Simulation Model of a Quality Control Laboratory in the Pharmaceutical Industry

Andrea Costigliola

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Susana Margarida da Silva Vieira
Dr. Filipe André Prata Ataíde

Examination Committee
Chairperson: Prof. Daniel Jorge Viegas Gonçalves
Supervisor: Prof. Susana Margarida da Silva Vieira
Members of the Committee: Prof. Carlos Baptista Cardeira
Dr. José Luis Santos

November 2016
To the memory of my grandmother Restituta
“Everybody is a genius. But if you judge a fish by its ability to climb a tree, it will live its whole life believing that it is stupid.”

Albert Einstein
Acknowledgments

Firstly, I would like to express my gratitude to my supervisors Prof. Susana Vieira and Dr. Filipe Ataide, for the continuous support given during the execution of this work. A special recognition goes to Prof. João Sousa for introducing me to this research field.

I would like to acknowledge Hovione Farmaciência S.A. for giving me the opportunity to work on a such challenging project. Thank to all the Knowledge Management group for the help provided me during those months.

A special thanks goes to my family, and in particular to my parents that gave me the possibility to complete my training abroad, and that always believed in me. To my brother, my very best friend and the most supportive person I know.

Thanks to my cousins Marco, Daniela and Pina for being always there every time I needed their help. Thanks to my aunts Teresa and Filomena, and to my uncle Alfonso.

Moreover, I would like to thanks my friends, Giulio, Mario, Teresa, Chiara and Salvatore. A special recognition goes to the wonderful people I met in the last two years at IST. To my colleagues Leonardo, Jakub, Antonio, Andrey and Miguel.

In conclusion, a special acknowledgment goes to the person that in the last three years have been always by my side. Thanks Mary!
Abstract

In the last decades the pharmaceutical industry has been through many changes, as a result of increased competition, patents expiration and increased drug quality standards requirements. In order to compete against peers, drug manufacturer started to be more concerned on achieving operational excellence through the optimization of all process involved in drug manufacturing. During its development, a drug must be constantly monitored with laboratory tests. In this scenario, quality control laboratories are critical components in drug manufacturing, and inefficiencies in laboratory management can have a major impact on the overall supply chain service level.

The aim of this thesis is to build a Discrete Event Simulation model of a Quality Control laboratory. To achieve this objective, a generic framework for information treatment and organization was built. In particular, information coming from different databases was organized into a single one that was used as input to a discrete event simulation model. The proposed model represents in detail the work flow of a quality control laboratory and it is intended as a support tool for planning, scheduling and decision making. The model was validated using real data, and it resulted effective to estimate performance parameters such as, system throughput, equipment usage rate, system responsiveness and tasks processing times. Furthermore, the simulation model was tested with an alternative scheduling policy to evaluate how modifications on the system may improve its performance.

Keywords: Quality Control laboratory, Discrete Event Systems, Discrete Event Simulation, Planning, Scheduling, Simulation study, Pharmaceutical industry
# Contents

Acknowledgments .............................................. v  
Abstract ...................................................... vii  
List of Tables ................................................ xii  
List of Figures ............................................... xiii  

## 1 Introduction

1.1 The Pharmaceutical Industry ........................................ 2  
1.1.1 Supply Chain and Contract Manufacturing Networks .............. 3  
1.1.2 GMP and GLP regulations ........................................ 4  
1.1.3 Drug Development ........................................... 5  
1.2 Quality Control ............................................... 6  
1.2.1 Quality Control in the Pharmaceutical Industry .................... 7  
1.2.2 Quality control laboratory management .......................... 8  
1.2.3 Planning and Scheduling methods ................................ 10  
1.3 Contributions ............................................... 11  
1.4 Thesis Outline ............................................... 12  

## 2 Discrete Event System Simulation

2.1 Systems and Models ........................................... 14  
2.1.1 Discrete Event Systems ...................................... 14  
2.1.2 Timed Petri Nets ........................................... 15  
2.2 Discrete Event Simulation ...................................... 18  
2.2.1 Steps in a Simulation Study .................................. 19  

## 3 Simulation model of a Quality Control laboratory

3.1 Quality control laboratory ...................................... 22  
3.2 System Overview ............................................. 22  
3.2.1 Analytical techniques ....................................... 23  
3.2.2 Available data and Information Systems ....................... 25  
3.3 Data processing ............................................... 26  
3.3.1 LIMS data ................................................ 27  
3.3.2 Information extraction algorithm .............................. 28
List of Tables

3.1 Percentage of analysis per analytical technique. ........................................ 28
3.2 Time information used to model analyses tasks. ................................. 31
3.3 Analyst Work shifts. ............................................................................. 32
3.4 Processing times distributions ($\text{Tr}(a, b, c)$: Triangular pdf; $\text{U}(a, b)$: Uniform pdf). Time unit is minutes ................................................. 37

4.1 Simulation model average processing times. Time unit is minutes .......... 44
4.2 Average time in system for each sample type. Time unit is hours ......... 47
4.3 Samples average time in system: comparison between models. Time unit is hours ................................................................. 48
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>New Molecular Entities (NME) approvals [41].</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Contract Manufacturing Organization (CMO).</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>Drug development phases.</td>
<td>6</td>
</tr>
<tr>
<td>2.1</td>
<td>A Petri Net with 3 places and 4 transitions.</td>
<td>16</td>
</tr>
<tr>
<td>2.2</td>
<td>A marked Petri net with ( m = {1, 0, 1} ).</td>
<td>17</td>
</tr>
<tr>
<td>2.3</td>
<td>A timed Petri net with two timed transitions.</td>
<td>18</td>
</tr>
<tr>
<td>2.4</td>
<td>Main steps in a simulation study.</td>
<td>19</td>
</tr>
<tr>
<td>3.1</td>
<td>Analysis Work-flow.</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>UML representation of the designed database.</td>
<td>28</td>
</tr>
<tr>
<td>3.3</td>
<td>Information extraction algorithm flowchart.</td>
<td>30</td>
</tr>
<tr>
<td>3.4</td>
<td>Time study manual form header.</td>
<td>30</td>
</tr>
<tr>
<td>3.5</td>
<td>High-level Quality Control laboratory model.</td>
<td>33</td>
</tr>
<tr>
<td>3.6</td>
<td>Discrete Event System model of a generic equipment.</td>
<td>34</td>
</tr>
<tr>
<td>3.7</td>
<td>Liquid Chromatography (HPLC) estimated times and probability distribution functions.</td>
<td>36</td>
</tr>
<tr>
<td>3.8</td>
<td>Particle Size Analysis (PSA) estimated times and probability distribution functions.</td>
<td>36</td>
</tr>
<tr>
<td>3.9</td>
<td>X-ray Powder Diffraction (XRPD) estimated times and probability distribution functions.</td>
<td>37</td>
</tr>
<tr>
<td>3.10</td>
<td>Simio equipment object and control inputs.</td>
<td>38</td>
</tr>
<tr>
<td>3.11</td>
<td>Quality Control laboratory model: Simio 3D view.</td>
<td>39</td>
</tr>
<tr>
<td>4.1</td>
<td>HPLC usage rate comparison between simulation and real data.</td>
<td>42</td>
</tr>
<tr>
<td>4.2</td>
<td>GC usage rate comparison between simulation and real data.</td>
<td>43</td>
</tr>
<tr>
<td>4.3</td>
<td>HPLC: tasks percentage on total time.</td>
<td>45</td>
</tr>
<tr>
<td>4.4</td>
<td>GC: tasks percentage on total time.</td>
<td>45</td>
</tr>
<tr>
<td>4.5</td>
<td>Other equipments: tasks percentage on total time.</td>
<td>46</td>
</tr>
<tr>
<td>B.1</td>
<td>Liquid Chromatography (HPLC) estimated times and probability distribution functions.</td>
<td>69</td>
</tr>
<tr>
<td>B.2</td>
<td>Gas Chromatography (GC) estimated times and probability distribution functions.</td>
<td>70</td>
</tr>
<tr>
<td>B.3</td>
<td>Particle Size Analysis (PSA) estimated times and probability distribution functions.</td>
<td>70</td>
</tr>
<tr>
<td>B.4</td>
<td>Karl Fischer titration (KF) estimated times and probability distribution functions.</td>
<td>71</td>
</tr>
<tr>
<td>B.5</td>
<td>Dynamic Scanning Calorimetry (DSC) estimated times and probability distribution functions.</td>
<td>71</td>
</tr>
</tbody>
</table>
B.6 X-ray Powder Diffraction (XRPD) estimated times and probability distribution functions.
Chapter 1

Introduction

In the last decades drug manufacture evolved driven by external economic forces, patents expiration and increased competition. In order to maintain their competitive advantage, pharmaceutical companies organized themselves into complex organizations (supply chain and contract manufacturing networks) and started to be more concerned on achieving operational excellence through the optimization of all the processes involved in drugs manufacturing: supply of raw materials, logistic operations, chemical processes modeling, management of quality control operations, etc.

The pharmaceutical industry is regulated by the Good Manufacturing Practices (GMP) and Good Laboratory Practices (GLP). These regulations were introduced to assure that every pharmaceutical product meets safety and quality requirements in a systematic fashion. The product life cycle must be constantly monitored with laboratory tests. Quality control laboratories are critical components in drug manufacturing and can have a major impact on the overall supply chain service level. The situation can be magnified in the case of a contract manufacturing organization, that produce goods under the brand of its clients and therefore has to deal with a large number of projects and materials.

Given the high mix of products and laboratory tests, it is important to develop effective strategies for laboratory management, which includes personnel and equipment scheduling, laboratory information management, stock optimization, etc. These tasks are often done by supervisors experience with the support of MS Excel worksheets or dedicated information system. However, laboratory information management systems (LIMS) used by pharmaceutical companies only track the analyses performed and lack of some features (i.e. information on processing times, work flow) essential for planning, scheduling and stock management.

From a research perspective, studies regarding the concrete production industry have been often limited to manufacturing processes. Advances in informatics, data analysis and knowledge management, made industries aware of the power of information. This information can be organized and generate knowledge towards the improvement of the quality of the services and manufacturing processes. As a result, informatics is being always more incorporated in the industrial setting, and nowadays industrial informatics represents an important field of study. Informatics is no more related just to the Information Technology (IT) services and infrastructures, but also to the design, simulation and modeling of
This chapter presents an overview of the pharmaceutical industry and discusses the contract manufacturing business model and its reasoning. It then discusses some of the key issues of quality control management and optimization, and in particular, different methodologies existing in literature will be reviewed. Finally, a description of the used methodologies and objectives of this thesis are presented.

1.1 The Pharmaceutical Industry

The Pharmaceutical Industry can be considered as a complex system whose main function is to discover, develop and manufacture drugs. A pharmaceutical drug is a chemical substance used in medicine for disease treatment, prevention, cure or diagnose. The World Health Organization (WHO) provides a more accurate definition of pharmaceutical preparation:

“A drug (or pharmaceutical preparation) is any substance or mixture of substances manufactured, sold, offered for sale or represented for use in:

(a) the diagnosis, treatment, mitigation or prevention of disease, abnormal physical state or the symptoms thereof in man or animal;

(b) restoring, correcting or modifying organic functions in man or animal.”

Over the last decades, the pharmaceutical industry has been through many changes, as a result of economic decline, patents expiration, increased costs for development and manufacturing of pharmaceutical products [3]. In addition to that, high quality and safety standards for drug products are required by Food and Drug Administration (FDA) and European Medicines Agency (EMA) authorities. In order to keep and improve quality of manufactured drug products, large investments for research and development programs and quality assurance are needed.

In this scenario, pharmaceutical companies started a new phase made of strong partnerships and licensing agreements. Some companies have joined supply chain structures, while others have decentralized their production activities. Similarly to what happened in other industries (i.e. aerospace, automotive, electronics, food manufacturing), the pharmaceutical industry is experiencing a new business model: Contract manufacturing. With this model, pharmaceutical brands make use of something similar to “outsourcing”.

As a result, new business opportunities arose and new challenges are there to be overcome. In recent years the number of New Chemical Entities (NCEs, also referred to as new molecular entities) approved by the Food and Drug Administration authority has increased [41], passing from 22 new molecules approved in 2006 to the 45 of 2015 (Figure 1.1). This led pharmaceutical companies to deal with a large number of products to manufacture and develop.
1.1.1 Supply Chain and Contract Manufacturing Networks

Nowadays, a supply chain structure is widely used in practice, and the most part of production industries follows this model [2]. This structure allows a higher level of specialization, more flexibility, lower costs and in general an increased productivity. This can be explained by the fact that every company has a more limited set of tasks to perform, and it can leave the auxiliary tasks to the partners.

Although the concept of a supply chain could be intuitive at a higher level of abstraction, there is not a unique definition that allows to fit all of existent supply chains. In general, a supply chain is a kind of network with facilities and distribution entities (suppliers, manufacturers, distributors, retailers). The supply chain performs the functions of procurement of raw materials, transformation of raw materials into intermediate and finished products, and distribution of finished products to customers [35]. From the logistic viewpoint, the management of supply chains involves a set of complex and interdependent combinatorial problems (e.g. acquisition of raw materials, scheduling of production facilities, management of quality control system, routing of transport vehicles, stock optimization, etc.) [39].

Thus, we can not fit every supply chain in a unique model. Also, the appropriate model depends on the level of detail in which we are defining the system, and what part of the system we are considering. In this context, the focus will be on the contract manufacturing organization, its suppliers and its clients (Figure 1.2).

