BULGARIAN AKADEMY OF SCIENCES
INTITUTE OF INFORMATION AND
COMMUNICATION TECHNOLOGIES

Radoy Strezimirov Dukovski

DECISION MAKING FOR PROCESS CONTROL MANAGEMENT OF TECHNOLOGICAL OBJECTS

ABSTRACT OF PhD THESIS

for acquiring the educational and scientific degree PhD

scientific field: 5. Technical sciences

Professional field: 5.2. Electrical engineering, electronics and automation

PhD program: 02.21.07. Automated information processing and management systems

Supervisor: acad. Vasil Stoyanov Sgurev

Sofia, 2025

Introduction

Topic relevance

Decision-making processes in the management of technological objects is a current scientific task, with a strongly expressed industrial character. The optimal management mode can lead to high profits from production and provide a comfortable working environment for personnel. The problem of managing objects with an insufficient number of control effects has been raised in the last 30 years.

In this PhD thesis, original results have been achieved related to the analysis of decision-making processes in the management of actually functioning technological facilities in the energy and chemical industries, and analytical and experimental models have been developed.

Goals and objectives of the PhD thesis

The present PhD thesis aims to develop, using modern design and automatic control systems, models of real technological objects to support the decision-making process in their management.

To achieve the goal set, the following tasks have been formulated:

1. To conduct a thorough theoretical analysis of decision-making systems and the challenges that arise in their application, presented using the means of modern theory.

2. To analyze the operation of a heat-energy technological control object - a steam generator, by developing models of decision-making systems for controlling the level of the steam-water mixture and the level in the drum.

3. To propose a decision-making system for automatic regulation of the level of the steam-water mixture in the drum during real-time operation of an industrial steam generator.

4. To design a membrane bioreactor for the production of yeast Hansenula polymoprha, by determining the necessary aeration rate to achieve a minimum deformation rate of the material in the membrane boundary layer of 0.8 [s^{-1}]. Also, determine the necessary stirring speed to achieve the desired maximum deformation rate in the cell growth chamber of 15 [s^{-1}].

5. Analyze the operation of a chemical technological control object - a cricket filter, by performing simulation modeling of the decision-making systems for its control.

6. Propose a decision-making system for real-time operation of an industrial cricket filter.

7. To analyze the operation of a chemical technological control object - a dual-fired heater, by performing simulation modeling of decision-making systems for its control.

8. To propose a decision-making system for real-time operation of an industrial double-layer heater.

Content structure

The PhD thesis is structured in an introduction, four chapters, a conclusion - a summary of the results obtained, a declaration of originality and a bibliography.

The list of publications on the PhD thesis includes five publications, of which one article in a journal with JCR-IF (Web of Science) and SJR (Scopus) - Bulgarian Chemical Communications Journal, three articles in the journal Engineering Sciences and one report included in the proceedings of the international conference - 9th IEEE International Conference on Big Data, Knowledge and Control Systems Engineering - BdKCSE'2025, which is in the IEEE Xplore Digital Library database and is referenced in Scopus. All publications are referenced and indexed in world-renowned databases of scientific information, which meets the requirements of the law on the development of academic staff in the Republic of Bulgaria.

CHAPTER ONE

THEORETICAL BASIS OF THE DECISION-MAKING PROCESS

Decision-making under uncertainty, including aircraft collision avoidance, forest fire management, and disaster response, is a highly important process. When designing automated decision support systems, it is important to consider the various sources of uncertainty in making or recommending decisions. Taking these sources into account and carefully balancing the multiple goals of the system can be a very difficult task.

These challenges will be discussed from a computational perspective, presenting the theory behind decision-making models and algorithms, and then illustrating the theory on a set of real-world problems.

1.1. Decision making using intelligent agents.

An agent is something that acts based on observations of its environment. Agents can be physical objects, such as humans or robots, or they can be non-physical objects, such as decision support systems that are fully implemented in software.

As shown in Figure 1.1, the interaction between the agent and the world follows an "observation-action" cycle [1, 10].

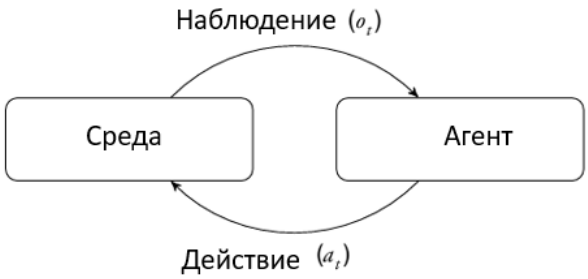


Figure 1.1 Relationships between the environment and the agent

There are many different methods for designing decision-making agents. Depending on the application, some methods may be more appropriate than others. The most straightforward method for designing a decision-making agent is to anticipate all the different scenarios in which the agent may find itself, and then explicitly program the agent to do what is desired. The explicit programming approach can work well for simple problems, but it places a great

deal of burden on the designer to provide a complete strategy. Various agent programming languages and frameworks have been proposed to facilitate their programming [2, 7, 8].

1.2. Decision making using probabilistic models.

Rational decision-making requires reasoning about one's own uncertainty and goals. Probabilistic model solutions are based on representing uncertainty as a probability distribution. Real-world problems require reasoning about distributions over many different variables. The literature presents tools for constructing such models and how to use them to make inferences [11, 16].

Uncertainty can also arise from practical and theoretical limitations in our ability to predict future events. For example, predicting exactly how a human operator will respond to advice from a decision support system would require, among other things, a detailed model of how the human brain works. Even the paths of satellites can be difficult to predict. Although Newtonian physics allows for highly accurate predictions of satellite trajectories, spontaneous failures in attitude control engines can lead to large deviations from the nominal path.

A limitation of using the Gaussian distribution is that it is unimodal, meaning that there is a point in the distribution at which the density increases monotonically on one side and decreases monotonically on the other. There are various ways to represent continuous distributions that are multimodal. One way is to mix a collection of unimodal distributions [4, 11, 16].

1.3. Decision making using a Bayesian network.

A Bayesian network is a compact representation of a joint distribution. The structure of the network is represented as a graph consisting of nodes and directed edges. Each node corresponds to a random variable. Directed edges connect pairs of nodes, with cycles in the graph prohibited.

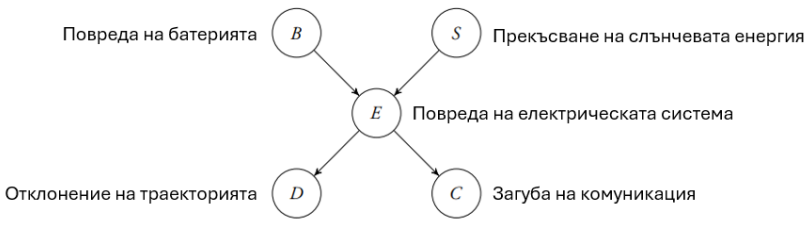


Figure 1.2 Example structure of a Bayesian network

Figure 1.2 shows an example Bayesian network for a satellite observation problem involving five binary variables. Fortunately, battery failure and solar panel failures are rare, although solar panel failures are slightly more likely than battery failures [1, 4, 13, 16, 18].

The reason a Bayesian network can represent joint distributions with fewer independent parameters is because of the conditional independence assumptions encoded in its graphical structure. If the conditional independence assumptions made by the Bayesian network are invalid, then we risk not modeling the joint distribution correctly.

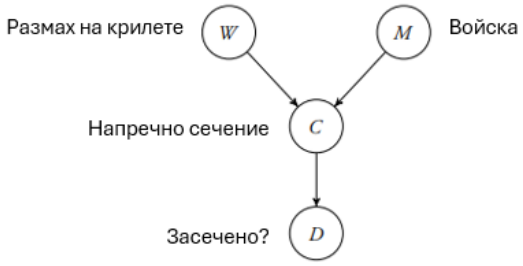


Figure 1.4 Example of a Hybrid Bayesian Network

The examples in this chapter so far have involved binary variables, but Bayesian networks can contain a mixture of discrete and continuous variables. Bayesian networks with discrete and continuous variables are often called hybrid Bayesian networks. Figure 1.4 shows an example hybrid Bayesian network representing the relationship between aircraft characteristics, their radar cross section, and the radar's ability to detect a target.

1.4. Decision making using temporal models.

A temporal model represents how a set of variables evolves over time. A simple temporal model is a Markov chain, where the state over time t is marked as S_t. A Markov chain can represent, for example, the position and velocity of an airplane over time.

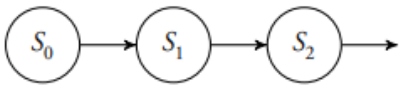


Figure 1.6 Markov chain

Figure 1.6 shows the structure of a Bayesian network, which is a Markov chain. Only the first three states are shown in the figure, but the Markov chain can continue indefinitely. The initial distribution is given by $P(S_0)$. The conditional distribution $P(S_t \mid S_{t-1})$ is often called a state transition model. If the state transition distribution does not change with t, then the model is called stationary [22, 23].

The state in a Markov chain does not have to be a scalar. For example, if we want to model the random behavior of an airplane over time, the state can be a vector $s = (h, \dot{h})$, where h is the altitude of the plane, $\dot{h}$ is the linear velocity. The initial distribution $P(S_0)$ can be represented by a multivariate Gaussian distribution parameterized by a mean, vector μ and covariance matrix Σ with density:

$$p(s) = N(x \mid \mu, \Sigma), \tag{1.17}$$

where $N(s \mid \mu,\Sigma)$ is a k-dimensional generalization of the Gaussian distribution in equation (1.6):

$$N(s \mid \mu, \Sigma) = \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(s-\mu)^T\Sigma^{-1}(s-\mu)\right) \tag{1.18}$$

A simple probabilistic model often used in classification tasks is the simplified Bayesian model, which has the structure shown in Figure 1.10. An equivalent, but more compact, representation is shown in Figure 1.11. The value i = 1 : n at the bottom of the box indicates that the index i in the subscript of the variable is repeated from 1 to n [16, 17, 19, 22]. In this Bayesian model, the class C is the query variable, and the observed features O_1,. .. ,O_n are the variables of the proofs. For compactness in this book we will use colon notation sometimes in the indices. For example, $O_{1:n}$ is a compact way to record O_1 ,. .. ,O_n. The Bayesian model is called simple because it assumes conditional independence between the evidence variables for which the class is given.

Filtering in linear dynamical systems that assume that the state transition distributions and observations are linear Gaussian can actually be performed exactly. If b_{t-1} is represented as a normal distribution, then it can be shown that integrating row 5 leads to the posterior b_t is Gaussian. The Kalman filter is a well-known filter for linear dynamic systems that simply updates the mean and covariance of b_t in an appropriate manner.

By the definition of conditional probability, we know that

$$P(b^1|d^1, c^1) = \frac{P(b^1,d^1,c^1)}{P(d^1,c^1)} \tag{1.34}$$

1.5. Parametric and nonparametric learning.

The Dirichlet distribution is a generalization of the beta distribution and can be used to estimate the parameters of a discrete distribution. Suppose that X is a discrete random variable that takes on integer values from 1 to n. We define the parameters of the distribution as $\theta_{1:n}$,

where $P(x^i) = \theta_i$. Of course, the parameters must sum to 1, and therefore only the first n − 1 parameters are independent. The Dirichlet distribution can be used to represent both the prior and the posterior distribution and is parameterized by $\alpha_{1:n}$.

The density is:

$$Dir(\theta_{1:n}|\alpha_{1:n}) = \frac{\Gamma(\alpha_0)}{\prod_{i=1}^{n}\Gamma(\alpha_i)}\prod_{i=1}^{n}\theta_i^{\alpha_i-1} \tag{1.72}$$

where α_0 is used to denote the summation of parameters $\alpha_{1:n}$.

If $n = 2$, then it is easy to see that equation (1.72) is equivalent to the beta distribution. It is common to use a uniform prior distribution, where all Dirichlet parameters $\alpha_{1:n}$ are set to 1. A symmetric Dirichlet distribution is one in which all parameters are identical. As with the beta distribution, the parameters in the Dirichlet are often called pseudocounts.

1.6. Structured learning.

In the previous sections, it was assumed that the structure of the Bayesian network is known a priori. This section discusses methods for learning the structure from data. The maximum likelihood approach to the structure of a Bayesian network involves finding the graph structure G that maximizes *P(G|D)*, where D represents the available data. First, the Bayesian network score is calculated based on *P(G|D)*, and then searching the network space for the network with the highest score. Similar to inference in Bayesian networks, it can be shown that for general graphs and input data, learning the structure of a Bayesian network is NP-hard..

