Hierarchical Workflow Management System for Life Science Applications

Dissertation

zur

Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik

der Universität Rostock

Submitted by:

Xiangyu Gu, born on 25th, January 1987 in Dalian, China

Rostock, Germany, 2018

https://doi.org/10.18453/rosdok_id00002396

Gutachter:
Prof. Dr.-Ing. habil. Kerstin Thurow, Universität Rostock, IEF, Institut für Automatisierungstechnik
Dr. Heidi Fleischer, Universität Rostock, IEF, Institut für Automatisierungstechnik
Dr. Mo-Yuen Chow, North Carolina State University Raleigh

Jahr der Einreichung: 2015

Jahr der Verteidigung: 2018

Acknowledgment:

This dissertation cannot be completed without the guidance and help form the following several persons.

First of all, I would like to express my deep sense of gratitude and respect to my supervisors Prof. Dr. Kerstin Thurow and Prof. Dr. Norbert Stoll for their guidance, support and scientific foresight. Their knowledge helped me in the whole researching and dissertation process. Without their trust and motivation, it is impossible for me to complete this doctoral research.

Furthermore, I would like to thank the leader of the research group "Life Science Automation - Information Technologies” Dr.-Ing. Sebastian Neubert for his continuous encouragement and support.

Besides, I would like to extend my thank all my colleagues of the Institute of Automation and the Center for Life Science Automation (celisca). They are always so nice to support me when I met problems. Especially I’d like to thank Prof. Dr. Hui Liu, Dr. Steffen Junginger, Dr. Thomas Roddelkopf and Dipl.-Ing. Volker Gatz and for their professional guidance and helps in my work.

Last, but most importantly, I would like to dedicate this thesis to my beloved family: my wife M.Sc. Nan Li, my father Mr. Chuanren Gu, my mother Mrs. Lanying Lu and my wife's mother Huiling Li, for their endless love, unparalleled support and massive encouragement throughout my life.

Abstract

In modern laboratories, an increasing number of automated stations and instruments are applied as standalone automated systems such as biological high throughput screening systems, chemical parallel reactors etc. At the same time, the mobile robot transportation solution becomes popular with the development of robotic technologies. In this dissertation, a new superordinate control system, called hierarchical workflow management system (HWMS) is presented to manage and to handle both, automated laboratory systems and logistics systems. The new innovative points of this method are listed as follows:

Firstly, a flexible hierarchical system structure is built for the HWMS, including workflow control layer, process control layer and instrument layer. This structure is compatible with various automated systems, devices and transportation management systems, which contain suitable web services for the communication and management.

Secondly, a laboratory material management system (LMMS) is presented to integrate the environment and resources of the laboratories including locations, slots, automated systems, labware etc. This system offers necessary information to create processes, transportations and workflows via workflow planning system (WPS).

Thirdly, a scheduling model was developed based on the strategy and constraint condition of the workflow execution. The fitness function of this model contains two parts: total execution duration and the balance score of the human and robot transportation. This balance score can adjust the scheduling result to adapt the changing of transportation resources.

Fourthly, various algorithms for scheduling are provided to search for the optimal solution, including genetic algorithm and two further modified genetic algorithms. They are evaluated with two groups of parameters, which are simulated in static and dynamic scheduling scenarios. Based on the distribution of the scheduling results, the GASA and GAPSO are selected for static scheduling and dynamic scheduling respectively.

Fifthly, a dynamic scheduling indicator is established to trigger the rescheduling procedure and offer the transportation-balance factor based on the status of the executing resources. The history data are used to train the artificial neural networks (ANN) and the ANN can generate suitable balance factor to adapt the resource changing due to the unexpected situations.

Finally, a group of experiments is provided to demonstrate and check the performance and functionality of HWMS. A typical life science application was used to evaluate and verify the execution stability of HWMS. The results show that, HWMS enhances the efficiency and stability of the laboratories observably.

Zusammenfassung

In modernen Labors werden immer mehr automatisierte Stationen und Instrumente als eigenständige automatisierte Systeme eingesetzt, wie beispielsweise biologische High-Throughput-Screening-Systeme und chemische Parallelreaktoren. Mit der Entwicklung der Robotertechnologien wird gleichzeitig die mobile Robotertransportlösung populär. In der vorliegenden Arbeit wurde ein hierarchisches Verwaltungssystem für Abeitsablauf, welches auch als Workflow-Management-System (HWMS) bekannt ist, entwickelt. Das neue übergeordnete Kontrollsystem kann sowohl automatisierte Laborsysteme als auch Logistiksysteme verwalten und behandeln. Die innovativen Punkte dieser Methode sind wie folgt aufgelistet:

Erstens wird eine flexible hierarchische Systemstruktur für das HWMS erstellt. In dieser Systemstruktur sind die Workflow-Kontrollschicht, die Prozesskontrollschicht, und der Instrumentenebene enthalten. Diese Struktur ist kompatibel mit verschiedenen automatisierten Systemen, Geräten und Transport-Management-Systemen, welche die geeignete Web-Services für die Kommunikation und die Verwaltung (das Management) beinhalten.

Zweitens wird ein Labormanagementsystem (LMMS) vorgestellt, um die Umgebung und die Ressourcen der Labors zu integrieren. Davon sind beispielsweise Standorte, Steckplätze, automatisierte Systeme, Labware eingeschlossen. Durch das Workflow-Planungssystem (WPS) bietet dies System die notwendigen Informationen zum Erstellen von Prozessen, Transporten und Workflows.

Drittens wird ein auf der Strategie- und Einschränkungsbedingung der Ausführung des Abeitsablaufs basierendes Planungsmodell entwickelt. Die Fitness-Funktion dieses Modells besteht aus zwei Teilen: der Gesamtdauer der Ausführung und dem Balance-Score zwischen den Mensch- und Robotertransports. Dieser Ausgleichswert kann das Planungsergebnis anpassen, um die Änderung von Transportressourcen zu adaptieren.

Nach der Einführung des Planungsmodells werden verschiedene Algorithmen für die Planung vorgestellt, um die optimalen Lösung zu suchen. In der Arbeit werden ein genetisches Algorithmus und zwei weiteren modifizierten genetischen Algorithmen dargestellt. In der Auswertung wird zuerst eine Parameterguppe bei der Simulation der statischen Planung benutzt. Zum Vergleich wird auch eine Simulation in dynamischen Planungsszenarien durch eine andere Parametergruppe durchgeführt.. Aufgrund der Verteilung der Planungsergebnisse werden GASA und GAPSO für die statische Planung bzw. dynamische Planung ausgewählt.

Im Anschluss daran wird ein dynamischer Planungsindikator angezeigt, um das Umplanungsverfahren auszulösen und den auf dem Status der Transportressourcen basierenden Transportbilanzfaktor anzubieten. Um die künstliche neurale Netzwerke (ANN) zu trainieren werden die Verlaufsdaten verwendet. Das ANN kann einen geeigneten

Gleichgewichtsfaktor erzeugen, um die sich aufgrund der unerwarteten Situationen ändernden Ressourcen anzupassen.

Abschließend wird eine Reihe von Experimenten bereitgestellt, um die Leistung und Funktionalität des HWMS zu demonstrieren und zu überprüfen. Beispielweise wurde eine typische Life-Science-Anwendung durchgeführt, um die Ausführungsstabilität des HWMS zu bewerten und zu verifizieren. Aus den Ergebnisse ist es ersichtlich, dass HWMS die Effizienz und Stabilität der Labore deutlich erhöht.

Content

List of Figures

List of tables

Abbreviations and Symbols

AGL-LIMS	Applied Genomics Laboratory - Laboratory Information Management System
ANN	Artificial Neuron Networks
API	Application Interface
APN	Arbitrary Processor Network
BNP	Bounded Number of Processors
BP	Back Propagation
BFGS	Quasi-Newton Back Propagation
BPM	Business Process Management
BPMS	Business Process Management System
BU	Bottom Up
BSA	Bubble Scheduling and Allocation
CAM	Content Addressable Memory
CPM	Critical Path Method
DAG	Directed Acyclic Graph
DCP	Dynamic Critical Path
DLS	Dynamic Level Scheduling
DSC	Dominant Sequence Clustering
ELN	Electronic Laboratory Notebook
ETF	Earliest Time First
EZ	Edge Zeroing
FIFO	First In First Out
FJSP	Flexible Job shop Scheduling Problems
GA	Genetic Algorithm
HACS	Human Assistance Control System
HCI	Human Computer Interaction
HLFET	Highest Level First with Estimated Times
HN	Hopfield Networks
HWMS	Hierarchical Workflow Management System
ISH	Insertion Scheduling Heuristic
ISPyB	Information System for Protein Crystallography Beamlines
LAST	Localized Allocation of Static Tasks
LC	Linear Clustering
LIMS	Laboratory Information Management System
LMMS	Laboratory Material Management System
LSA	Life Science Automation
MAE	Mean Absolute Error

MAPE	Mean Absolute Percentage Error
MCP	Modified Critical Path
MD	Mobility Directed
MH	Mapping Heuristic
MLP	Multi-Layer Perception
MMP-LIMS	Maize Mapping Project - Laboratory Information Management System
MRMS	Multiple Robot Management System
MTP	Micro Titer Plate
OLF	Open Lab Framework
PCAS	Process Control Adapter System
PCS	Process Control System
PMS	Process Management System
PSO	particle swarm optimization
RCP	Resource Constrained Scheduling Project Problem
RPPOP	Resilient Back Propagation
SA	Simulated Annealing
SGD	Stochastic Gradient Descent Back Propagation
SOA	Service Oriented Architecture
SWMS	Scientific Workflow Management System
UNC	Unbounded Number of Clusters
WPS	Workflow Planning System
WES	Workflow Execution System

Chapter 1 Introduction

Life sciences are consist of interdisciplinary field of sciences which involve not only the scientific study of organisms, but also the considerations [1], [2]. They include many specific disciplines, such as bioengineering, biochemistry, genetics, medicine, biomedicine, medical imaging and so on. The life sciences are used to improve the quality and standard of life by applications in agriculture, medicine, health and pharmaceutics. All these applications cannot be invented or developed without the life science laboratory.

Life science laboratories are distributed widely in research institutes, universities and technology companies. In these laboratories, more and more automated instruments appear to increase the throughput and the accuracy [3]–[6]. Therefore, life science automation (LSA) becomes an important factor for assessing the life science laboratory.

With the development of high-throughput technologies and laboratory robotics, LSA has major advances in the last two decades. As a typical LSA system, there are two key concepts: consolidation and integration.

- Consolidation: Combining different analytical technologies or strategies in one instrument or in a group of connected instruments.
- Integration: connecting analytical instruments or groups of instruments with pre- and post-analytical devices.

Because of consolidation and integration, an increasing number of automated stations and instruments are used in LSA system such as biological high throughput screening systems, chemical parallel reactors etc. They offer reaction and analysis solution with high performance and efficiency. In addition to economic reasons and time constraints, a further goal of automation in biotech and pharmaceutical companies or institutes is to improve experimental results and workflow [7]. Research and development have been working on LSA systems over last few years [2], [8], [9]. However, complex automation demands on comprehensive investigate have often met their limits in the integration.

On one hand, consolidation and integration increase performance and bring more functions in life science laboratory. On the other hand, they require more accurate controller than the stand-alone device because the complex constraint and environment. Therefore, automated systems are normally controlled and monitored by a self-contained process control system (PCS), which provides users with specific control panels for the definition and control of processes. Local process control via PCS for heterogeneous components in the automated system with industrial robots and increasingly, dynamic local optimization of procedure have become the state of the art [10]–[12]. Process modeling and execution languages are only locally valid for certain automated systems, which are proprietary and manufacturer-dependent [13]. As a result, each automated system has individual PCS and there is no direct communication between PCSs.

Moreover, increasing comprehensive investigates require more than one processes, which are executed in various automated systems distributed in different laboratories. To transport samples reliably and efficiently, a suitable transportation management system is essential for the laboratories and this system stands the same layer as the automated system. More and more mobile robots are used to perform the transportation tasks [14]–[16]. The onboard robotic arm realizes the automatic transportation. It not only reduces the stress of worker, but also avoids the human errors. Although there is special design of the mobile robot for the laboratory environment, they cannot instead of human totally to do transportation tasks because of the limitation of the moving mechanisms and the narrow path in some laboratories. A human and mobile robot mixed transportation solution is adopted in the life science laboratory.

So far, the automated systems and the transportation solution compose the “body” of LSA. As the “brain”, a superordinate management system, which controls and monitors PCSs and transportation management system, is directly determining the performance and efficiency of the laboratory. Laboratory information management system (LIMS), business process management systems (BPMS) and workflow management system (WFMS) are typical superordinate management systems for a laboratory. Most of LIMS focus on the data chain and they are good at data and information management. WFMS concentrates on the control chain to handle various processes. BPMS has both data and control chain and it is also the most complex solution for the laboratory management. These superordinate management systems serve not only laboratories, but also hospitals, famous companies and organizations. Most of these systems are highly specialized for their own requirements.

In the superordinate management system, an investigate can be described as a workflow, which consists of a sequence of processes and transportations. It is normal that more than one workflow is executed at the same time. It is possible to operate them one by one depending their priority or applying FIFO (first in first out) strategy. On one hand, it has high stability, because it reduces maximum competition between automated systems [10], [11]. On the other hand, the efficiency of the laboratory becomes very bad and it leads the samples to lose efficacy possibly by reason of the long waiting time. The mainstream solution is to operate multiple workflows in different automated systems in parallel [12], [17], [18].

Inevitably, resource contention problems arise during the execution, sometimes leading to failure. In order to allocate various resources before execution, the static scheduler is used to increase efficiency of the laboratory and avoid clash. As the brain of the laboratories, the scheduler creates the time plan to guide all automated systems and the sample transportation system, which is used to support the resource allocation.

A customized algorithm for schedule optimization is developed in laboratory environment. Different from the most of the commercial or existing open source schedulers, the aim of optimization is not only to force on the shortest operation time or lowest resource requirement, but also to balance transportation task distribution between the mobile robots and human operators.

Different form single automated system, the laboratory environment is more complex and dynamic. Sometimes, there are unexpected situation arising during execution, such as unexpected delay of the transportations and even failed tasks. The uncertainties lead to more delay and failures, which create potential disruptions of the execution.

In response to the problems posed by the uncertainties, static scheduling is no longer suitable because of the huge computation effort, which guarantees the global optimization. It is not agile to face various unexpected situations. Dynamic scheduling, which is operated during the execution can increase the system stability dramatically.

Dynamic scheduling is defined under two categories: completely reactive scheduling and predictive-reactive scheduling. Completely reactive scheduling does not generate time schedule in advance and the execution procedure is made locally in real-time. Predictive-reactive scheduling, which is used in this approach, is an event-driven rescheduling process based on the current situation. It can take full advantage of the models and environment of the static scheduling.

In addition, when and how to rescheduling will influence the performance and efficiency of the laboratory. Not all unexpected situation requires rescheduling. An indicator is used to determine whether it is necessary to trigger the procedure of dynamic scheduling. Moreover, the indicator should also offer the currently status of transportation factor, which is the core parameter to generate new schedule for the execution. This factor comes from the statistics of the transportation task record. An intelligent algorithm is used to find it with an affordable cost.

The purpose of this thesis is to create a superordinate management system to integrate and consolidate the automated systems, automated instruments and transportation management system and make them running with high performance and stability.

In chapter 2, the state of the art for automated systems will be explained and the algorithms of the scheduling optimization are analyzed and compared.

In chapter 3, the goals and concept of this thesis are proposed. The current state and the requirements of laboratories are analyzed. The definition of the system HWMS is presented and the developed method is discussed also.

Chapter 4 explains the system development, which includes the data structure, the planning editor and the execution module. The laboratory environment is digitized and integrated into a database following the specific data structure. This is the base of laboratory management system. The planning editor offers the function to the research for the workflow designing and management. The execution module has two parts: the monitor and the controller. Furthermore, the monitor not only serves for the user interface, but also supports the scheduler for the dynamic scheduling.

In chapter 5, the intelligent scheduler, which includes static and dynamic scheduling, is proposed. The scheduling model, strategy and algorithms are presented. In the end, an experiment is used to evaluate the performance of the algorithms and the suitable algorithms will be selected to support the scheduling service for HWMS.

In chapter 6, the rescheduling indicator is presented. The indicator is used to trigger and guide the rescheduling to realize the dynamic scheduling in an efficient way. A specific example is used to compare the performance and efficiency of various algorithms and parameter settings. Finally, the suitable solution is selected for the indicator.

The chapter 7 shows the conclusion and further works.

Chapter 2 State of the Art

In this chapter, two research areas are concerned: laboratory management systems and scheduling methods. The modern automated laboratories are filled with automated workstations and instruments. The laboratory management system, which is used to control these workstations and instruments determines the performance and efficiency of the laboratory. Moreover, the scheduler of the management system plays a resource allocating roller in the laboratory management system.

2.1 Laboratory Management Systems

In modern laboratories, an effective management is imperative to guarantee the stability and performance during planning, execution, storage and analysis. As the mainstream solution, there are three kinds of approaches: Laboratory Information Management System (LIMS), Business Process Management System (BPMS) and Workflow Management System (WFMS).