A contract manufacturing organization (CMO) is a company that agrees to produce goods under the brand of another firm. This business model is already well assessed in many industries. For instance, in the electronics industry original equipment manufacturers (OEMs) outsource manufacturing to companies who are specialists in electronic manufacturing services. The success of this model can be in part attributed to the advent of the Internet that has made possible continuous, ubiquitous and correct information flow between client and manufacturers. Its influence is evident not only at the strategic level, but
also at tactical and operational level [14]. This form of outsourcing offers new opportunities to companies for the improvement of their core business through the conversion of fixed costs into variable costs. In fact, the productive activity can be reduced or eliminated, and replaced with contract manufacturers.

Nowadays, pharmaceutical contract manufacturing organizations offer a large number of services, such as synthesis and production of active pharmaceutical ingredients (API) [32]. In general, a pharmaceutical contract manufacturer offers two main service types: primary and secondary manufacturing. The first refers to the synthesis of the active ingredient, while the latter refers to the transformation (by addition of excipients) of the active pharmaceutical ingredient produced at the primary site into final drug product (i.e. pills, injectable) [38].

This scenario is very complex and involves different entities cooperating and communicating for common objectives: increase products quality and reduce costs. However, efficiency relies mostly on decision making processes, which are, in many cases, based on experience and instinct. This, often leads to non optimal decision and affects negatively the efficiency of the overall supply chain. Difficulties in supply chain management come from the complexity of the system itself and from the level of integration needed to share informations, coordinate and synchronize different processes in the supply chain.

1.1.2 GMP and GLP regulations

In order to ensure drug quality and safety, the pharmaceutical industry is regulated by the Good Manufacturing Practices (GMP) and Good Laboratory Practices (GLP).

GMP regulations contain minimum requirements that a pharmaceutical product manufacturer has to meet to ensure that the developed drugs have the intended quality and purity characteristics, and do not pose risks to the consumer health. Compliance with GMP regulatory is necessary for marketing authorization. It defines the guidelines which govern every stage in the drug development life cycle, including production and quality control. Good manufacturing practices have their roots in the first half
of the 20th century when a series of food and drug scandals in the United States cleared up that it was required a regulation of the food and drug market. In 1938, the U.S. congress passed the Federal Food, Drug and Cosmetic Act, which required companies to prove the quality and safety of their products before marketing them. During the second World War, companies had to submit to the Food and Drug Administration authority samples from each produced batch in order to get a permission for their release. However, the first GMPs for finished drugs were compiled only in 1963. Finally, good manufacturing practices for pharmaceutical products were detailed and finalized in 1978 with the Code of Federal Regulations [10].

The modern good laboratory practices regulations ensure the quality and reliability of laboratory tests. It provides a set of principles that have to be followed when laboratory studies are performed, analyzed and reported. In 1970's, a survey conducted by the U.S. Government highlighted that pharmaceutical companies clinical research documentation was poor and not reliable. As a consequence of this study, modern GLPs regulations were established in 1976 [1]. In Europe, those regulation were introduced by the European Medicinal Agency (EMA), that started to control the drug manufacturing industry compliance with GMPs and GLPs regulations.

Both GMPs and GLPs states that a pharmaceutical company must have a Quality assurance unit (QA). Quality assurance have been designed with the scope of preventing defected manufactured products. In drug manufacturing it assures the compliance of equipment, facility management, methods, practices, laboratory records with the regulations imposed by the Food and Drug Administration and by the European Medicinal Agency.

1.1.3 Drug Development

*Drug Development* is the process that leads to the marketing of a new pharmaceutical drug. It can be divided into five main steps (Figure 1.3) [7]:

1. Drug Discovery;
2. Preclinical Development;
3. Clinical Development;
4. Drug Approval;
5. Market.

The first step comprises the discovery of a new promising chemical compound to develop and a detailed study of the molecule in order to gather preliminary information on its benefits, toxicities, mechanisms of action, etc. Preclinical development includes a series of laboratory tests on the drug that must be performed to assess drug kinetics, dosage and safety. At this stage, FDA requires researchers compliance to the GLPs regulation. The third step includes clinical research and it is performed into three different phases which vary based on the number of patients involved in the study. During the clinical development, GMPs regulation ensures that there is no risk for patients safety. Once the clinical
development has been completed and no harms or toxicities have been detected, the drug developer submits an application to market the product. FDA authorities review the submitted data and decide on its approval. When a drug is accepted, it goes to the market. During the commercial phase, FDA and the manufacturer must continue the investigation on the compound to have a better picture on its safety and stability [9].

![Drug development phases diagram]

**Figure 1.3: Drug development phases.**

### 1.2 Quality Control

Quality refers to a series of product features which meet customer requirements and conformance to desired specifications. Quality control (QC) is the process that ensures that a manufactured product meets a defined set of quality criteria and requirements. The ISO 9000 addresses different aspects of quality management and define quality standards that serve as guidance and tools for companies who want to ensure consistency in manufactured product quality. As referred by Juran [20], in order to achieve quality in a manufactured product, three main tasks are needed:

- Quality planning;
- Quality control;
- Quality improvement.

Planning refers to the identification of quality goals, customers needs and to the development of specific manufacturing and control processes to meet the requested quality standards. Control refers to the evaluation of the manufacturing process performance towards quality goals. Quality improvement is the process that creates new needs and procedures for higher quality standards.

As referred, the pharmaceutical industry is the most regulated of all industries. During its production life cycle a drug has to be constantly monitored with laboratory tests to ensure compliance with quality and safety standards. The process that leads from the discovery of a new promising compound to its marketing, also referred to as cycle time, may take many years. In this period, a continuous and recursive process of quality planning-control-improvement is needed.
In drug manufacturing, quality control unit develops and improves specific analytical methods to test chemical compounds. Analytical chemistry is a measurement science which develops and uses methods in order to perform quantitative and qualitative analysis. Given a certain compound, a specific analytical method has to be designed to identify and quantify matter. The identification process is often referred to as qualitative analysis. A quantitative analysis is performed every time there is the necessity to quantify the mass or the concentration of a particular compound.

### 1.2.1 Quality Control in the Pharmaceutical Industry

Quality control plays a crucial role in the drug development industry. It is responsible for quality, safety and efficacy of new medicines. Throughout the whole drug development process, analytical methods development and validation are key elements. A method should be developed with the goal to rapidly test preclinical samples, formulation prototypes and commercial samples [6]. In this phase all the information gathered is necessary for methods design and improvement. An analytical method is ready to be used if it is capable to produce results with acceptable accuracy and precision. Thus, once a method has been developed it must be validated in order to fulfill a series of acceptance criteria. Given an analytical method, an analysis must produce the “same result” independently from the analyst that performs the work, the equipment or the laboratory where the analysis is conducted. The validation of an analytical method proves that the method is scientifically sound and effective in measuring correctly the substances both from the quantitative and qualitative point of view [17]. Although analytical methods development and validation are fundamentals for any drug research and development program, there are other operations of quality control directly related to the manufacturing process: In-Process Control (IPC) and testing product stability.

In-process control are tests performed during the manufacturing process. Its main function is to monitor the manufacturing process and provide insights for its on-line correction when necessary. This operation can be performed at predefined steps during the manufacturing or at the end of a process task. Also, extraordinaries IPC tests may be executed any time there are some doubts about the quality of the manufacturing batch. In particular, quality control evaluates that the product attributes agree with its specifications. Any information gathered during the in-process control process must be recorded and detailed:

“Finished product assessment should embrace all relevant factors, including production conditions, results of in-process testing, a review of manufacturing (including packaging) documentation, compliance with Finished Product Specification and examination of the final finished pack” [13].

Stability tests are performed on active pharmaceutical ingredients (APIs) and finite pharmaceutical products (FPPs) to recover information on how the quality of the drug varies with time under the influence of external factors, such as temperature, humidity and light exposure. A stability program must also assess existing interferences with product-related factors that may influence the drug quality (i.e. packaging materials, interaction of API with excipients). The objective of stability testing is to identify the degradation of the drug product and the intrinsic stability of the molecule. For pharmaceutical products
whose shelf-life is at least of 12 months, the frequency of stability tests should be every three months over the first year, every six months over the second year and annually from the third year on [31].

Pharmaceutical analytical methods can be of four main types:

- identification tests;
- potency assay;
- impurity tests;
- specific tests.

The first three tests types are universal and depending on the particular compound different analytical techniques can be used. On the other hand, specific tests (i.e. x-ray diffraction, particle size analysis) are used to control specific properties of the drug product.

A quality control laboratory represents a complex system composed by personnel and equipment. It performs a wide range of analytical techniques on a large number of different samples. The situation of a laboratory of a contract manufacturer is even more complex due to the high mix of product coming from different active projects with its clients. In quality control laboratories a large number of tests has to be executed on product samples, and restrictions coming from compliance with GLP and GMP regulations, make them critical components in drug manufacturing. In fact, inefficiencies at quality control level may lead to bad analytical methods design, delays in the delivery of the manufactured product to a client, delays in the production process and, in the worst case, to the degradation of a production batch.

1.2.2 Quality control laboratory management

Due to the challenging scenario described, the optimization of all the operations involved in the process of drug manufacturing is nowadays a necessity. Pharmaceutical companies are always more concerned on the achievement of operational excellence which is an element of organizational leadership that tends to the improvement of key performance metrics. Optimization methods can not only improve quality but also minimize process operational costs and reduce variation in the product [34]. Products quality and operational efficiency depend on timely delivery of quality analysis results. Laboratory management is a complex task which includes resources (personnel and equipment) scheduling, analysis prioritization, results evaluation and documentation. Inaccurate scheduling policies may delay obtaining results, affect negatively their quality and may generate financial losses. Quality control laboratory efficiency can be essentially improved by analysis rational scheduling [12]. Key factors in lab management are resources planning and scheduling [33].

Planning is a process that must be done before scheduling and its main goal is to evaluate how many resources and what resources are needed to perform a job. In a laboratory environment, planning defines the correct number of analysts and instrumentations needed to perform quality control operations. Without the proper modeling tool, it is difficult to predict the correct number of resources. Mistakes in laboratory dimension may increase costs and generate inefficiencies at the quality control level.
Scheduling refers to the allocation of jobs to available resources and its prioritization. It defines “what job should be done first and by who”. In a complex scenario as a quality control laboratory, a large number of samples need to be analyzed every day, and those analyses often concur for the same laboratory instrument. A good schedule could improve by far quality control labs performances. Although not very used in pharmaceutical quality control laboratory management, operations research (OR) provides methods and tools for the optimization of similar planning and scheduling problems [12].

Planning of Quality Control labs is a very complex task, since it involves the forecasting of the number of samples entering the laboratory and the types of analyses to perform. Thus, it is difficult to design the laboratory with the desired capacity. Historical data may provide some insights that can be used for planning purposes but, given the non homogeneity of drug development projects, a stochastic approach that takes into account also the nature of the projects, their expected cycle time and other relevant features may result in a better strategy for laboratory planning. Nowadays, in pharmaceutical companies, planning is often done based on senior managers’ experience, rather than with a verified data driven model. As suggested by Maslaton [28], resource analysis should be done at regular intervals (i.e. monthly, yearly), based on the dynamics of the forecast. The author proposes the development of a generic framework to organize laboratory information to be used for laboratory workload forecasting. In particular, the process of data collection/organization must be done following some key steps:

1. Develop list of products and raw materials and group them into product families.
2. Identify representative product for each family.
3. Characterize each representative product (i.e. types and number of tests, tests frequency).
4. Define naming convention for every test.
5. Identify/estimate analysis processing time for each test.

Once the correct amount of resources has been determined, the second step towards the optimization of a quality control laboratory is the scheduling of the analyses. As referred above, quality control activities may vary a lot depending on the sample to be analyzed. Independently from the analytical technique that needs to be used for a particular test, samples can come from one of five main groups:

- Raw material samples;
- Development samples;
- In process control samples;
- Finite product samples;
- Stability test samples.

To perform a test it is needed a specific laboratory equipment and a specialized analyst that has adequate formation to run the test. However, even if two samples have different nature they can still concur for the same resources. An optimized scheduling will ensure assigning the samples/tests to the
best available resources. Moreover, it ensures that the right tests are started at the right time and there are no delays. On-time delivery and, as consequence, the cycle time can be significantly affected by scheduling [29].

An interesting solution for laboratory personnel scheduling is the one proposed by Boyd et al. [5]. In this paper, the authors applied a genetic algorithm for analysts scheduling. Their program maximizes the value of a fitness function that measures how well a given work shift scheduling of analysts and their skills matches a set of work tasks. Dudnikov et al. [12] developed in 2011 a laboratory analysis planning system to schedule analyses on the available resources. The authors considered two types of samples for their system: periodic samples and unplanned samples. They created a ranking in order to prioritize the analyses and prepared a 24 hours scheduling considering only the periodic samples. However, the produced schedule needs to be rearranged in case of unplanned samples or changes in priority of periodic samples. Although the information system and the theoretical framework are interesting and of easy application, the paper lacks of references on the scheduling algorithm used to generate the work schedule. In literature does not exist a generic framework applied to the pharmaceutical industry that is capable of addressing all the issues involved in quality control laboratory management. However, Schäfer proposes a set of terms and definitions that may be used as basis to build a consistent framework for laboratory management [36]. The author proposes the modeling of all components interacting on the workbench (i.e samples, instruments, sensors, results, information systems) and provides a description of a schematic scheduling process that can be applied in quality control laboratories. This scheme comprises the following steps:


2. Information treatment.

3. Generation of working plan elements and relative constraints.

4. Schedule generation.

5. Scheduling execution.

6. Instrumental control and Data storage.

Modeling can be identified as the very first step for a correct laboratory management. In this context, modeling refers to the identification of all the relevant features of the system under study, and to the collection of relevant information that allows a good characterization of the system behavior (i.e. samples arrival frequency, number of tests, analyses steps, processing times). Once a robust laboratory model has been constructed, classical planning and scheduling methods can be used to optimize the system.