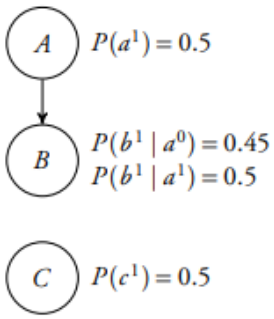


Figure 1.20 Example Bayesian Network

To illustrate how Bayesian estimation balances model complexity, consider the simple Bayesian network in Figure 1.20. The value of A has little effect on the value of B, and C is independent of the other variables. We sample this “true” model to generate data D and then

try to learn the structure of the model. There are 25 possible network structures involving three models:

• The true model with 1 + 2 + 1 = 4 independent parameters,

• The fully connected model A → B, A → C , B → C with 1 + 2 + 4 = 7 independent parameters,

• The completely unrelated model with 1 + 1 + 1 = 3 independent parameters.

It is common for K2 to impose an upper bound on the number of parents for each node to reduce the required computations. The original K2 algorithm assumes that the prior Dirichlet parameters $\alpha_{ijk} = 1$ for all i, j and k, but any a priori can be used in principle [5, 16, 18, 21, 24, 25]. A Bayesian network encodes a set of conditional independence assumptions. An important observation to make when trying to learn the structure of a Bayesian network is that two different graphs can encode the same independence assumptions. As a simple example, consider a network with two variables A → B..

Instead of searching the space of directed acyclic graphs, we can search the space of Markov equivalence classes represented by partially directed graphs. The space of Markov equivalence classes is smaller than the space of directed acyclic graphs, and therefore the search can be performed more efficiently. The Bayesian result is defined for directed acyclic graphs. To evaluate a partially directed graph, we need to generate a member of its Markov equivalence class and compute its result.

1.7. Conclusions on the first chapter.

1. A review of the theoretical foundations of the decision-making process is conducted, and well-known methods and algorithms for decision-making with wide application in practice are presented.

2. It is shown that a particular challenge in the decision-making process can be uncertainty, which arises from incomplete information or the inability to predict future events due to the presence of practical or theoretical limitations. Proper accounting for uncertainty is important in building robust decision-making systems.

3. The analysis shows that when using network models, the structure of the network encodes assumptions about conditional independence, and that probabilistic inferences can be effective if the structure of the network is used.

4. It is shown that Bayesian networks compactly represent the distributions of variables and are suitable for flexible representation for encoding a wide variety of models, and that Bayesian methods and maximum likelihood methods can be used to derive the parameters and structure of the model.

CHAPTER TWO

DECISION-MAKING SYSTEMS

The previous chapter focused on uncertainty, including how to construct probabilistic models of uncertainty and use them to make inferences. This chapter focuses on how to make rational decisions based on a probabilistic model and a utility function. We will focus on single-step decisions, reserving the discussion of sequential decision-making problems for the next chapter. This chapter begins by introducing the basics of utility theory and showing how it forms the basis for rational decision-making under uncertainty. We then show how concepts from utility theory can be incorporated into the probabilistic graphical models introduced in the previous chapter to form so-called decision networks. Since many important problems related to the decision-making process involve interaction with other agents.

2.1. Utility theory.

The previous chapter discussed the need to compare the degree of conviction regarding two different statements. In this paragraph, we compare the degree of desirability of two different outcomes. We state our preferences using the following operators:

• A≻B, if we prefer A to B.

• A∼B, if we are indifferent between A and B.

• A≥B, if we prefer A to B or are indifferent.

Just as beliefs can be subjective, so can preferences. In addition to comparing events, our preference operators can be used to compare preferences over uncertain outcomes. A lottery is a set of probabilities associated with a set of outcomes. For example, if $S_{1:n}$ is a set of results and $p_{1:n}$ are the associated probabilities, then the lottery involving these outcomes and probabilities is written:

$$[S_1: p_1; \dots; S_n: p_n] \tag{2.1}$$

This section discusses how the existence of a realistically valued utility function arises from a set of preference assumptions. From this utility function, it is possible to define what it means to make rational decisions under uncertainty [26, 27].

When building a decision-making system or a decision-support system, it is often useful to derive a utility function from a person or group of people. This approach is called utility extraction or preference extraction. One way to do this is to fix the utility of the worst outcome $S_\perp$ over 0 and the best result S_T to 1. While the utilities of outcomes are bounded, we can transform and scale the utilities without changing our preferences. If we want to determine the utility of an outcome S, then we determine the probability p, so that $S \sim [S_\top : p; S_\perp : 1 - p]$. It follows that:

$$U(s) = p \tag{2.7}$$

2.2. Decision-making networks.

We can extend the concept of Bayesian networks to decision networks that involve actions and utilities. Decision networks are composed of three types of nodes:

• A random node corresponds to a random variable (denoted by a circle);

• A random node corresponds to any decision that needs to be made (denoted by a square);

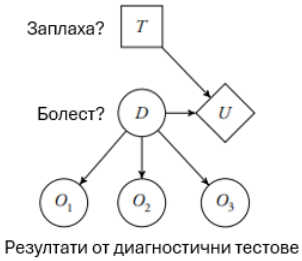
• A utility node corresponds to an additive component of utility (denoted by a diamond).

There are three types of directed edges:

• A conditional edge ends at a random node and indicates that the uncertainty in that random node is conditioned by the values of all its parents;

• An information edge ends at a decision node and indicates that the decision associated with that node is made with knowledge of the values of its parents;

• A functional edge ends at a utility node and indicates that the utility node is determined by the outcomes of its parents.



T	D	$U(T,D)$
0	0	0
0	1	−10
1	0	−1
1	1	−1

Figure 2.3 Decision network for diagnostic test and utility function

Decision networks are sometimes called influence diagrams. Like Bayesian networks, decision networks cannot have cycles. Representing a decision problem as a decision network allows us to take advantage of the structure of the problem when computing the optimal solution in terms of a utility function.

The optimal solution is the one that, when initialized in the decision network, provides the highest expected utility. Over the years, various methods have been developed to more efficiently evaluate decision networks. One method involves removing action and contingency nodes from decision networks if they have no successors, as defined by conditional, informational, or functional edges.

Decision networks are a powerful framework for building decision support systems. So far, we have discussed the key elements in building decision networks and how to use them to make optimal one-time decisions. We will briefly discuss the procedure for creating a decision network. The first step is to identify the space of possible actions.

2.3. Game theory.

This chapter focuses on rational decision making with an assumed model of the environment. The methods presented in this chapter so far can, of course, be applied to environments containing other agents, as long as the probabilistic model captures the effects of the other agents' behavior. However, there are many cases in which we do not have a probabilistic model of the other agents' behavior, but we do have a model of their utilities. Decision making in such situations is the subject of game theory, which will be discussed briefly in this section.

When building a decision-making system that must interact with humans, information about Nash equilibriums is not always useful. People often do not play a Nash equilibrium strategy. First, it may not be clear which equilibrium to adopt if there are many different equilibria in the game. For games with only one equilibrium, it may be difficult for a person to compute the Nash equilibrium due to cognitive limitations. Even if agents can compute the Nash equilibrium, they may doubt that their opponents can perform this computation.

A field known as behavioral game theory aims to model agents. There are many different behavioral models, but the level-k logit model, sometimes called the quantum level-k model, has become popular recently and tends to work well in practice. The level-k logit model captures the assumption that people are:

• more likely to make mistakes when those mistakes are less costly, and

• limited in the number of steps of strategic foresight (as in "I think you think I think...").

The model is defined by:

• *accuracy parameter* $\lambda \geq 0$, which controls sensitivity to differences in utility (0 is insensitive); and

• *a depth parameter* $k > 0$ that controls the depth of rationality.

2.4. Basic concepts in decision theory.

Decision making is the process of choosing the best course of action from a set of possible alternatives. This process can be intuitive, based on experience and knowledge, or structured, based on analysis and data. Examples of decision making include:

- Choosing a travel route to avoid traffic.
- Deciding to invest in certain stocks.
- Planning a production schedule to maximize output.

Optimization is the scientific discipline that deals with finding the best (optimal) solution to a problem, taking into account certain constraints. It seeks to maximize or minimize a given objective function [31, 36, 46, 47, 48].

- **Objective function**: This is a mathematical expression that quantitatively measures the quality of the solution (e.g., profit, cost, time, distance).
- **Constraints**: These are the conditions or rules that must be met (e.g., available budget, resources, time frames).

Optimization is a tool used to make rational decisions, especially in complex situations where intuition is not enough. Any solution that tries to be the “best” by some criterion is actually an attempt at optimization.

Imagine you are the manager of a logistics company. You have to decide on the routes of your trucks.

- **Problem:** How to deliver goods to all customers with minimal cost?
- **Goal:** Minimize fuel and time costs.
- **Constraints:** Each truck has limited capacity, workers have working hours, and deliveries must be on time.

In this case, decision making is the selection of specific routes. Optimization is the process of finding the most efficient routes that respect all constraints and minimize costs.

Therefore, optimization provides a methodology and framework for systematic decision making that ensures that the chosen option is not just good, but the best possible under the given conditions. It transforms decision making from a matter of judgment to a scientific process supported by data [36, 41, 44, 47, 48].

Three of the main information processing abilities are attention, cognitive abilities, and memory. Information is perceived by a person, and the ability to pay attention allows a portion of this perceived information to be processed. The portion of information that is paid attention to is then cognitively processed by the person, using memory as an aid to make sense of the information.

Another way that expert users simplify the situation to ensure that cognitive processing is manageable is by using the structure of the context to constrain the possible outcomes of the situation. Structure is defined as shared knowledge about the functioning of a system that inherently constrains the evolution of the system state [70, 71].

2.5. Trust and transparency of decision logic.

The first problem that must be overcome when implementing a decision support system is that users must be able to trust the information and recommendations that the system provides [66, 72, 73]. The definition of trust is selectively adapted from Muir [65, 74].

Trust is the expectation of a system member that there will be reliability and technically competent performance from another system member, which is related to, but not necessarily isomorphic to, objective measures of these qualities.

“Reliability” refers to consistent and predictable performance. “Technically competent” refers to the capable performance of functions assigned to that member, within certain limits and constraints. The fact that trust is not directly representative of these qualities indicates that there is a difference based on human perception of these attributes, which may be biased.

“Misuse” of a decision support system can take several forms. A frequently discussed form is when designers over-automate a function so that the user’s role is primarily a supervisory, controlling, or monitoring role.

While often innovative and useful, these unintended uses extend the boundaries of the technical competence of the system, whose use in these areas was not foreseen by the designers [76].

When the decision support system has a clear answer, it should provide:

1. *Information or advice regarding the decision.* When designing a decision support algorithm, designers often consider information in terms of the problem to be solved, not necessarily in the terms that the user requires to interpret that information. Once the information and/or advice is generated by the decision support system, the designer must take care to translate it into contextual terminology and form..

2. *Timely information or advice.* Similarly, when the system provides information or advice, it must be provided at the appropriate time. Some decisions must be made hours before implementation, while others are time-critical and the user has only seconds to react appropriately.

3. *Reliable information or recommendation.* As discussed in the section above, the reliability of a system is critical to the user's ability to develop trust in the system. Reliability in many circumstances must be assessed before the system is implemented..

4. *Reviewability of information or recommendation reliability.* In order for designers to properly calibrate trust in a decision support system, information about the reliability of the system must be regularly provided to the user.

5. *The rationale for the decision support recommendation.* Some decision support systems help decision-making in the longer term by providing additional information or a decision support recommendation.

Progressive decision making is defined as the periodic review of information and the adaptation of strategic decisions to updated information as it becomes available. This type of decision making involves making sound strategic decisions that anticipate the need for subsequent adaptation, ensuring that multiple tactical options are available later in the operational time horizon, and reviewing decisions made as information updates.

2.6. A systems view of implementation.

Figure 2.11 Systems development process

Implementing an effective decision support system is an iterative process (Figure 2.11). Ideally, the design of a decision support system should begin with field observations and analysis of operational data to understand the operation and the role of the users.

The designer's understanding of the operation, users, and constraints forms the operational model. From this operational model, the need for decision support can arise and requirements are defined.

The design and development of the decision support system then takes place, resulting in a procedure, human-machine interface/graphical interface, and training.

