'
  \[ \text{FlowMaximum} > (60) \text{ and SetpConst} > (60) \]

- **FlowLimit2**: 'Flow is at Minimum rate'
  \[ \text{FlowMinimum} > (60) \text{ and SetpConst} > (60) \]

Finally, a gain up request while the control gain is at the maximum level is a symptom of ineffective control:

- **GainLimit**: 'Gain up request while gain is at maximum'
  \[ \text{GainUpRequest and GainMaximum} \]

Hence ineffective control is detected as follows:

- **IneffectiveControl**: 'Control is maybe ineffective'
  \[ \text{NoProgress1 or NoProgress2 or FlowLimit1 or FlowLimit2 or GainLimit} \]

### 9.3 Gain change request versus gain change adaptation

Several actions are taken in case of ineffective control. The flow is limited between its maximum and minimum, and similarly the gain is limited to its maximum. A more general action is to disable gain up adaptation in all cases of ineffective control. When the gain is enlarged during a zero flow period (because the MAP is below the setpoint for a long period) a dangerous situation might occur when the MAP increases to a level above the setpoint at a later time.

Gain adaptation mechanisms request a gain up or gain down adaptation; usually this request is acknowledged and followed by a gain change. However in case of ineffective control or when a transient is detected, the gain change request is not acknowledged, as the following rules show:

- **GainUp**: 'Gain up request acknowledged'
  \[ \text{NoTransient} > (180) \text{ and GainUpRequest and not FlowMaximum and not IneffectiveControl and not GainMaximum} \]
  \[ \text{then do gain := 3 * gain} \]

- **GainDown**: 'Gain Down request acknowledged'
  \[ \text{NoTransient} > (180) \text{ and GainDownRequest and not GainMinimum} \]
  \[ \text{then do gain := gain / 3} \]

- **GainChanged**: 'Resets Last Gain Changed timer'
  \[ \text{GainUp or GainDown} \]
9.4 Final remarks about ineffective control

It turned out from the clinical cases that ineffective control was detected mostly because the flow reached its maximum or minimum, and further gain up adaptation was disabled. However in some cases the flow rate was not at its maximum or minimum for only a few samples, and an unnecessary gain up adaptation resulted. This problem is solved when the implementation of the mechanism is changed. Instead of disabling gain up adaptation if the flow is at its maximum for a minute, gain changes should be enabled only if the flow has not been at its maximum for a minute.

At low flow rates, gain down adaptation should be disabled also. Due to noise influences, gain down adaptation is requested in some cases. Since even for the most sensitive patient a low flow rate cannot cause danger, the gain needs not to be decreased by any gain down mechanism. In summary it is recommended to modify the ineffective control mechanism as follows:

EnableGainUp: 'Gain up adaptation is enabled'
FlowNotMaximum > (60) and FlowNotMinimum > (60) and not GainMaximum and not NoProgress1 and not NoProgress2

EnableGainDown: 'Gain down adaptation enabled'
FlowNotMinimum > (30) and not GainMinimum

Rules NoProgress1 and NoProgress2 are given in section 9.2. The gain change rules are then

GainUp: 'Gain up request acknowledged'
NoTransient > (180) and GainUpRequest and EnableGainUp
then do gain := 3 * gain

GainDown: 'Gain Down request acknowledged'
NoTransient > (180) and GainDownRequest and EnableGainDown
then do gain := gain / 3

Ineffective control is also detected when the MAP is in a region for a long time. It turned out from the tests that in more than 50% of the cases that rule NoProgress1 or NoProgress2 was true, the flow reached its maximum level some minutes later, without much further result. More knowledge should be acquired in order to limit the flow at a lower level than its maximum under circumstances in which the MAP shows less progress.
10 Invalid measurements

10.1 Validation of the arterial pressure

Sometimes the pressure measurement is disturbed, and in such cases the measurements do not reflect the real pressure. To prevent invalid measurements from causing unreliable control, in each heart cycle the shape of the arterial pressure curve is compared with previous heart cycles [Melissen, 1989; Zwart, 1990]. Comparison is based on the slopes, the systolic maximum and diastolic minimum, the cycle time and some other parameters; only when these parameters have physiological values and differ little from parameters of previous cycles, the heart cycle is considered valid (figure 10.1).