1.2.3 Planning and Scheduling methods

During the XIXth century, the manufacturing systems were simple and produced a small number of products. Factories were relatively small and the production was made of large batches of few products.
As factories grew, manufacturing operations became more complex. New and effective methods to manage such systems were needed. In response to these needs, planning and scheduling techniques rose during the first two decades of the 20th century. At the time of World War I, Frederick Taylor proposed the production planning office to create plans, manage inventory, and monitor processes. However, a universal rigorous method for production scheduling was proposed by Henry L. Gantt in 1919 that designed an innovative chart diagram for production control [18].

Over the 20th century many techniques and algorithms have been developed for the solution of planning and scheduling problems. Some of these techniques are: loading, planning boards and line of balance. Loading is a scheduling technique that defines the specific day or week when a job is performed by a machine (or a group of machines). Planning boards, can be attributed to Taylor, and define the available time ("space") existing on each machine in order to allocate jobs to that machine [25]. Line of balance is a technique used to determine how far ahead a production line can meet a certain demand.

The development of algorithms for creating scheduling can be located in time around the 1940s, when linear programming started to be developed and applied to production planning problems. In the 1950s, research into scheduling problems led to important algorithms and scheduling rules that are still used. Examples of these rules are the Earliest Due Date (EDD) which minimize the maximum lateness and the Shortest Processing Time (SPT) for minimizing average flow time. However, many famous mathematical programming problems, such as Job Shop Scheduling problem, Facility Planning and Travel Salesman Problem, were found to be NP-hard [11]. Conventional mathematical programming methods are inappropriate to solve those problems, especially taking into account the computational time needed to solve them. In the 1960 it was developed an algorithm known as Branch and Bound that implicitly enumerates all the possible solutions to find the optimal solution. This approach, still effective is inappropriate for large problems.

During the 1980s and 1990s, many operations research methods were developed and resulted able to found sub-optimal solutions to NP-hard problems in a limited amount of time. Such methods included local search algorithms such as simulated annealing and tabu search [18]. In recent years, a new class of nature-inspired methods for the solution of large problems has emerged. These methods are the so called metaheuristics. Examples of metaheuristics are: Genetic Algorithms [30], Ant Colony Optimization [27], Artificial Bee Colony Optimization [21].

1.3 Contributions

This work has been developed in a partnership agreement between Instituto Superior Técnico and Hovione Farmacêutica, S.A.. The aim of this thesis is to build a generic framework for modeling and simulating a quality control laboratory of the partner company. In particular, this work uses the concepts of Discrete Event System theory and Petri nets for modeling purposes. A model of the system is simulated using Simio simulation software, and the analysis of the outputs will serve as basis for planning, scheduling and decision making in quality control laboratories.
In the state of the art review conducted, it has been evidenced the lack of papers that propose a systemic way to solve problems related with the quality control laboratory management in the pharmaceutical industry. For this reason, this thesis proposes an innovative way for data management that can be used for modeling purposes, but also to easily gather information on the operations in quality control.

An article of this thesis has been submitted to the track “Supply Network Engineering, Dynamics, and Control” for the conference IFAC World Congress 2017.

1.4 Thesis Outline

Chapter 1 presents an overview of the pharmaceutical industry and quality control systems.

In chapter 2 the theory of Discrete Event Systems is presented and the Petri nets formalism is introduced to be used as a modeling tool.

Chapter 3 is dedicated to the development of a simulation model of a quality control laboratory. In particular, it will be focused on two main aspects: data processing and simulation model implementation.

Chapter 4 presents the results of the simulation model and its validation.

Finally, chapter 5 will conclude this thesis providing insights on the quality of the proposed solution, and it will evaluate the performance of the quality control system under study. Moreover, suggestions for future work are presented.
Chapter 2

Discrete Event System Simulation

A simulation can be defined as the imitation of a real world process or system over time. Simulation involves the generation of an artificial history of a system (i.e. process, facility) and its observation in order to get insights on system operating conditions [4]. Simulation has been widely used over the last 50 years and it is nowadays one of the most used operations research (OR) and management-science techniques, if not the most widely used [24]. In 1993 Lane et al. [23] performed a study of the most used methods in operations research and identified discrete event simulation as one of the three most used techniques together with mathematical programming and statistics. A successive study analyzed 1294 papers from 1970 through 1992, and found out that simulation was second only to mathematical programming among 13 operations research techniques [16].

Following the definition proposed by Schmidt and Taylor in 1970, a system can be defined as a collection of entities that interacts towards the accomplishment of some goal [37]. Although this definition is of ease understanding, it is still very generic. In general, the system (and its definition) depends on the objectives of a particular study. There are two main techniques used to study and simulate a system:

- System Dynamics (SD);
- Discrete Event Simulation (DES).

The first approach uses mathematical methods (i.e. calculus, algebra, probability) to describe analytically systems dynamics. A system is modeled as a set of stocks and flows in pseudo-continuous time, and it is described by differential equations. This approach is effective in a large number of engineering, social, economic and environmental systems. However, most real world systems are too complex to allow realistic models to be evaluated analytically.

Discrete event simulation models, on the other hand, are stochastic and the dynamic of the system is driven by events happening at discrete points in time [40]. In simulation a system model is evaluated numerically with a computer, and relevant information is collected to estimate the behavior of a real world system [24].

The availability of special-purpose simulation languages made possible the large diffusion of computer simulation methods. Discrete event simulation represents an economic and flexible choice that
allows to recover information on a system, simulate changing scenarios and get insights on its dynamics. The main difference with optimization models is that simulation models are “run” rather than solved. Given a particular set of input and model features a model is “run” in different simulation scenarios and its behavior is observed.

Next sections formalizes the concept of Discrete Event System and Petri net paradigm is presented as a modeling tool for this class of systems. Furthermore, this chapter presents an overview on discrete event simulation models and a description of the main steps of a simulation study.

2.1 Systems and Models

When it comes the time to study a system to understand its behavior or to predict its performances under some new conditions, two main approaches can be used in practice:

- Experiments with the actual system;
- Experiments with a model of the system.

The first approach is a good way to study a system every time it is possible to modify the operating conditions of the real system. However, this approach can be costly and unfeasible in other cases. For example, if a company wants to measure the return of investment for equipment purchase it can not modify the real system to “try” what would happen with the new machinery. When experiments on the real system are not feasible or too expensive, a model of the system provides a representation of the system that can be studied.

As defined above, a system consists of interacting “entities” that perform a function. In order to make a quantitative analysis of a system it is necessary to develop a model that represents a description of the real system and its dynamics that can be observed through a series of variables necessary to describe its state. The state can be defined as the collection of variables necessary to describe a system at a particular point in time. According to the nature of the state variables a system can belong to one of two types: discrete time system and continuous time systems. A discrete time system is one for which the state variables change instantaneously at separated points in time. On the other hand, the state variables of a continuous time system change continuously with time.

2.1.1 Discrete Event Systems

A discrete event system is a particular type of dynamic system whose state space is naturally described by a discrete set, and state transitions are observed at discrete points in time associated with “events” [8]. An event can be seen as “something that happens” and can cause a transition from one state value to another.

Definition 1. A discrete event system (DES) is characterized by:

- Set of possible events: \( E = \{e_1, e_2, ..., e_n\} \);
• Discrete state space: $X$;

• Event driven dynamics: $x_{k+1} = x(\delta_k, e_k), k \in \mathbb{N}$, where:
  
  $x_{k+1}$: next state after the $k$th event happens;
  
  $e_k$: $k$th event happened from the considered point in time;
  
  $\delta : X \times E \to X$: state transition function.

From the definition above, it is clear that the behavior of a system depends on events that drive the system into a particular state. During the study of a discrete event system it is possible to consider different types of event sequences:

• Logic event sequences: the time when an event occurs is not explicit. This kind of sequence is used to model the logic behavior of a system;

• Timed event sequences: the time of occurrence of the events is explicit and it can be deterministic or stochastic. Those sequences are used to evaluate the performance of a system considering its evolution in time.

Different tools can be used to build and represent a discrete event system model. Relying on the definitions above we can have logic models, that are usually modeled as Automata or Petri nets, and Timed models. Tools for timed models are timed Automata, timed Petri nets, $(\max, +)$ Algebra and Markov Chains.

In discrete event simulation time is a fundamental feature. Thus, timed event sequences are more appropriate for modeling. In this context, the main focus will be on timed Petri nets with stochastic timed sequences.

### 2.1.2 Timed Petri Nets

Petri nets were first developed by Carl Adam Petri in the early 1960s. They provide an explicit representation of the transition function of a system. A Petri net is a device that manipulates events according to predefined rules. A system modeled as a Petri net can be easily described in graph form, resulting in a Petri net graph, that is very intuitive and self-explanatory [8].

A Petri net is a bipartite graph with two types of vertexes: places and transitions (Figure 2.1). In a Petri net events are associated with transitions. Places contain information on the state of the Petri net. In order for a transition to “fire” (occur), the Petri net must meet particular conditions.

**Definition 2.** A *Petri net graph* is a 4-tuple $(P, T, A, w)$, where:

- $P$ is the finite set of places;

- $T$ is the finite set of transitions;

- $A \subseteq (P \times T) \cup (T \times P)$ is the set of arcs from places to transitions and from transitions to places;
Figure 2.1: A Petri Net with 3 places and 4 transitions.

$w : A \mapsto 1, 2, 3, \ldots$ is the weight function on the arc set.

To describe the evolution of a Petri net it is necessary to define the concept of *token*. A *token* is an entity that occupies a place in a Petri net, and it is represented with a black dot within a place of the net. The way tokens are assigned to a Petri net graph defines a *marking*.

**Definition 3.** A *marking* is a function $m : P \mapsto \mathbb{N} = 0, 1, 2, \ldots$ that assigns a non negative number of tokens to places in a Petri net.

A *marking* represents explicitly the state of a Petri net graph and defines the condition under which an event (transition) “is enabled” and can fire.

**Definition 4.** A *marked* Petri net is a five-tuple $(P, T, A, w, m)$ where $(P, T, A, w)$ is a Petri net graph and $m$ is a $P \times 1$ vector containing the number of tokens in each place of the net:

$$m^T = [m(p_1), m(p_2), \ldots, m(p_n)] \in \mathbb{N}^n.$$  

**Definition 5.** A transition $t_i \in T$ in a Petri net is said to be *enabled* and can fire if

$$m(p_j) \geq w(p_j, t_i) \forall p_j \in I(t_i)$$

where $I(t_i)$ is the set of places that are input to the transition $t_i$.

In other words, a transition is enabled and can fire if the number of tokens in the places upstream the transition contain a number of tokens greater or equal than the weight of the arcs that links those places to the considered transition. In figure 2.2 is represented a marked Petri net. Given the definition above, in the example the transitions $t_1$ and $t_2$ are enabled and can fire if the associated event happens. On the other hand, transition $t_4$ can not fire because the place $p_3$ contains only one token and the arc that links $p_3$ to $t_4$ has weight equal to two.

When a transition fires, it leads the system in a new state.

**Definition 6.** Given a marked Petri net and its marking $m'$, the firing of an event associated with a transition $t_i \in T$ generates a new marking $m''$ given by:

$$m'(p_j) = m(p_j) - w(p_j, t_i) + w(t_i, p_j), \quad j = 1, \ldots, n$$
When a transition is enabled and fires, all tokens contained in places that are input to the firing transition are transferred to the places that are output of the fired transition. For instance, in the example of figure 2.2, if the transition \( t_2 \) fires, the new marking of the Petri net will be \( m'' = \{0, 0, 2\} \).

This formalism provides a generic framework for the study of discrete event systems. A system can be modeled as a Petri net where transitions correspond to events directly correlated to the system. In a real world system, an event can be an input command, the end or the start of an operation, the activation of a sensor, etc. To simulate the evolution in time of a discrete event system, it is necessary to define a clock structure associated with a set of transitions \( T_D \subseteq T \). Transitions in this set are called timed transitions. When a timed transition is enabled, it does not fire immediately, but after a time delay given by the clock. In this case, the clock regulates the happening of the events and thus, the firing of timed transitions in the Petri net.

**Definition 7.** The clock structure associated with timed transitions of a marked Petri net is a set:

\[
V = \{v_i : t_i \in T_D\}
\]

of time sequences

\[
v_i = \{v_{i,1}, v_{i,2}, \ldots\}, \quad t_i \in T_D, \quad v_{i,k} \in \mathbb{R}^+, \quad k = 1, 2, \ldots
\]

It is now possible to define a timed Petri net, which will be used for the modeling of the discrete event simulation system.

**Definition 8.** A Timed Petri Net is a six-tuple \((P, T, A, w, m, V)\), where \((P, T, A, w, m)\) is a marked Petri net, and \(V\) is a clock structure associated with transitions \( t_i \in T_D \).