The implementation of the decision support system impacts operations, and its impact can be measured through analysis of operational data and field observations.

When designing a decision support tool, there is often a basic idea for the system that is translated into at least a functional requirement to support decision making.

In the broad sense of a decision support system, the decision support provided may include a technical algorithm supported by a visualization for the user, a new procedure, new or modified training, or some combination of these three. Effective implementation of the decision support (tool, procedure, training) results in measurable benefits.

2.7. Conclusions on the second chapter.

1. It is shown that the process of rational decision-making combines probability theory and utility theory, and we can build rational decision-making systems based on utility functions derived from people.

2. The existence of a utility function follows from the constraints on rational preferences. A rational decision is one that maximizes expected utility. People are not always rational.

3. It has been found that in making rational decisions with an assumed model of the environment, Game Theory, involving multiple agents, can be very useful. This is appropriate when there is no probabilistic model of the behavior of agents, but there is a model of their utilities.

4. The concept of trust is considered as a key component, both for the design of algorithms and their acceptability, and a systems approach is presented as a framework for developing an effective iterative design method that includes training and systematic measurements in addition to the design work.

5. A concerted effort to truly understand the problem, the context, and the experts involved is needed before designing a decision support system. This understanding will make the difference between a system that provides true decision support and one that either hinders the process or is not used at all.

CHAPTER THREE

DECISION MAKING IN THE MANAGEMENT OF A HEATING TECHNOLOGICAL OBJECT

A significant part of modern automatic control systems ensure stable operation of technological facilities in the energy sector. Such facilities have a significant share in the economy of each country and are of strategic importance for the country's national security.

This chapter analyzes the operation of a key facility in the electricity generation system, namely the steam generator. The object of the study is the processes of steam generation and the operation of the equipment used. The principles of operation and different types of steam boilers are briefly considered. The process and stages of steam production are described and the components in the production system are mentioned.

A mathematical model of the drum is created and various control strategies used in the automation of the process are briefly presented. The control processes and decision-making stages related to the operation of the technological facility are schematically presented.

3.1. Principle of operation of a steam generator.

The basic principle of operation of a steam generator (boiler) is very simple and easy to understand. Essentially, a boiler is a closed vessel with water. Fuel is burned in the furnace chamber and hot gases are produced.

These hot gases come into contact with the so-called downpipes, where the heat of the hot gases is transferred to the water and as a result steam begins to form in the boiler. The steam is then reheated several times until the required parameters are reached and fed to the turbine or to a collecting manifold.

When the steam pressure decreases, the water level rises because the steam bubbles expand (swell). Conversely, when the steam pressure increases, the water level decreases due to the compression (contraction) of the steam bubbles [85].

This makes the object non-minimally phase (Figure 3.2), which leads to control problems when regulating the level in the steam generator drum. The “barriers” in Figure 3.6

are separators that separate the dry steam from the steam-water mixture and provide a guided path for the dry steam.

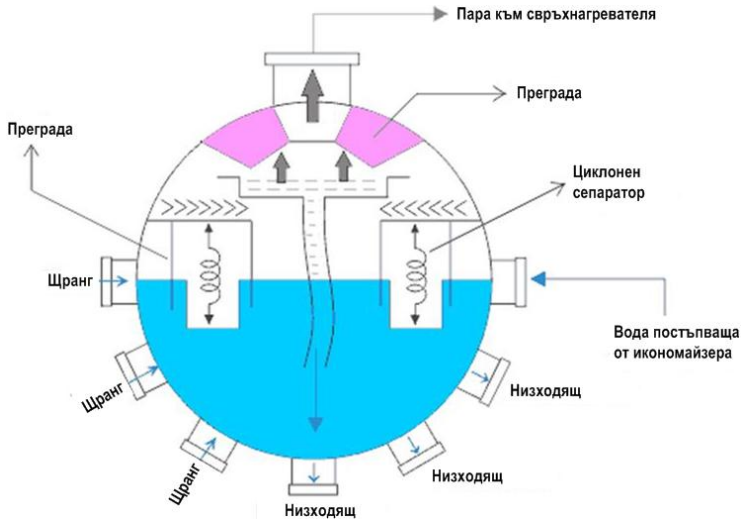


Figure 3.6 Schematic of the steam drum of a water tube boiler..

3.2. Mathematical modeling of a steam generator.

In the process of modeling a steam boiler, the goal is to represent the phenomenon of contraction and swelling: the increase or decrease in the boiler's steam consumption, which leads to a change in level, exactly the opposite of what is intuitively expected. If there is a sudden increase in steam consumption, the level temporarily increases due to the momentary drop in steam pressure caused by the increase in consumption.

This initial drop in pressure leads to greater evaporation and an increase in the size of the steam bubbles (the swelling phenomenon), which causes the level in the drum to rise until the heat flow increases enough to restore the pressure value.

3.3. Steam-water mixture level control in a steam generator.

In the scheme of Figure 3.18, the level is stabilized by a single-loop system that varies the amount of blowing air F_B depending on the current value of the difference $\Delta\theta = \theta_2 - \theta_1$ between the temperatures in the vapor and liquid spaces. When the foam reaches the sensing element at the top of the column, the temperature θ_2 starts to drop and the regulator reduces the air flow F_B, as the height of the foam layer is reduced.

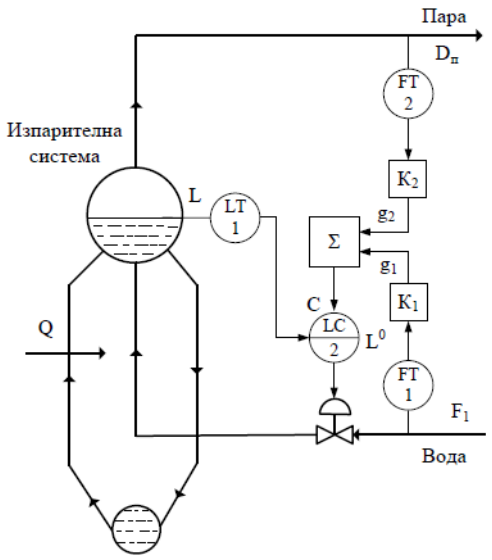


Figure 3.18 Combined control scheme

The level control in the separators of the boilers (Figure 3.14a) is carried out according to the combined control scheme shown in Figure 3.18. In addition to the main signal via level feedback L in the level regulator $C(p)$ two compensating signals are also supplied g_1 and g_2. The compensator K_1 is designed as a non-inertial unit and takes into account random changes in water consumption F_1, due to pressure changes in the supply line. The compensator K_2 is also implemented as a proportional unit and takes into account changes in steam consumption D_n.

3.4. Level control in the steam generator drum.

A combined drum level control system (Figure 3.21) is ideal for cases where the boiler plant consists of multiple boilers and multiple feed water pumps or the pressure drop across the control valve is variable. The drum level, the steam flow rate and the feed water flow rate must be measured. In order for the system to be in equilibrium (the level does not change), the steam flow rate must be equal to the feed water flow rate. If there is a difference between the two flows, the level will either increase or decrease continuously (the object has integrating properties). The difference between the steam flow rate and the water flow rate is fed as a correction value to the drum level setpoint. In this way, any disturbance in either flow rate is reflected in the control action before the level has changed. The efficiency of the three-pulse control system under transient conditions makes it very useful for general industrial and domestic steam generator applications. It handles loads with wide and rapid rates of change [101, 102].

3.5. Simulation modeling of steam generator control systems.

For the purposes of the study, a gas boiler (Figure 3.25) F-1011 with four burners, with forced and induced draft and flue gas recovery and with a capacity of 70000 Lbs/Hr was selected. It has a feedwater economizer that uses flue gas exhaust [110].

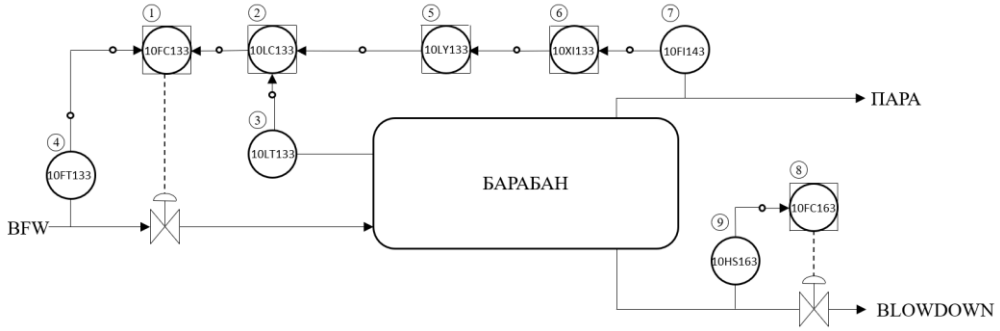


Figure 3.25 Design scheme for drum level control in boiler F - 1011

(1 - feed water control regulator; 2 - drum level regulator; 3 - level transmitter; 4 - water flow transmitter; 5 - ratio regulator; 6 - mode indicator of the corresponding regulators; 7 - steam flow; 8 - water flow continuous purges; 9 - manual switch)

In the MATLAB programming environment, using Simulink, a model (Figure 3.26) was built for level control in the F-1011 boiler, with the data for Level block 1, 2 and Level integrator being real and taken from a refinery in El Paso, USA..

We have a non-minimum phase object with the following transfer function:

$$W(s) = \frac{30s-1}{10s+1}\ \frac{-2}{5s+1}\ \frac{1}{30s} \tag{3.39}$$

It can be seen that we have an integrating unit with a 15 second integration time and a non-minimum phase part, which can be approximated with a pure delay of 45 seconds. This makes the control object too complex, compensation for the non-minimum phase is necessary, which can be largely achieved by using the three-element control scheme discussed in the literature review. The equilibrium point of the exemplary simulation, at which the level is zero with 70% steam consumption and 70% water consumption. At the entrance to the object with the shown transfer function, the difference between the water consumption and the steam consumption enters. Due to the integrating properties of the object, if this difference is positive, the level will increase constantly, and if it is negative, the level will decrease constantly.

To compensate for the swelling and shrinkage of the level when changing the steam consumption or water consumption, the difference between the steam and water consumption is added to the setpoint with a weighting coefficient, which is part of the control scheme setup.

As discussed in the literature review, when increasing the steam consumption, the level increases at the first moment and in the given model it will start to fall after 45 seconds. This fact is sufficient for inadequate response of the regulator in a conventional regulation scheme, without compensation of non-minimum phase.

In the present scheme, the difference between steam and water, weighted through the coefficients Gain_S and Gain_S1, is fed to the setpoint (Setpoint block). Thus, the increase in steam consumption instantly increases the setpoint, which leads to an increase in water consumption, despite the fact that the level increases, not decreases.

In other words, compensation is achieved for the increase in level due to the pressure drop in the drum and the expansion of the steam bubbles. In a conventional scheme, this fictitious higher level leads to the stoppage of the feed water supply at a time when more water is really required.

The controller used is PI. We do not use PID because there is a valve at the output of the controller that is not able to handle an output containing a differential component and may be damaged.

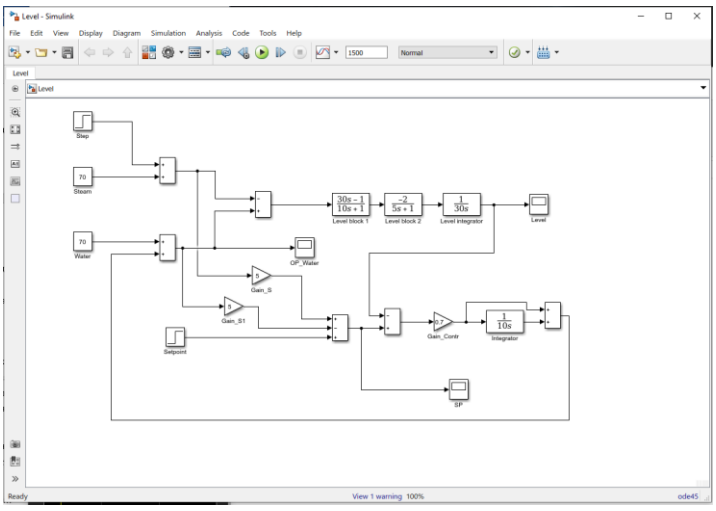


Figure 3.26 Model for level control in boiler F-1011 (MATLAB Simulink)

Figure 3.29 shows a control module 10LC133 for controlling the level in the drum, where LEADLAG, DATAACQ, ADD, MUL, SUB, NUMERIC and PIDA function blocks are involved..