![Figure 10.1](image)

Valid arterial pressure measurements

For control purposes, the 5 second averaged Mean Arterial Pressure (MAP) is used as the feedback signal; only valid heart cycles contribute to this 5 second MAP average. The 5 second average MAP can be computed reliably as long as there are at least two valid cycles; otherwise no new MAP can be computed and the controller must take appropriate actions.

10.2 Actions taken in case of invalid measurements

When no valid MAP measurement is available, the controller can act in three ways: return to manual mode, keep the flow rate at a constant level, or continue control using the last valid MAP. Actually a combination of these is actions is taken. When invalid measurements occur first, the controller continues and uses the last valid MAP; when valid measurements are available again within a short period, it continues using the new valid MAP.
The longer the MAP stays invalid, the more doubtful the appropriateness of
the control is. Therefore after 60 seconds (if the MAP is close to the target;
30 seconds otherwise) the system returns to manual mode and as a result the
flow rate is frozen. Since measurement validation is performed both in
manual and automatic mode, it is not possible to return to automatic mode
unless the MAP is valid again.

Simulations show that the controller can work accurately for 2 minutes if the
MAP is close to the target, and for 1 minute otherwise (the model delay
equals 50 seconds). For safety reasons the system returns to manual after a
shorter period: there is a certain risk that the last valid MAP does not
completely reflect the pressure of that time, thus control should not continue
for a long period based on that pressure.

In the trend display a valid MAP is displayed by a red vertical line, an invalid
MAP as a red point instead (see figures 8.1 and 11.1). If the measurement is
invalid for 15 seconds, a message is displayed on the screen.

Invalid measurements have no serious consequences as long as no switch to
manual results. A return to manual does have some consequences (see
chapter 11), but these are not serious: just a return to automatic when the
measurement is valid again is adequate in most of the cases.

10.3 Final remarks about validation

During the first introduction of the system, MAP validation has been
improved [Zwart, 1990]. After that, validation worked accurately; it protects
the controller against measurements that do not reflect the true pressure.
Although in all cases invalid measurements occurred frequently, returning to
manual mode due to a long period of invalid measurements occurred only in
5 of the 30 cases, and at the correct moments.
11.1 Contents of the display

The display shows the current flow rate and the current MAP numerically, as well as their history over the last 30 minutes graphically. The upper right window shows the mode of operation, which is either manual or automatic. The lower part of the display is used for system messages and for displaying the active function keys (for clarity, in figure 11.1 all the function keys are shown). Ergonomics have guided the design of the display and gained some compliments of the medical staff.

The system starts up in manual mode, and the SNP flow rate can be adjusted manually. When the user switches to automatic mode, the SNP flow rate is adjusted automatically, so that the MAP moves to a previously defined target level (setpoint). From automatic mode, manual mode can be entered again at any time. In this chapter these two modes of operation, and the consequences of switching between them, are discussed in more detail.

![Figure 11.1](image-url)  
*Layout of the display*
11.63 The human interface

11.2 Manual mode of operation

In manual mode four actions can be performed: change the flow rate, adjust the setpoint, switch to automatic mode, and stop.

(1) Adjustment of the SNP flow rate

The display shows which three keys are used for adjustment of the flow rate. Pressing the "Flow Up" key will increase the flow rate, and pressing "Flow Down" will decrease it. The third key is "Zero Flow", which immediately shuts off the flow.

(2) Adjustment of the setpoint

The setpoint is the desired MAP level, that the controller will try to reach and maintain after switching to automatic mode. Two keys labelled "Setpoint Up" and "Setpoint Down" are used to change the setpoint. The setpoint is displayed on the screen, but not yet actively used because the system is in manual mode of operation.

(3) Switching to automatic mode of operation

After switching to automatic mode, the controller will start to adjust the SNP flow rate in order to reach and maintain the previously defined setpoint. The setpoint can be readjusted at all times.

(4) Stopping the system

Finally if the "Stop" key is pressed, the flow is shut off and the system will shut down.