As referred, a timed transition, when enabled, fires only after a clock delay has expired. In a timed Petri net graph, timed transitions are labeled using the notation \( v_i \), while zero delay transitions are labeled keeping the notation presented for Petri net graphs \((t_i)\). In figure 2.3, transitions \( t_1, t_4 \) are associated to clocks \( v_1, v_2 \) and are then timed transitions.
2.2 Discrete Event Simulation

A discrete event simulation model is a representation of a real world system. It consists in driving entities through a logic structure in order to obtain a desired behavior and observe such behavior. Following the approach used by Law [24], it is possible to identify a series of features that are common to all simulation models:

- **System State**: Set of state variables used to describe the system at a particular time;
- **Simulation clock**: Counter variable containing information on the actual simulation time;
- **Event list**: Set of events that drive the evolution of the system;
- **Statistical variables**: Output variables used to collect statistical information about system performances.

Simulation is used to evaluate the performance of a system and predict the effect of changes to existing systems. It represents an useful tool for the design and the emulation of a system. As referred, a simulation can be stochastic if one wants to evaluate the effect of randomness on a real system. Randomness can be introduced to model unpredictable system features and components that are random in nature. Examples of random components can be: processing times, service times, entities arrival rate, transportation times, equipment failures, etc. In those cases, the usage of estimated deterministic values (i.e. average, median, mode) can generate errors and invalid models. Randomness in a simulation model can be expressed using random variables. A random variable $X$ is a function whose value is determined by the outcome of an experiment. Its probabilistic behavior is described by the cumulative density function (CDF), $F(x) = Pr(X \leq x)$. This function represents the probability that the random variable $X$ takes values smaller or equal to $x$. The probability density function (PDF) of a random variable describes the relative likelihood for a random variable to assume a given value. For discrete variable the probability mass function (PMF) must be considered [22].

In a simulation study several features are needed to take into account events logic and randomness. Some of those features are, generation of random numbers and distribution, time simulation, events firing logic and observation of relevant state variables. General-purpose languages such as C, C++, Java
can be used to build a simulation model, but for complex systems it is difficult to include and manage effectively all the mentioned features [24]. For this reason, in the last decade special-purpose programming languages for discrete event simulation became popular. The main reason for this, is that the implementation of a simulation model is relatively simple and features such as time simulation management, generation of random numbers, are controlled by the simulation software itself. Examples of simulators are: Arena, FlexSim and Simio. To the scope of this thesis, Simio will be used to build and simulate the model. Simio is a special-purpose programming language based on intelligent objects that provides a new object-based paradigm. Every model in Simio is an intelligent object, with its own properties and can act and interact with other objects to produce the desired result. Simio objects are created using graphical process flows and the operation of building a model is totally oriented to the system logics [22].

2.2.1 Steps in a Simulation Study

Independently from the simulation software used, modeling can be an hard task and it is important to follow a rigorous approach, as the one presented in figure 2.4. During the kickoff of a simulation study the very first step should answer the following questions:
• Which part of the system are we interested in?

• What information do we want to obtain?

Even if those might seem simple questions to answer, in practice it is difficult to identify the objectives of a simulation system. Usually, the only thing that is known a priori is that “there is a problem with the real system” or with “the design of a new system”. For this reason, the very first step is to identify the stakeholders, and “only” after several meetings it is possible to define the objectives of the study (i.e. measure the throughput of a system, identify bottlenecks, evaluate a scheduling, measure the responsiveness of the system). Once the objectives have been set, it is important to formulate the problem and plan the study. Formulation includes a description of the system: identification of the main tasks, operations, work flow and constraints. Planning is important to define the modeling tools to be used, and to identify what data are needed to model the system. Next steps include data collection and treatment, and definition of a first model. In general, it is difficult to model a real world system in its entirety, and it is then important to make assumptions to “simplify the model”. The next task is to implement the model using the chosen simulation software and verify its logic. If the model behaves as expected then it is possible to design experimental scenarios and test the system under desired condition. On the other hand, if the logics can not be verified a review of the previous steps could be useful to refine the model. Once the model has produced the desired results, the last step is the analysis of the output data.
Chapter 3

Simulation model of a Quality Control laboratory

The main objectives of this project are to gather knowledge on the Quality Control system of a contract manufacturing organization, build a discrete event system model and perform simulations to mimic its behavior. Following the steps described in section 2.2.1, the first part of the project has been dedicated to the study of a quality control laboratory and to the identification of the objectives of the study. In particular, all the stakeholders were interested in estimating equipment and labor utilization, and the responsiveness of the laboratory (i.e. the time needed to perform an analysis and the time it takes with the existing configuration). From an operational perspective, different aspects have been addressed during this study: available resources (equipment, laboratory instruments, personnel) and their characteristics/specificities, inputs to the system (jobs, activities), and outputs (results of an executed job/activity).

In a discrete event simulation software paradigm, such as Simio, it is important to define the entities that moves across the machines, and the resources used to process those entities. Also, the processing logic must be robust in order to achieve a good model of the real system. Processing logic is related not only to the work-flow, but also to the resources that are necessary to process a particular job, the timing and sequencing of all the operations.

The study of a dynamic system should be performed in a systemic way. It is important to understand how products and information flow through the system, and to identify all the processes and variables involved. According with the Object-Oriented programming paradigm provided by modern simulation softwares, the system can be modeled as a collection of different objects and entities that interact to provide the desired output. Following this paradigm, the main processes have been modeled as independent systems with their own dynamics, logics and properties, and have been later added to a hierarchical model in order to build the entire system.

The following sections will describe the quality control system, the available data for modeling and its treatment. As conclusion of this chapter, it will be presented the discrete event simulation model.
3.1 Quality control laboratory

As discussed in section 1.2.1, a quality control laboratory of a contract manufacturing organization performs a large number of analysis on a large number of samples. Those samples compete for the same resources (analysts and equipment), have a limited validity and may have different priorities. Quality control laboratories are not very responsive in general, and delays in an analysis may lead to the degradation of a sample, delays in the manufacturing process, slow down method's development. A contract manufacturer should be able to control all those issues in order to provide on-time quality responses to its clients. The ideal scenario is to complete all the operations involved in a single analysis within a short time in order for the system to be responsive to company's needs.

Usually analyses priorities are related with the type of sample to be analyzed and it is possible to identify six main work types within a quality control laboratory:

- In-process Control;
- Stability tests;
- Analytical validation tests;
- Finite product tests;
- Raw material tests;
- Routine and development tests.

In the considered quality control laboratory there exist four different groups of analytical chemistry. Each group has its own equipment and analysts, and manages a limited number of projects. However, it is not uncommon that those groups have to share resources in particular cases. An in-process control analysis has usually higher priority with respect to other types of samples. Stability tests are mostly programmed tests and must be scheduled in advance, in order to be performed on time. Validation tests, are usually more complex to be scheduled. In fact, methods validation is a process that lasts many weeks with various tests that have to be performed to meet a series of criteria. Analytical validation is performed by a specific group which is not included in this study and can be then neglected at this stage. Other tests, related with the process of drug and analytical method development, represent a large percentage of the work in quality control laboratories. In this context the interest will be on the analyses performed in a GMP laboratory, so it will include in-process control, stability, final product, raw materials, routine and development (for projects from phase I to phase III) tests.

3.2 System Overview

The quality control laboratories of the contract manufacturing organization under study process, every year, more than 70000 samples, using limited resources. The management of the laboratories can be difficult and inefficiencies at laboratory level may affect negatively the overall supply chain service level.
Among several quality control laboratories, this study focuses on a particular one, that is located in the Portuguese facility of Loures (Lisbon). It is furnished with 70 equipment of 6 different types and employs 20 analysts working on 3 work shifts. The six classes of equipment are related with particular analytical techniques.

The quality control laboratory is open 24 hours a week. A reason for this is the need to support the manufacturing process with in-process control tests. Other types of tests (i.e. stability, development, validation) are performed 5 days a week, from Monday to Friday.

3.2.1 Analytical techniques

The term analytical technique is related with the specific equipment, while the term analytical method is sample specific and describes how the analysis must be performed. For this study, six analytical techniques have been considered critical to the laboratory efficiency and the relative instrumentations have been modeled. Those techniques are: High Performance Liquid Chromatography (HPLC), Gas Chromatography, Particle Size Analysis (PSA), Karl Fischer Titration (KF), Differential Scanning Calorimetry (DSC) and X-Ray Powder Diffraction (XRPD).

Chromatography is one of the most widely used analytical techniques in quality control of drug manufactured. It is a separation technique, and its objective is to separate a liquid (HPLC) or gas (GC) sample into several components. Particle size analysis (PSA) is a specific test used to determine the size range of the particles in a powder or sample. Karl Fischer titration (KF) is an analytical technique used to determine the water content in solid, liquid or gas samples. Differential scanning calorimetry (DSC) is one of the most important thermal analysis techniques, and it is used to recover information on a sample by heating it in a controlled scenario. Finally, X-Ray powder diffraction is an analytical technique used to identify the molecular structure of a crystal.

Even if those techniques are very different from each other, and so are the relative equipment, it is possible to identify steps that are common to all those analyses (see Figure 3.1): system preparation, equipment setup, equipment verification (or system suitability), sample preparation, analysis and analysis of the results. Some of these steps are performed exclusively by the analyst or by the equipment, while others are performed on the equipment in presence of the analyst. When a sample is received in the laboratory and it is scheduled to be analyzed, the analysis starts with bench work performed by the analyst to prepare the sample and additional solutions/materials necessary to execute the test. The next step is the setup of the equipment performed by the analyst. Once the equipment is ready, it has to be verified to ensure that the equipment is adequate to run the specific test. This step is also referred to as System Suitability and it is performed by the equipment autonomously, with the analyst that has to verify several parameters at the end of the System Suitability run. Once the equipment is verified the equipment is ready for the analysis. At the end of the run, an analyst disassembles the equipment (unless there are new samples to be analyzed with the same method), and analyzes the output data.

Analysts and equipment can be considered as the processing units or resources of the system. Jobs and activities are defined by the samples and by the analytical methods that have to be used to analyze
them. Thus, samples can be considered as the input of the system. In the simulation software paradigm they can be defined as entities that flow through the system and need to be processed by a given number of processing units according to the analytical techniques and methods to be used.

A sample may need more than one test and then, it should be analyzed on different machines. Although, this situation is not very frequent and usually different samples of the same product are available to be analyzed on various equipment. In case there is only one sample available that has to be tested with different analytical techniques, it is separated and it must be analyzed following a precise sequence to avoid its degradation. Assuming that a sample needs to be analyzed by all the machines it should be then processed as follows:

1. Karl Fischer titration

2. Gas chromatography

3. Other analytical techniques

This sequence refers to the start time of an analysis. The analyst has to guarantee that when a sample is opened it must prepare it to be analyzed first on a Karl Fischer equipment, and only after this analysis has started it can be analyzed by a gas chromatograph and so on. It is important to remark that there is no need to wait for its completion, before starting with another analysis. For other analytical techniques there is no such risk and it is not very important to prioritize one over another. However, this situation happens rarely and for modeling purposes every analysis is considered as a different entity (even if it is referred to the same sample) and it is only processed by one equipment.
3.2.2 Available data and Information Systems

In order to develop the simulation model of the system, a detailed study of the available data and information systems has been performed. In the section above, the main components of the system have been identified: analysis work flow, entities (samples) and processing units (analysts and equipment). The next step in the modeling process is to estimate the entities arrival rate, the process logics and the processing times.

In the pharmaceutical industry, due to the compliance with GMPs and GLPs regulations, every product manufactured, analysis performed or even incidents are documented. In particular, quality control makes use of information tools and databases provided by the Laboratory Management Information System (LIMS). A LIMS provides mechanisms to build automated tasks and to integrate them [26]. It is also a powerful tool used to track, document and review analyses performed in a laboratory. The intended scope of such information systems is to increase sample throughput, reduce turnaround times and output quality. LIMS provides support to the analytical chemistry area, and in times of big data analysis, knowledge work automation and in general knowledge management, it represents an opportunity to improve the quality of the analytical operations.

In this context, LIMS database has been used to recover information on samples, projects, equipment and analyses. Every analysis performed in the laboratory can be retrieved from LIMS where the following fields, amongst many others, are stored:

- **Sample number**: primary key field that identifies uniquely a product sample;
- **Project**: project or raw material identifier;
- **Analytical method**: analytical method used to analyze the sample;
- **Instrument**: name of the equipment/instrument used in the analysis;
- **Location**: name of the laboratory where the analysis has been performed;
- **Identifier**: text code that contains information about: type of analysis (i.e. IPC, Stability, Raw material), project, substance;
- **Received date**: date and time at which the sample is created in LIMS;
- **Completed date**: date at which all the tests on a sample have been completed;
- **Analyst**: analyst that performed the test.

From this database it is possible to retrieve information on the number of samples processed within a time period, and it can be used to determine the samples arrival rate. Unfortunately, in the LIMS database there is no detailed information about processing times and analytical methods. For what concerns analytical methods, they still exist in LIMS, but it is not easy to extract information about process logics and operations.

Every equipment has its own dedicated computer and relative software, where it is stored information on every run the equipment does. All chromatograph machines are linked to the same database.
where information about the analyses and the projects are stored. In particular, chromatographic technique runs are made of multiple sample injections through a column to separate the sample components. Information about number of injections and run time of each injection can be retrieved from the chromatograph equipment software. However, this information is partial and it is tricky to get to the processing time of all the analyses.

To retrieve information on processing times and logics a (document management system (DMS)), containing files with the description of the analytical methods, have been studied. This document repository contains information in text form and thus, not structured. From these analytical methods documents it is possible to extract information about:

- **Analytical method name**: Identifier of the analytical method;
- **Equipment**: description of the appropriate equipment to be used to perform the analysis;
- **Solutions and materials**: name and quantity of the solutions needed to perform the analysis;
- **Description of the sample preparation**: description of the steps to do to prepare a sample for the run, and number of samples/solutions to be prepared;
- **Number of injections for the System Suitability**: information about name and number of injections needed to verify the system before a run;
- **Number of injections of the sample solution**: number of injections needed to perform the analysis following the analytical method;
- **Injection run time**: minutes needed to perform a single injection of the described method.