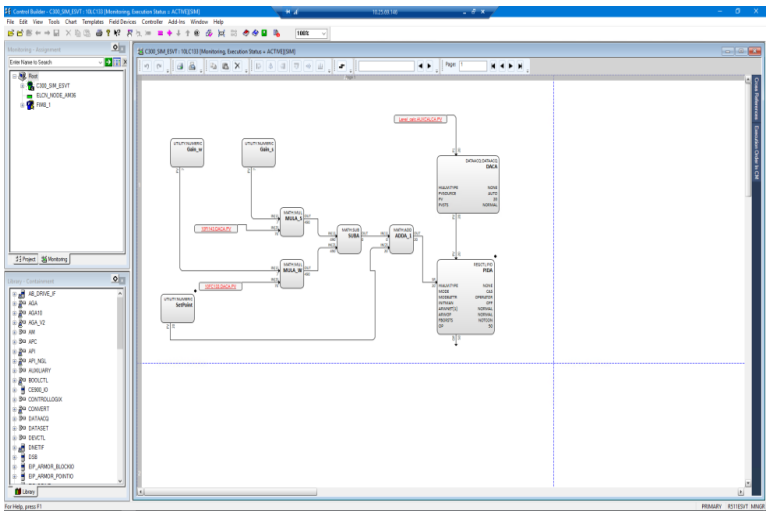


Figure 3.29 Control module 10LC133

3.6. Experimental results from the analysis of a steam generator.

In the plant control system (original configuration) boiler F-1011, which is presented in Figure 3.39, a PI controller with a gain of 0.85 and an integration time of 7 minutes is used. The reason for this is that before the modernization, the control scheme was single-loop, considered in the literature review. It does not allow sharp adjustments of the controller.

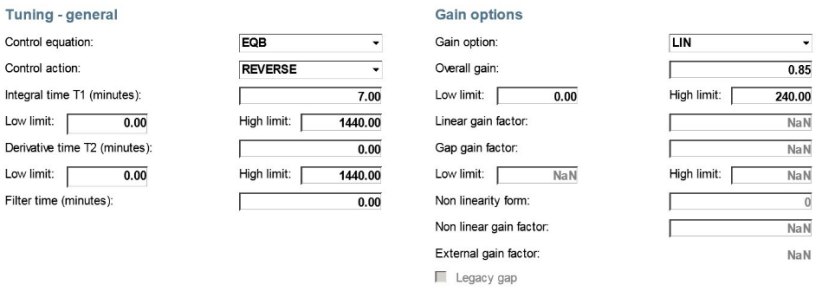
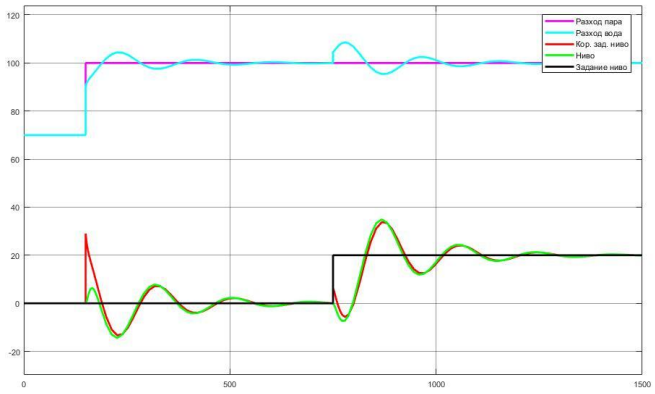


Figure 3.39 Original settings of control module 10LC133

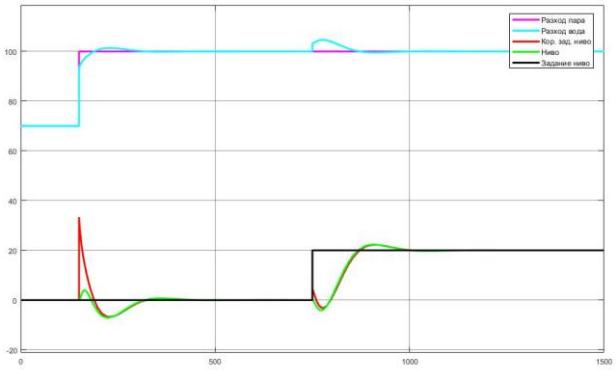
For the purpose of the study, the three-component scheme was used, as already mentioned with a PI controller, due to the presence of a slow valve and with weighting coefficients for the difference between the steam and water flow rates supplied to the setpoint.

The weighting coefficients for the difference between steam and water were determined experimentally. In the simulations conducted, they ranged from 1 to 7. The controller settings are a gain coefficient: k=0.7 and integration time t=10 seconds. We see a significant difference in the integration time compared to that of the plant. Figure 3.40 presents the results obtained.

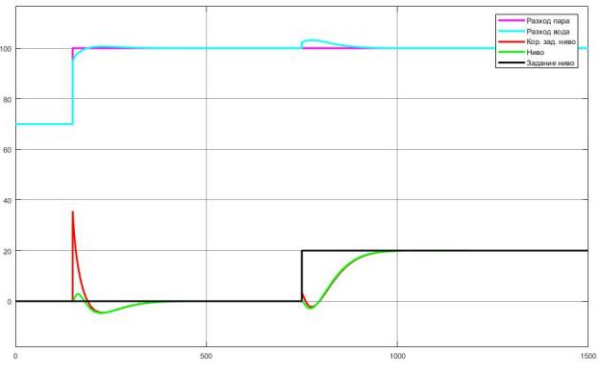
The best results were obtained with weighting factors k=7. It can be seen that the correction to the setpoint is sharpest, which leads to a stronger correction in the water flow before the level has deviated significantly.



а)



б)



в)

Figure 3.40 Level adjustment at different compensation coefficients

а) - at *k*=3; б) - at *k*=5; в) - at *k*=7

Figure 3.41 shows the open-loop scheme (the controller output does not affect the feedwater flow). The scheme is used to illustrate the open system under steam and water disturbances. The results are shown below in the text and compared with those obtained in EPKS.

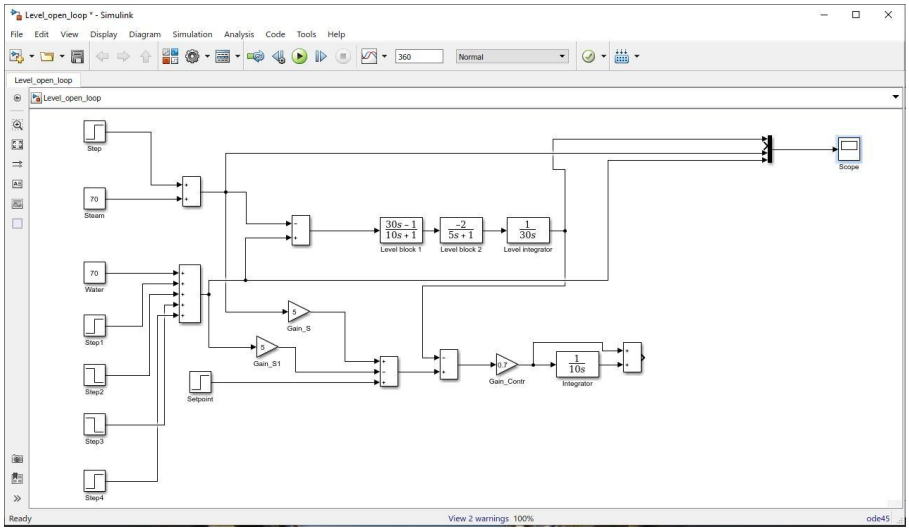


Figure 3.41 Open-loop level control model

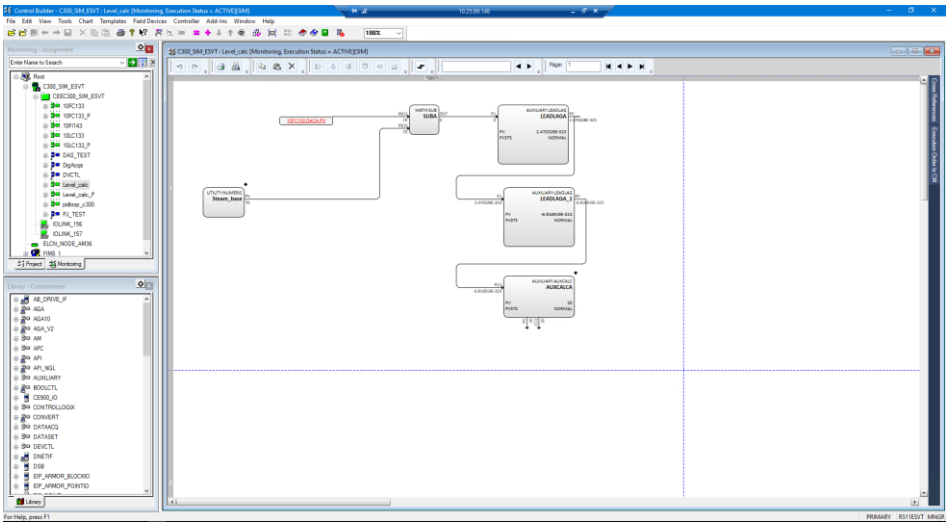


Figure 3.45 Level_calc control module

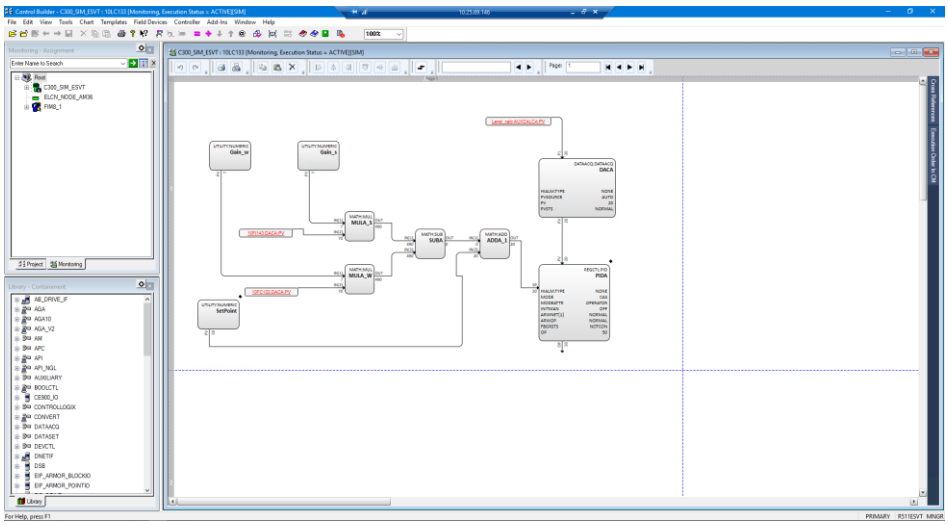


Figure 3.46 Control module 10LC133

The 10LC133 control module, which is responsible for regulating the level in the drum, is configured with the weighting factors and controller settings obtained in MATLAB. MONITORING mode is shown in Figures 3.45 and 3.46 for the Level_calc and 10LC133 control modules, in which the level in the drum is calculated and regulated.

Figure 3.51 shows the behavior of the open-loop and closed-loop control object implemented in Honeywell's Experion PKS software environment. The figure is divided into three parts: the first part shows the behavior in open loop with a working water flow controller and a change in its setpoint. The change in level as a result of these disturbances is shown. The non-minimum phase and integrating nature of the object is clearly visible.

The second part shows a setpoint disturbance in a closed loop. The level drop is seen as the feedwater flow rate increases and the new setpoint is reached, at which the water flow rate again becomes equal to the steam flow rate.

The third part shows a steam flow rate disturbance (the steam flow rate decreases stepwise). The sharp correction added to the level setpoint is seen, causing the immediate decrease in water flow rate, whose value equals the steam flow rate when the level is established at its previous value [120, 122].