2.1.1 Laboratory Information Management Systems (LIMS)

Laboratory information management systems (LIMS) have been successfully established on the market. The customized solution of LIMS plays a vital role to fulfill the individual requirements of life science applications [19]. LIMS is a software, which bases on the laboratory environment. It is an information management system that offers a set of key features such as workflow and data tracking support, smart data exchange interfaces and flexible architecture [20], [21]. It is used to track the sample, monitor the procedure of the execution, store the data associated with the sample analysis, schedule the tasks or events and analyze the data and trends [22], [23].

Beteva *et al.* presented a data-collection pipeline (DCP) to automate data collection at synchrotron beamlines as well as putting in place protocols to automatically align optical elements, improve beamline diagnostics and optimize the usage of available beam time. The DCP contains two systems, the information management system and the instrument control system. Completing the pipeline series is a LIMS for diffraction data collection which has been developed in collaboration and named ISPyB (Information System for Protein Crystallography Beamlines) [24]–[26].

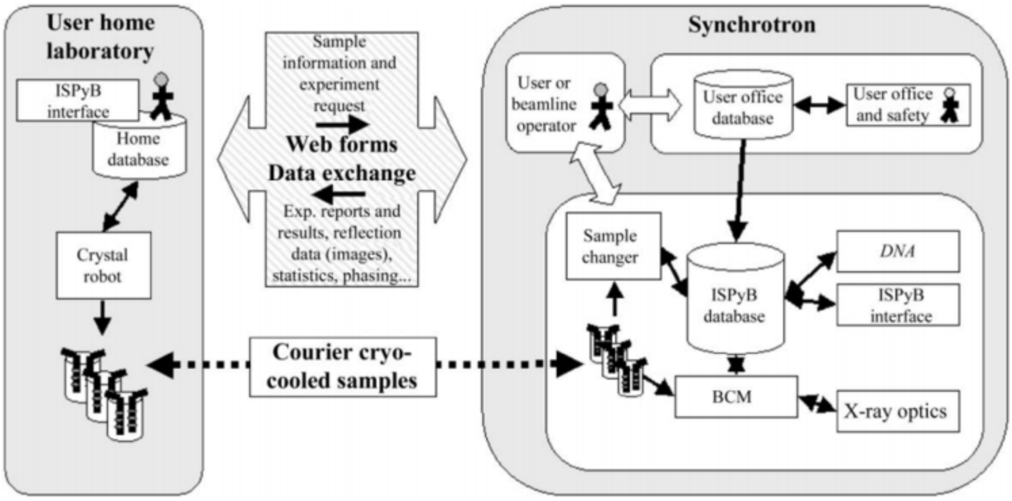


Figure 1 *Schematic of DCP for a user experiment [24]*

Sample data can be transferred into the ISPyB at different levels of sophistication depending on the user's needs.

1. Direct manual input via a web-based browser interface.
2. Input of data using a personal digital assistant device that runs a pocket version of ISPyB developed to provide a portable desktop solution for storing data as you mount your crystal.
3. Robotic crystallogenesis systems can produce large numbers of samples and LIMS exist to manage these data.
4. Once a user logs into ISPyB, he/ she can retrieve the sample information stored in the ESRF User Office database.

The creation of a control software framework, which brings together all the components in a form which is simple to control and efficient, was a major part of the development of the DCP. This overarching control software distributes the management of hardware over several computers whilst maintaining a principal control point on one machine where all the necessary information can be displayed and accessed.

Delageniere *et al.* developed the ISPyB and the main functions of ISPyB were developed into these three [27], [28]:

1. Sample management
2. Experiment recording and planning
3. Searching and reporting

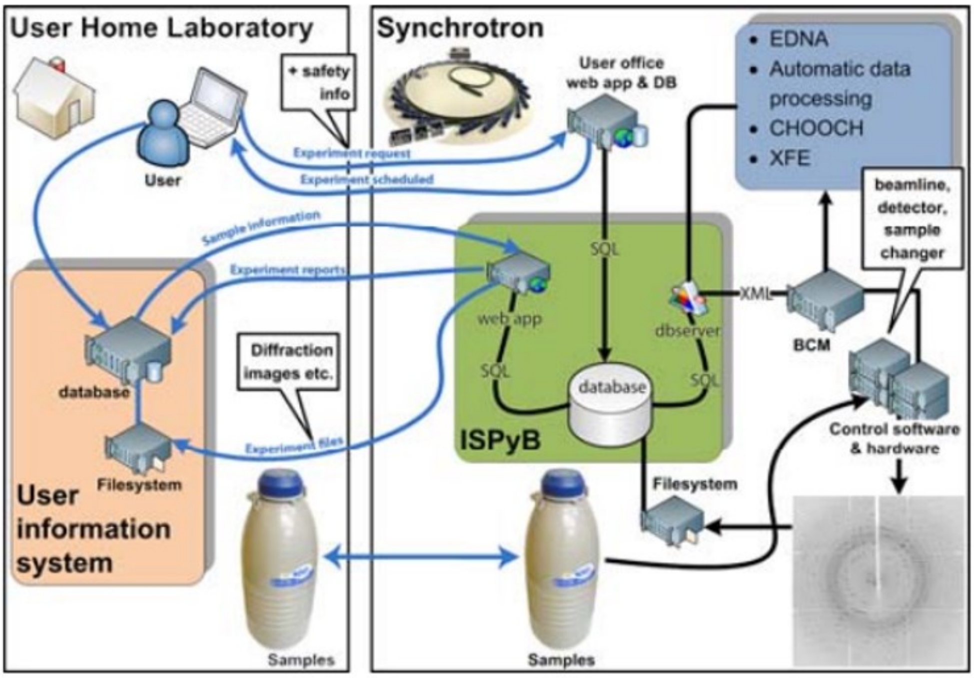


Figure 2 *The data flow and application environment of ISPyB [27]*

ISPyB was designed to optimize the information transfer between home laboratories and synchrotron-based MX beamlines before, during and after experimental sessions. The particular strength of ISPyB relates to sample management and the recording of information during an experiment. The possibility to rank the diffraction characteristics of samples means that, in principle, full data collections should only be carried out on the samples best suited to the experiment in hand. Moreover, the accessibility of results and output files from the automatic, on-line data processing protocols means that experimenters can also monitor the success of data collection experiments in real-time (i.e. while still at the beamline) and decide whether or not more data need to be collected for a specific project [12].

List *et al.* presented the OpenLabFramework (OLF), a laboratory information management system (LIMS) primarily targeted at advanced sample and storage management in mid-sized laboratories with less than 50 users. It facilitates a seamless integration of virtual and real world storage handling by making use of mobile devices, which are carried by lab personal anyways, in combination with cheap and fully integrated barcode labeling technology [29].

At first, they compared different kinds of LIMS, which are browser-based solution as shown in following Table.

Table 1 *Some examples of existing browser-based LIMS solution [29]*

Project Name	Main purpose	Built with
MMP-LIMS	Genome mapping in maize	Java
AGL-LIMS	Genotyping workflow	Java
SMS	Gene mutation screening & biobanking	Java
PiMS	Sample & exeperiment tracking	Java
LISA	Protein crystallography	PHP
FreeLIMS	Sample management & report	Java
Open-LIMS	Exerimental workflow, sample & document amnagement	PHP
Lablog	Project documentation	Java
SIMBioMS	Linking experimental, patient and high-throughput data	Perl

Any LIMS that involves sample management on a large scale should fulfill a number of requirements listed in the following as R1-15. Existing open-source LIMS fulfill these requirements to varying degrees (Table 2).

Table 2 *Feature comparison of requirements across browser-based LIMS solutions [29]. EnzymeTracker (ET), Free-LIMS (FL), SLIMS (SL), Open-LIMS (OL), ProteinTracker (PT), SMS (SM), OpenLabFramework (OLF)*

Requirements	ET	FL	SL	OL	PT	SM	OLF
Open-source	√	√	√	√	√	√	√
Modularity	X	X	X	√	X	X	√
Sample Management	√	√	√	√	√	√	√
Sample Tracking	X	X	X	X	X	√	√
File Management	X	X	X	√	X	X	√
Reports	√	√	√	X	√	√	√
Local Deployment	X	X	X	X	X	X	√
Cloud Deployment	X	X	X	X	X	X	√
Documentation	√	X	√	√	√	√	√
Barcode	√	X	X	X	X	√	√
Data Analysis	√	X	√	X	X	X	X

There are four modules in OLF,

1. GeneTracker: is intended to fulfill requirements specific to the hierarchical organization of genes, gene variants, vector constructs and genetically engineered cell lines, thus helping to keep track of extensive sample libraries in the field of targeted genomics.

2. Sample Storage: The Storage module adds options for tracking and organizing samples in a customizable storage infrastructure. This infrastructure is hierarchical, starting from buildings and rooms and ending in individual freezers and storage boxes.
3. File Uploads: The FileAttachments module allows users to up- and download arbitrary files, allowing for a better organization of their results and documents.
4. Barcode and Label Support: The functionality of the Storage module is complemented by the Barcode module, with which a user can create and print barcode labels.

In order to retain an overview over large sample libraries typically found in nowadays laboratories, an efficient system for management and tracking of samples is required. OpenLabFramework (OLF) has been developed with focus on vector construct and cell line libraries. Thanks to its modularity, however, it can be adapted to new scenarios.

Table 3 *Mainstream commercial solutions of LIMS*

Name	Main purpose	Web based	Labor customization
QuaLIS LIMS	Agaram Tech [30]	Yes	Yes
Matrix Gemini LIMS	Autoscibe Informatics [31]	Yes	Yes
ApolloLIMS	Common Cents Systems [32]	Yes	Yes
LabCollector LIMS	AgileBio SARL [33]	Yes	Yes

Some commercial LIMS are also really popular in the laboratories as presented in Table 3. They have similar functions and all of them need a high customized level. Common Cents Systems bring their flagship product ApolloLIMS, which owns the structure as below.

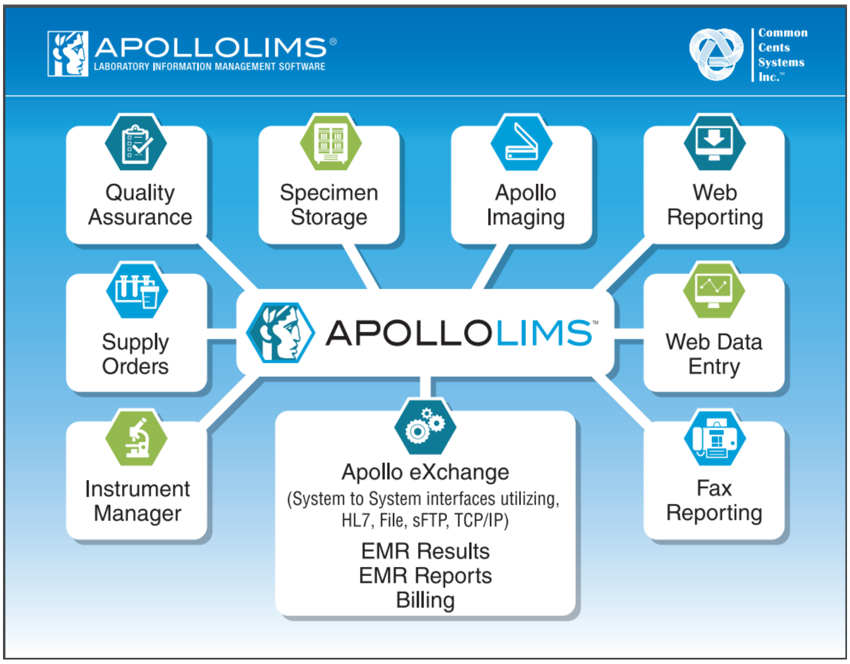


Figure 3 *Structure of ApllоLIMS [32]*

ApolloLIMS includes almost every aspect in a typical laboratory [32]. With the help of native components, ApolloLIMS offers a more stable and higher levels of integration than the tools, which are created by third parts. Customizations and Localizations are handled at the Database level, which allows the core product to remain standard. The Apollo Schema is developed around a concept known as a visible database. ApolloLIMS offers two implementation options: stand-alone software and a cloud-based solution. Therefore, the LIMS can be accessed and handled in anywhere and anytime.

2.1.2 Business Process Management Systems (BPMS)

The definition of Business Process Management (BPM) is "Supporting business processes using methods, techniques, and software to design, enact, control, and analyze operational processes involving humans, organizations, applications, documents and other sources of information" [34].

The BPM life-cycle has four phases as shown in Figure 4 [35]:

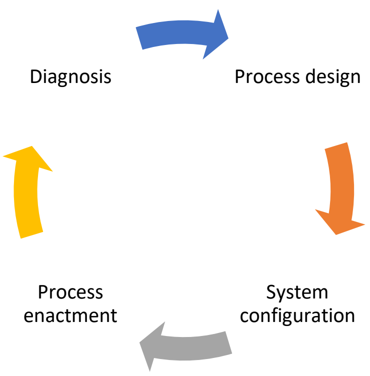


Figure 4 *The life-cycle of Business Process Management [35]*

1. Process design: Any BPM effort requires the modeling of an existing ("as-is") or desired ("to-be") process, i.e., a process design. During this phase process models including various perspectives (control-flow, data-flow, organizational, sociotechnical, and operational aspects) are constructed. The only way to create a "process-aware" enterprise information system is to add knowledge about the operational processes at hand.
2. System configuration: Based on a process design, the process-aware enterprise information system is realized. In the traditional setting the realization would require a time-consuming and complex software development process. Using software from the second and third layer shown in Figure 1, the traditional software development process is replaced by a configuration or assembly process. Therefore, we use the term system configuration for the phase in-between process design and enactment.
3. Process enactment: The process enactment phase is the phase where the process-aware enterprise information system realized in the system configuration phase is actually used.

4. Diagnosis: Process-aware enterprise information systems have to change over time to improve performance, exploit new technologies, support new processes, and adapt to an ever-changing environment. Therefore, the diagnosis phase is linking the process enactment phase to a new design phase.

The following figure shows an example of BPM.

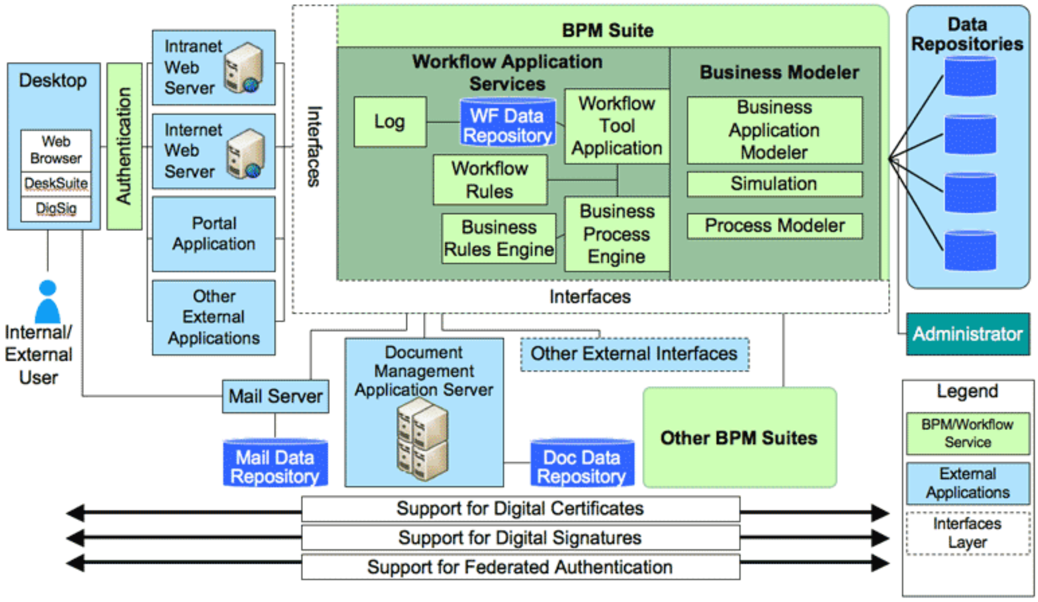


Figure 5 *A typical structure of BPMS [35]*

This pattern shows how business process management (BPM) tools can be used to implement business processes through the orchestration of activities between people and systems.

This pattern can be viewed in layers:

- The User Interface layer can be implemented either through the web-based tools provided with the BPM software, or via custom user interface development which interfaces with the BPM software.
- The BPM Tools layer provides the core BPM functionality (see Workflow/BPM Brick).
- The Storage layer provides repositories for both business process models and corporate data.
- The Interfaces layer provides means to exchange data with the BPM Tool.

This solution provides the user with the ability to access the business process management tools through a Web-enabled interface or through the organization's email system. Access is controlled through the mechanisms specified in the security architecture. BPM tools can exchange task information with each other by adhering to established BPM data exchange standards. Inter-application communication and data exchange is accomplished through the enterprise services bus technology [36].

The benefits of BPM are that:

- BPM tools provide business users with the ability to model their business processes, implement and execute those models, and refine the models based on as-executed data. As a result, BPM tools can provide transparency into business processes, as well as the centralization of corporate business process models and execution metrics.
- BPM suite software provides programming interfaces (web services, application program interfaces (APIs)) which allow enterprise applications to be built to leverage the BPM engine.
- Modeling and simulation functionality allows for pre-execution "what-if" modeling and simulation. Post-execution optimization is available based on the analysis of actual as-performed metrics.

Yang *et al.* introduced the API-based customization and component-based customization of an open source BPMS uEngine, with some illustrative examples, for the flexible coverage of various requirements when deploying BPM applications. Considered requirements are friendly GUI and its related logic, and supportive system functions [37].