Those databases and information systems provide good quality information about operations in quality control laboratories that can be used for modeling purpose. However, the information is mostly related to the equipment and to the general work flow, but still a little is known about the labor efficiency and the processing times of bench operations. This information have been retrieved “on field” with a time study in the laboratory.

### 3.3 Data processing

One of the biggest challenges of this project was gathering and processing information. In fact, information was distributed on different databases, and in some cases it was stored in a document repository in text form. For the purpose of this project it was necessary to extract the relevant information from all these data sources and merge it into a unique database to be used for modeling.

This simulation study refers to the time period that goes from January 2015 to August 2016. As it will be detailed further on, the first year has been used to calibrate and validate the model, while the last six months have been used to perform simulations and evaluate different simulation scenarios.

The couple sample/analysis can be identified as the central figure of the quality control laboratory system. In fact, a sample should carry on all the information necessary to analyze it. Thus, it seemed...
appropriate to organize data by the unique key given by the couple sample/analysis. Every entry of the
dataset contains informations about:

- **Sample Number/Analysis**: identifies uniquely an analysis performed in the laboratories;
- **Analytical Method**: name of the analytical method used for the analysis;
- **Project**: name of the project the sample belongs to;
- **Equipment**: name of the equipment where the analysis was performed;
- **Sample release date**: time at which a sample is ready to be analyzed;
- **Analysis due date**: time at which the analysis was completed in the real system;
- **Run Time**: time needed to perform an injections for chromatography machines, or the overall
  processing time for other class of machines.
- **Number of System Suitability preparations**: number of preparations needed to verify the system
  (only for chromatographic techniques, in other cases a value of 0 is considered by default);
- **Number of analysis preparations**: number of samples/materials to be prepared to run an analysis;
- **Number of System Suitability injections**: number of injections needed to perform the System
  Suitability (only for chromatographic techniques, in other cases a value of 0 is considered by de-
  fault);
- **Number of Analysis injections**: number of injections needed to perform an analysis (it is equal
  to 1 by default, except for chromatographic techniques that may need more than one injection);

As referred, this information comes from four different data sources: Document Management Sys-
tem (DMS), Equipment dedicated Software, manual records retrieved in the laboratory, and Laboratory
Information Management System (LIMS). LIMS have been considered as the central information source
in the design of a more complex database to be used for modeling purposes, as it is shown in figure 3.2.
Other tables, such as *Analytical Methods and Instruments* have been created using other information
sources. *Sample Entity* table condenses all the information needed to process a sample and it could be
effectively used for planning and scheduling. In fact, it completely characterizes an analysis providing
information on the sample, the analytical method to be used and the type of equipment needed.

### 3.3.1 LIMS data

The Laboratory Information Management System (LIMS) is a powerful tool for laboratory management.
As described before, it can be used to recover information about samples that have been analyzed in the
laboratory. This information is very useful to estimate the arrival rate of the samples and the number of
samples per analytical technique. For the considered laboratory, and set of equipment, the percentages
Figure 3.2: UML representation of the designed database.

of each analytical technique in terms of number of analysis performed during the period January 2015 - August 2016 is shown in table 3.1.

For this study have been considered: 36 Liquid chromatography equipment, 17 Gas Chromatography equipment, 3 Particle Size Analyzers, 8 Karl Fischer equipment, 1 Dynamic Scanning Calorimeter and 1 X-Ray powder diffractometer.

<table>
<thead>
<tr>
<th>Analytical Techniques</th>
<th>HPLC</th>
<th>GC</th>
<th>PSA</th>
<th>KF</th>
<th>DSC</th>
<th>XRPD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage</td>
<td>40.7%</td>
<td>25.7%</td>
<td>18.8%</td>
<td>7.1%</td>
<td>4.0%</td>
<td>3.7%</td>
</tr>
</tbody>
</table>

Table 3.1: Percentage of analysis per analytical technique.

LIMS represents a complete database, with structured data that is easy to access. However, it lacks of information and tools that could be used for planning and scheduling, or in general for modeling and optimization purposes. Additional information regarding processing times and tasks have been retrieved from other databases.

3.3.2 Information extraction algorithm

Information extraction (IE) is a task used to transform information from an unstructured form to structured data. It often consists in processing human language text with the objective to find and extract data and concepts. Information extraction algorithms are usually document specific or template specific.
IE assumes the existence of a pattern to be identified, and even little variation in the documents (i.e. misspelled words, different terminology, difference in paragraph numbers) can affect the result of the information extraction process.

In this context, analytical methods document existing in the document management system were stored in two different languages (Portuguese and English), and some differences were found in the way the documents are written. To guarantee robustness of the IE algorithm an iterative process was needed to detect those differences and extract correctly the information.

As referred, an analytical method report contains information about: run time, analysis type, solutions to be prepared and number of injections. It was then implemented an information extraction algorithm using R programming language, to identify and extract information. The algorithm converts a pdf document into a string array variable and finds table structures from a docx file of the same document storing them into a data frames. It then identifies the language of the document and choses the relative vocabulary to be used to match words and patterns. The algorithm extracts analytical method name and run time from the string array variable. It then searches in the table structures within the document the following information: number of analysis injections, number of analysis solutions, number of system injections, number of system solutions and sequence of injections. In case this information is not found, it searches those values analyzing the text stored in the string array from the pdf file. A description of the algorithm can be found in figure 3.3. The algorithm has been developed only for chromatographic methods, and it has processed more than 900 analytical methods reports. Data collected with this algorithm have been used to generate the Analytical Method table (see figure 3.2). The code of the information extraction algorithm can be found in the appendix A.

3.3.3 Time study and work measurement

Work measurement is a technique designed to estimate the time for a worker to perform a specific manufacturing task. It must be performed for many reasons, such as the detection of potential inefficiencies in the manufacturing process. It is not simple to measure workers time, and this measurement is often performed on empirical basis. In fact, it is rare to have information systems capable to measure the time a worker is producing active work [15]. When it comes to the laboratory environment, labor represents an important percentage of the activities performed in quality control. Measuring laboratory personnel work can be even harder, given the limitations that exist in an environment regulated by Good Laboratory Practices. Also, time studies can have a negative impact on workers performance and this aspect must be considered during the analysis of the results of a time study.

In a quality control laboratory, analysts are responsible for sample preparation, system setup, equipment cleaning and analysis data processing. To estimate those parameters a time study has been designed and it has been performed with the following objectives in mind:

- Understand better the work-flow in analytical chemistry laboratories;
- Identify analysts' tasks;
- Divide equipment “hands-on” and “hands-off” tasks;
• Estimate times for those tasks.

It was asked to analysts to collect data, during a period of one month, by filling a manual form model, as the one presented in figure 3.4.

During the time study, 20 analysts filled the forms, and data of more than 500 analyses performed on 6 different equipment types have been collected. From this study it was possible to obtain an estimative of the processing times of analyst related tasks.

This information completes the one obtained with the information extraction algorithm. For instance, considering the task of sample preparation, given an analytical method, from the Analytical Method table it is possible to retrieve information on the number of samples to prepare. This data can be combined with the estimated processing time for sample preparation in order to get the overall preparation time for a given method. Also for analytical techniques different to the liquid and gas chromatography, this data can be used to estimate also the equipment processing times that were not available otherwise. In table 3.2 is shown the source of information for every task and for every class of equipment.
3.4 Quality Control laboratory model

The developed simulation model of the quality control laboratories is composed by the following objects:

- Equipments;
- Analysts;
- Work locations;
- Samples.

An equipment can be seen as a series of processes needed to perform an analysis. Those processes are: sample preparation, system suitability, analysis preparation, analysis run and data processing. Even if not all those tasks are performed at the equipment location, the modeling strategy chosen is process-oriented. The spatial components have been added to those processes selecting the physical places in the quality control laboratory model where the analyst has to go to perform the requested tasks.

An analyst has been modeled as a secondary resource in the system. His/her presence is necessary to perform specific tasks. Moreover, to do any operations the analyst has to go to a specific work location (bench, equipment or computer). The term work location is referred to the physical position of a laboratory bench, where solution are prepared, or to the location of an equipment, or to the location of a computer, where the analyst can process results data from the analyses. An analyst works in shifts. Simio has a nice feature that allows to implement different work shifts for every worker in the system, defining the hours an analyst is available to perform active work.

A total of twenty analysts have been considered for this study: six analysts work on 12 hours rotating shifts, four rotates on two different 8 hours shifts, and ten analysts work 5 days a week on a fix 8 hours shift. These shifts are summarized in table 3.3.

Samples have been modeled as input entities that travel in the system to perform certain tasks. A sample entity carry on information about the equipment where it is scheduled, the number of samples that an analyst has to prepare, the number of injection of the system suitability, the number of injections of

<table>
<thead>
<tr>
<th>Equipment type</th>
<th>Sample Preparation</th>
<th>Analysis</th>
<th>Data Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPLC</td>
<td>IE Algorithm and Time Study</td>
<td>IE Algorithm</td>
<td>Time Study</td>
</tr>
<tr>
<td>GC</td>
<td>IE Algorithm and Time Study</td>
<td>IE Algorithm</td>
<td>Time Study</td>
</tr>
<tr>
<td>PSA</td>
<td>Time Study</td>
<td>Time Study</td>
<td>Time Study</td>
</tr>
<tr>
<td>KF</td>
<td>Time Study</td>
<td>Time Study</td>
<td>Time Study</td>
</tr>
<tr>
<td>DSC</td>
<td>Time Study</td>
<td>Time Study</td>
<td>Time Study</td>
</tr>
<tr>
<td>XRPD</td>
<td>Time Study</td>
<td>Time Study</td>
<td>Time Study</td>
</tr>
</tbody>
</table>

Table 3.2: Time information used to model analyses tasks.
the analysis and the injection run time. When it is allocated to an equipment, the equipment reads those values and processes the sample accordingly. However, the system suitability is not always needed. In particular, it has been assumed a validity of 24 hour for a system suitability. In this time window an equipment can analyze samples that makes use of the same analytical method, without having to run again the system suitability. For this reason it is preferable that a sample goes to an equipment that is already running the same analytical method.

Samples informations are stored into a data table, that is provided as input to the model. The arrival date of a sample to the laboratory is assumed to be the Received date from LIMS and its due date is assumed to be the Completed date from LIMS. Information on the analytical method and relative run time, solutions and injections are retrieved from the Analytical Method table.

In figure 3.5 is presented an high level flow chart of the simulation model. The simulator generates samples according to the input data provided, and it allocates a job to the equipment group that can process it. The scheduling is done within the equipment group where the sample entity is allocated.

### 3.4.1 Generic equipment model

A generic equipment model has been implemented to simulate the analysis process. The idea of this model is to have a general process that can be applicable to every analytical technique. An analysis is composed by different tasks: system preparation (includes preparation of solutions and equipment setup), system suitability, analysis sample preparation, analysis and data processing. A timed Petri net have been used to represent the underlying discrete event system (figure 3.6). The Petri net is characterized by a set of places $P$, that represents the five tasks described above, more auxiliary tasks needed to avoid situation like performing the system suitability and the analysis at same time.

When samples enters the system they wait in a queue and are scheduled according to a policy that estimates the lateness of the sample and gives higher priority to samples with higher lateness. Lateness $L_i$ is defined as the difference between the expected sample completion time $C_i$ and the due date $D_i$:

$$L_i = C_i - D_i.$$  

The scheduling policy tends to minimize the maximum lateness. With this approach, the completion time of an analysis in the simulation model should be closer to the one existing in the real data. Furthermore, the Petri net is characterized by four timed transition corresponding to processing times of each task.
Note that the transition $v_4$ takes into account two processing times related respectively with the system suitability and the analysis preparation. The initial marking in the Petri net $m_0 = [1, 0, 0, 0, 0, 1, 0, 0]$ represents the system empty and ready to process samples.

As referred, the system suitability is needed only if the sample to be processed needs to be analyzed with a different analytical method with respect to the last method performed on the equipment. Also, the system must be performed every 24 hours, and the internal variable $last\_method$ is set to zero when this time expires. In order for the equipment to decide if the system is needed, every time a sample enters the equipment, the model evaluates if the sample has to be processed using the same analytical method. If this is the case, it sets to zero a boolean variable $system\_needed$. This variable is used as multiplier to the processing times of system preparation and system suitability tasks. Thus, the sample will still pass through every processing station but it will have a processing time equal to zero in those that refers to the system suitability.

### 3.4.2 Processing time distributions

Many of the processing times in the model are stochastic in nature, while others are deterministic. For instance, workers processing times are stochastic in nature, since it is not probable for a human workforce to replicate a work exactly in the same amount of time. Deterministic processing times are mostly associated with equipments tasks. However, part of those processing times are assumed to be stochastic given the empirical nature of the available data. Processing times for HPLC and GC equipment are assumed to be deterministic, since they depends on the analytical method and exact
Figure 3.6: Discrete Event System model of a generic equipment.
information was retrieved using the *information extraction algorithm*. For other equipments and for tasks that depends on the work of an analyst (i.e. equipment setup, sample preparation, data processing), the processing times have been obtained fitting curves to the collected data.