Two function keys are left, which are active in manual as well as in automatic mode of operation: "To Perfusion" and "From Perfusion". These keys are used to change the measurement validation algorithm between normal validation (when an arterial pressure curve is measured) and perfusion validation (when the heart function is taken over by a machine and the pressure measurement is just a straight line). Since the knowledge of how to switch over automatically is still missing, this needed to be done manually.
11.3 Automatic mode of operation

When switching to automatic mode of operation, the controller starts with the last flow rate that was adjusted in manual mode. The user can change the setpoint, shut off the flow, return to manual and stop.

(1) Adjustment of the setpoint

Before switching to automatic mode, a setpoint has been defined; this setpoint can be changed in automatic as well as in manual mode. In automatic mode, the new setpoint is immediately active and overrules the previous one.

(2) Emergency stop

The "Zero Flow" key is an emergency stop. The flow is shut off immediately, and the system returns to manual mode.

(3) Returning to manual mode

In manual mode, the knowledge base is in a another context than in automatic mode. Some specific knowledge that has to do with control is meaningless in the manual mode of operation. As a consequence, after a return to manual, some details about the control are forgotten, but other details are stored for later use. For instance the flow rate is kept constant at the last value the controller computed, and also the setpoint stays the same, but is not active anymore. The control gain that is employed in automatic mode is maintained for future use in automatic mode.

Transient detection is still active, but in manual mode the flow rate is not influenced as in automatic mode. Switching to manual during a transient (e.g. after a down transient that results in a zero flow) allows manual adjustment of the flow. When returning to automatic mode after such an adjustment, control continues with the new flow rate, but keeps the flow rate constant until the transient is over.

In manual mode the adaptation mechanism is not active, and details about gain adaptation are forgotten. For instance, suppose that the MAP has been in a region for such a long time, that after a few more seconds a gain up adaptation would be requested. A switch to manual and back to automatic resets all Region history times, and thus adaptation is delayed until the MAP is again in a region for a long period. The SetpConst and GainConst timers are also reset.
Besides pressing the "To Manual" key, invalid measurements and pump failure (chapter 13) can also bring the system into manual mode; as long as the MAP is invalid or as long as the pump fails, the system cannot return to automatic mode.

In automatic mode, the user cannot directly stop the system; this is only possible from manual mode. When the "Stop" key is pressed, the system returns to manual; a second key press will then stop the system. Thus the "Stop" key must be pressed twice in order to prevent an accidental key press from causing an unwanted system stop, which could have serious results (starting up again takes some time, the control restarts at the minimum gain, etc). Accidentally pressing the "Stop" key instead lets the system return to manual mode, which has far less serious consequences.
In the previous description, several times phrases were used like "if the MAP exceeds ... then ... "; these phrases are the human intention, but they cannot be implemented in that way directly. Reliable decisions cannot be taken based on only one MAP measurement; more measurements must be taken into account. One way to do so is to provide some hysteresis: enter a certain state if a parameter exceeds a border value, and stay in that state until it becomes less than another border value. Decisions are taken when that state is active for some period; for regions border crossings (chapter 2) and the flow slope (chapter 4), hysteresis is used. Another method to reduce random influences such as noise is filtering; moving average filters are used at different places all through the rule base.

12.1 Moving average filters

A moving average filter implements a low pass filter. The filter is described with:

\[ x_k = A \cdot x_{k-1} + (1-A) \cdot y_k \]  \[ 8 \]

The output \( x \) is the moving average filtered value of \( y \). The filter parameter \( A \) is between 0 and 1 (generally it is close to 1) and determines the characteristics of the filter. For example when \( A \) equals 0.8 the new value of \( x \) (denoted as \( x_k \)) is determined for 80% by the old value of \( x \) (denoted as \( x_{k-1} \)) and for 20% by the measurement \( y \) (denoted as \( y_k \)).

Parameter \( A \) can be interpreted in a different way: it specifies the number of samples that play a major role for the current value of \( x \). Computation of \( A \cdot (1-A) \) results the number of input samples that determines the output for 65%. For the example above (\( A = 0.8 \)) the filter output is for 65% based on the last 4 samples and for 95% on the last 12 samples (3 times \( A \cdot (A-1) \)).