At first they compared several academic and open source BPMS as shown below, uEngine is one of the top-evaluated open source BPMS with most of the features of commercial products [38]. It delivers all core BPM functionalities and special solutions to maintain the consistency of existing application architecture, reducing implementation problems and the total cost of ownership. There is the framework of uEngine:

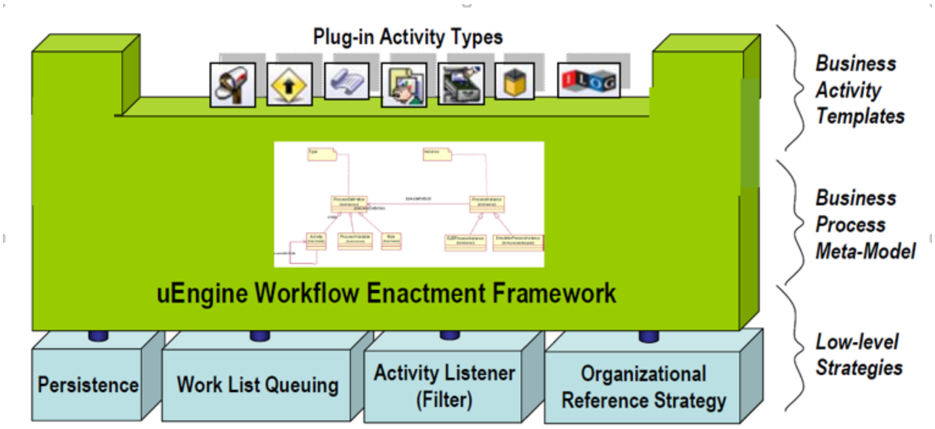


Figure 6 *Component model of uEngine [37]*

There are also many BPMS software products, e.g. WorkflowGen, KiSSFLOW, VisionFlow. The major function of them are quite similar. But the capacity of customization is totally different. WorkflowGen will be explained as an example.

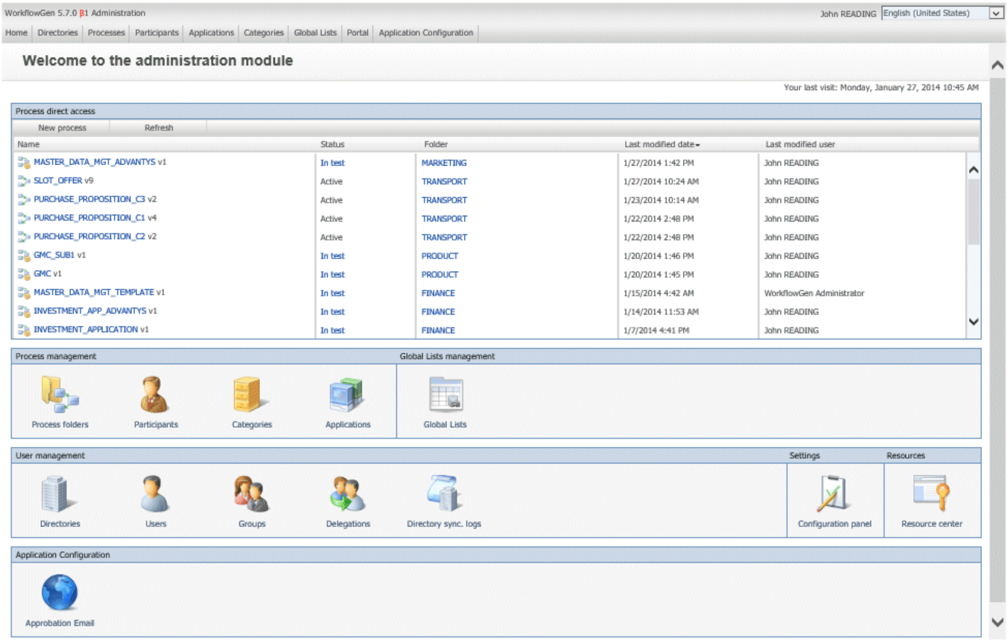


Figure 7 *Administration module of "WorkflowGen"*

The figure above shows the structure of WorkflowGen in administration module. When user wants to manage processes, the process will be showed as process workflow.

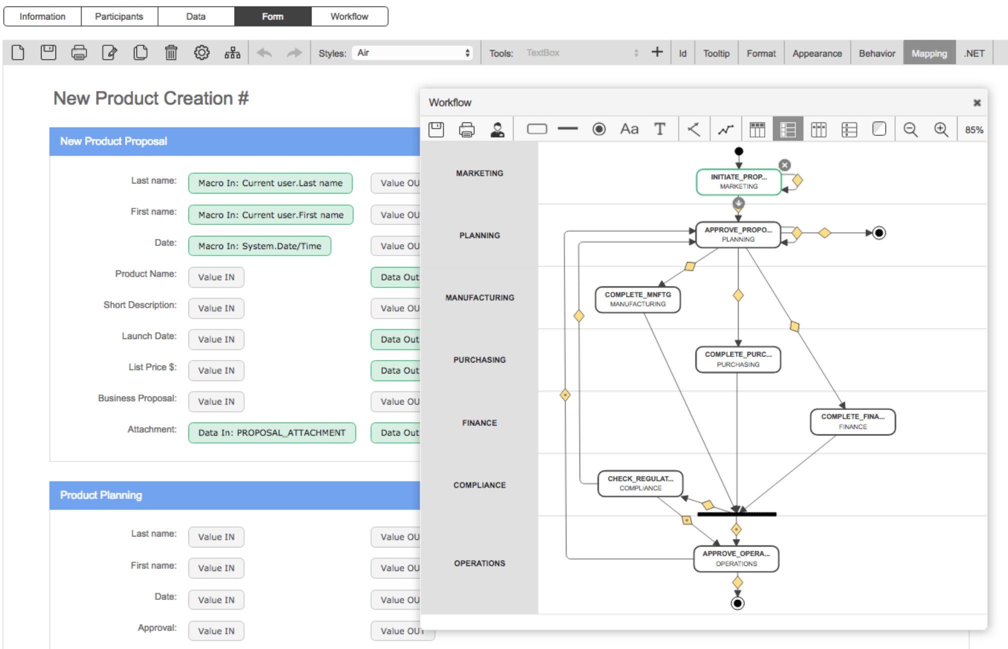


Figure 8 *Workflow for new product creation*

This workflow shows the steps, participators and their relationship. There means also a hierarchical structure. The functions and duties can be find really clearly. During the process user can easily find the point of problems.

With the help of WorkflowGen, The Department of Facilities Management of the University of Texas Health Science Center at San Antonio provides services (maintenance, repair, etc.) to support more than 2.8 million gross square feet of space in buildings with a staff of 315 [39].

The Department of Facilities Management designed the work order process with WorkflowGen in one month. A pilot was first organized for a few selected participants before the solution was rolled out campus-wide to 3,000 people by sending a simple email.

The response time has been optimized from two days to 15 minutes. The workload for the operators in the Department of Facilities Management has been reduced dramatically and their ability to monitor and retrieve process data has been maximized. The clients enjoyed both the productivity gains and the possibility to monitor their requests in real time.

Holzmüller-Laue *et al.* developed a fully web-based laboratory information management system (LIMS) at first. This LIMS is application-independent by a user-defined parameter library and a method repository. Currently chemical, biological, and preventive medical research groups work with this system. Core functionalities are the administration, the organization and documentation of the laboratory workflows and research projects on user-defined abstraction level and with any structuring. An electronic laboratory notebook (ELN) is integrated for the unstructured process documentation in the development as well as operation phase. The progressive technical automation inclusive the generated data amounts require an automation of data management, data manipulation and knowledge extraction. Therefore, the basic LIMS are expanded from a pure measurement management system to an integration platform. This is a framework solution for a user-defined process description, a configurable process communication for the automated exchange of process data, the process visualization as well as a template based process data manipulation [13].

Later they presented a BPM approach because of the limits of LIMS in consequence of its interactivity, and these fully-automated sub-processes, manual process steps and activities, which are generating decisions and knowledge, will be integrated. The comprehensive automation of the overall workflow, which covers all involved, any structured automation components across the application fields is the logical further development of the automation of individual working steps. BPMS takes over the role of a systems integrator or rather the role of an integration platform for the process control and support so the interoperability.

They compared different approaches for the overall workflow automation, such as ExpWF, SIGLa, LES and so on. The properties of these workflow automation focus the generic workflow for data capture and documentation and follow the specialization of the system suppliers and their involved information system and automation solutions. In comparison with the above-mentioned variants, the following BPM approach presents a particularly open, standard- driven integration path to arbitrarily complex workflow automation. The approach is completely system- and process independent. The used graphical process modeling language provides a high chance for broad interdisciplinary establishment. By using open standard-based BPM approach new workflow modules are generated with each realization and are managed in libraries of workflow automation. This solution library is comparable to methods libraries of hierarchical automation system, but without regard to complexity and interdisciplinary.

Business Process Management Systems (BPMS) connect process modeler and process engine enhanced by functions for process monitoring, process administration, and for the human task management [40].

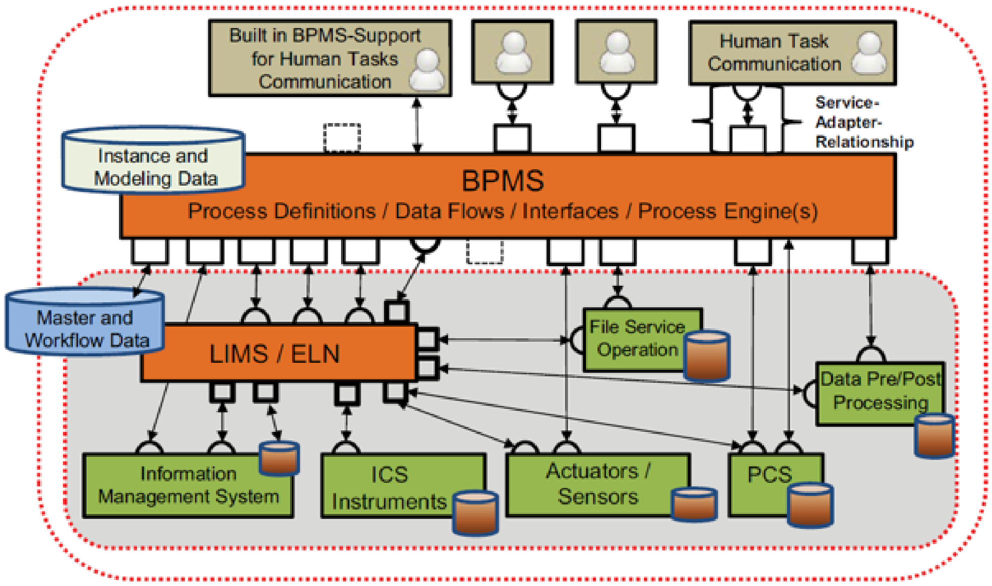


Figure 9 *Representative architecture for the workflow automation [40]*

The presented solution for end-to-end workflow automation prefers a generic LIMS as a documentation platform and instance data store for workflow runs. Thus, with the BPMN 2.0–based workflow automation using a LIMS, a complete, overall process documentation can be generated and controlled [9]. The favored approach of a combination of generic LIMS and BPMN 2.0 as an automation language offers the possibility of using the systems integration that already exists in the LIMS. This particular system position of LIMS results in the architecture illustrated in Figure 9 for the BPM-based laboratory automation. The adaptation effort relating to information technology (the service adapter drawn in Figure 8) between the BPMS-driven workflow as a service consumer and more or less compatible components should be reduced in standards-based service-oriented architecture (SOA) environments.

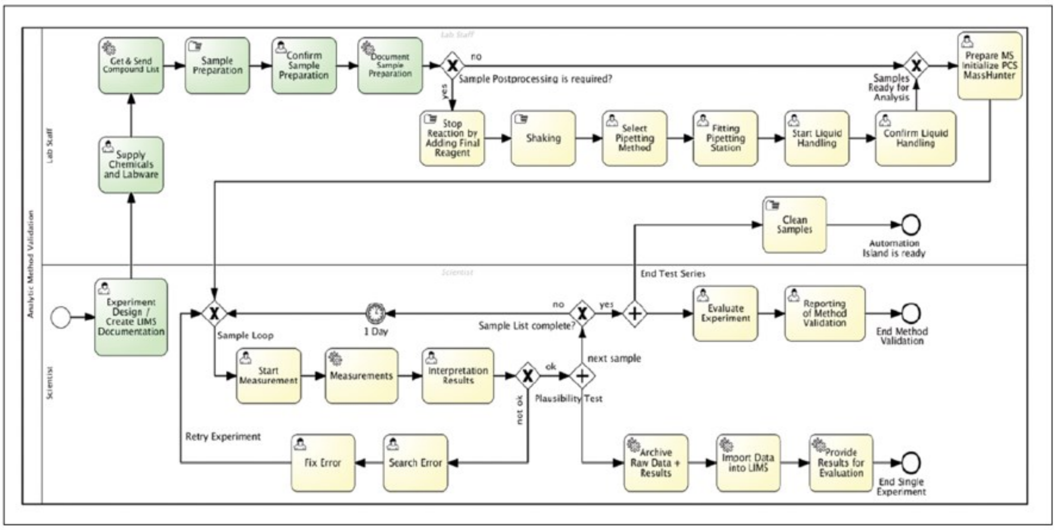


Figure 10 *The process model of the example application [9]*

Figure 10 shows the process model of the described workflow in BPMN 2.0 syntax. There are two process participants who are also provided “roles”: the scientist (chemist) as investigator and the laboratory assistants (staff). The role includes a group of persons that is represented in the process model by the lanes. During runtime, an activity instance is normally offered to all people available who are members of the role and thus capable of performing the task [11].

The process modeling using the graphical notation standard BPMN 2.0 meets the requirements of a high-process transparency for both robust end-to-end workflow controls and for knowledge management. All participants in research and in customer order processing benefit from the high-level standard-based and process-driven IT applications. The BPM-driven laboratory automation combines both manual and automated activities (respectively, sub processes). End-to-end process models promote understanding of the overall relationship and dependencies between some steps as well as overcome boundaries.

Results:

1. Benefits for scientists: flexible processes, workflow reliability and data and knowledge availability
2. Benefits for customers in internal and external business relationships
3. Benefits for operators
4. Benefits for IT and the automation engineering team

2.1.3 Workflow Management Systems (WFMS)

Before the description of the Workflow Management Systems (WFMS), the primary object for this system – workflow will be explained at the beginning of this chapter. A workflow is the part of a work process, which contains the sequence of functions and information about the resources involved in the execution of these functions. The workflow is defined by the *Workflow Management Coalition* as “An automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules". Therefore, a workflow is a partial automation of a business process [41], [42].

Supersede database management and user-interface management, workflow management extracts the business processes out of applications. Similar to an operation system, WFMS controls the workflows between the various resources, which are confined to the logistics of case handling. On the other hand, WFMS consists of several functions which are used to define and graphically track workflows. It brings great convenience and high flexibility to revise the progress of a case via a new workflow and the structure of the flow itself [43]–[45].

In general, the reference model of WFMS is shown in Figure 11. In this model, the heart of a WFMS is the so-called Workflow Enactment Service. The Enactment Service ensures that the right activities are implemented in the right order and by the right device or people. In order to realize this, the Process Definition Tools is used to produce the process definitions and resource classifications. Moreover, these tools can offer facilities for analysis techniques such as simulation. The work item is offered to the staffs or devices through Workflow Client Applications. By starting a work item, a staff should begin performing a specific task for a specific case in an application. All the application software which can be implemented by the WFMS is known as Invoked Applications in the reference model. The Administration and Monitoring Tools are used to track workflows, control cases and manage the staffs and devices. The various interfaces connect the Workflow Enactment Service and other tools and applications. The figure shows the various components of WFMS and it explains the detail contents of the tools and applications in the reference model [45].