When data comes from direct observation of the real system a theoretical distribution is fitted to the data and it is used in the simulation to generate random values. In literature different types of distributions are used in simulation studies, such as *normal distribution*, *gamma distribution*, *uniform* and *triangular*. Uniform distribution can be used for a quantity that seems to variate randomly between a minimum(a) and a maximum(b). The probability density function of a uniform distribution is defined as:

\[
f(x) = \begin{cases} 
\frac{1}{b-a}, & \text{if } a \leq x \leq b \\
0, & \text{otherwise.}
\end{cases}
\]

The triangular distribution is often used in simulation, especially when the available data is scarce. The triangular probability distribution is attractive, and it only requires the knowledge of three parameters: minimum (a), maximum(b) and most likely(b). One difficulty with the triangular distribution approach is that it requires subjective estimates of the parameters. Usually the data is collected for limited period, and it could not be representative of the real situation [24]. A probability density function for a triangular distribution is defined as:

\[
f(x) = \begin{cases} 
\frac{2(x-a)}{(b-a)(m-a)}, & \text{if } a \leq x \leq m \\
\frac{2(b-x)}{(b-a)(b-m)}, & \text{if } m \leq x \leq b \\
0, & \text{otherwise.}
\end{cases}
\]

Given the limited data available for estimating the processing times, uniform and triangular distributions have been used to approximate the data. The most likely value has been estimated using the *mode* or the *median*. In particular, the *mode* was used when the number of occurrences of the most frequent value was two times greater than the number of occurrences of the second most frequent value. In all other cases the *median* was considered better to deal with variations in the empirical data. In figure 3.7, times distribution for sample preparation, equipment setup, system verification and data processing are shown. The histograms have been computed from the values registered in the manual forms by the analysts for each task performed. As expected, there is a big variation in the data that is given by diversity of operations but also from inaccuracy in the process of data collection.

For analytical techniques different from the gas and liquid chromatography, the time studied served also to estimate the processing time of the equipment. For those techniques a big variance for some of the tasks has been found. The processing time distributions for the particle size analysis (PSA) are shown in figure 3.8. Note that for the setup of the equipment the data do not seem to follow a triangular, normal or any other known distribution. In this case seemed appropriate to approximate the data with a uniform distribution between the minimum and the maximum value.

The X-ray powder diffractometer (figure 3.9) presents data that are easier to fit with triangular distributions. However, for the analysis task it presents high values that were validated with laboratory experts, and can not be neglected.
Figure 3.7: Liquid Chromatography (HPLC) estimated times and probability distribution functions.

Figure 3.8: Particle Size Analysis (PSA) estimated times and probability distribution functions.
Figure 3.9: X-ray Powder Diffraction (XRPD) estimated times and probability distribution functions.

The processing time distributions for all tasks and equipments are presented in table 3.4. All the processing times have been assumed stochastic except for the chromatographic equipments, where the analysis time have been retrieved for each method from the respective analytical method report. Time distribution figures for other equipments are shown in Appendix B.

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Sample Preparation</th>
<th>Equipment Setup</th>
<th>Analysis</th>
<th>Data Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPLC</td>
<td>$Tr(5, 40, 100)$</td>
<td>$Tr(25, 50, 65)$</td>
<td>Deterministic</td>
<td>$Tr(2, 5, 10)$</td>
</tr>
<tr>
<td>GC</td>
<td>$Tr(5, 10, 35)$</td>
<td>$Tr(2, 5, 30)$</td>
<td>Deterministic</td>
<td>$Tr(10, 20, 45)$</td>
</tr>
<tr>
<td>PSA</td>
<td>$Tr(5, 7.5, 21)$</td>
<td>$U(1, 32)$</td>
<td>$Tr(5, 10, 30)$</td>
<td>$Tr(2, 6, 10)$</td>
</tr>
<tr>
<td>KF</td>
<td>$Tr(1, 6, 25)$</td>
<td>$Tr(4, 8, 16)$</td>
<td>$Tr(2, 5, 56)$</td>
<td>$Tr(10, 20, 30)$</td>
</tr>
<tr>
<td>DSC</td>
<td>$Tr(5, 10, 30)$</td>
<td>$Tr(1, 3, 12)$</td>
<td>$U(12, 180)$</td>
<td>$Tr(4, 9, 16)$</td>
</tr>
<tr>
<td>XRPD</td>
<td>$Tr(5, 10, 20)$</td>
<td>$Tr(1.3, 20)$</td>
<td>$Tr(11, 25, 203)$</td>
<td>$Tr(2, 7, 30)$</td>
</tr>
</tbody>
</table>

Table 3.4: Processing times distributions ($Tr(a, b, c)$: Triangular pdf; $U(a, b)$: Uniform pdf). Time unit is minutes.
3.5 Simio Model

To deal with the high number of machines and analysts, a hierarchical model has been created using a dedicated library that implements the following objects:

- **Equipment**: is the implementation of the single machine model presented in section 3.4.1. This object has been used to build complex objects, such as machine groups.

- **HPLCs**: equipment group containing all the HPLC machines considered in the study. In this model the physical location of the machines and the lab benches have been implemented.

- **GCs**: equipment group formed by the Gas chromatography machines.

- **KFs**: as for the GC and HPLC, a machine group of the Karl Fischer titration machines have been implemented as an object.

The implementation of the generic equipment Simio object is shown in figure 3.10. The proposed implementation follows the Petri net chart model of figure 3.6. Moreover, processing times, secondary resources needed for processing (i.e. analysts) and physical locations of bench, computer and equipment are set as inputs to the model and can be easily modified externally.

The complete Simio model was developed under the following assumptions:

1. *The couple sample/analysis has been considered as a single entity*. Every sample entity is processed using only one machine;

2. *The system suitability has a validity of 24 hours*. This means that the system suitability does not need to be re-executed within 24 hours, unless a new sample using a different analytical method is scheduled to the same machine;

3. *Analysts can perform every type of work regardless the work shift they are allocated to*. In the real system every work shift is allocated only to few operation types (i.e. in-process control, stability, development).
This library has been used in a new Simio model, which implements the complete quality control laboratory system (figure 3.11). The model runs using a data table as input, which represents an arrival table for the samples and it is based on LIMS data.

The implemented model imitates the original system by driving each sample towards the equipment that was used in the real system to process that particular sample. With this configuration it is possible to estimate which equipments have been used the most and also it is possible to compare the throughput of the simulation model with the real system.

![Figure 3.11: Quality Control laboratory model: Simio 3D view.](image)

For simulation purposes, a modified model has been implemented to evaluate the effect of changes in the scheduling policy. In particular, it has been estimated that an equipment is capable to process at most five samples with the same analytical method within a system suitability validity. Therefore, an *ad-hoc* scheduling rule has been created for this model. This rule tries to allocate a sample to a machine that has a shorter queue, in case the machines that are already running the same method have four or more samples in queue. The scheduling rule can be described by the following equation, which computes a rank for all the possible machines and allocates the entering sample to the one with the lowest rank value.
Given the set of available equipments \( Eq = \{ Eq_1, Eq_2, ..., Eq_n \} \), and a candidate sample entity \( Sam \), \( \forall Eq_i \in Eq \) computes:

\[
\text{rank}(Eq_i) = \begin{cases} 
\text{nInQueue}(Eq_i) - 4 \times 1, & \text{if } \text{LastMethod}(Eq_i) = \text{Method}(Sam) \\
\text{nInQueue}(Eq_i) - 4 \times 0, & \text{if } \text{LastMethod}(Eq_i) \neq \text{Method}(Sam).
\end{cases}
\]

Where \( \text{nInQueue}(Eq_i) \) and \( \text{LastMethod}(Eq_i) \) represent respectively, the number of samples waiting to be processed on equipment \( i \) and the last analytical method performed by the equipment \( i \). \( \text{Method}(Sam) \) is the analytical method to be used to process the sample \( Sam \).

The two models have been simulated in different scenarios to evaluate the quality of the developed model and to get insights on the efficiency of the system. In the next chapter, the simulation results will be presented and a comparison between the two scheduling techniques will be performed.
Chapter 4

Simulation results

The main objectives of a simulation study are to measure the performance of a real system and to evaluate how modifications on the system affect its performance. In this study, the following aspects were considered: system throughput, tasks processing time, equipment usage rate, system responsiveness and employee utilization. The first three metrics were compared with the available data to validate the model. In particular, it was considered a one year period data (2015) for validation purposes. Statistical independent replications of the simulation model, using the same input parameters, were run to retrieve information on the behavior of the model.

The throughput of the system, in this context, represents the percentage of the number of samples processed over the total number of samples that entered the system. Considering the total number of samples created by the Source object \((NIS)\) in Simio, and the total number of samples destroyed by the Sink object \((NOS)\) this metric can be easily computed as:

\[
\text{Throughput} = \frac{NIS}{NOS}.
\]

The processing time for each task of an analysis can be computed by dividing the total processing time of each process by the number of analysis performed. This time represents an average and it must be complemented with its standard deviation. This value has been computed for all the class of machines and it can be compared with the estimated processing times to check if the model behaves as expected. A metric that is related with the total processing time, is the Equipment usage rate. It represents the amount of time that the equipment is performing active work over the total time available, and can be estimated by the formula:

\[
\text{Usage}\% = \frac{EPT}{TRT},
\]

where \(EPT\) is the Total Equipment Processing Time, and \(TRT\) is the Total Run Time.

The last two metrics that have been considered are: system responsiveness and employee utilization. The first one is related with the time the laboratory takes to produce a finished job (analysis). It represents an average of all the total system performance and it tells us how much time is needed in average to perform an analysis if the system is already in a steady state. Employee utilization refers to
the amount of time an analyst is busy performing bench work (i.e. preparing samples, processing data) over the total time available.

In the next sections those metrics will be used to assess the quality of the simulation model and to measure the performances of the system.

4.1 Model Validation

To validate the model, several simulations have been performed over a time period of one year (2015). The first aspect that was considered was the throughput of the system. The model was capable to perform all the analyses existing in the arrival table for the referred period, having than a throughput of 100%. This means that all the samples that entered the system during the year 2015 were processed and the relative results were produced on time.

To get more information about the quality of the simulation model, other parameters have been taken into account. During the period from October 2015 to January 2016, it was performed a time study in the Quality Control laboratory that was used to assess the usage rate of a pool of equipment (HPLCs and GCs). This data has been compared with data from the simulation model considering the same machines. In figure 4.1 are shown the usage rates for the selected HPLC equipments coming from the simulation model and from the real data.

![Figure 4.1: HPLC usage rate comparison between simulation and real data.](image)

It can be noticed that there are not many differences between real and simulated data. In some cases, machines from the simulation model present values that are slightly lower than the ones coming from the real data. This can be explained by the fact that not all the analyses are in LIMS database (i.e. analyses for non GMP projects, fast analyses, calibration analyses).

Similar results have been found also for data from Gas Chromatography equipment (figure 4.2). Again, the simulation model presents values slightly lower than the real data, but in general, except for
the equipment GC8, the difference between real and simulation data is around 1.5%.

![Figure 4.2: GC usage rate comparison between simulation and real data.](image)

Given the lack real data to use for comparison purposes, it is difficult to validate the model behavior for other classes of equipments. However, in the next section, the overall performance of the real system will be evaluated and the processing times from the simulation model will be compared with the expected processing time estimated in section 3.4.2.

### 4.2 Real system model simulation

To simulate the real system, data coming from LIMS was used to define the number of analyses performed. The model tries to imitate the real system by replicating the analyses on the same machines that were used during the considered period. For instance, if a given analysis was performed on a given equipment, the simulation model will allocate this analysis to the same equipment. The period considered for the simulation was from January 2016 to August 2016.

**Throughput**

For the considered period, the simulation provided a throughput of 100% in all the simulation runs (50 statistical independent runs). The model was able to process all the samples, in the system with no delays on the due date.

**Tasks processing time**

As referred in section 3.4.2, the times were estimated using both real data (for chromatographic techniques) and empirical data obtained observing the real system. In this context, an analysis will be
divided into five main tasks, according to the number of processing units existing in the machine model presented in 3.4.1:

- **System preparation**: includes samples preparation and equipment setup;
- **System suitability**: refers to the verification of the equipment suitability;
- **Analysis preparation**: refers to the preparation of samples for the analysis;
- **Analysis**: refers to the analysis of the sample;
- **Data Processing**: task performed by an analyst to analyze the results of an analysis.

In table 4.1 are presented the average time per task obtained on each equipment type. Comparing those values with the estimated processing times of table 3.4, it can be noticed that the model is generating similar values of the processing times. However, at first it may seem that the times for the HPLC and GC equipments for the preparation tasks are higher than the one estimated in section 3.4.2. This is given by the fact that a chromatography analysis requires the preparation of more than one samples according to the analytical method. For example, if we consider that typically five solutions are prepared for the system suitability and two for the analysis, by multiplying these numbers for the central value of the sample preparation triangular distribution (40 minutes) we get 200 minutes for the system preparation and 80 minutes for the analysis preparation, which are not so different with the values shown in table 4.1.