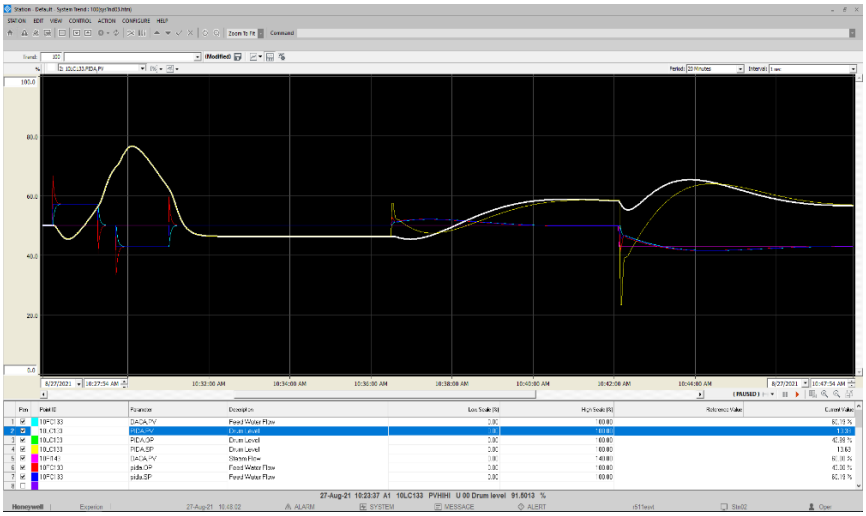


Figure 3.51 Behavior of the control object in open and closed loop, implemented in the Experion Process Knowledge System programming environment

After creating the level control model in the MATLAB programming environment and implementing it in the Experion PKS programming environment, experiments were conducted to compare the behavior of the model used in the two systems. The graphs in Figures 3.52 and 3.53 show the level change in an open loop.

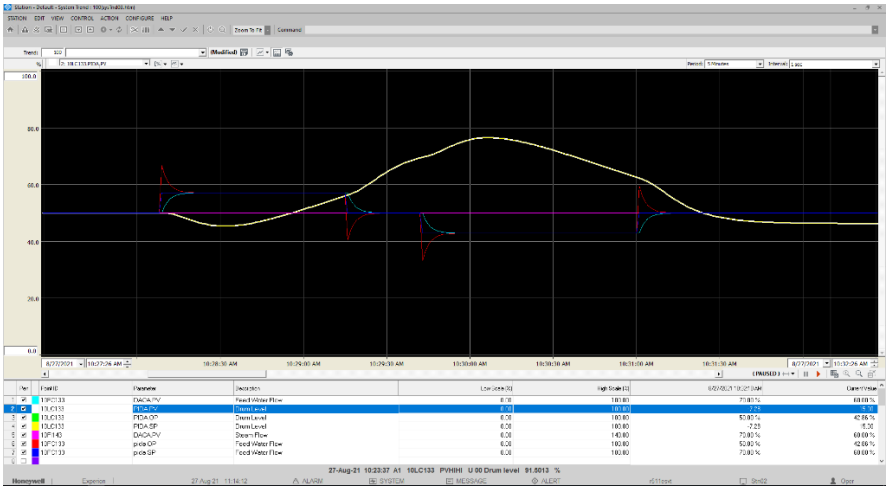


Figure 3.52 Open Loop Drum Level Behavior with Water Disturbance (Experion Process Knowledge System)

a) 10FC133 PV (feed water) - turquoise line; b) 10FI143 PV (steam quantity) - purple line; c) 10LC133 SP (drum level setpoint matching the actual level in open loop operation) - yellow line; d) 10LC133 PV (actual drum level) - white line

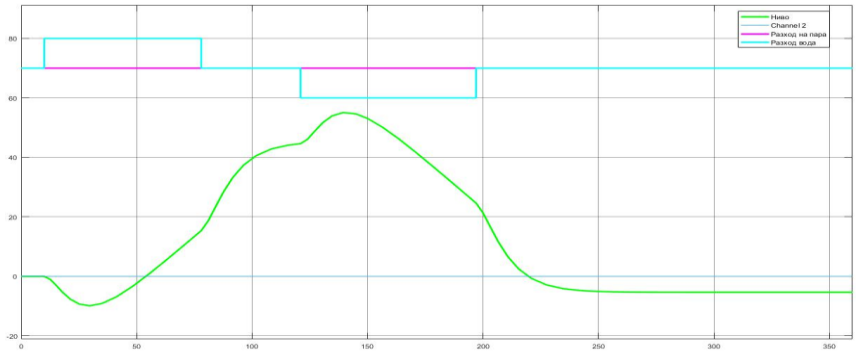


Figure 3.53 Open-loop drum level behavior under water disturbance (MATLAB)

a) 10FC133 PV (feed water) - turquoise line; b) 10FI143 PV (steam quantity) - purple line; c) 10LC133 SP (drum level setpoint matching the actual level in open-loop operation) - red line; d) 10LC133 PV (actual drum level) - green line

The simulations performed show that in MATLAB the change in feed water is an instantaneous process. In the Experion PKS environment, the amount of water changes a little slower, since a valve is also simulated. A valve is not simulated in MATLAB, since the simulation there was conducted to verify the idea and operability of the proposed scheme, which is implemented in full scale in Experion PKS.

The results obtained from the analysis performed prove the operability of the control scheme in real conditions with the model of the control object set in this way.

3.7. Conclusions on chapter three.

This chapter discusses the types of steam generators and their operating principle. The mathematical description, transfer functions and transient characteristics of a non-minimally phase object, such as the level in the steam generator, are presented. The following conclusions can be drawn from the conducted research:

1. An analysis of the processes occurring in a steam-water mixture for the purposes of level control is made and the features of the steam-water mixture as a regulated variable are presented. The results of the studied systems for automatic control at different gain coefficients of a static disturbance compensator are presented.

2. Mathematical dependencies describing the dynamic characteristics of the processes occurring in the steam generator are presented and simulation schemes for real-time level control in the MATLAB Simulink and Experion Process Knowledge System environments are developed to obtain a mathematical model of a real steam generator.

3. The proposed system for automatic regulation of the level of the steam-water mixture in the drum with the Honeywell Experion Process Knowledge System decision-making tools, during the operation of a steam generator at the El Paso refinery in the USA, increases its productivity.

CHAPTER FOUR

DECISION MAKING IN MANAGEMENT OF CHEMICAL TECHNOLOGICAL FACILITIES

This chapter will present results from the analysis of the decision-making process in the management of various chemical technological facilities.

4.1. Membrane bioreactor control.

Membrane technologies are increasingly used in biotechnological production and allow the separation of components contained in the liquid phase of the bioreactor according to their size, which can vary from a few angstroms to a few millimeters. Of particular interest are membrane bioreactors with a submerged membrane module, in which conditions are created for achieving high product quality with reduced energy consumption and a reduction in the amounts of solvents used. These advantages lead to lower operating costs. The optimal operation of the membrane bioreactor is related to: the position of the membrane module in the reactor; the gas/liquid ratio; the transmembrane pressure; the creation of sufficient shear stresses that favor the friction of the membrane surface.

Submerged membrane reactors and aeration of the type of bubble or airlift columns with an integrated membrane module use the favorable influence of the two-phase gas-liquid flow on the shear stresses around the membrane. Increasing the shear stress near the membrane is considered one of the most effective ways to control fouling. Of course, care should be taken to avoid possible deactivation of shear-sensitive components. It is important to investigate the tangential stress field and its homogeneity near the membrane surface, for which systematic information is lacking..

4.2. Design of a membrane bioreactor.

For the purpose of the study, based on the literature review, a bioreactor with aeration and stirring and a submerged membrane module was selected, shown in Figure 4.15. This bioreactor is made up of two chambers (1-outer chamber and 2-inner chamber), with the substrate necessary for cell growth entering chamber 1 through the lower opening (9), and the solution outlet (10). The substrate passes through the membrane surface (5) into chamber 2, in

which cell growth takes place, the bottom and cover of the membrane module (6). Stirrers (4) are installed in it, which are driven by an electric motor (13). A toroidal-shaped bubbler (3) is placed in the lower part of chamber 1, through which air is introduced for mixing the materials and cleaning the membrane module. The air necessary for aeration is introduced through the supply pipe (11), released onto the free surface of the solution in chamber 1 and exits through a ventilation hole (12). The membrane module can be dismantled by first dismantling the drive motor (13) and the gearbox (14) and removing the fastening elements of the membrane module (8) from the membrane block bed (7), shaft (15).

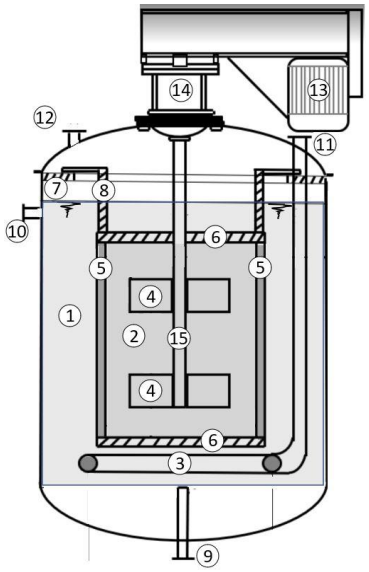


Figure 4.15 Submerged membrane module bioreactor with aeration and agitation

The required volume and height of the membrane bioreactor were determined by a numerical experiment in the MATLAB programming environment, so as to reach the desired cell throughput of 6 [g/h]. To determine the height, the ratio H/d=1.5 from the standard geometry for a laboratory reactor was taken, and the diameter was calculated based on the volume obtained from the experiment.

Cell growth in chamber two, as well as the amount of substrate in both chambers, was calculated using the ode23 function in the MATLAB programming environment, which allows for the calculation of up to three differential equations. The equations used are as follows:

- substart in 1st chamber: $\frac{dS_1}{dt}V_1 = \dot{v}S_0 - \dot{v}S_1 - Per(S_1 - S_2)$ (4.16)

- substart in 2nd chamber: $\frac{dS_2}{dt}V_{12} = Per(S_1 - S_2) - \frac{1}{Yx/s}\mu X V_2$ (4.17)

-cells: $\frac{dX}{dt}V_2 = \mu X V_2$ (4.18)

For the parameters of this model we use experimental data from Figure 4.15 and Figure 4.16 (cell growth): kinetics of Mono with μ_{max}= 0.5 [h^{-1}]; Ks=0.12 [g/l]; $Y_{x/s}$=0.52; initial substrate concentration S_0= 2 [g/l]; flow rate in 1st chamber $\dot{v}$= 2[l/h]= 0,00056 [kg/s]; membrane data Per=0.52 [dm/h] и A=5.29 [dm^2].

From the results obtained, it was found that the required productivity of 6 [g/h] is achieved by a volume of the membrane module 1.4 [dm^3] (Figure 4.24). The minimum aeration rate was determined by means of a numerical experiment (computer simulation) implemented with ANSYS, version 16.0. The present experimental work was implemented at the Center for Mathematical Modeling and Computer Simulation at UCTM with the ANSYS CFX product, for which UCTM has a permanent license.

The computer simulation was carried out on the working volume of the reactor, shown in Figure 4.17.

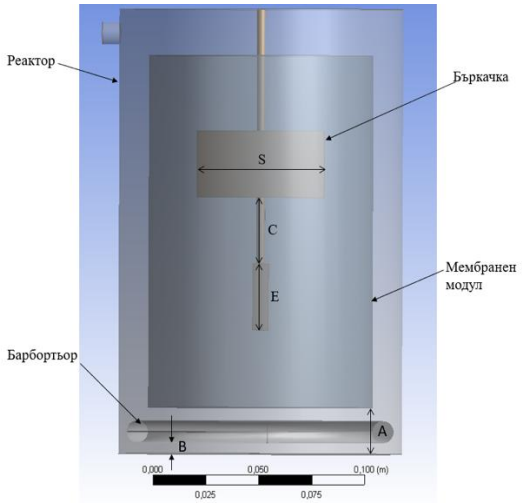


Figure 4.17 Schematic representation of the experimental bubble reactor

4.3. Simulation analysis of a membrane bioreactor.

Initial conditions of the numerical study:

- amount of air supplied by the bubbler equal to 0.05 volume fractions, which for the specific dimensions of the bubbler corresponds to 100 holes creating bubbles with a diameter of 6 [mm];

- amount of the mixture, which includes two solutions - Mixture and Kletki, at the inlet is equal to 0.95 volume fractions;

- for the gas outlet, the Degassing function is used, which is provided by the ANSYS CFX software product;

- equations 4.15-4.17 are set for the calculation of cell growth in the membrane module;

- water concentration is set with Constant.

The experiment was conducted at an air velocity varying in the range from 0.1 [m/s] to 0.8 [m/s] through 0.1 [m/s].