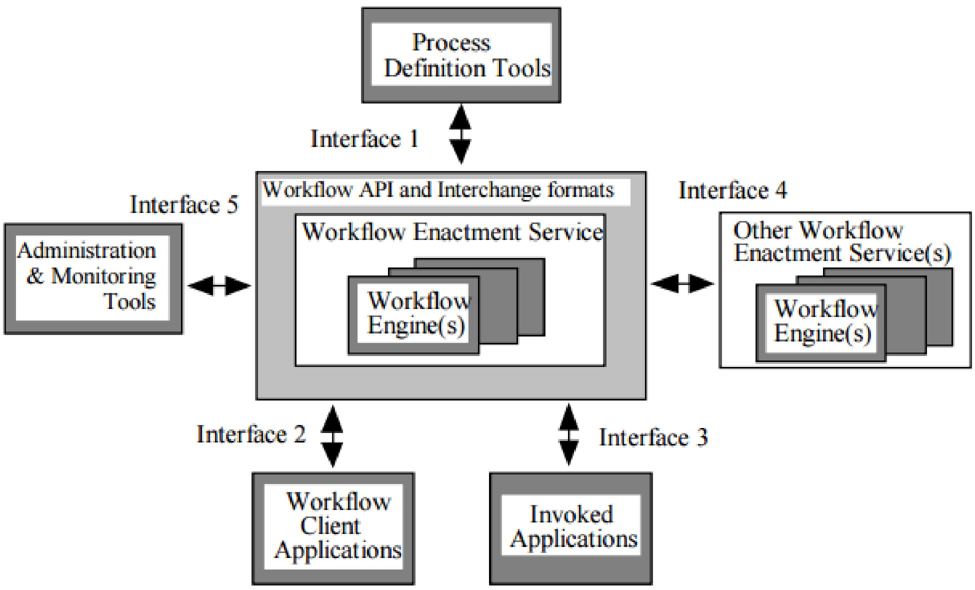


Figure 11 *Reference model of WFMS [45]*

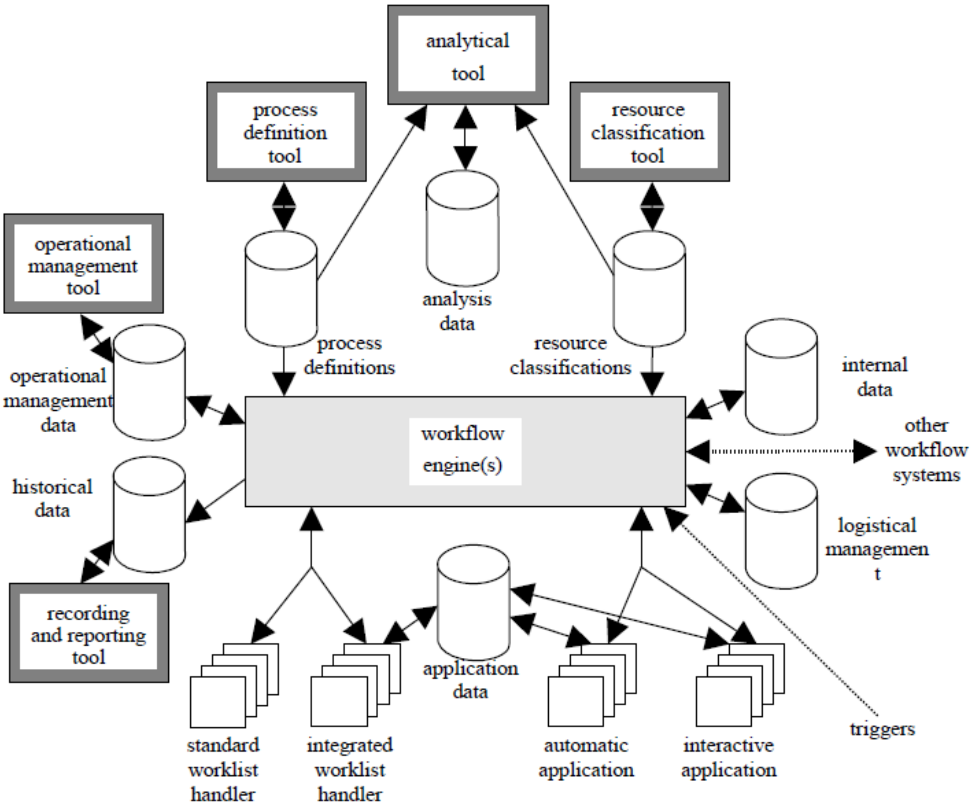


Figure 12 *The various components of WFMS [45]*

According to the Figure 12, the involved roles in WFMS are shown clearly. There are four types of the roles: the workflow designer, the administrator, the process analyst and staff (device). The workflow designer uses the Process Definition Tools to create the processes and build the structure of the workflow. The administrator implements new processes, monitor workflows and maintain the system with the help of operational management tool.

WFMS is applied to various domains. As a commercial WFMS – Ceiton, for example, has a long customer list such as Sony DADC in USA, Red Bull Media House in Austria, AT&T in USA, ZF Friedrichshafen in Germany and so on. In the life science area, there are more and more laboratories running with WFMS.

Kochut *et al.* presented a distributed workflow system - IntelliGEN in the university of Georgia as shown in Figure 13 [46]. IntelliGEN was used for a large genomic project of mapping protein-protein interactions of fungi. This system conducted and coordinated genomic experiments in a heterogeneous and distributed environment. An intelligent adaption agent was designed to maximize the coverage given the resources and time constraints.

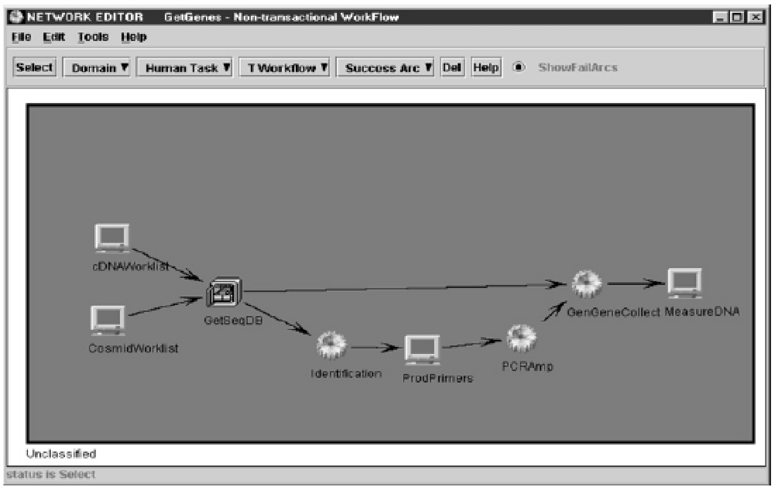


Figure 13 *Graphical design tool displaying top-level protein interaction workflow.*

Achilleos *et al.* proposed an open source workflow systems in life sciences informatics [47]. This system was based on Scientific Workflow Management System (SWMS), which was used in many application domains such as chemistry, biology, genetics and so on. The SWMS approach offered for data and resource management transparently and it provided a friendlier visual working environment supporting easy use. However, there is no controller for the laboratory automation system. Its functions are similar to a typical LIMS.

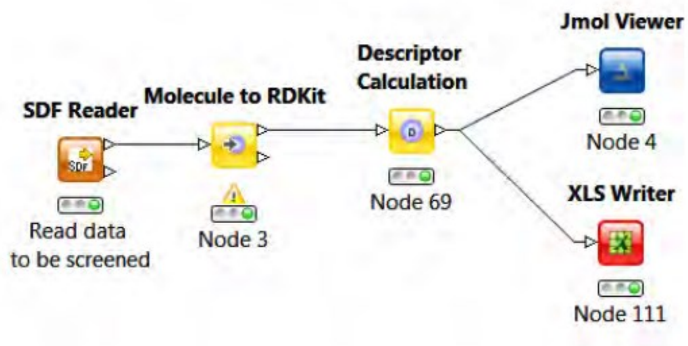


Figure 14 *An example of SWMS [47]*

Gabor *et al.* presented a workflow management module named Exp-WF which can be incorporated into a typical LIMS [48]. It is not necessary to change the original LIMS and the data management. Exp-WF adapted the LIMS and extracted the data directly from the database management of LIMS but without any modification of the original data. This system tracked the operating workflow and provided a genetic interface for agent communication.

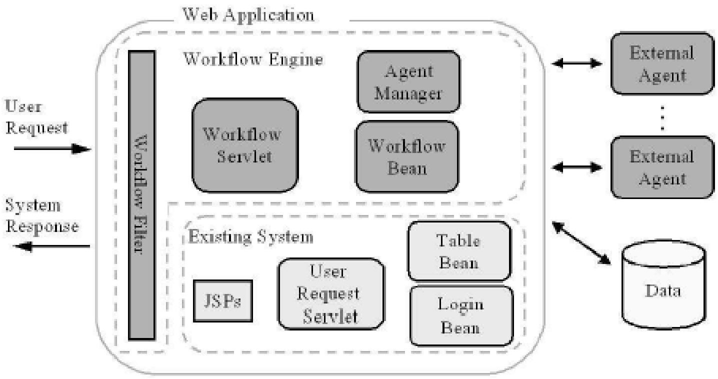


Figure 15 *Workflow engine architecture of Exp-WF [48]*

Deelman *et al.* developed a workflow management system for science automation named Pegasus since 2001 [49]–[52]. Pegasus has been used in large collaborations such as the Southern California Earthquake Center, the National Virtual Observatory and so on. Pegasus workflows are based on the Directed Acyclic Graph (DAG) which can represent tasks to be executed as nodes and the data- and control- flow dependencies between tasks as edges. Through the abstraction of DAGs, the wealth of research in graph algorithms can be utilized to optimize performance and to improve reliability and scalability [53].

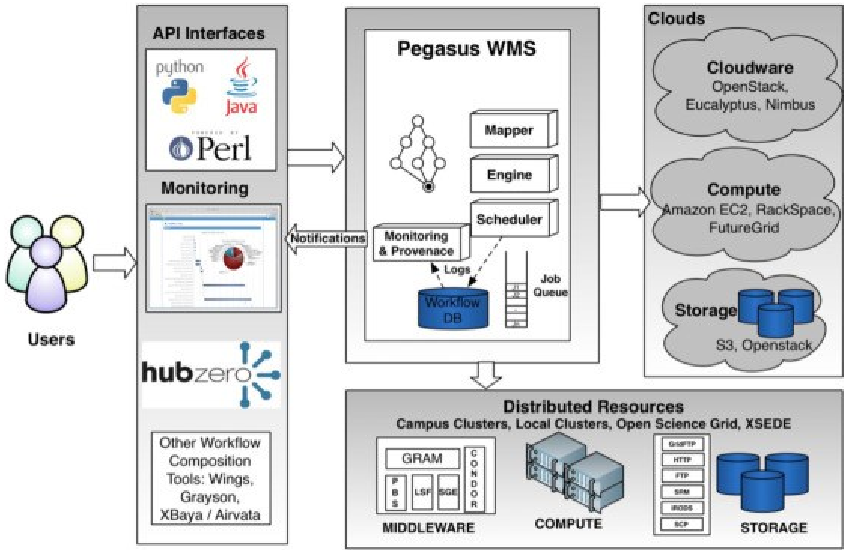


Figure 16 *Pegasus WMS architecture [49]*

Moreover, there are some commercial WFMS serving cell culture, drug discovery assay, DNA purification and analysis, etc. As a world leader life sciences company, Thermo Fisher Scientific can offer a complete solution for clinics, laboratories and pharmaceutical companies. Their products cover labware, analysis instruments, complete workstations, etc. Based on their hardware platform, Momentum is provided to execute and monitor complex processes in a powerful yet easy-to-use visual environment [54].

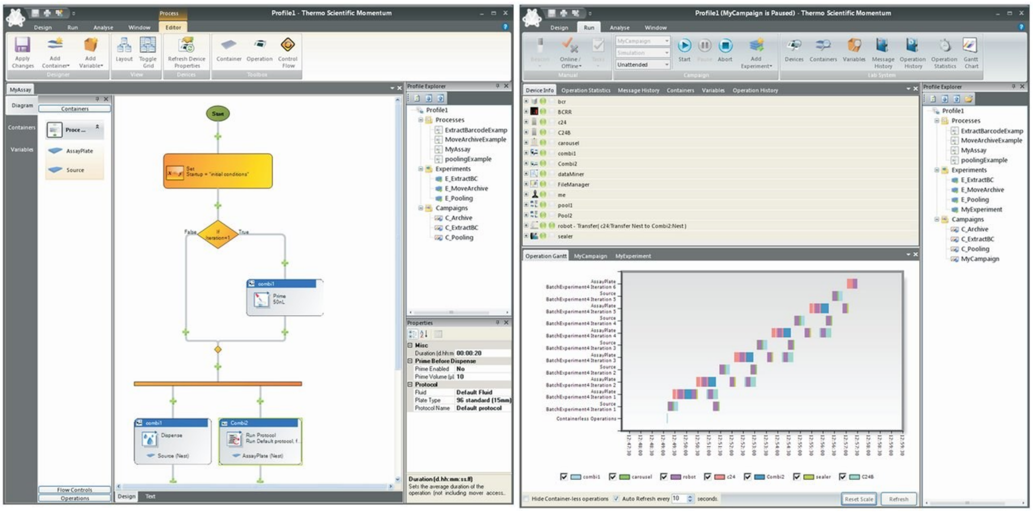


Figure 17 *Workflow definition and execution in Momentum [54]*

As shown in Figure 17, a set of processes can be defined and combined as a workflow and they can be executed under the control of Momentum. It is helpful to generate a workflow via the graphic user interface for a researcher, who has not so much experience. Momentum also provides the dynamic scheduling and rescheduling to face errors and unexpected situations during the execution. Figure 18 provides an automated workstation for nucleic acid execution. It integrates various instruments and a benchtop mover to transport the microplate among instruments. It realizes the automated execution in this workstation. It increases throughput, storage capacity and operational flexibility [55].

However, the limitation of Momentum is the lack of flexibility and compatibility. It is well suited with their own devices and workstations. Furthermore, it does not offer the connecting solution for multiple workstations. It is not suitable for a distributed environment.

Figure 18 *Automated nucleic acid extraction workstation with Orbitor BenchTrak [55]*

2.1.4 Comparison

Based on the description of these three typical laboratory management system, a simple analysis is used to compare them. The similarities of them are basic functionalities, which are suitable to manage complex laboratories or companies. They collect and store relevant information based on the central database. The reference models offer high flexibility for various environment. However, there are still difference between them.

LIMS is almost the best solution for the laboratories, which have requirement for collecting data, analyzing results and generating reports. The core function of LIMS is managing the information in the whole produce of the investigation. They include preparing samples, experiment execution, result analysis and so on. LIMS concentrate on the information and data management. However, it misses the hardware handling function or interface. Although some LIMS have additional modules for the hardware controlling, it is limited by the architecture of the LIMS because these modules still work under the LIMS. Therefore, the functionality and the stability cannot be ensured.

BPMS and WFMS have similar structure and functionality, but they are implemented in different platforms and scenarios. BPMS is more complex than WFMS since the BPMS tries to evolve from “end to end” to automate business process. It means that the current BPMS includes the resource and technologies for supporting human tasks along with automate processing applications in a way, which allows laboratories and companies to flexibly manage their works [56]. Similar to the LIMS, BPMS can offer the analysis and report support. Therefore, BPMS can handle both manual and automated activities at the same time. On the other hand, BPMS cannot offer a common interface for all the activities. It needs high customization for each functions and sub-modules. In the laboratory, a customized BPMS can take the place of LIMS but with a lower compatibility.

Table 4 *Comparison of different solution*

Solution	LIMS	BPMS	WFMS
Domain	Lab	Lab & Company	Lab & Company
Object	Information	Human & Automated system	Automated system
Complexity	Middle	Complex	Simple
Customization	Easy	Hard	Middle
Expansion capability	Limited	Good	Good
Maintainability	Middle	Bad	Good
Compatibility	Low	High	Middle

WFMS is regarded as a simplified BPMS. It can be integrated into LIMS or BPMS as a special sub-system, which handles the automated devices or stations. Moreover, it can be implanted as a stand-along application to handle the whole company or laboratories in the automation domain. It means that the WFMS is focus on the automated systems. The analysis and report function are not as the core function in WFMS in order to keep the stability and robustness of the management system. The comparison of these three management system is shown in Table 4.

2.2 Scheduling Methods

The limitation of the modern laboratory is the contradiction between the requirements of the throughput and the capacity of the automated system. When the quantity of devices is constant, the utilization rate of the automated systems is the most important factor to increase the throughput. The performance of schedulers is determined by the scheduling method and it is used to enable and optimize the execution procedure in the life science laboratories.

According to the schedule model and strategy, the scheduling problem can be transformed to a NP-hard (Non-deterministic Polynomial-time hard) problem. Typically, there are various heuristic algorithms for NP-hard to search the optimum solution, which guides the workflows orders during the execution [57]. In this chapter, three heuristic algorithms are explained. They can be used in different domains for combinatorial optimization problems. It means that these methods can solve various problems in different domain. However, one method cannot suit all problems and it is necessary to find one suitable for life science laboratory.

2.2.1 Hopfield Network (HN)

A Hopfield Network (HN), is a typical recurrent artificial neural network proposed by John Hopfield in 1982 [58], [59]. It is constructed from neurons which have N inputs and with each input i there is a weight w_i associated. Each neuron has one or more outputs, which are maintained, until the neuron is updated. The updating operations are as follows:

- The value of each input, x_i is defined and the weighted sum of all inputs is calculated as $\sum_i \omega_i x_i$.
- There are two output state of the neuron: +1 and -1. The state is set to +1 if the weighted input sum is equal or larger than 0. It is set to -1 if the weighted input sum is less than 0.
- The neuron keeps the output state until the next updating.

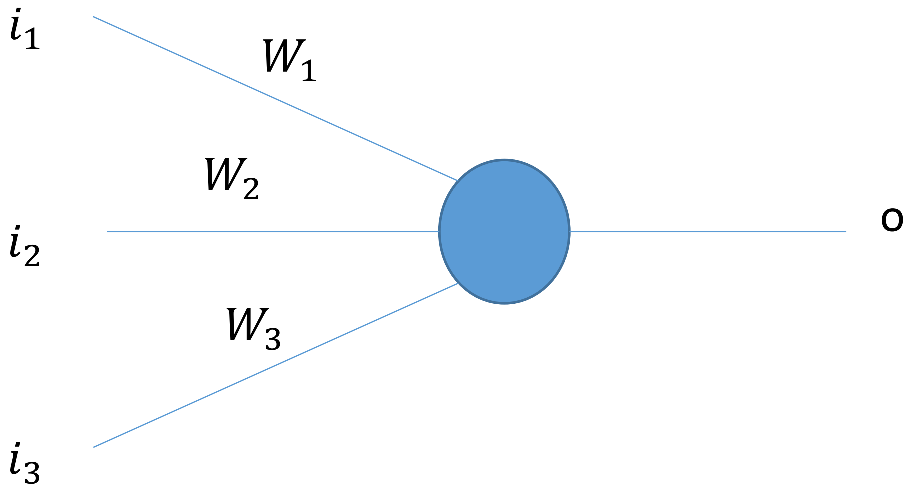


Figure 19 *Theory structure of HN*

Shown as a formula:

$$o = \begin{cases} +1 \mid \sum_i \omega_i x_i \geq 0 \\ -1 \mid \sum_i \omega_i x_i < 0 \end{cases}$$

A typical HN is a network of N such artificial neurons, which are completely connected. The weight of connection from neuron j to neuron i is defined as ω_{ij}. All connection weights constitute the weight matrix W.