<table>
<thead>
<tr>
<th>Equipment</th>
<th>System preparation time</th>
<th>System suitability time</th>
<th>Analysis preparation time</th>
<th>Analysis time</th>
<th>Data processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPLC</td>
<td>207 ± 14</td>
<td>227 ± 11</td>
<td>96 ± 3</td>
<td>85 ± 1</td>
<td>59 ± 2</td>
</tr>
<tr>
<td>GC</td>
<td>67 ± 5</td>
<td>152 ± 5</td>
<td>48 ± 1</td>
<td>62 ± 1</td>
<td>39 ± 1</td>
</tr>
<tr>
<td>PSA</td>
<td>n.a.</td>
<td>n.a.</td>
<td>31 ± 1</td>
<td>32 ± 1</td>
<td>16 ± 1</td>
</tr>
<tr>
<td>KF</td>
<td>n.a.</td>
<td>n.a.</td>
<td>45 ± 3</td>
<td>30 ± 2</td>
<td>29 ± 2</td>
</tr>
<tr>
<td>DSC</td>
<td>n.a.</td>
<td>n.a.</td>
<td>47 ± 2</td>
<td>101 ± 3</td>
<td>24 ± 2</td>
</tr>
<tr>
<td>XRPD</td>
<td>n.a.</td>
<td>n.a.</td>
<td>46 ± 1</td>
<td>88 ± 3</td>
<td>30 ± 1</td>
</tr>
</tbody>
</table>

Table 4.1: Simulation model average processing times. Time unit is minutes.

Using these values it is possible to infer the total time of an analysis for each class of equipment. For the liquid chromatography, a complete analysis (including system suitability and system preparation) may last 673 minutes. If the system suitability is not needed, the analysis will only perform three out of five tasks (analysis preparation, analysis and data processing), and it may last 240 minutes. The same reasoning can be used for the gas chromatography, where a complete analysis can take around 460 minutes, while an analysis without system suitability takes around 149 minutes. As it can be seen in figure 4.3 and in figure 4.4, both for gas and liquid chromatography, the system suitability and its
preparation has a major impact on the time of analysis and a good scheduling strategy should try to minimize the number of system suitability needed, and maximize the number of samples processed within the same system validity.

Figure 4.3: HPLC: tasks percentage on total time.

Figure 4.4: GC: tasks percentage on total time.

For what concerns other classes of equipment, the analysis is composed by three main tasks. In figure 4.5 it can be seen the percentage of time taken by each of the tasks for Particle size analysis, Karl Fischer titration, Dynamic scanning calorimetry and X-ray powder diffraction.
Figure 4.5: Other equipments: tasks percentage on total time.

Equipment Usage Rate

Equipment usage rate is a metric used to understand how much time the equipment is performing active work. This value is calculated over 24 hours a day, 7 days a week. However, this is not the real scenario, since those machines are not able to perform analyses in a complete autonomous way, and they still need an analyst to start an analysis. With this considerations in mind we introduce a few values to understand how much any class of equipment is used:

- **HPLC**: Liquid chromatography equipments are used in average the 4% of the time. This low value is given by a large number of equipment that did not process any samples (or that processed just few samples) during the considered period. Neglecting those equipment the average usage rate is around the 10%. This is still a low value, but as referred, in the database used to generates inputs to the system, some type of work is missing. As a consequence, in the real system we expect those value to be slightly higher. The equipment with the highest usage rate works 20% of the time;

- **GC**: Gas chromatography equipment have usage rates similar to the one found for the HPLC equipments. In this case the average usage rate is 11%, with the most used equipment occupied the 21% of the time;

- **PSA**: For this equipment the usage rate have been estimated around 8%.

- **KF**: The class of equipment with the lowest usage rate have been found in the Karl Fischer equipments. In this case the usage rate is around the 4% per equipment. This can be explained
by the fact that a Karl Fischer analysis is quite fast and follows the processing time distribution: $Tr(2,5,56)$. For this reason many analyses on this machine are processed in few minutes;

- **DSC**: Dynamic scanning calorimetry is the equipment with the highest usage rate, that is around the 13%;
- **XRPD**: Similar to the DSC case, also for the X-ray powder diffraction analysis the usage rate is around the 12%.

**System Responsiveness**

To evaluate the responsiveness of the system Simio provides a metric called *Time in System*, which counts the total time each entity spends in the system. For each type of sample entity (*HPLC sample, GC sample, PSA sample, KF sample, DSC sample* and *XRPD sample*) the software counts the average time in system which can be interpreted as the average time needed to perform an analysis considering all the samples that are already in the laboratory. This value depends of different factors, such as system capacity, number of analysis in the system, processing time, available analysts, etc. The time in system obtained in simulation for each sample type is shown in table 4.2:

<table>
<thead>
<tr>
<th>Sample Type</th>
<th>Average Time in System</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPLC sample</td>
<td>31 ± 1.2</td>
</tr>
<tr>
<td>GC</td>
<td>10 ± 0.8</td>
</tr>
<tr>
<td>PSA</td>
<td>4 ± 0.5</td>
</tr>
<tr>
<td>KF</td>
<td>3 ± 0.2</td>
</tr>
<tr>
<td>DSC</td>
<td>5 ± 0.8</td>
</tr>
<tr>
<td>XRPD</td>
<td>6 ± 0.6</td>
</tr>
</tbody>
</table>

Table 4.2: Average time in system for each sample type. Time unit is hours.

**Employee utilization**

In quality control laboratory, employees represent an important part of the system. They are necessary to perform different tasks during an analysis. Their work has been calculated considering the time they perform active work, such as preparations, setup on a given equipment, data processing, etc. In average, an analyst is busy doing active bench work the 35% of the time. Separating this value for each of the three work shifts considered in section 3.4, we obtain for the *Shift 1* a utilization of 51%, for the *Shift 2* a value of 26%, and for the *Shift 3* a utilization rate of 22%. This value highlights the fact that analysts perform a great amount of hidden work, that is not directly related with bench work.
4.3 Capacity experiment

In this section it will be described another simulation performed on the two implemented models for comparison purposes, considering the time period January 2016 - August 2016. The first model is the model of the real system, while the latter is a modified version that schedules samples according to a policy that tries to minimize the waiting time for a sample. The objective of this experiment is to evaluate the capacity of the two systems in processing samples. At this purpose, it has been designed an experiment varying the number of samples entering the system. In particular three scenarios were created:

- **Scenario 1**: the two models have been simulated with the number of samples entering the system equal to the one existing in the original input arrival table;
- **Scenario 2**: the number of samples entering the system is increased by the 25%;
- **Scenario 3**: the number of samples entering the system is increased by the 50%.

Those two models have been tested in order to get information on the responsiveness of the system. As expected, the different scheduling techniques used in the two models can have a big impact on the responsiveness of the system. The values of the time in system in the two models, around the three scenarios are shown in table 4.3:

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Scenario 1</th>
<th>Scenario 2</th>
<th>Scenario 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPLC</td>
<td>31 ± 1.0</td>
<td>47 ± 1.0</td>
<td>57 ± 1.1</td>
</tr>
<tr>
<td></td>
<td>15 ± 1.0</td>
<td>31 ± 2.3</td>
<td>69 ± 7.9</td>
</tr>
<tr>
<td>GC</td>
<td>10 ± 0.9</td>
<td>18 ± 1.1</td>
<td>29 ± 1.2</td>
</tr>
<tr>
<td></td>
<td>7 ± 0.9</td>
<td>18 ± 0.9</td>
<td>52 ± 10.8</td>
</tr>
<tr>
<td>PSA</td>
<td>4 ± 1.1</td>
<td>8 ± 1.0</td>
<td>10 ± 1.0</td>
</tr>
<tr>
<td></td>
<td>4 ± 0.1</td>
<td>19.5 ± 3</td>
<td>152 ± 42.1</td>
</tr>
<tr>
<td>KF</td>
<td>3 ± 0.2</td>
<td>3.3 ± 0.5</td>
<td>3.7 ± 1.0</td>
</tr>
<tr>
<td></td>
<td>3 ± 0.2</td>
<td>8.5 ± 1.0</td>
<td>18 ± 1.5</td>
</tr>
<tr>
<td>DSC</td>
<td>5 ± 0.2</td>
<td>8 ± 0.5</td>
<td>10 ± 0.7</td>
</tr>
<tr>
<td></td>
<td>7 ± 0.2</td>
<td>27 ± 4.9</td>
<td>237 ± 42.1</td>
</tr>
<tr>
<td>XRPD</td>
<td>6 ± 1.0</td>
<td>9 ± 1.2</td>
<td>12 ± 1.5</td>
</tr>
<tr>
<td></td>
<td>8 ± 0.3</td>
<td>39 ± 8.0</td>
<td>312 ± 41.1</td>
</tr>
</tbody>
</table>

Table 4.3: Samples average time in system: comparison between models. Time unit is hours.

In the first scenario, the second model performs better. In fact, it is able to perform HPLC samples two times faster than the original model. Since the second model tries to use more equipment, there are many samples that can be processed in parallel. Same consideration can be used for the gas chromatography samples, that are processed 30% faster. For other equipment, the situation is similar except for the DSC and XRPD machines where the performance decreases. This is given by the fact that
only one X-Ray and three DSC machines exist in the laboratory, and changes in the scheduling does not affect directly the performance of these machines. In Scenario 2, the modified model is still able to perform faster HPLC samples, but it has worse performances on other types of equipment. The situation gets even worse in the third scenario, where the second model is very slow in processing samples. For X-ray powder diffractometer the average time in system for the modified model is 312 hours against the only 12 hours of the real model. This situation can be explained by the fact that since the modified model tries to allocate more samples to empty machines, and there are not enough analysts to process them all, part of the samples suffers a huge delay. On the other hand, the first model that uses always a lower number of machines has to perform less system suitabilities and analysts are often available to end a task.

Comparing the two models and the different scenarios it is clear how scheduling has a huge impact on the performance of the system. Rational scheduling can improve the productivity of a quality control laboratory, while poor scheduling technique can lead the system to a state in which it is no more able to process jobs on time.
Chapter 5

Conclusions

The main objective of this project was the development of a simulation model of a quality control laboratory. The model represents in detail the entire work flow of the quality control laboratory and it is intended as a support tool for planning, scheduling and decision making. To achieve this objective, a generic framework for information treatment and organization was build. In particular, different databases were merged into a single one. This allows an easy access to relevant data that can be used to describe the operations in quality control and that can serve as basis for planning and scheduling. This data, organized in a worksheet is used as input to the simulation model, and it provides not only the history of the analyses but also informs on the type of work, processing times and sequence of operations.

As presented in chapter 2, Discrete Event Simulation can be an appropriate tool for the study of manufacturing systems and it can be applied to a quality control system. Moreover, Petri net formalism provides a graphical way to design and represent the system that can be used as guideline to implement the model in a simulation software. To the scope of this thesis, the academic version of Simio software was used both for modeling and simulating. In order to make easy modifications on the model, the inputs have been structured in data tables. Simio objects for equipment and groups of machines have been implemented into an external library that can be expanded and used to model other laboratories or to test modification on the proposed laboratory model. Also, the implementation of a generic machine model applicable to different types of equipment make the model flexible to be rearranged in different scenarios.

5.1 Data processing framework

As presented in the literature review in chapter 1, in order to develop a generic framework for quality control laboratory management, data organization is a critical task. Information is often spread around many databases and information systems often lack of integration mechanisms. Laboratory Management Information Systems are useful to track samples and analyses, but are not appropriate for planning, scheduling and decision making. With this purpose, the very first step in the development of a simulation model was the organization of the relevant information into a unique data table. A database structure like
the one proposed in this thesis will allow the development of a simulation model, but also easy access to relevant information (i.e. processing times, number of tests, analytical method description) that is stored in pdf documents or in others databases.

5.2 Quality Control model performance

The development of a simulation model of the quality control laboratory can be seen as a first step towards the automation of management tasks such as planning and scheduling. The model represents a powerful tool for the estimation of performance parameters such as task processing times, equipment usage rate, system throughput and responsiveness. Given a well structured input table, it can be used for planning purposes, such as the prediction of the resources needed to meet the due dates and the evaluation of the capacity of the laboratory.

The validation of part of the model related to High Performance Liquid Chromatography and Gas Chromatography equipments with real data suggests the good quality of the developed model. In fact, as stated in chapter 3, analyses on those classes of equipment represent the 68% of the total numbers of analyses. However, the lack of information for other equipment types made difficult the validation of processing times and equipment usage rate. The data collected manually refers to a one month period and could not represent the normal scenario. Although, they were assumed to be correct and in fact the system throughput was not affected negatively by this assumption.

The performance of the quality control system was evaluated on different levels. In first analysis, the model was able to replicate the history. The simulation model of the laboratory performed all the analyses meeting the due dates. For what concerns equipment usage rate, it was found to be very small (around 10%). This is given by the fact that data coming from LIMS were not complete, and as consequence part of the analyses (i.e. validation and non GMP tests) was not considered. Also, the system resulted not very responsive, since a sample is completed in a time that is three times higher than the total processing time needed to perform an analysis. As it was shown in the first simulation scenario, a different scheduling policy could help to improve the system responsiveness.

5.3 Future Work

While this thesis has demonstrated the effectiveness of the developed simulation model, many opportunities for improving the quality of the model exist.

5.3.1 Quality of Data

The quality of data is a fundamental aspect that has to be considered in every simulation study. In particular, a review of the processing times is important for the improvement of this work, especially for equipment and tasks that were estimated using data coming from a time study. The main limitations of a time study are the short period of time available to collect data and the possibility of having human
errors. With this purpose, it would be interesting to extend the information extraction algorithm to other analytical methods in order to characterize better other analytical techniques.

Also, there is the need to explore how to retrieve information about the analysis performed in the quality control laboratory that are missing in the Laboratory Information Management System (LIMS). A suggestion is the study of the equipment dedicated softwares that could provide a good basis to fill the gaps existing in the main database.

5.3.2 Model Improvement

As referred in the above section, improvement to the model can be achieved by improving the quality of data.