Filtering has some advantages and some disadvantages; the main reason for filtering is the reduction of meaningless signal variations. Decisions based on noisy signals are generally influenced by that noise, and thus may be incorrect; decisions based on filtered signals are more reliable. The closer the parameter \( A \) is to 1, the more fluctuations are suppressed.

The main disadvantage of filtering is the time lag between the input and the output of the filter. The filtered signal reflects the original signal of some time ago. The faster the signal changes, and the closer parameter \( A \) is to 1, the larger the difference between the filtered signal and the original signal.
A large noise reduction conflicts with a small time lag. Generally for decision taking (such as gain adaptation) noise suppression is important, and larger delays are allowed then for control purposes, where time delays should be limited and thus filtering should be preferably absent.

Moving average filters can also be used to implement a high pass filter. The difference between the input and the output of a moving average filter represents the slope of the input signal; whether the flow is constant, increasing or decreasing, can be detected that way (section 8.2).

12.2 Filters used in the knowledge base

The error signal is the difference between the setpoint and the current MAP; region border crossings are based on this signal. To prevent noise that appears in the error signal from causing unnecessary region switches, filtering is provided (note that also hysteresis is provided). The error signal can change rapidly, for instance if the control gain is high. The filter parameter A equals 0.8, and is chosen so that filter output follows the error signal closely even when the control gain is too high.

The Mean Arterial Pressure could be filtered before it is used in the P- and D-terms of the controller. Measurement noise that appears in the flow rate due to the P- and D-terms will then decrease considerably. However, generally filtering affects control stability; when the filter parameter A equals 0.7, its influence is limited and the control stability is not affected. However for safety reasons the MAP is not filtered when it is used for control purposes. The D-term is set to zero instead, which is a more powerful method to reduce noise influences.

When a transient occurs, the pre-transient flow is computed (section 8.2). A transient always starts some time before it is detected, and the control signal will have already shown some undesired reactions before the transient handling control regime is entered. The pre-transient flow is therefore based on the moving average filtered ($A = 0.93$) flow rate, in order to compensate for that undesired reaction.

Finally the flow rate is filtered ($A = 0.98$) to detect a large change of flow. When the relative difference between the flow and filtered flow exceeds a certain limit, a large MAP change may be expected (section 8.3).
During automatic control, the SNP supply is between 0 and 2 [μg/kg/min]; for a (default) 80 kg adult this is a dose of about 10 [mg/h] (in manual mode, more can be administrated). The pump rate equals the dose divided by the SNP concentration; the concentration equals 0.1 [mg/ml] and thus the pump rate is a real number between 0 and 100 [ml/h]. However, the infusion pump only accepts integer pump rates; in order to realize the flow more accurately, the remaining fraction is remembered, and added to the next pump rate. For example, if the pump rate should be 0.75 [ml/h], the resulting rate is alternately 0, 1, 1 and 1.

In the knowledge base as well as on the display, the quantity flow rate is used; the flow rate is 1/10 of the pump rate. The maximum flow rate of 10 agrees with the maximum dose of 2 [μg/kg/min]. The system uses the quantity flow rate because it agrees with the pump rate of the SNP infusion pumps that are normally used in the Catharina hospital.

Normally the pump accepts the new rate, but not in case of pump failure. Pump failure can occur due to a computer/pump communication error, an occlusion of the infusion line, air in the line, an exhausted battery, an empty infusion bag and some other causes. During pump failure, the pump either continues with the last flow, or the pump stops; in both cases the expert system reports the reason for pump failure on the screen. When the system is in automatic mode of operation and the pump failure continues for more than 30 seconds, the system returns to manual mode. Automatic control cannot continue for a longer time because a zero flow might occur. In that case the MAP will increase after a minute, and the controller will increase the flow rate, but without results because of the pump failure.

The trend display shows pump failure in a similar way as invalid measurements: the flow rate is displayed as a blue dot rather than as a solid vertical line. Pump failure did not occur during the tests.
Before automatic control tests were started, tests with manual control were performed. The anaesthetist was asked to choose a target pressure level and to adjust the flow rate manually so that the MAP reached and remained at that target level. In the background the controller was active; it also computed a flow rate. The controller computed an *advise*, that was stored for afterward inspection and comparison.