Before the updating of neurons, the updating order should be decided [60]. There are two ways of updating:

- Asynchronous: one neuron is picked and its weighted input sum is calculated. This neuron will be updated immediately. Neurons can be picked in a fixed order or at random (asynchronous random updating).
- Synchronous: the weighted input sums of all neurons are calculated at the same time and updated together.

Based on these feathers of neuro, the way in which the HN is used as follows.

- A pattern is putted in the network by setting all nodes or only part of the nodes to a specific value.
- The network is updated for a number of iterations following asynchronous or synchronous ways.
- The neurons can be read out to see which pattern is in the network.

The core idea of HN is that patterns are stored in the weight matrix. The input is part of these patterns and the dynamics of the network search the patterns which are stored in the weight matrix. This is called Content Addressable Memory (CAM). The patterns, which are stored in the network are divided in two parts: cue and association. With the help of the cue, the entire pattern, which is exist in the weight matrix, can be retrieved. The training of network can help the network to create a regular weight matrix. The energy function is used to enhance the stability of pattern recognition.

Willems *et al.* presented a optimization criterion in neural networks for scheduling [61]. This criterion is always used as a range pole to assess the performance of the new approaches. They created the model for HN to solve the job-shop scheduling problem. An object-oriented HN was developed to enable the modelling of HN as a collection of connected parallel processing processes. The implementation and the results shown that, HN had a better performance for both local and global optimization criteria than others.

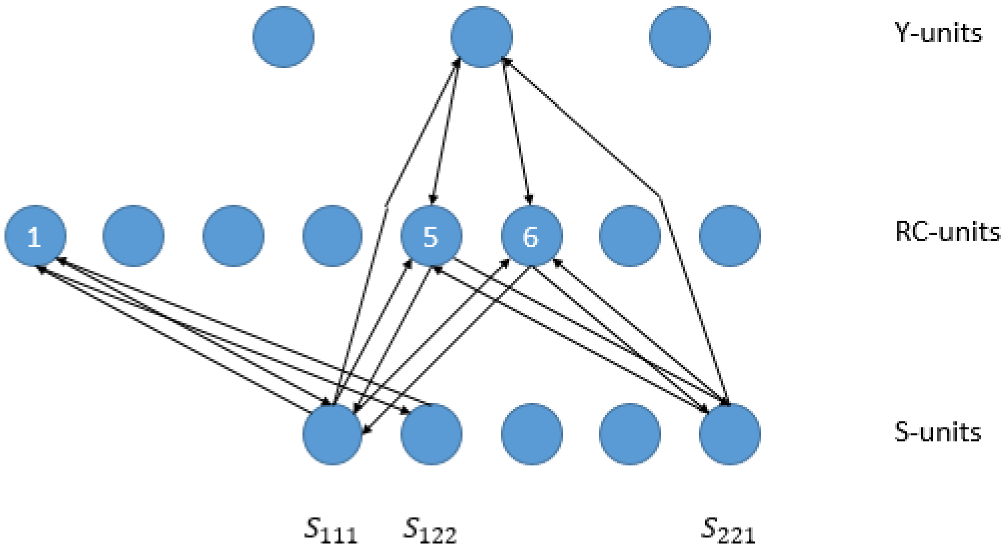


Figure 20 *Neural network for the job shop problem*

Fnaiech *et al.* proposed a new HN for optimization of scheduling [62]. Based on the method of Willems, they combined the heuristic of initialization proposed by Yahyaoui. This HN model was able to solve the scheduling problem with defect machines. This method can determine different starting times, which satisfy resource constraints, sequence constrains and unavailability constraints for flexible maintenance task. It made the system to realize a dynamic scheduling and optimized the tasks with low cost [63].

2.2.2 Simulated Annealing (SA)

Simulated annealing (SA) is a probabilistic technique for approximating the global optimum with one function or a group of functions. Specifically, it is a metaheuristic to approximate the global optimization in the solution [64].

The name of SA origins from the simulation of cooling process of heated solids. In condensed matter physics, annealing means a procedure in which a solid is heated up by increasing the temperature of the heat bath until all the solid randomly arrange themselves in the liquid phase, and then cool the heat bath through slowly lowering the temperature. In this way, all particles arrange themselves back to the ground state of a corresponding lattice [65]. SA interprets slow cooling of the hot metal as a slow decrease in the probability to select worse solutions during it explores the solution space [66]–[68].

SA is a very effective practical algorithm for scheduling optimization. In order to obtain the global optimum solution, two tricks has been used: Metropolis algorithm and Cooling Schedule [69].

Metropolis algorithm is the method for acceptance of a search step.

- Assume the performance change in the search direction is Δ.
- If it is a favorable direction, set $\Delta \leq 0$ and accept it.
- Otherwise, this step is accepted only if it meets this condition $\exp\left(-\frac{\Delta}{T}\right) > \mathrm{random}(0, 1)$

Cooling Schedule is used to guide the process of temperature reducing.

- T, the annealing temperature, is the parameter to control the frequency of acceptance of ascending steps.
- The temperature is decreased by the function T(k), and at each temperature, it is allowed to proceed certain steps following L(k).

The pseudocode is shown as figure below.

- Let $s = s_0$
- For $k = 0$ through k_{max} (exclusive):
  - $T \leftarrow \mathrm{temperature}(k / k_{max})$
  - Pick a random neighbour, $s_{new} \leftarrow \mathrm{neighbour}(s)$
  - If $P(E(s), E(s_{new}), T) \geq \mathrm{random}(0, 1)$, move to the new state:
    - $s \leftarrow s_{new}$
- Output: the final state s

Figure 21 *Pseudocode of Simulated Annealing*

Das *et al.* compared several SA variants for solving the resource constrained scheduling project problem (RCP) [70]. RCP is categorized as a NP-hard problem [71]–[74]. They toke three SA variants and applied them to solve the same scheduling problem. The results shown that SA also had better optimum search ability than other meta-heuristics such as Genetic Algorithm, Tabu Search and so on and SA incorporated with Greedy Selection Heuristic and Tabu List (GTSA_RCP) was the best method in getting optimum with maximum hit and minimum fluctuations.

Bouleimen *et al.* proposed a new SA adaptations for RCP and its multiple mode version [75]. The results showed that SA had a very good performance for optimum searching. The multiple mode version was used to improve the performance and to adapt different kinds of problem complexity and different restrictions of applied experiment conditions.

2.2.3 Genetic Algorithm (GA)

GA was proposed by John H. Holland in 1975 at the University of Michigan [76]. It is a search heuristic in the field of artificial intelligence and this heuristic is used to generate feasible solutions to optimization and search problems [77]–[79]. It is a search method based on principles of natural selection and genetics.

A typical GA requires two premises:

- The schedule problem can be transformed into a genetic representation, which is the solution domain of GA.
- A fitness function is used to evaluate the solution domain.

Based on these two premises, the particular schedule problem can be adapted to a model, which can be integrated into GA and the result of GA can be translated to the schedule solution. These processes are also called encode and decode. They are the bridges between the real-world problems and algorithms [80], [81].

The conventional GA approach has five steps as follows.

- *Initial population generating:* A certain quantity of chromosomes is created following the scheduling constraints. Each of chromosomes is an effective solution for scheduling.
- *Chromosomes evaluation:* The chromosomes are evaluated according to fitness functions. These functions will be explained in the following subsection.
- *Chromosomes selecting:* Based on the result of evaluation the chromosomes with highest fitness value are selected as the seeds for the next generation.
- *Crossover and mutation:* With the help of crossover and mutation to the selected chromosomes, a new generation is generated.
- *Return to Chromosomes evaluation and continue until meeting the stopping conditions:* Repeat the Chromosomes evaluation to Crossover and mutation until the approach is completed. The chromosome with the highest fitness is selected as the result of the scheduling.

The steps of chromosome selection, crossover and mutation are originated from the principles of natural selection and genetics. After the evaluation, the stronger individual has a higher possibility to be selected, and the procedure of crossover and mutation simulates the process of gen evolution. As a mainstreamed method of optimization, GA is widely used to solve the scheduling problems.

Hartmann *et al.* presented a competitive GA for RCP [82]. They created a new GA approach and compared it to other two concepts which used a priority value and a priority rule representation. Furthermore, this approach was used by a real-world medical research project after the transformation from the real-world data to an instance of the RCP with time varying resource request and availability [83]. The GA approach was capable of searching good schedules within moderate computation around one minute and it saved more than 10% of the duration of the project according to the previous hand-made schedule.

Driss *et al.* proposed a GA for the Flexible Job shop Scheduling Problems (FJSP) [84]. FJSP is a well-known NP-hard problem, and it is an extension of the classical job shop problem which can be represented as allocate the operations to machines and sequencing the operations on the machines in order to maximize the performance [85]–[91]. They applied a new chromosome representation and suitable crossover and mutation operators. A numerical experiment shown that this approach was effective.

Table 5 *Results' of Brandimart's data*

Problem	n*m	M&G	GENACE	Zhang	Chen	Pezzella	HGTS	NGA
		C_m	C_m	C_m	C_m	C_m	C_m	C_m
Mk01	10*6	40	40	40	40	40	40	37
Mk02	10*6	26	32	26	29	26	26	26
Mk03	15*8	204	N/A	204	204	204	204	204
Mk04	15*8	60	67	60	63	60	60	60
Mk05	15*4	173	176	173	181	173	172	173
Mk06	10*15	58	67	58	60	63	57	67
Mk07	20*5	144	147	144	148	139	139	148
Mk08	20*10	523	523	523	523	523	523	523
Mk09	20*10	307	320	307	308	311	307	307
Mk10	20*15	198	229	198	212	212	198	212

Zhang *et al.* developed a multistage-based GA for FJSP [92]. They proposed a new multistage operation-based representation to make the chromosome simpler. The comparison indicated that this algorithm (moGA) had high efficiency and the objective values were better than others as shown below.

Table 6 *Result Comparisons (8x8) [92]*

	Heuristic method (SPT)	Classic GA	Kacem's Approach	moGA
t_M	19	16	16	15
W_T	91	77	75	73
W_M	16	14	14	14

Table 7 *Result Comparisons (10x10)* [92]

	Heuristic method (SPT)	Classic GA	Kacem's Approach	moGA
t_M	16	7	7	7
W_T	59	53	45	43
W_M	16	7	6	5

These experiments show that moGA has the best performance with the lowest makespan and total workload. The advantage of this algorithm is obvious.

2.2.4 Comparison

Comparing with SA and GA, HNN has high performance for classification and pattern recognition. For optimization, HNN will return non-feasible solutions sometimes and it cannot be accepted by the laboratory management system because it elevates the failure risk.

SA has absolute advantage for global optimum comparing with HNN and GA. With regular parameters, it can get the optimum with a high probability. The disadvantage, which cannot be ignored is the huge computation effort requirement. In SA processing, there are a lot of exponentiation and logarithm operation. With the increase of processes, the computation effort is growing dramatically. The operation time is a significant problem for scheduling.

GA is a classic and wide using algorithm because its high flexibility and robustness. There are five major steps in GA: chromosome generation, chromosome evaluation, chromosome selection, chromosome crossover and chromosome mutation. All steps can be customized for specific problems and they are independent of each other. Although it is sensitive about parameters and gets local optimum, these problems can be solved by optimization of GA steps.

Based on the flexibility of GA steps, some researchers combine classical GA with other algorithms such as SA, particle swarm optimization(PSO) etc. The modified GA algorithms obtain multiple features and some of them is usable to improve the performance and stability. On the other hand, it is possible that the modified GA algorithms bring new disadvantages. Adequate evaluations and comprehensive comparison with other algorithm is necessary to find and understand new features and they are also the basis to determine whether to use the modified algorithm.

Furthermore, the intelligent algorithms are not only used to optimize the time schedule for the execution, but also for the indicator of rescheduling, which realizes the dynamic scheduling for the system. This indicator will trigger and guide the rescheduling operation according the real time system status and the unexpected situations. The discussion for indicator algorithm is presented in the chapter 5.

Chapter 3 Goals and Concept

In this chapter, the general structure characteristics of life science laboratories are described and analyzed. Based on the analysis, the goals and requirements for a flexible workflow management are proposed. Combined with existing control management systems, the concept of a hierarchical workflow management system (HWMS) is presented and developed and the overview about the concept is shown in the end of this chapter.

3.1 Current State Analysis

Chapter 1 describes the concept and development of life science automation. Accompanying this progress, the life science laboratories become more complex and multipurpose due to more and more automated instruments appearing in the laboratories. Some of them are integrated into automated workstations as a part of integrated system and others are used as standalone automated instruments. Most of them use proprietary process control systems. For example, some of the integrated systems are controlled by SAMI, which can control the processes in the integrated system and some of the automated instruments are controlled by some particular applications. These process control systems have no possibility to connect to other process control systems. Accordingly, all integrated systems and automated instruments are isolated to each other. Therefore, the first challenge is the management of different kinds of instruments and integrated systems in laboratories.

On the one hand, the complexity of the life science laboratories increases and the management of the laboratories becomes more difficult. On the other hand, pharmaceutical and biotechnological investigations require even more complex processes with a wide range of standardized and specialized sub-processes (e.g. analytical and sample preparation) on different instruments and systems in distributed laboratories. The management of these processes which are based on heterogeneous control systems is a new problem for life science companies or research institutes. The cooperation between different integrated systems is the second challenge.

Due to the recent development of localization and navigation, mobile robots play a very important role in the laboratories. Transportation tasks can be executed by mobile robots instead of human assistants. Therefore, the human work can be focused on cognitive tasks and the work stress of the staff can be reduced [93]. Thus, the interaction between robotic transportation system and automated instruments is the third challenge in the modern laboratories.

3.2 Goals of the Work

Based on the analysis of the current state and the requirements of the life science laboratories, the ultimate goal of the work is the development of an intelligent workflow management system for distributed life science laboratories to design, store and execute the investigation workflows. This system should contain the following properties:

- Super ordinated Control System

In order to control and combine the integrated systems, instruments and the transportation system at the same time, a super ordinated control system is required. This higher level system has the ability to control all the sub-systems to complete a complex workflow which requires different integrated systems and different instruments. It is used as a hub to relate various sub-systems and the transportation system in order to break the “wall” among systems and instruments. It is the base of the laboratory management system.

- Flexibility

Each integrated system or instrument has its own process control system. Not only the management structures but also the platforms of the control systems are completely different. Therefore, the integration flexibility of the superordinate control system plays an important role especially in complex laboratories. A suitable interface and a unified communication protocol are the base of the flexible integration. When there is a modification in the current system, just the interface of the sub-system should be changed for the HWMS.

- High Efficiency

In life science laboratories, one investigation requires more than one instruments or integrated systems. At the same time, more than one experiment is executed parallel in different automated systems or manual stations. The resource competition among the investigations is a tough problem. The resources in laboratories which can be competed concludes not only the visible things such as the lab wares, positions, experimental material and so on, but also the invisible resource such as the instrument operation time, mobile robots or human transportation time, monitoring attention and so on. On the one hand, if the usage is intense, it will bring more risk to reduce the stability. On the other hand, if the competition is insufficient, the operation time of the automated systems and human resource can be wasted. Accordingly, an intelligent scheduler is required to allocate the resource of laboratories and to guide the procedure of the executions.

- Stability

Chemical reactions and biological cultures are very common and basic in life science laboratories. Therefore, it is dangerous if the experiment is failed due to instable automation systems and solutions. Moreover, the high-value samples and materials are required high stability. Accordingly, the stability is an indispensable factor to be considered during the design of the super ordinated management system for life science laboratories. Typically, the stability is related to the error frequency. A system with lower error frequency runs more stable. From the planning to the real process execution there is a lone chain of the data transformation. It is necessary to build a model to avoid errors.

Moreover, the unexpected situations cannot be neglected. The mobile robot transportation delay because of collision avoidance and human-robot interaction is normal in the laboratory environment. But the delay cannot be forecasted or prescient. How to handle these unexpected situations is a great influence factor for the execution stability.

- Ergonomic Design

The ergonomic design can be reflected in two parts: planning and monitoring. A bad design of the planning user interface can confuse the user and bring more workload. The confusion and stress can lead the user to make more and serious mistakes. Inevitably, these mistakes can make the investigations failed. Moreover, they may damage the instruments, mobile robots and threat to the health of assistants. During the execution, a readable monitoring user interface can help the user to handle the procedure correctly.

3.3 Concept of this Thesis

In order to realize the goals, the concept of this thesis will be presented below. In general, there are four major parts system definition, structure, interface and scheduling.