After each experiment, the average angular deformation rate along the membrane surface on the side of chamber 1 was calculated using the calculator built into the system. Additional cutting planes were constructed to monitor process characteristics, such as flow rate and reactor pressure..

The experiment to determine the maximum stirring speed was conducted under the same conditions as the experiment to determine the minimum aeration speed. The observations were concentrated inside chamber 2, where the cell growth was taking place.

The main parameter in this experiment was the angular speed of the stirrers, which was changed in the range from 5 [rev/min] to 35 [rev/min] in 5 [rev/min] increments. To monitor the hydrodynamic characteristics of the process, the calculator of the software system and additional planes were used..

Numerical experiments were conducted in the MATLAB programming environment to size the operating parameters of the membrane module and the entire reactor, with the volume of the membrane module varying from 0.55 [dm^3] to 1.6 [dm^3] with a step of 0.1 [dm^3].

A computer simulation was conducted with different reactor volumes to track cell growth and the results are presented in Figure 4.21.

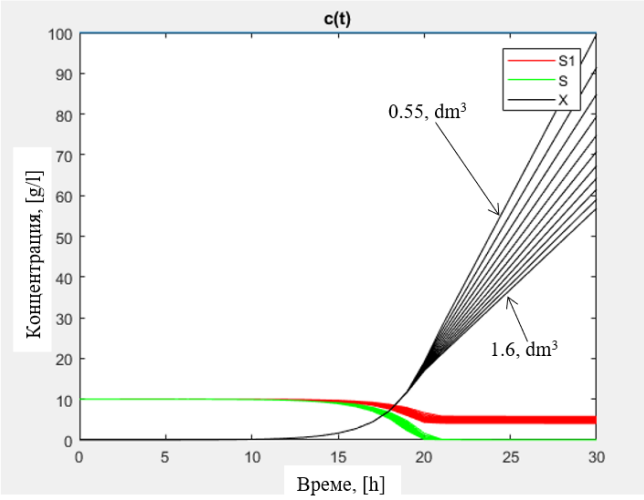


Figure 4.21 Cell growth

The graph shows that with increasing reactor volume, the growth rate of Hansenula polymorpha decreases (the slope of the graph lines decreases). The highest cell concentration is at a membrane module volume of 0.55 [dm^3], however, the productivity relative to the reactor volume (Figure 4.22) at this size is too low, which is explained by the smaller volume of the membrane module.

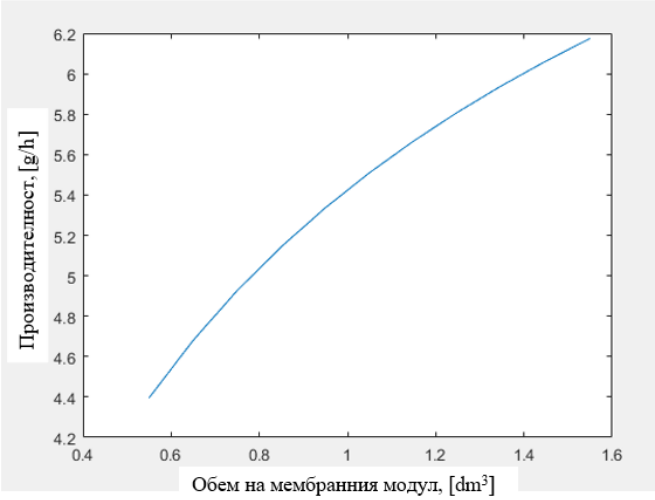


Figure 4.22 Productivity curve

Therefore, a new approximation is made, this time the working volume of the membrane starts from 1.2 [dm^3] to 1.6 [dm^3] with a step of 0.02 [dm^3]. A new result is obtained (Figure 4.23), in which the cell concentration is lower compared to the results obtained from the previous Figure 4.22, also due to the smaller step the number of curves showing cell growth increases, but at the expense of this the productivity increases, as can be seen from the graph in Figure 4.24.

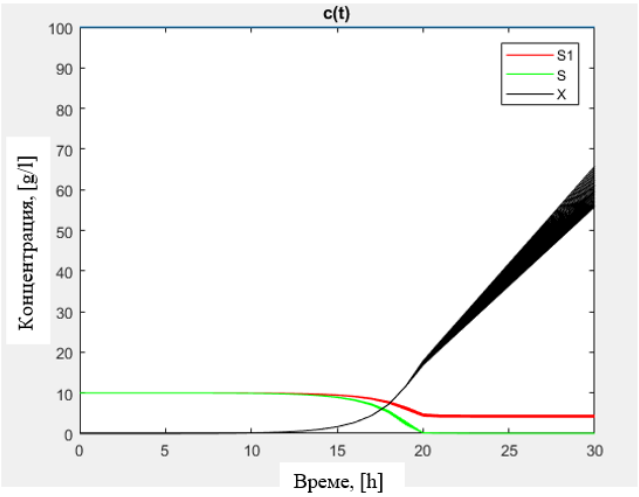


Figure 4.23 Cell growth at initial membrane size 1.2[dm^3]

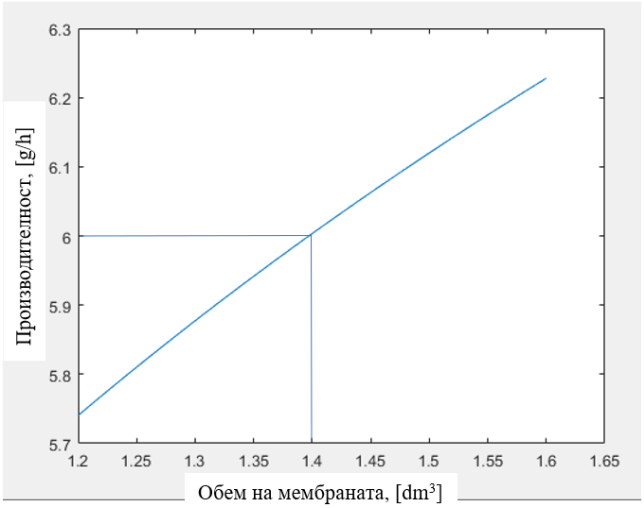


Figure 4.24 Performance curve for membrane size between 1.2 [dm^3] and 1.6 [dm^3]

From the graph shown in Figure 4.24 it is noted that the set productivity of 6 [g/h] is achieved at a volume of the membrane module of 1.4 [dm^3]. The two chambers of the reactor have the same volume, therefore the inner and outer chambers are 1.4 [dm^3] each and for the volume of the entire reactor we get 2.8 [dm^3] [146].

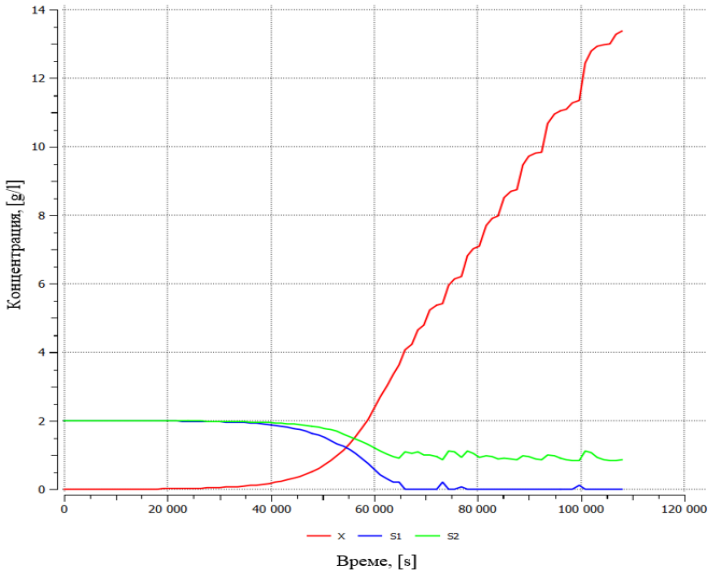


Figure 4.25 Cell growth (obtained with ANSYS CFX)

S2 (in green) - substrate concentration at the reactor outlet; S1 (in blue) - substrate concentration required for cell growth; X (in black) - cell concentration

After determining the dimensions and operating parameters of the reactor and the membrane module with MATLAB, computer simulations of cell growth in the selected reactor

were carried out in the ANSYS programming environment. The graphs in Figure 4.25 and Figure 4.26 show the change in the cell concentration in the inner chamber of the bioreactor.

The results shown in the above figures are at different dimensions of the time axis, as in MATLAB time is presented in hours, while in ANSYS CFX it is in seconds.

Also, the graph obtained with ANSYS CFX shows the presence of disturbances. These disturbances appear after the 18th hour (65000 seconds). In general, the two graphs are identical, which can be considered as a verification of the used model for computer simulation.

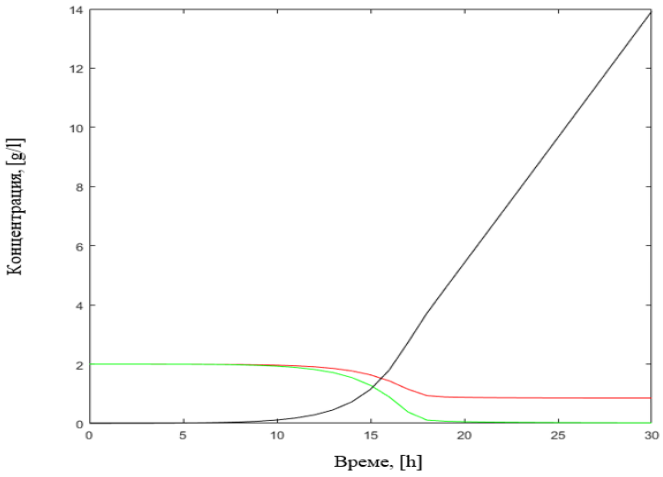


Figure 4.26 Cell growth (obtained with MATLAB)

S2 (in red) - substrate concentration at the reactor outlet; S1 (in green) - substrate concentration required for cell growth; X (in black) - cell concentration

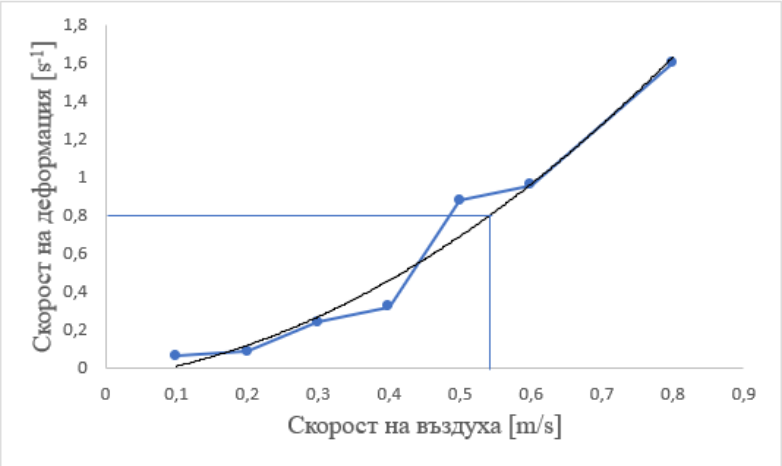


Figure 4.27 Strain rate in the boundary layer

Aeration has a significant effect on the shear rate, by increasing the degree of aeration, more intense turbulence is achieved inside the bubble column. In addition, aeration contributes to the higher efficiency of the membrane module by reducing the formation of sediment on the membrane surface.

The results presented in Figure 4.27 show the dependence of the deformation rate at the boundary layer on the outer side of the membrane. In the ANSYS CFX programming environment, the deformation rate in the boundary layer was calculated at different air inlet speeds of 0.1 [m/s] till 0.8 [m/s].

From the graph in Figure 4.27 it is visible that with increasing air velocity the deformation rate also increases. On the one hand this tendency prevents contamination of the membrane module, but it can have an adverse effect on the cells developing in the reactor. For the set minimum deformation rate of 0.8 [s^{-1}], the graph shows a minimum speed of the bubbling air of 0.55 [m/s].