This was called *open loop control*: the controller computes an advise only, and the user performs the real control task. However the controller must necessarily assume that its advise flow is realized, and it will judge the response of the patient as being the result of the control based on the advise.

For example, suppose that the target level is lower than the MAP. The control advise will show a slowly increasing flow rate. When the user does not increase the flow rate also, the MAP will not change and the controller will increase the advise flow even more. On the other hand when the user immediately adjusts the correct flow rate, the controller will increase its advise flow somewhat, but will find the MAP at the target level after a short time, and will keep its advise constant. In both cases a large difference between control advise and the real flow rate develops, which makes a valid comparison difficult.

Since the controller is designed for a system with a large time delay, direct feedback from the patient is not necessary to keep the control advise reliable. However if the difference between the control advise and real flow is large, actions must be taken to keep the control advise reliable, especially when the setpoint is almost reached.

In principle it is possible to perform open loop control in order to advise the user. However the method through which a reliable advise is computed will in some respects differ from the standard PID control, and thus will not accurately reflect the closed loop control behavior. Therefore the control advise has never been displayed, but nevertheless these tests were valuable: the model parameters were derived from the flow adjustments during this stage, the validation algorithm was improved, and specific knowledge about transients was obtained.
During the first introduction of the system, three significant changes were made. First, the pressure validation was improved; after the improvement, the system always returned to manual mode (due to invalid measurements) at the correct moment. Second, specific knowledge about transients was acquired; after the implementation of that knowledge, transient detection was accurate, and the correct actions were taken at the right moments. Third, the D-term of the controller was set to zero because the MAP fluctuations were sometimes larger than initially expected, and the controller was re-tuned in order to keep the same robustness. Control robustness was sufficient, because the inter- and intra-patient variations that were found for the dynamic parameters were less than the worst case that the controller was designed for.

*Gain up adaptation* is necessary to obtain a sufficient control gain for less sensitive patients. In order to employ a certain control gain, the pressure must be in a region for some period. The filtering and hysteresis, which is applied for regions, eliminates the influence of insignificant pressure variations, so that the pressure indeed was long enough in a region to let the controller develop the correct gain.

If the pressure *moves fast through* two successive regions, the control gain is decreased. This adaptation mechanism did not completely perform as desired. The regions close to the target level are quite narrow, so that a relatively small drift of the pressure can cause a gain down adaptation. Gain down adaptation occurs when the pressure is in a region shorter than a certain minimum period; this minimum period should be made shorter for the narrow regions.

Control never came to oscillation; nevertheless oscillation was detected several times. The method through which the number of region border crossings is counted should be changed so that small fluctuations do not cause detection of oscillation.
If the flow rate changes quickly, the control gain is decreased. The measure of *flow rate change* is a relative measure, thus the influence of the measurement noise is larger at lower flow rates. Frequently the gain was decreased unnecessarily at low flow rates because the relative flow change was too high, which was caused by measurement noise. To reduce the influence of the noise, flow increments should be distinguished from flow decrements, and the hysteresis between high and low flow rate change should be made larger.

More knowledge should be acquired in more extensive clinical studies in order to attempt to obtain a reliable estimate of the sensitivity class of individuals patients based on other clinical information. Up to now, mainly control engineering knowledge is incorporated; gain adaptation is based on following the MAP's progress to the target level. If a proper estimate of the sensitivity would be available, the gain down adaptation mechanisms that is now active for safety reasons could possibly be prevented from unnecessarily slowing down the control. A more specific question is: can a patient's sensitivity to SNP ever significantly increase again?

*Ineffective control* was often detected because the flow rate exceeded its maximum value, and therefore gain up adaptation was disabled. However when the flow rate is not at its maximum for only a few samples, sometimes a gain up adaptation results. The implementation of this mechanism can be changed as recommended. Furthermore at low flow rates gain down adaptation should also be disabled.