3.3.1 System Definition

According to the introduction in the chapter 2, there are some kinds of management system for life science laboratories. It is necessary to make a deeper discuss to make sure weather a known system for laboratories is suitable for the goals.

- LIMS is designed for laboratories which have automated instruments. Mostly, LIMS is used as information collector, data storage and result analysis. LIMS has limited or absolutely no ability to control the process execution in laboratories. Therefore, LIMS is not suitable for the planned development.
- BPMS is utilized to analyze, design, implement, optimize, and monitor the business processes. As the connecter between theory and practice, BPMS enjoys contributions from different research domains such as economics, engineering, computation and so on. In recent years, BPMS appeared in automated laboratories to evaluate and improve the procedure of experiments. BPMS does not focus on the control level of laboratories. Therefore, there is no possibility to integrate an automated system into BPMS directly. If it is necessary, an additional WFMS is required for the automated systems.
- WFMS is a perfect system to manage and control systems. The features of WFMS really match the requirements of life science laboratories. It is a high-efficiency solution to help the user to design, implement and control the processes. It is suitable for many domains such as chemistry, biology, genetics and so on. Of cause, more functions bring more possibility to the system, but the irrelevance can pare down the stability and usability of the system.

Following the features of these existing system, a hierarchical structure is shown as below.

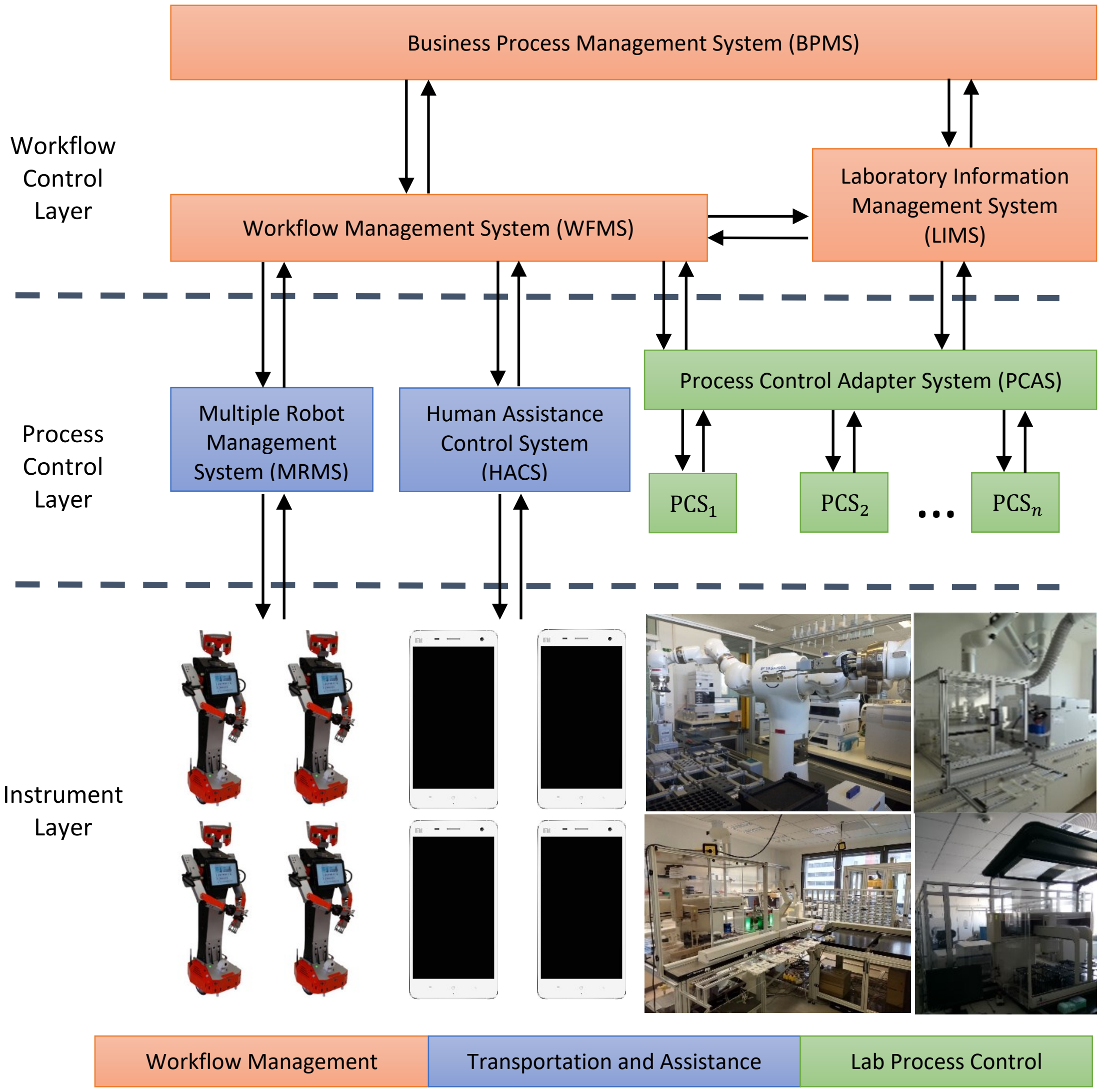


Figure 22 *Structure of laboratory management system*

There are three layers in this structure – the workflow control layer, the process control layer and the instrument control layer. As the environment of laboratories, the process control layer and instrument control layer are existing. A BPMS is used to control both human assistant and automated system at the same time. Based on the analysis result and the location distributed in the structure, a WFMS is suitable for our goals.

From this structure, the difference between LIMS and WFMS is clearly shown. First, LIMS has no possibility to integrate the transportation system because the object, which can be managed by LIMS is the information of investigation instead of the process controlling. Therefore, the second difference is the data type. Between LIMS and sub-systems, the data which is transmitted is about the state and result of the investigation. The data between WFMS and sub-systems is the command and feedback which is used to track and confirm the operation procedure of the sub-systems in order to allocate the resource reasonably and with high performance.

In the process control layer, there are three process control systems: Multiple Robot Management System (MRMS), Human Assistance Control System (HACS) and Process Control Adapter System (PCAS). The former two systems can carry out the transportation tasks. MRMS can integrate multiple various robots and handle them to transport samples in the laboratories [94]. HACS is not only used for the transportation task, but also for error handling in the laboratories [95]. PCAS integrates various process control systems of automated workstations and instruments.

Based on the introduction of WFMS in chapter 2, a typical WFMS is very huge. It is not necessary to keep the redundancy for the compatibility because in laboratory environment, the objects are unitary and fixed. The agile system requires the customization modeling. The model definition is the essential of the system.

There are four kinds of model in this system: *Process, Transportation, Workflow* and *Task*.

The functions of the automated systems are defined as *Processes* in the planning phase. The process model includes the parameter 'Position', 'Device', 'Method'; parameter and requirements as shown Figure 20.

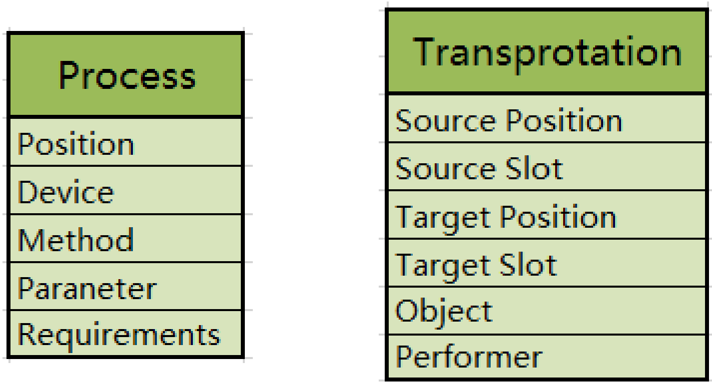


Figure 23 *Model of Process and Transportation*

As explained before, a process is an activity of the automated system. Automated systems are controlled by the local PCSs or ICSs, which store activities. Of cause, different local controllers have different forms and structures to save activities. The process model offers a universal architecture to represent various activities. Transportation is the connection between two processes. It includes the parameter ´Source Position´, ´Source Slot´, ´Target Position´, ´Target Slot´, ´Object´ and ´Performer´.

The transportation has two major functions. On one hand, it guides the transportation management system to complete the transportation task. It includes all the information for the sample transport: the source and target details, sample details and performer. On the other hand, it is the connection between two processes and make them in relationship. In other words, transportations combine various processes and determine the priority of the execution.

Several processes and transportations constitute a workflow and are stored in the system. The workflow is not only the media to connect the planning phase and execution phase, but also the information carrier in the system, which represents a set of processes and transportations. The processes and transportations are but only saved in a workflow form. Any single process or transportation will not exist in the system.

During the execution phase, the workflow splits in several tasks which can be handled by the execution controller and transportation management system. The task information is also used to operate scheduling procedure.

With the help of the customization modeling, a new WFMS named hierarchical workflow management system (HWMS) is definite. It is positioned as a materials flow management system. Therefore, the HWMS is close to sub-systems including various automated systems and transportation solutions.

3.3.2 System Structure

Similar to other management system, HWMS is a data-oriented system. As the main part of the system, the database plays an irreplaceable role. The system structure is shown as below.

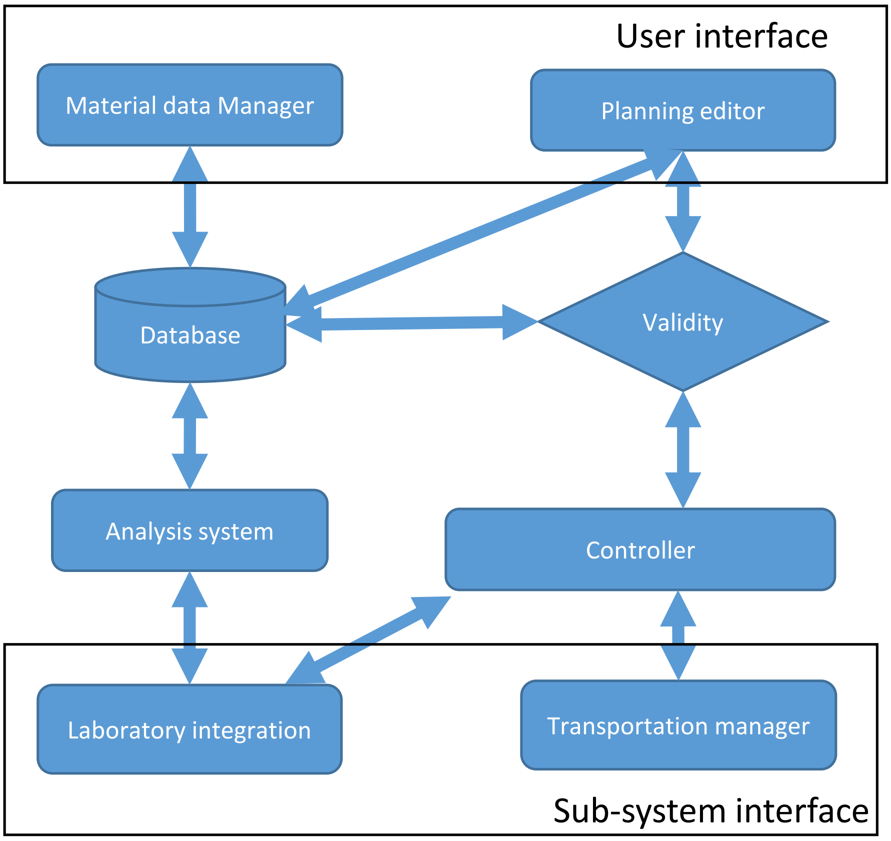


Figure 24 *Structure of Initial HWMS*

Following this architecture, the relationship of these modules is shown clearly. There are the procedures of HWMS:

1. The ´Material manager´ is used to create or edit the basic data in the laboratory such as to create a new slot or location. This data is stored in the database.
2. Based on the material data and the information form the local controller of the automated system, some basic activities are created in the database of HWMS, such as dilution.
3. Before users invoke an activity, the process should be instantiated by parameters. In the planning editor, users can create various processes and connect them with transportations as a combinational process group – workflow. This workflow will be stored into the database.
4. ´Validation module´ is used to guarantee the validity of the process which is created by planning editor. When the process is created successfully, that means the workflow layer is completed. Usually, the workflow will not be executed immediately after the planning. Between planning and execution, there might be some modifications of the methods which are contained in the workflow. Therefore, the data validation checking is necessary before and during the planning and the execution to ensure the system stability. The parameters of the processes and transportation endpoints information will be checked before the execution. Before the execution, the workflow details will be sent to the scheduler. Based on the priority and the parameters of the workflow, a time schedule can be generated to guide the procedure of execution.
5. Following the time schedule, the controller sends order to the laboratory integration system and transportation management system. On the other hand, the controller also collects the operation status and reports back to scheduler. When it is necessary, the scheduler will update the time schedule
6. If the process is an analysis operation, the result will link to the execution record in the database. When users search for the reports, they can directly find them in the execution history

As shown in Figure 21, there are two kinds of interface in HWMS, user interface and sub-system adapter interface. The user interface is used to help the user to definite the static environment and create workflows. The sub-system adapter interface is existing in the execution controller system. It is the bridge to connect the sub-system and HWMS.

3.3.3 User Interface

User interface is the connector between the user and HWMS. The usability of the user interface reflects the ergonomics of the system and includes the following points [49]:

- Learnability: The system should be easy to learn so that the user can rapidly start getting some work done with the system.
- Efficiency: The system should be efficient to use, so that once the user has learned the system, a high level of productivity is possible.
- Memorability: The system should be easy to remember, so that the casual user is able to return to the system after some period of not having used it, without having to learn everything all over again.

- Errors: The system should have a low error rate, so that users make few errors during the use of the system, and so that if they do make errors they can easily recover from them. Further, catastrophic errors must not occur.
- Satisfaction: The system should be pleasant to use, so that users are subjectively satisfied when using it; they like it.

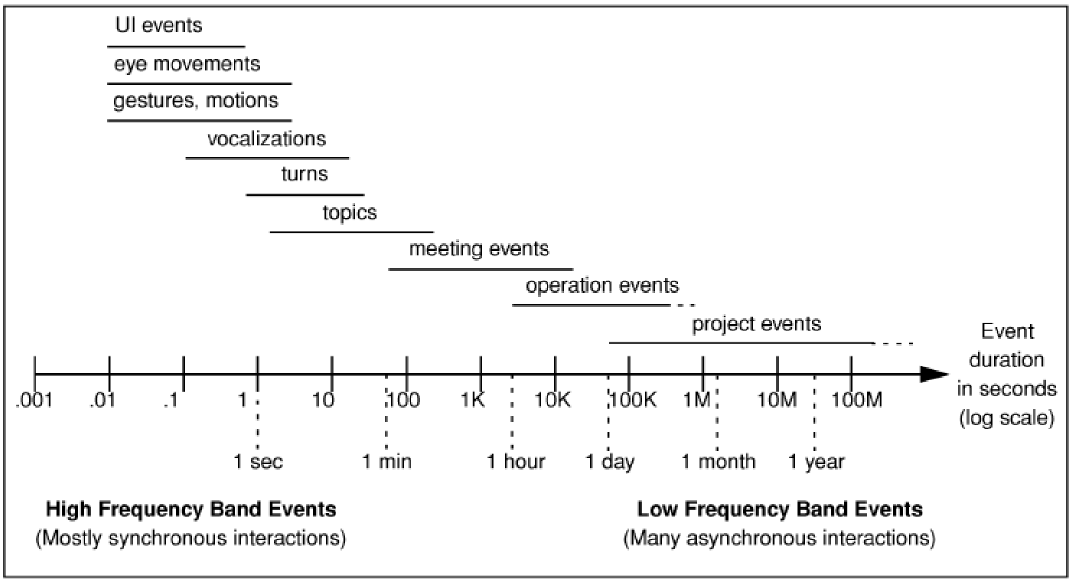


Figure 25 *The durations of different types of human computer interaction (HCI) events[11]*

This figure indicates the durations of different types of human computer interaction (HCI) events [11]. The horizontal axis is a log scale indicating event durations in seconds. It ranges from durations of less than one second to durations of years. The durations of UI events fall in the range of 10 milliseconds to approximately one second. The range of possible durations for each "type" of event is between one and two orders of magnitude, and the ranges of different types of events overlap one another.

UI events can exhibit much higher frequencies when in sequence, and thus might be referred to as high-frequency band event types. Likewise, event types of longer duration, such as project events, exhibit much lower frequencies when in sequence and thus might be referred to as low-frequency band event types.

UI of HWMS will be designed as following principles:

- UI bases on WEB technology. The main advantage of WEB UI is that users do not need to install or update the client part when they want to connect to the system. That means users can operate the system in flexible environment.
- The elements on UI will be as simple as possible, when the system has ability to acquire necessary information.
- The color and shape of element is considered in order to make user focus on planning

3.3.3.1 Development Solution Analysis

The major form to develop a user interface is to create particular applications in different platforms. Normally, the WinForms will be used as the form to develop the user interface when users work on the windows platform [96]. With the help of WinForms, the interface is easy to develop and running stable. The data interface for WinForms in windows platform is thorough and safe. The disadvantage of this form is also obvious. It can just work under windows operation system and the update needs a huge effort.

With the development of mobile computation technologies, more and more systems require the multi-platform possibility. In the life science laboratories, mobile devices are used everywhere. The multi-platform possibility of the user interface is also one of the requirements. The user interface based on web is one of the best solutions which can run on multi-platform. The comparison of these two development forms is shown below.