Further on, other improvements are related with the review of the assumptions made in section 3.5. For instance, every sample can be considered as an entity that need to be processed at more than one equipment. Also, assumptions on system validity and system preparation can be modified according to the particular analytical method.

An additional feature that could be considered in a next version of the simulation model is the stock management. In this project, solvents and solutions have been considered infinite, and it would be interesting to add constraints to the model to evaluate limitations coming from lack of stock.
Bibliography


Appendix A

Information Extraction Algorithm

This appendix contains the R code of the developed information extraction algorithm. The algorithm aims to extract automatically data from text files that contains the description of the analytical methods. The first function is used to extract information from table structures found in the report, while the latter is used to analyze the text when the first function can not find the information needed.

Listing A.1: Information Extraction Algorithm (table)

```r
#*****************************************************************************#
# Extract Data from Method Reports #
#*****************************************************************************#
extractEN <- function(dat, dat_tbl){
  #*****************************************************************************#
  # Initialize variables #
  #*****************************************************************************#
  RunTime = -1
  Method = ""
  Sequence = ""
  nSystemSol = 0
  nAnalysisSol = 0
  nSystemInj = 0
  nAnalysisInj = 0

  #*****************************************************************************#
  # Extract Method #
  #*****************************************************************************#
  Test <- substr(dat, regexpr("TESTS", dat, ignore.case = TRUE), nchar(dat))

  index <- regexpr("CRGC|CRLC|CRTG|CRTL|CRIC|CRCE", Test)
```
if (index==−1){
  # Search for Chromatography
  index <- regexpr("CR.TG|CRTG|LC[0−9]+",dat)
  Method <- substr(dat,index,index+7)
  Method <- gsub("\","",Method)
} else Method <- substr(Test,index,index−2+regexpr(\.,
    substr(Test,index,nchar(Test))))

rm(index)

#*******************************************************************************
# Extract Run Time
#*******************************************************************************
# Review particular cases
# Select string between "Run time" and "min"
index1 <- regexpr("Run time|Acquisition time",dat,ignore.case=TRUE)
if (index1!=−1){
  index2 <- index1+regexpr("min|NA|AP",substr(dat,index1,nchar(dat))))
  string <- substr(dat,index1,index2)
  index1 <- regexpr(\,:",string)
  string <- substr(string,index1,nchar(string))
  # Extract Run time (int)
  index3 <- regexpr("[0−9]+",string)
  if (index3!=-1) RunTime <- c(as.numeric(regmatches(string,index3)))
  rm(string,index2,index3)
}

if (length(RunTime)==0){
  RunTime = −1
}

rm(index1)

#*******************************************************************************
# Extract injections
#*******************************************************************************
# Extract tables from document
datInj <- docx_extract_all_tbls(dat_tbl)
# Extract injections table

```r
index1 <- which(regexpr("Nr. of injections | Number of injections", datInj, ignore.case = TRUE) > 0)

if (length(index1) > 0) {
  nSequence = length(index1)

  index2 <- which(regexpr("LIMS TESTS | LIMS Tests", datInj) > 0)
  if (length(index2) == 0) {
    datInj <- datInj[index1[1] : length(datInj)]
  } else datInj <- datInj[index1[1] : index2[1]]
}

rm(index1)

index <- which(unlist(lapply(datInj, function(x) is.null(x[1]$V2))) == TRUE)
if (length(index) > 0) datInj <- datInj[!index]

datInj <- lapply(datInj, function(x) as.data.frame(x[1:2]))
datInj <- data.frame(unlist(lapply(datInj, "[[", 1)),
                      unlist(lapply(datInj, "[[", 2)))

colnames(datInj) <- c("Sol", "Inj")

# Eliminate special characters from Number of injections

datInj$Inj <- gsub("\u003c | \u003e | at least | a | b | [0-9] | - [0-9]", "", datInj$Inj)

# Extract injections

ind1 <- which(regexpr("[0-9]$ | NA | Nr. of injections", datInj$Inj, ignore.case = TRUE) < 0)

if (length(ind1) > 0) datInj <- datInj[!ind1,]

index1 <- which(is.na(as.numeric(datInj$Inj)) == FALSE)

datInj$RunTime <- NA
datInj$RunTime[index1] <- RunTime

rm(index1)

rownames(datInj) <- 1:nrow(datInj)
```
id <- which(regexpr("Nr. of injections | Number of injections", datInj$Inj, ignore.case = TRUE) > 0)[1]

id <- c(id, length(datInj$Inj))

Seq <- datInj[(id[1]+1):(id[2]),]
rownames(Seq) <- 1:nrow(Seq)

# Eliminate other rows that do not present injection values
index <- which(is.na(c(as.numeric(Seq$Inj))))
if (length(index) > 0) Seq <- Seq[-index,]
rownames(Seq) <- 1:nrow(Seq)

SamIndex = which(regexpr("(sample | SPL)", Seq$Sol, ignore.case = TRUE) > 0)[1]
StopIndex = which(regexpr("(sample | SPL)", Seq$Sol[SamIndex:length(Seq$Sol)],
                   ignore.case = TRUE) <= 0)[1]

if (is.na(StopIndex)) {
  StopIndex = length(Seq$Sol)
} else StopIndex = StopIndex + SamIndex - 1
Seq <- Seq[1:(StopIndex),]

nSystemInj <- sum(as.numeric(Seq$Inj[1:(SamIndex - 1)]))
nAnalysisInj <- sum(as.numeric(Seq$Inj[SamIndex:length(Seq$Sol)]))
nSystemSol <- length(unique(Seq$Sol[1:(SamIndex - 1)]))
nAnalysisSol <- length(unique(Seq$Sol[SamIndex:(length(Seq$Sol) - 1)]))

} else{
  print(’No table detected’) 
  out <- ExtractInjEN(dat)
  Sequence = out[1]
  Seq = out[2]
  nSystemSol <- out[3]
  nAnalysisSol <- out[4]
  nSystemInj <- out[5]
  nAnalysisInj <- out[6]
  if (Sequence==”Found”) Sequence=Method
Listing A.2: Information Extraction Algorithm (text)

ExtractInjEN <- function(dat){

  nSystemSol = 0
  nAnalysisSol = 0
  nSystemInj = 0
  nAnalysisInj = 0
  Sequence = ""

  # Extract injections

  # Select Table Starting Point (Sequence of injections)
  datInj <- substr(dat, regexpr("Sequence of injections|Injection sequence", dat)−7, nchar(dat))

  # Isolate Number of paragraph
  index1 <- regexpr('[0−9]\\[0−9]', datInj)
  if (index1!=-1){
    paragraph <- c(as.numeric(regmatches(datInj, index1)))

    # Calculate Next paragraph
    index2 <- regexpr('Retention times|TESTS', substr(datInj, index1+3, nchar(datInj)))
    if (index2==−1){
      index2 <- regexpr('[0−9]\\[0−9]', substr(datInj, index1+3, nchar(datInj)))
    }
  }

  # Substring of Sequence of Injections paragraph
  datInj <- substr(datInj, index1, index2−3)
}

if (Sequence==""){
  Sequence=Method
}

output <- list(Method, as.character(Sequence), as.numeric(RunTime),
               as.numeric(nSystemSol), as.numeric(nAnalysisSol),
               as.numeric(nSystemInj), as.numeric(nAnalysisInj))

Listing A.2: Information Extraction Algorithm (text)
# Delete Known patterns that generate mistakes

datInj <- gsub("S/N\u22643","",datInj)
datInj <- gsub("3<S/N\u226410","",datInj)
datInj <- gsub("S/N>10","",datInj)
datInj <- gsub("S/N\u226510","",datInj)
datInj <- gsub("S/N\u226510","",datInj)
datInj <- gsub("S/N\u226510","",datInj)
datInj <- gsub("S/N>3","",datInj)
datInj <- gsub("S/N<3","",datInj)
datInj <- gsub("\-\[0-9]\]","",datInj)
datInj <- gsub("\at least","",datInj)
datInj <- gsub("\[0-9].\[0-9]\]\r\n","",datInj)
datInj <- gsub("\[0-9].\[0-9]\]","",datInj)

# Remove special characters

datInj <- gsub("\u2264","",datInj)
datInj <- gsub("\u2265","",datInj)
datInj <- gsub("\−−−","",datInj)
datInj <- gsub("\<","",datInj)
datInj <- gsub("\>","",datInj)
datInj <- gsub("\=","",datInj)
datInj <- gsub("\r","",datInj)
datInj <- gsub("\n","",datInj)

datInj <- gsub("\<\[0-9]\>\[0-9]\]","",datInj)

datInj <- gsub("\[0-9].\[0-9]\]\r\n","",datInj)

datInj <- gsub("\[0-9].\[0-9]\]","",datInj)

# Split columns

testInj <- c(as.character(strsplit(datInj,"\+"))[1])
#testInj <- testInj[-(which(regexpr("\n","testInj"><1))]

rm(index1,index2,paragraph)

# Extract numbers

index = which(regexpr("^[0-9]\$|^[0-9]\r\n$","testInj">0)

if (length(index)>2){
  # Total Number of Injections
  injections = c(as.numeric(gsub("\D","",testInj[index])))

  # Separate Sample injections

  64
BlankIndex = \texttt{which(regexpr("Blank\ |\ Diluent\ |\ Branco\ |\ Dissolution", pastetestInj[index]-1, testInj[index]+1), ignore\_case=TRUE)>0)\\}

StdIndex = \texttt{which(regexpr('\(standard\ |LOD\ |LOQ\ |Std\) ', paste(testInj[index]-1, testInj[index]+1), ignore\_case=TRUE)>0)\\}

SamIndex = \texttt{which(regexpr('\(sample\ |SPL\) ', paste(testInj[index]-1, testInj[index]+1), ignore\_case=TRUE)>0)\\}

Sol <- NULL\\
Sol[SamIndex] <- "Sample"\\
Sol[StdIndex] <- "Standard"\\
Sol[BlankIndex] <- "Blank"

\textbf{if} (\texttt{length(Sol)}\texttt{>=length(injections)}){
  Injections <- \texttt{as.data.frame(Sol)}\\
  Injections$\texttt{Inj} <- injections\\
  id <- \texttt{is.na(Injections$Sol)}\\
  \textbf{if} (\texttt{length(id)}>0) Injections$Sol[id]<-"Standard2"
} \textbf{else}{
  Injections <-\texttt{as.data.frame(injections)}\\
  Injections$Sol <- \texttt{NA}\\
  Injections$Sol[SamIndex] <- "Sample"\\
  Injections$Sol[StdIndex] <- "Standard"\\
  Injections$Sol[BlankIndex] <- "Blank"\\
  id <- \texttt{is.na(Injections$Sol)}\\
  \textbf{if} (\texttt{length(id)}>0) Injections$Sol[id]<-"Standard2"
  Injections <- Injections[,c(2,1)]\\
  \texttt{colnames(Injections) <- c('Sol\','Inj')}\\
}\n
idx <- \texttt{as.numeric(which(gregexpr("Sample", Injections$Sol)==1)[1])}
\textbf{if} (!\texttt{is.na(idx))}{
  idx2 <- \texttt{length(Injections$Sol)}\\
  \textbf{if} (idx2-idx==0){
    Seq = Injections[1:idx]\\
  } else {
    Seq = Injections[idx2:length(Injections)]\\
  }\\
}\n
65
else {
  idx2 <- which(gregexpr("Standard|Blank", 
    Injections$Sol[idx:length(Injections$Sol)])==1)[1]

  if (!is.na(idx2)) Seq = Injections[1:(idx+idx2-1),]
}

nSystemInj <- sum(as.numeric(Seq$Inj[1:(idx-1)]))
nAnalysisInj <- sum(as.numeric(Seq$Inj[idx:length(Seq$Inj)]))
nSystemSol <- length(unique(Seq$Sol[1:(idx-1)]))
nAnalysisSol <- (length(Seq$Sol)-idx)

Sequence = "Found"
}

if (length(Sequence)==0) Sequence=""
}

# If injection sequence is not detected
if (Sequence == ""){
  index <- regexpr("According_to_CR", dat)
  if (index!=-1){
    dat1 <- substr(dat, index+13, nchar(dat))
    rm(dat1)
  } else {
    index <- regexpr("Follow_the_sequence_in_the_CR", dat)
    if (index!=-1){
      dat1 <- substr(dat, index+27, nchar(dat))
    rm(dat1)
  }
}
else if (length(Sequence)==0) Sequence = "Sequence_not_found!"

output <- list(Sequence, seq, as.numeric(nSystemSol),
  as.numeric(nAnalysisSol),
  as.numeric(nSystemInj),
  as.numeric(nAnalysisInj))

# Return Sequence
return (output)
}

Appendix B

Processing time distributions

In this section are shown the processing time distributions for every class of equipment. As referred, in chapter 3 those times have been obtained with a time study for work measurement performed in the quality control laboratory, and represent the duration of the main tasks that are related with the analysis.

![Sample Preparation](image1)

![Equipment Setup](image2)

![System Verification](image3)

![Data Processing](image4)

Figure B.1: Liquid Chromatography (HPLC) estimated times and probability distribution functions.
Figure B.2: Gas Chromatography (GC) estimated times and probability distribution functions.

Figure B.3: Particle Size Analysis (PSA) estimated times and probability distribution functions.
Figure B.4: Karl Fischer titration (KF) estimated times and probability distribution functions.

Figure B.5: Dynamic Scanning Calorimetry (DSC) estimated times and probability distribution functions.
Figure B.6: X-ray Powder Diffraction (XRPD) estimated times and probability distribution functions.