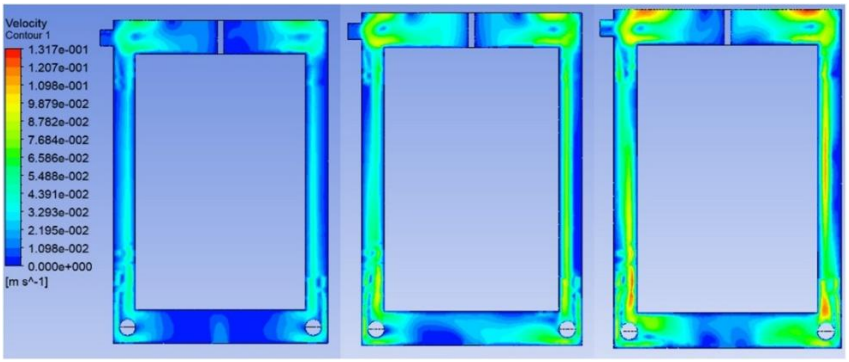


Figure 4.28 Shear rate in the outer chamber

(a) - at aeration rate 0.2[m/s]; (b) - at aeration rate 0.4[m/s]; (c) - at aeration rate 0.6[m/s]

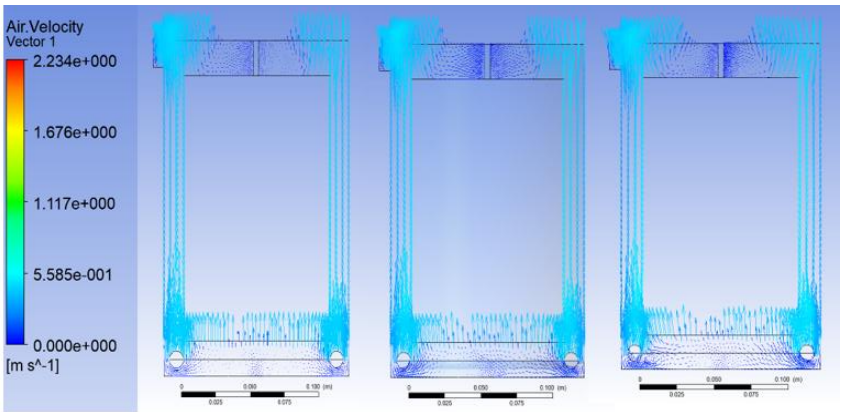


Figure 4.29 Shear rate in the outer chamber (vector representation)

(a) - at aeration speed 0.2[m/s]; (b) - at aeration speed 0.4[m/s]; (c) - at aeration speed 0.6[m/s]

The ANSYS CFX environment allows the obtained numerical results for the deformation rate in the outer chamber of the reactor to be represented vectorially and in the form of colored contours, shown in Figure 4.28 and Figure 4.29.

The following Figure 4.30 shows a close-up of the vector direction of fluid movement.

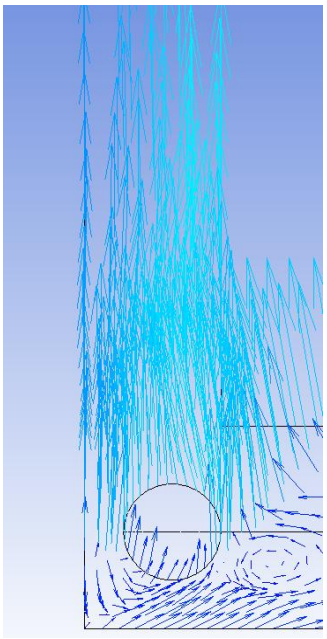


Figure 4.30 Direction of fluid movement in the outer chamber (vector representation)

From the above figure it is clear that at higher aeration rates greater turbulence is achieved around the membrane module. Figure 4.31 presents the results for the shear rate distribution in the inner chamber of the reactor, induced by the stirrer at different rotation speeds from 5 [rev/min] to 30 [rev/min].

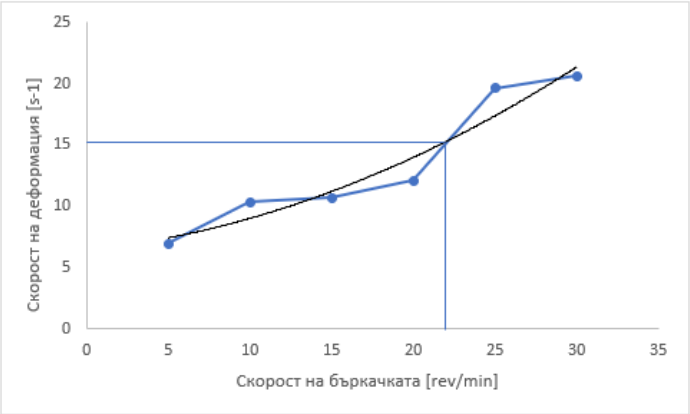


Figure 4.31 Deformation rate induced by the stirrer

To reach the set maximum deformation rate of 15 [s^{-1}], it is determined from the graph that the maximum speed of the stirrer should be 22 [rev/min]. In the following figures, Figure 4.32 and Figure 4.33, the deformation rate at different stirring speeds in the inner chamber of the reactor is represented by colored contours and vector.

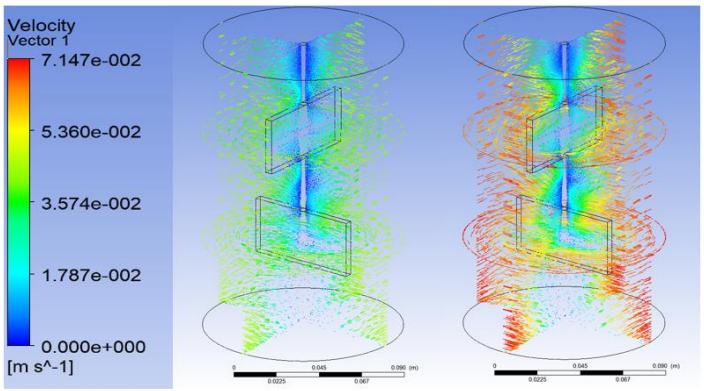


Figure 4.32 Strain rate in the inner chamber
(c) - at stirring speed 20 [rev/min]; (d) - at stirring speed 25 [rev/min]

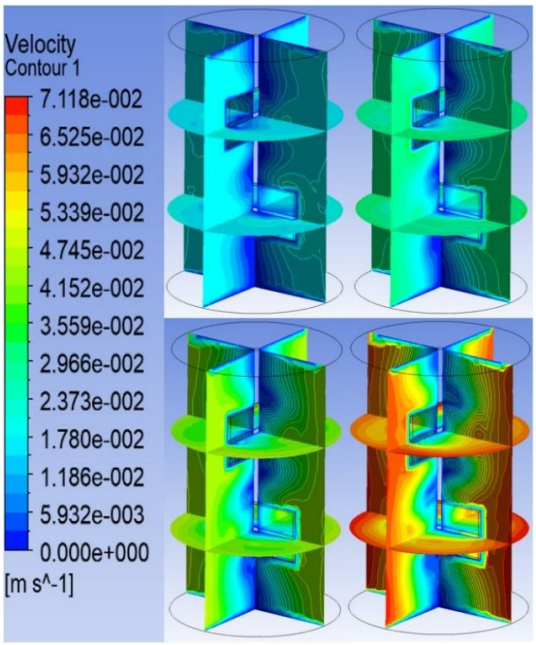


Figure 4.33 Strain rate in the inner chamber
(a) - at stirring speed 10[rev/min]; (b) - at stirring speed 15[rev/min]; (c) - at stirring speed 20[rev/min]; (d) - at stirring speed 25[rev/min]

It can be seen from Figure 4.32 and Figure 4.32 that with increasing the speed of the stirrer, the deformation rate also increases, which leads to better mixing, but can also lead to the destruction of living cells in the inner chamber of the reactor.

The deformation rate presented in Figure (d) at the speed of rotation of the stirrer 25 [rev/min] is most clearly expressed, with the highest deformation rate reached being colored in red and the lowest in blue [146]. In Figure 4.34, the direction of rotation of the stirrers is represented vectorially.

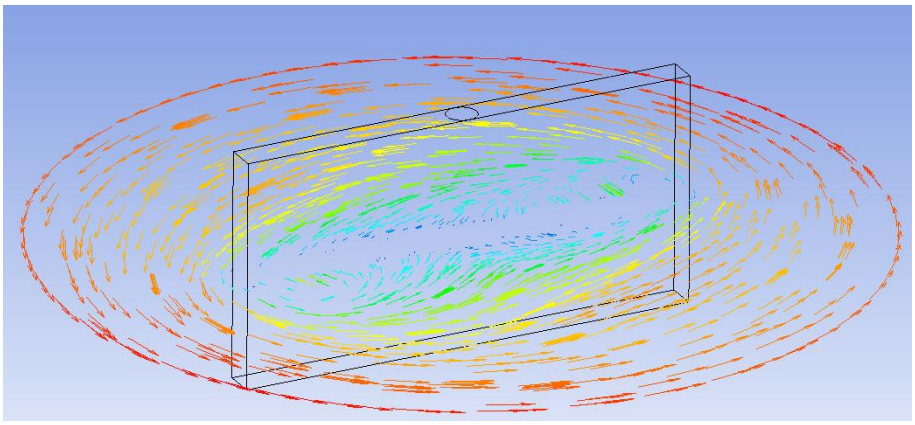


Figure 4.34 Direction of stirring (vector representation)

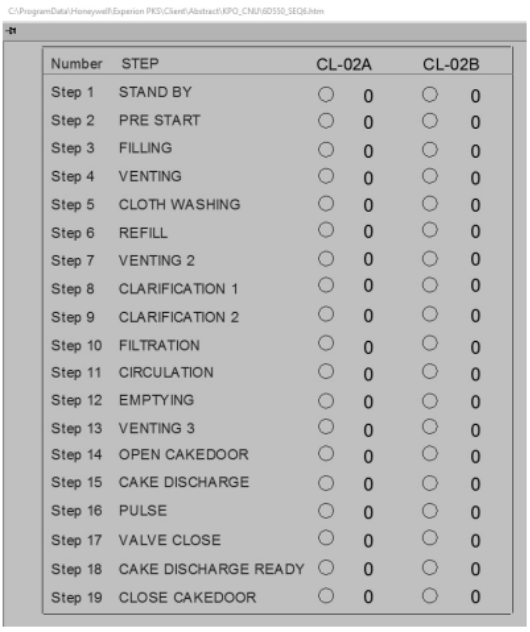
4.4. Cricket filter management.

The Cricket filter shown in Figure 4.35 was developed by amafilter® Filtration group® and was introduced in 1990 and has since been installed in over 1000 industrial applications worldwide. It has become an established pressure filter, designed to combine the advantages of pressure leaf filters and cartridge/candle filters while avoiding their disadvantages. It is particularly suitable for separating fine solids from liquids, widely used in applications such as edible oil, gelatin, cocoa butter, sugar, sweeteners, oleochemicals, mining and amine cleaning.

4.5. Simulation analysis of cricket filter.

The above-mentioned cricket filter is implemented at the Karachaganak Petroleum Operating B.V. (KPO) refinery in Kazakhstan. KPO. The cricket filter control strategy was developed in Honeywell Experion Process Knowledge System (PKS),

The station allows the operator to see all active alarms, which can be viewed in the alarm list. Figure 4.41 shows the steps in controlling the cricket filter with the currently active display in place. This display allows the operator to see the actual step, but does not provide any other process information.



Number	STEP	CL-02A	CL-02B
Step 1	STAND BY	0	0
Step 2	PRE START	0	0
Step 3	FILLING	0	0
Step 4	VENTING	0	0
Step 5	CLOTH WASHING	0	0
Step 6	REFILL	0	0
Step 7	VENTING 2	0	0
Step 8	CLARIFICATION 1	0	0
Step 9	CLARIFICATION 2	0	0
Step 10	FILTRATION	0	0
Step 11	CIRCULATION	0	0
Step 12	EMPTYING	0	0
Step 13	VENTING 3	0	0
Step 14	OPEN CAKEDOOR	0	0
Step 15	CAKE DISCHARGE	0	0
Step 16	PULSE	0	0
Step 17	VALVE CLOSE	0	0
Step 18	CAKE DISCHARGE READY	0	0
Step 19	CLOSE CAKEDOOR	0	0

Figure 4.41 Steps to implement the cricket filter

4.6. Dual fired heater control.

Dual-fired heaters, also known as dual-fired process heaters, are advanced thermal units capable of operating on two fuel sources, typically natural gas and fuel oil. This dual-fuel flexibility increases energy security, operational reliability, and profitability in industries such as oil extraction, petrochemicals, and power generation.