Table 8 *Comparison of development forms*

Solution	WinForms	WEB
Develop environment	Visual Studio	Poor
Platform	Windows Platform	All
Data interface	thorough	complex custom
Update	User download and install	server update

The advantage of the web technology is not only the multi-platform possibility, but also for the updating. In the laboratory environment, users access the system with serval computers or mobile devices. With the web technology, when the server is updated, all the user interface is also updated.

The disadvantage is the lack of development environment for the alternative solutions. The solution selection will be discussed later.

3.3.3.2 Development Solution Selection

Based on the web technology, there are several solutions for the development of user interface. The figure below shows three mainstream solutions.

Table 9 *Comparison of development solutions*

Development Tool	Adobe Flash	Silverlight	HTML5
Develop environment	Flash CS4	Visual Studio	Eclipse
Platform	without mobile system	Windows Platform	All
Data size	big	middle	small
Flexible	bad	normal	good
Data interface	bad	thorough	self-custom

Adobe Flash is very popular in the PC platform, but it is too heavy to run on the mobile platform. Actually, the Silverlight can support to the mobile device. However, it required still windows mobile operation system. HTML5 is suitable for our requirements.

According to the development solution analysis part, another imperfection of the web form is the data interface. Our database is based on the SQL server and it is better to choose a Windows based development environment.

ASP.NET MVC offers input logic, business logic and UI logic in one template.

MVC is an architectural pattern which enables the development of an application having loosely coupling between each of these elements. According to MVC system should be divided as M (Model), V (View) and C (Controller).

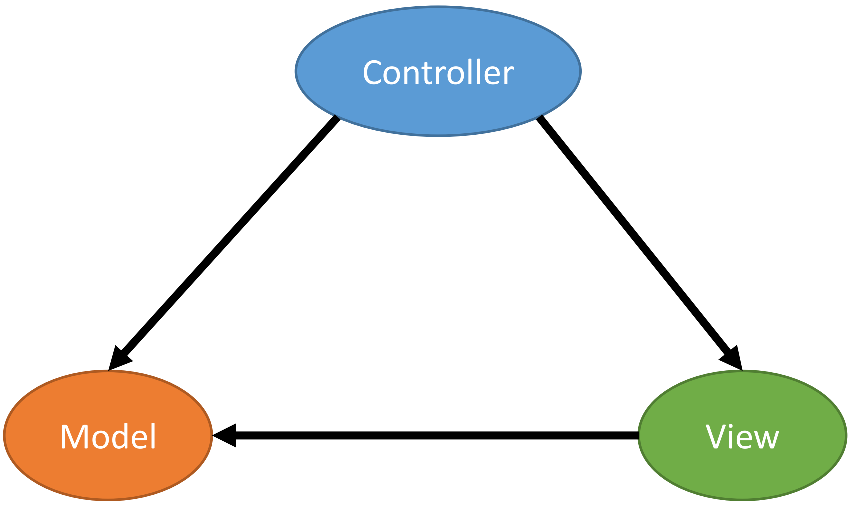


Figure 26 *MVC structure*

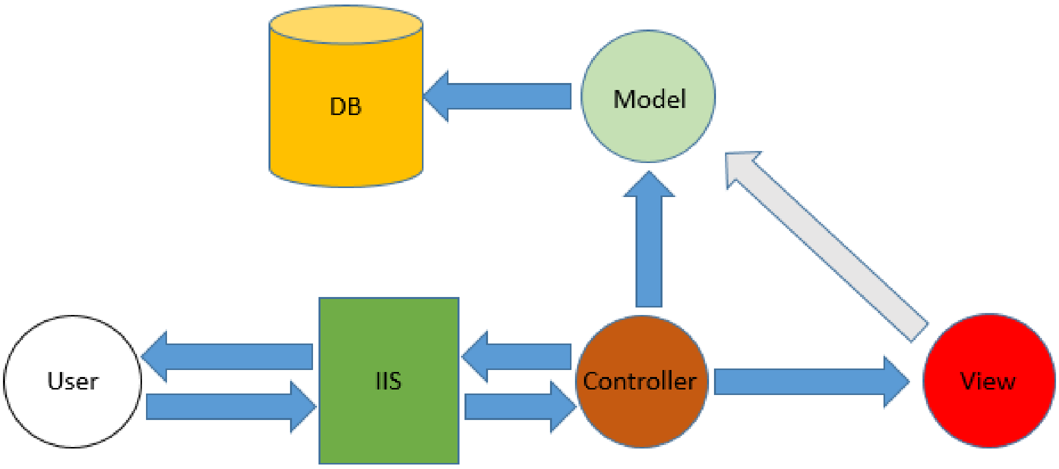


Figure 27 *MVC running procedure*

The MVC is running under the procedure as below:

- User makes the request for some resource in server.
- Request comes to controller first.
- Controller if required talks to the model for data.
- Model operates on database and returns data to the controller.
- Controller chooses the appropriate view.
- Controller passes the data to chosen the place where data will be populated as per convenience.
- Controller sends back view to the user.

The Advantages of MVC are shown below:

- Test Driven development and Reusability.
- Performance. It doesn't have support for view state.
- Full control over HTML.
- Support for parallel development.
- Extensibility. It supports multiple view engines like aspx, razor and if required we can create our own.
- The only disadvantage is more learning effort.

The disadvantage is the huge effort for development. Because this solution requires not only the web development knowledge such as JavaScript, Cascading Style Sheets, but also needs the knowledge on the ASP.Net platform. The most difficult part is how to combine this knowledge to create customized user interface.

3.3.4 Sub-system Adapter Interface

As the analysis before, the requirements of the sub-system adapter interface are embodied in two parts– flexibility and compatibility.

The network of laboratories guarantees the possibility of connection between the sub-systems to HWMS. Based on this network, several protocols can be used to create the communication for information transmission. Normally, the underling protocols such as TCP and UDP are popular in the laboratory environment because of their high performance and flexibility. UDP is also be used to transmit files and huge data. There is no guarantee of delivery, ordering, or duplicate protection. Therefore, it has high performance but poor stability. TCP increases the stability by a more complex protocol mechanism. Although the performance of TCP is not as high as UDP, it can keep the connection to make sure the usability of the network.

However, in HWMS, it is not suitable to use the underling protocols. There are two reasons:

The scale of the information is small. Thank of the customization modeling, the information, which needs to be transmitted is quite succinct. Although the underling protocols are flexible, it is hard to build and unify form for heterogeneous controllers of sub-systems.

Interface based on service-oriented architecture (SOA) is the best solution. The of this method is to build unify models for each sub-system. In these models, the functions of sub-system are packed as service to be integrated into HWMS. Of cause, the communication model is included in the model. Following this way, the stability of HWMS can be increased and at the same time, the compatibility is also guaranteed.

3.3.5 Hybrid Scheduler

For the tiny and simple laboratory with one or two automated systems, the scheduler is not necessary since the execution can be operated gradually in one thread. This strategy has high robustness and stability. However, for a group of life science laboratories, this strategy is no longer suitable. It brings unbelievable time waste for waiting and reduces efficiency advantage of the automated systems.

In order to increase the performance of the laboratories, an intelligent scheduler is required. The scheduler is used to guide the execution procedure. With the help of some algorithms, the static scheduler can find an optimized solution to allocate the resources of the laboratories which are comprised of automated systems, mobile robots, slots, lab wares, human assistants and so on. One more thing, the transportation task can be served by human or mobile robots. How to keep the balance of the task assigning will be also considered. The static scheduler improves the efficiency of the laboratories and reduces the cost.

Cause of the complexity of the laboratories, the processes which are operated in sub-system cannot always finished on time successfully. During the execution, the events occur all the time. Some of them are just follow the static schedule but others are not. How to handle the events out of the schedule is a critical problem for the system because they influence the stability and robustness of the system directly. Some of the exceptional event, which are not so serious such as the delayed transportation task can be solved by dynamic scheduling [97], [98].

In response to the problems posed by the uncertainties, static scheduling is no longer suitable because of the huge computation effort, which guarantees the global optimization. It is not agile to face various unexpected situations. Dynamic scheduling, which is operated during the execution can increase the system stability dramatically.

Dynamic scheduling is defined under two categories: completely reactive scheduling and predictive-reactive scheduling [99]. Completely reactive scheduling does not generate time schedule in advance and the execution procedure is made locally in real-time [98], [100]. Predictive-reactive scheduling, which is used in this approach, is an event-driven rescheduling process based on the current situation [101]–[104]. The result of the rescheduling is used to update the primary time schedule. Predictive-reactive scheduling is the most common approach because it can be transformed into a static scheduling problem easily and fully utilize the scheduling environment, which includes the data interface and even schedule engine. Inevitably, static and dynamic scheduling require different operation features, which can be obtained by using various scheduling algorithms. That means, with different scheduling algorithms, the hybrid scheduler can be operated as a static scheduler before execution and as a dynamic scheduler during execution. Several algorithms are discussed and evaluated in this study to select the reasonable method for these two scenarios.

Chapter 4 HWMS Realization

In this chapter, the detailed information of HWMS is explained. There are three sub-systems in the data area - Laboratory Material Management System (LMMS), Workflow Planning System (WPS) and Workflow Execution System (WES). Each of them faces one kind of data and they consist also a hierarchical structure. LMMS is the base of the WPS and WES just used the output of WPS.

4.1 Laboratory Material Management System (LMMS)

LMMS is the base of the whole system. It is used to store the core data of the laboratories including location, device, slot and container.

'Location' is the information to define an area. There are three levels of location in HWMS: Building, floor and room/laboratory. As a parameter of 'device', the location information should be the level of room/laboratory. 'Device' represents various automated systems. One or a combination of several functions can constitute a 'method', which is stored in a specific 'project'. It is a unit to defined a process. 'Slot' is based on the information of location or device. It is defined as a formed plate, which is fixed in the device or in a specific location. The form of the plate and the height of the free space limit the usability of a slot. 'Container' is the carrier of experiment materials. It includes various labware. However, it also can be used to carry other containers. For example, tube rack is the container for tubes. The container should contain the slot information instead of the location. In a sense, the slot is a lower and more detailed location for a container.

In order to enhance the performance of the database, all of them have be classified by different properties.

4.1.1 Data Structure

There are four parts of data in LMMS – Container, Slot, Location and Device as shown in Figure 26.

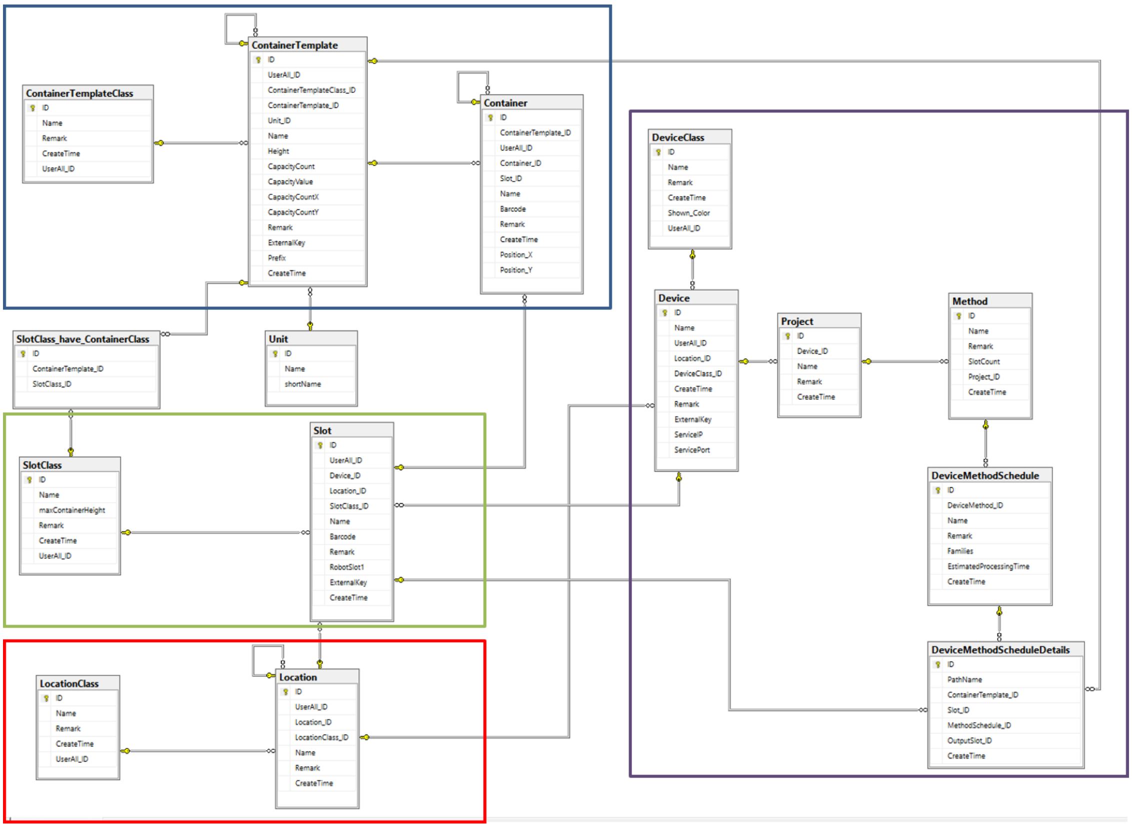


Figure 28 *Data structure of LMMS*

Container includes the labwares and their rack. The table of ContainerTempleClass defines the template type and ContainerTemplate explains different kinds of Containers. The Container table stores the particular labware or rack. The Device table has the core function data of the instruments and automated systems. They pack one or a series of functions as a method. Instead of the specific contains, the requirement of a method is the combined information of ContainerTemplates and slots. That means the method is not bound with one or some specific labware. It requires containers with correct template in the specific slots.

As shown in Figure 29, the labware is located in a storage frame named “hotel”. It has dozens of slots, which contain the labware with the standard footprint of micro titer plate (MTP). Each slot has its own ID and barcode. In this figure, there is a micro titer plate with 96 wells named “MTP 96x300µl”. It is a normal labware for biological test reactions and chemical analysis. As a container of samples, this labware is managed with the help of ID and barcode. Moreover, the slot information of each labware can track the sample all the time.

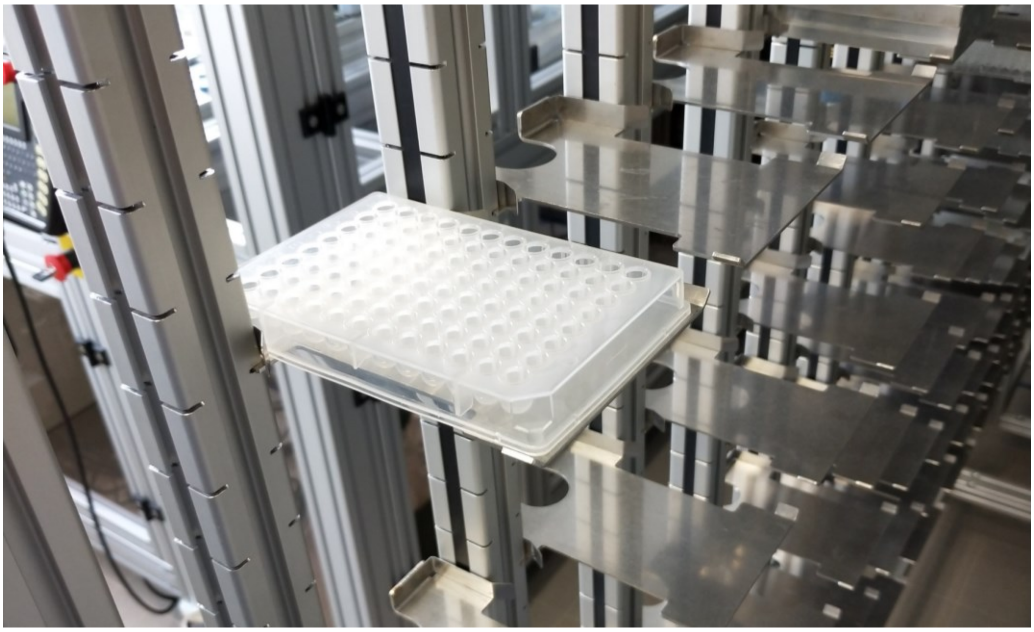


Figure 29 *Labware and slots in the laboratory*

Figure 30 *Automated workstation "Reformatter"*

As shown in Figure 30, the automated workstation “Reformatter” is defined as a ‘Device’ in LMMS. There is a PCS in the workstation to control and manage the automated devices and desktop transport robots. Moreover, the PCS offers functions and ‘methods’ information, which is stored in the database following the hierarchical data structure ‘project’, ’method’, ‘DeviceMethodSchedule’ and ‘DeviceMethodScheduleDetails’. This information is the most important component for the definition of the process as explained later.

4.1.2 Data Management

The data management system for the LMMS is a web-based system. The user, who has the extent of authority can edit the core data of the laboratories via any computer which connects to the internet. Of cause, this system shows the core data on the website (shown in Figure 31), which is helpful for the workflow designing.