Ineffective control is also detected if the pressure does not make sufficient progress. It turned out that in some cases after detection of insufficient progress, the flow reached its maximum without further result. More knowledge should be acquired in order to limit the flow rate at a lower level then its maximum when the pressure does not show progress.

In summary the *safety mechanisms* worked well; especially actions taken in case of transients were correct. However, the *control performance* was not as accurate as expected. Due to spontaneous fluctuations, the mechanism that acts on a high flow rate change sometimes ordered unnecessary gain down adaptations, which resulted in a too slow control. The *clinical evaluation* was successful; the current pressure controller is a good basis for more extensive studies in order to obtain more knowledge about blood pressure control.
References

Blom J.A. (1990)


Slate, J.B. (1980)

Zwart, R.M.P. (1990)
(222) Józwiak, L.
THE FULL-DECOMPOSITION OF SEQUENTIAL MACHINES WITH THE SEPARATE REALIZATION OF THE NEXT-STATE AND OUTPUT FUNCTIONS.

(223) Józwiak, L.
THE BIT FULL-DECOMPOSITION OF SEQUENTIAL MACHINES.

Ed. by Han Vinck.

(225) Hoeijmakers, M.J.
A POSSIBILITY TO INCORPORATE SATURATION IN THE SIMPLE, GLOBAL MODEL OF A SYNCHRONOUS MACHINE WITH RECTIFIER.

(226) Dehitya, R.P. and E.M. van Veldhuizen, L.H.Th. Bietiens
EXPERIMENTS ON INITIAL BEHAVIOUR OF CORONA GENERATED WITH ELECTRICAL PULSES SUPERIMPOSED ON DC BIAS.

(227) Basting, R.H.A.
TOWARD THE DEVELOPMENT OF AN INTELLIGENT ALARM SYSTEM IN ANESTHESIA.

(228) Hekker, J.J.
COMPUTER ANIMATED GRAPHICS AS A TEACHING TOOL FOR THE ANESTHESIA MACHINE SIMULATOR.

(229) Oostrom, J.H.M. van
INTELLIGENT ALARMS IN ANESTHESIA: An implementation.

(230) Winter, M.R.M.
DESIGN OF A UNIVERSAL PROTOCOL SUBSYSTEM ARCHITECTURE: Specification of functions and services.

(231) Schiemann, M.F.C. and H.C. Heyker, J.J.R. Kwaspers, Th.G. van de Roer
MOUNTING AND DC TO 18 GHz CHARACTERISATION OF DOUBLE BARRIER RESONANT TUNNELLING DEVICES.

(232) Sarma, A.D. and M.H.A.J. Herben
DATA ACQUISITION AND SIGNAL PROCESSING/ANALYSIS OF SCINTILATION EVENTS FOR THE OLYMPUS PROPAGATION EXPERIMENT.

(233) Nederstigt, J.A.
DESIGN AND IMPLEMENTATION OF A SECOND PROTOTYPE OF THE INTELLIGENT ALARM SYSTEM IN ANESTHESIA.

(234) Philippens, E.H.J.
DESIGNING DEBUGGING TOOLS FOR SIMPLEXYS EXPERT SYSTEMS.

(235) Ueffels, J.J.M.
A PATIENT SIMULATOR FOR ANESTHESIA TRAINING: A mechanical lung model and a physiological software model.

(236) Lammers, J.O.
KNOWLEDGE BASED ADAPTIVE BLOOD PRESSURE CONTROL: A Simplexys expert system application.

(237) Ren Qingchang
PREDICTION ERROR METHOD FOR IDENTIFICATION OF A HEAT EXCHANGER.
(238) Lamers, J.O.  
THE USE OF PETRI NET THEORY FOR SIMPLEXYS EXPERT SYSTEMS PROTOCOL CHECKING.  

(239) Wang, X.  
PRELIMINARY INVESTIGATIONS ON TACTILE PERCEPTION OF GRAPHICAL PATTERNS.  

(240) Lutgens, J.M.A.  
KNOWLEDGE BASE CORRECTNESS CHECKING FOR SIMPLEXYS EXPERT SYSTEMS.  

(241) Brinker, A.C. den  
A MEMBRANE MODEL FOR SPATIOTEMPORAL COUPLING.  