4.7. Simulation analysis of a double-layer heater.

The control strategy for the double-layer heater was developed in the Honeywell Experion Process Knowledge System (PKS).

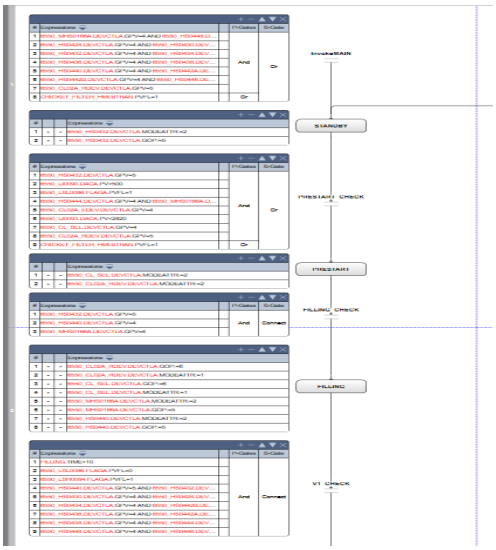


Figure 4.44 Sequence Control module

For the dual-fire heater system, due to the number of steps that had to be performed, it was done with a sequential control module (Figure 4.44). The HMI library also provides an SCM form (Figure 4.50) where the steps can be seen, as well as the commands that are executed during the step.

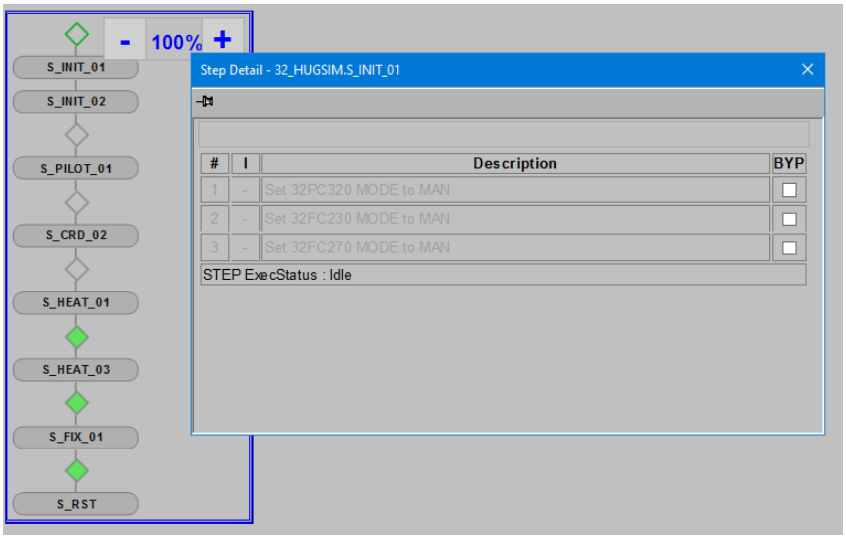
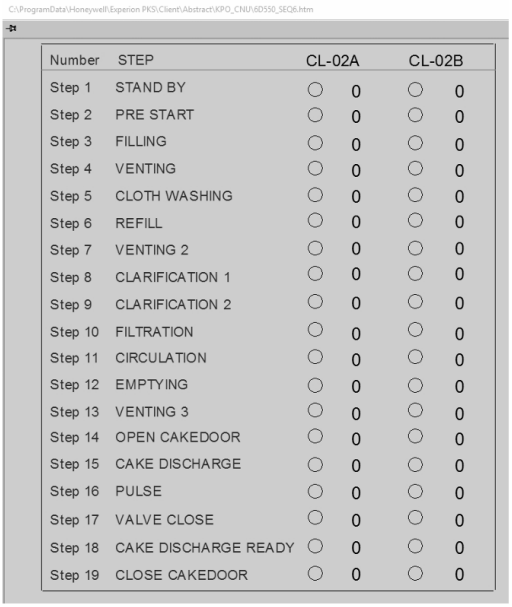


Figure 4.50 Honeywell solution pack SCM shape



Number	STEP	CL-02A		CL-02B	
Step 1	STAND BY		0		0
Step 2	PRE START		0		0
Step 3	FILLING		0		0
Step 4	VENTING		0		0
Step 5	CLOTH WASHING		0		0
Step 6	REFILL		0		0
Step 7	VENTING 2		0		0
Step 8	CLARIFICATION 1		0		0
Step 9	CLARIFICATION 2		0		0
Step 10	FILTRATION		0		0
Step 11	CIRCULATION		0		0
Step 12	EMPTYING		0		0
Step 13	VENTING 3		0		0
Step 14	OPEN CAKEDOOR		0		0
Step 15	CAKE DISCHARGE		0		0
Step 16	PULSE		0		0
Step 17	VALVE CLOSE		0		0
Step 18	CAKE DISCHARGE READY		0		0
Step 19	CLOSE CAKEDOOR		0		0

Figure 4.51 Popup Display

From here, the operator can easily understand the process and can determine if a condition is not met and from there decide whether it is necessary to override it or not, without having to open the configuration studio.

Karachaganak Petroleum Operating B.V. (KPO) previously used a pop-up display (see Figure 4.51) before adopting the SCM format. Although the display is still active on site and

shows the current steps, its usefulness is minimal: it only shows the actual step and does not provide the operator with additional data. However, this simplicity offers a critical advantage: it does not degrade system performance. In contrast, the standard form of SCM is very data-intensive and can cause delays or crashes in the Experion station, especially when multiple SCMs are present on one display [117-120].

4.8. Conclusions on chapter four.

From the conducted research and the obtained results of the analysis in the fourth chapter, the following conclusions can be drawn:

1. The MATLAB working environment allows numerical solution of the mathematical model from ordinary differential equations describing the material balance by cells and substrate. The results obtained with the program code allow calculation and sizing of a bioreactor at a given productivity.

2. The ANSYS CFX program environment allows for description of the hydrodynamic picture in the bioreactor and around the membrane module. The created computer model can be verified by comparing with results obtained from calculations with MATLAB.

3. At different reduced gas velocity (0.4 [m/s] and 0.6 [m/s]) and the same other conditions (location of the membrane module relative to the bubbler), the deformation rate increases at a higher reduced gas velocity. It has also been found that to achieve the specified minimum deformation rate of 0.8 [s^{-1}], the minimum speed of the bubbling air needs to be 0.55 [m/s].

4. When the stirring speed is changed, the deformation rate in the inner chamber of the bioreactor increases, which can adversely affect cell growth. To achieve the specified maximum deformation rate of 15 [s^{-1}], the maximum speed of the stirrer needs to be 22 [rev/min].

5. It is shown how to control a cricket filter with a dry sludge outlet, manufactured by Amafilter® Filtration group®, using the Honeywell Experion Process Knowledge System environment. By improving Sequence Control and thanks to the HMI Display Builder, it is easier for users to understand the system and thus increase its productivity. The Sequence Control Module allows users to see in real time what is happening with the system and using the different Handlers, every possible scenario can be controlled.

6. The proposed system for controlling technological objects with the help of Honeywell Experion Process Knowledge System for decision-making in the operation of a cricket filter for the Karachaganak Petroleum Operating B.V. (KPO) refinery in Kazakhstan increases the productivity of the system.

7. The control of a double-layer heater using the Honeywell Experion Process Knowledge System environment is shown. By improving the sequence management and thanks to the use of HMI Display Builder and Electronic Work Instructions, users are able to more easily understand the system, as well as avoid interruptions and reduce operator errors, thereby increasing productivity. The sequence management module allows users to see in real time what is happening with the system, and with the help of the various tools provided by it, any possible scenario can be controlled.

8. The proposed system for controlling technological objects with the help of Honeywell Experion Process Knowledge System for decision-making in the operation of double-fired process heaters for the Karachaganak Petroleum Operating B.V. refinery (KPO) in Kazakhstan, increases system performance.

Conclusion - summary of the obtained results

List of publications on the PhD thesis

1. Iliev, V., Moutafchieva, D., **Dukovski, R.**, Numerical Investigation of Hydrodynamics in Submerged Membrane Bioreactor with Aeration. Bulgarian Chemical Communications, Journal of the Chemicals Institutes of the Bulgarian Academy of Sciences and of the Union of Chemists in Bulgaria, 52, 4, Bulgarian Academy of Sciences, 2020, ISSN:0324-1130, DOI:1034049/bcc.52.4.MP10, 561-568. SJR (Scopus):0.17, JCR-IF (Web of Science):0.349

2. Dukovski, R., Implementation and Process Control of a Cricket Filter. Engineering Sciences, LXI, 2, Prof. Marin Drinov Academic Publishing House, 2024, ISSN:1312-5702, DOI:10.7546/EngSci.LXI.24.02.01, 3-14.

3. Dukovski, R., Drum Level Control of a Boiler System. Engineering Sciences, LXII, 3, Prof. Marin Drinov Academic Publishing House, 2025, ISSN:1312-5702, DOI:10.7546/EngSci.LXII.25.03.05, 46-67.

4. Dukovski, R., Implementation and Control of Boiler System. Engineering Sciences, LXII, 2, Prof. Marin Drinov Academic Publishing House, 2025, ISSN:1312-5702, DOI:10.7546/EngSci.LXII.25.02.03, 42-55.

5. Dukovski, R., Implementation and Process Control of a Dual Fired Heater. Proceedings of the 9th IEEE International Conference on Big Data, Knowledge and Control Systems Engineering – BdKCSE'2025, 06-07 November 2025, Bankya, Bulgaria, IEEE Xplore, приета за печат: 2025.

Major scientific and applied scientific contributions

The set goals and objectives of the PhD thesis are aimed at achieving highly efficient management of complex technological objects, offering the possibility of using various modern design and management methods in order to coordinate local management systems of individual processes and achieve effective management of technological objects.

The obtained results, described in this PhD thesis, can be summarized in the following scientific and scientific-applied contributions:

1. An in-depth theoretical analysis of decision-making systems and the challenges that arise in their application, presented using the tools of modern theory, has been conducted.

2. The operation of a thermal energy technological control object - a steam generator - has been analyzed, and models have been developed for control at the level of the steam-water mixture and at the level in the drum, in the presence of a static disturbance compensator.

3. Simulation modeling of decision-making systems for controlling a steam generator was carried out, and simulation models were developed in the MATLAB Simulink and Experion Process Knowledge System programming environments for a mathematical model of a real steam generator.

4. A system for automatic regulation of the level of the steam-water mixture in the drum using Honeywell Experion Process Knowledge System decision-making tools is proposed during real-time operation of a steam generator at the El Paso refinery in the USA..

5. A membrane bioreactor for the production of yeast Hansenula polymorpha has been designed, with a productivity of 6 [g/h] at a cultivation temperature of 30 [^{0}C] and a growth medium of methanol. The required aeration rate has been determined by computer simulations of the hydrodynamic behavior around the membrane boundary layer at a minimum strain rate of 0.8 [s^{-1}]. The required stirring rate has been determined by computer simulations of the hydrodynamic behavior in the cell growth chamber at a maximum strain rate of 15 [s^{-1}].

6. A simulation modeling of a chemical technological control object - a cricket filter has been carried out and a control system has been proposed using the Honeywell Experion Process Knowledge System for decision-making in real-time operation of a cricket filter for the Karachaganak Petroleum Operating B.V. (KPO) refinery in Kazakhstan.

7. A simulation modeling of a chemical technological control object - a double-layer heater was carried out and a control system was proposed using the Honeywell Experion Process Knowledge System for decision-making in the real-time operation of double-fired technological heaters for the Karachaganak Petroleum Operating B.V. (KPO) refinery in Kazakhstan.

Declaration of originality

Declaration of originality of results

I declare that this PhD thesis contains original results obtained during my research {with the support and assistance of my supervisor}. The results obtained, described and/or published by other scientists are duly and in detail cited in the bibliography.

This PhD thesis has not been applied for the acquisition of a scientific degree at any other higher education institution, university or scientific institute.

Signature:

Acknowledgements

I would like to express my heartfelt gratitude to my supervisor – Acad. Vasil Sgurev, for the methodological assistance provided, for the advice and discussions on topics that significantly enriched my understanding of the world of science and its various manifestations. This was an invaluable time for me and allowed me to broaden my horizons not only in the engineering field, but also broadened my worldview as a person.

I would like to express my gratitude to my family, who have been not only my support but also my inspiration. They have encouraged me to broaden my horizons and develop my potential. I thank my family for their unconditional support and understanding!

Thank you!