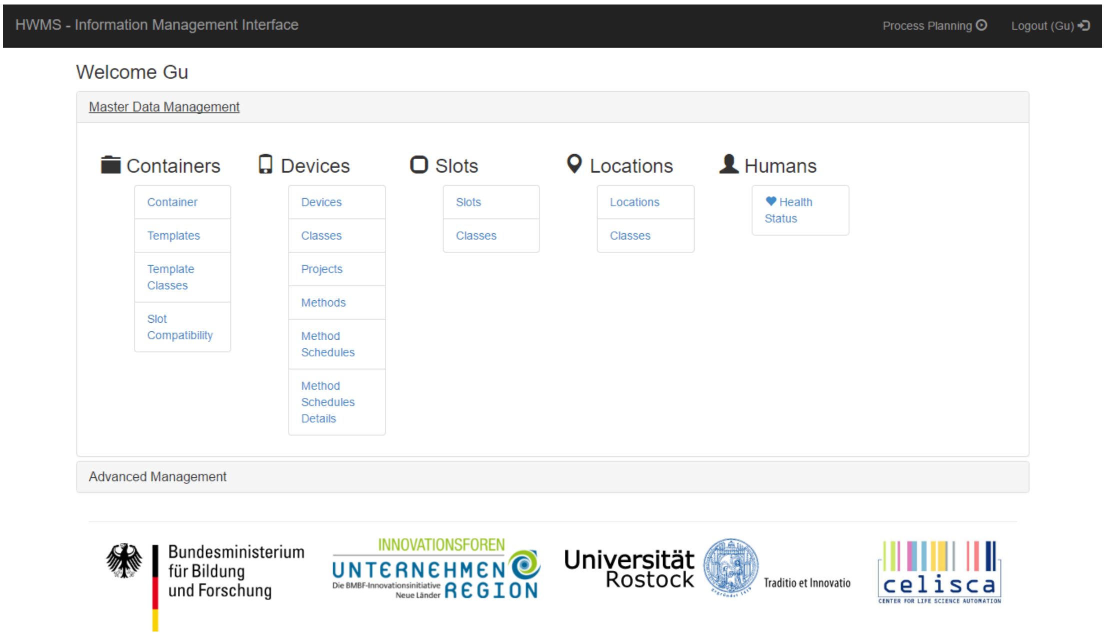


Figure 31 *Website for data management in LMMS*

With the help of LMMS, user can check all the material data easily, which includes the information of current status of containers, slots and devices. These information is the base to design a workflow for the investigation. For the data security, not all the user can modify the core data of LMMS totally. The users are divided into three groups: administrator, researcher and assistant. Administrator has the highest authority to handle the whole database. Researcher can create and edit workflow data, which is not including the material data. Assistant is able to access all the data but cannot change anything. During the login process, LMMS recognizes the group information of the user and gives the user corresponding permission and this permission is also usable in the whole HWMS.

The advantage of LMMS is that users can operate the system in anywhere and anytime when they have the internet connection. It is not limited by the operation system or platform. Even with a smartphone, users are able to search for their interested information. On the other hand, LMMS is a stand-alone system. Even during the execution, the access and storage of the system are not influenced.

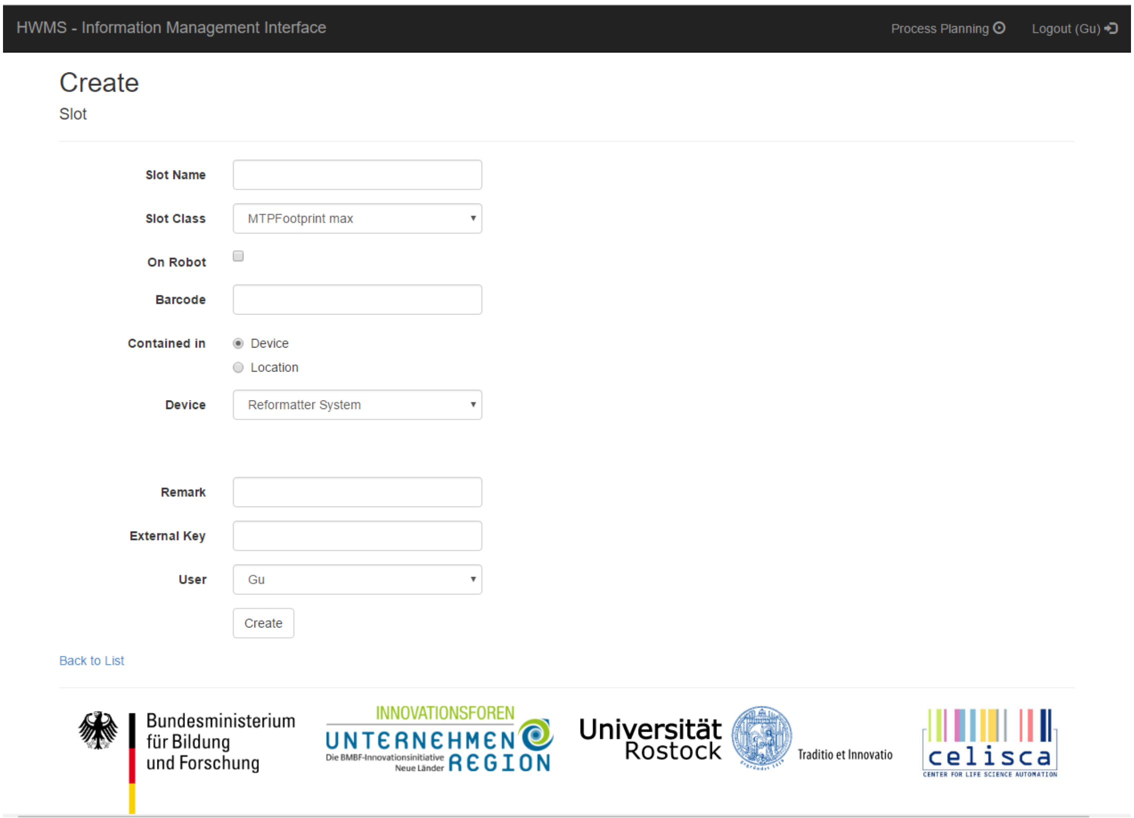


Figure 32 *Creation of new slot*

The laboratory environment is dynamic and flexible. There are always changes for the material data before and during the experiment. LMMS can be used to create new item such as container, slot, device, location etc. When there is a new slot, the new information can be insert into the database via the interface(webpage) as shown in Figure 32. The customized template and ergonomic design guide the user to make a creation easily and to avoid error. For example, the slot is related to a location or a device and the radio button for device and location decides which combo box is available, device or location. There are a lot of locations in the laboratory, so they are in a hierarchical structure. LMMS guides the user to select a correct location with related combo boxes. Of cause, when the user type in a wrong information, LMMS will notice the user to correct it before the creation.

4.2 Workflow Planning System (WPS)

Based on the LMMS, user can create specific processes via the Planning Editor. In the WPS, there are three kinds of objects: Process, Transportation and Workflow. Several processes and transportation form one workflow and the data structure is shown below.

4.2.1 Data Structure

The data structure of WPS is shown in Figure 21.

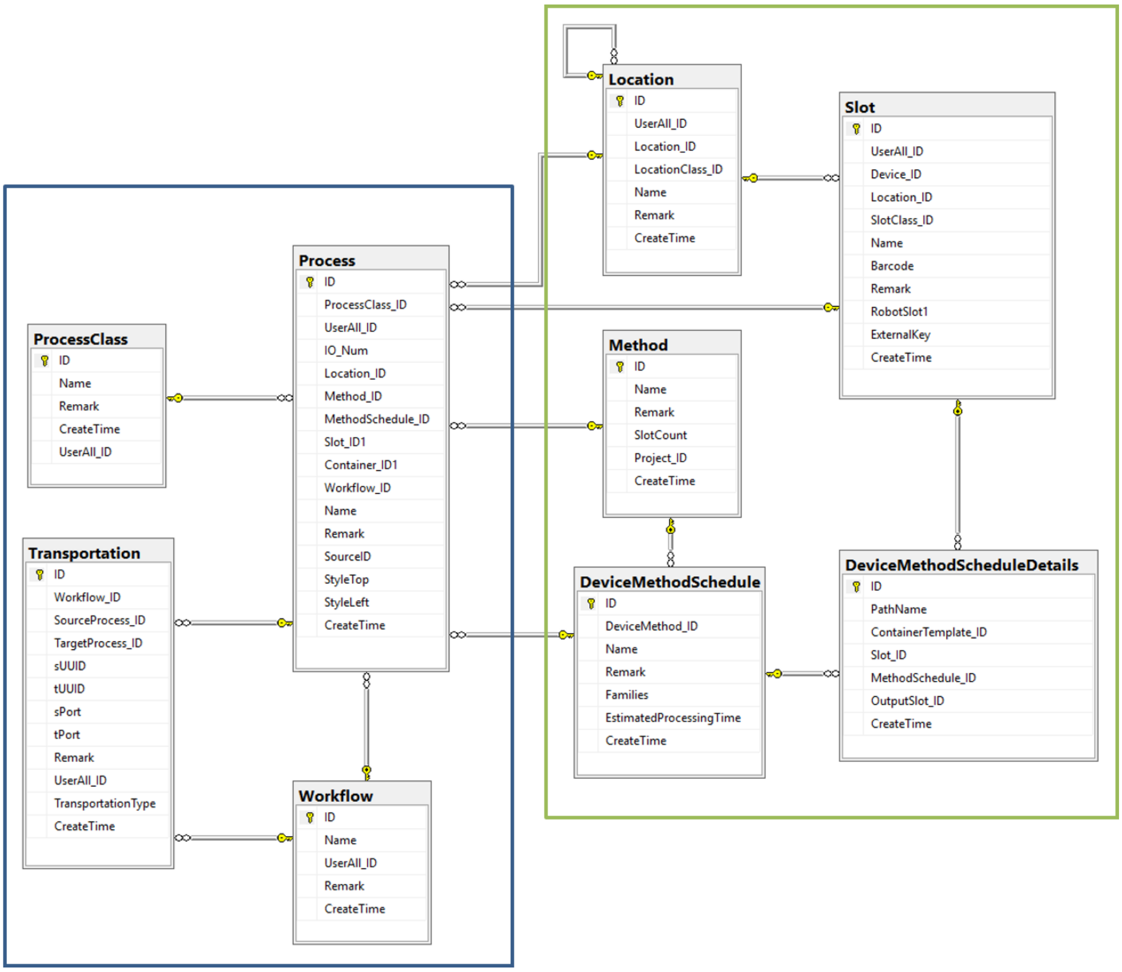


Figure 33 *Data structure of WPS*

The 'Process table' contains all kinds of core data and it is the adapter to combine LMMS and WPS. There are two types of processes in this system – normal process and IO process. IO process is used to define the input and output for the workflow. It is a special kind of process, which is created in pare – one input and one output. The normal process has all the properties of the functions of the automated instrument or station. The connection between two processes is a transportation. It can be used to track samples and guide the transportation tasks.

4.2.2 Process Management

This section explains how to create and edit the 'Process'. There are two kinds of processes in HWMS: normal process and IO process. The normal process is the representation of the activity of the automated systems and the IO process defines the input and output.

As shown in Figure 34, one input and one output are created at the same time. Usually, when the user wants to add a sample to the workflow, the sample should be contained in a container such as "*MTP 96x300µl*" with the name "*C02*". In the end of the workflow, it is necessary to store the sample in a specific slot. There are too many slots in the laboratory, so the location information is used to select the slot.

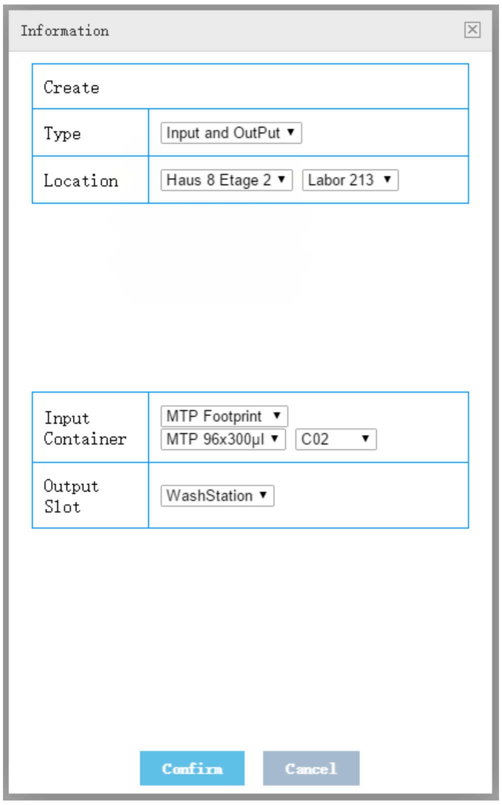


Figure 34 *UI for IO Process creation*

Based on the information from LMMS, the slot information of the input container can be found in the background of WPS. Therefore, the user has not to type in the slot information of the input. The output slot is the final slot of the sample after the experiment.

In addition, the valid check is implement before the creation. As the most important properties of container, footprint and height will influence the slot selection. Definitely, a tub is not able to be stored on the slot with “MTP Footprint” directly. In order to store more containers in a limited area, the height space of the slot has also limit. When the height of the container is higher than the limit of the slot, WPS will warn the user to change the slot.

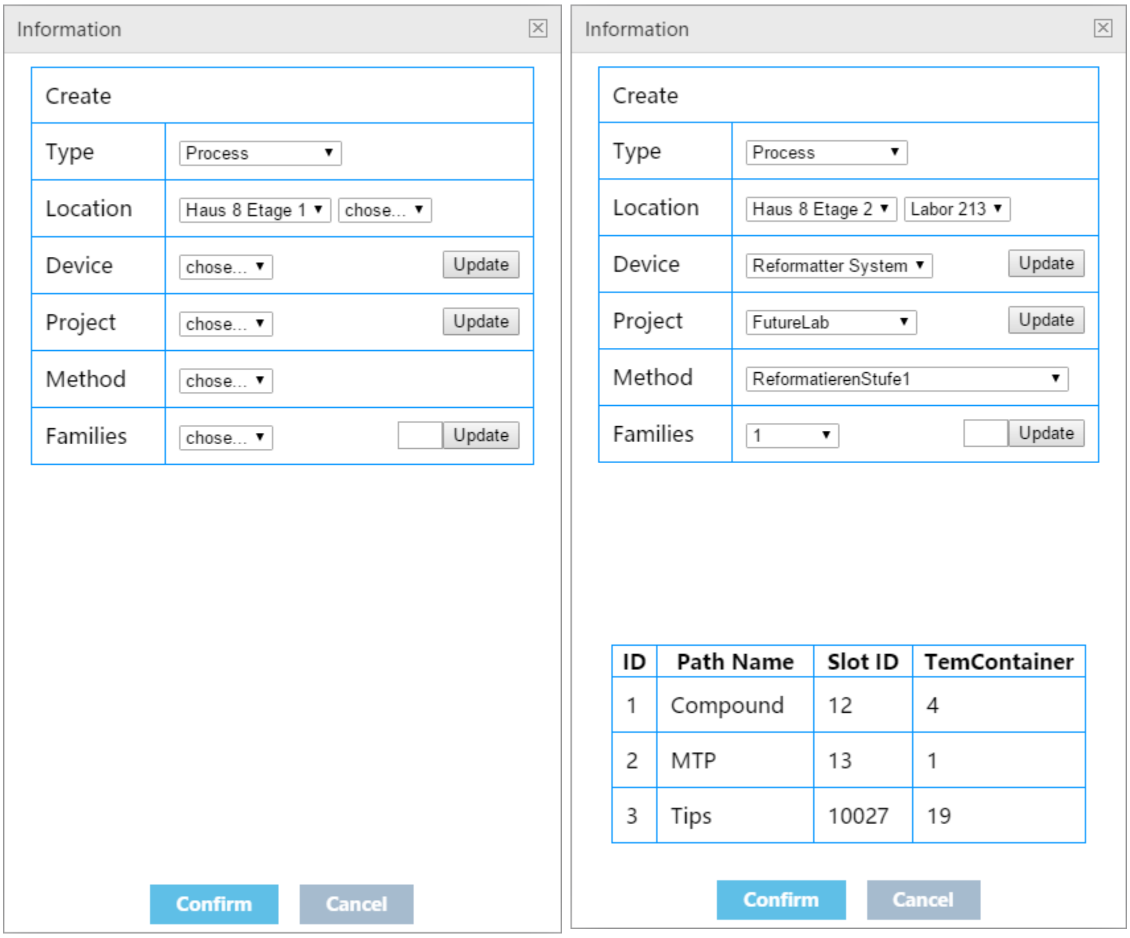


Figure 35 *UI for normal Process Creation*

Figure 35 presents the UI for normal process creation. In the created dialog, the table above includes the basic information of the new process such as location, device, project, method and families. The table below is appearing after the family's number is selected, which shows the details of the method schedule from PCS.

The essence of the normal process creation is to select one activity schedule form the data table 'DeviceMethodSchedule' and each item correspond several items from the data table 'DeviceMethodScheduldeDetails'. These details are the core information for automatic execution and transportation tasks. There are over one thousand 'DeviceMethodScheduldeDetails' in our laboratory and the number is still increasing. It is unbelievable to search them directly by users. They are stored in a hierarchical data structure with layers of location, device, project and method. Based on this structure, users can easily find the schedule of the activity.

In order to avoid mistakes, the related combo boxes are used to search for the correct activity based on the hieratical structure. Furthermore, there is additional table showing the list of 'DeviceMethodScheduldeDetails'. This table can notice the user to choose the correct schedule.

When the process is already created and needs to be changed, the process edition UI as shown in Figure 36 can modify the process.

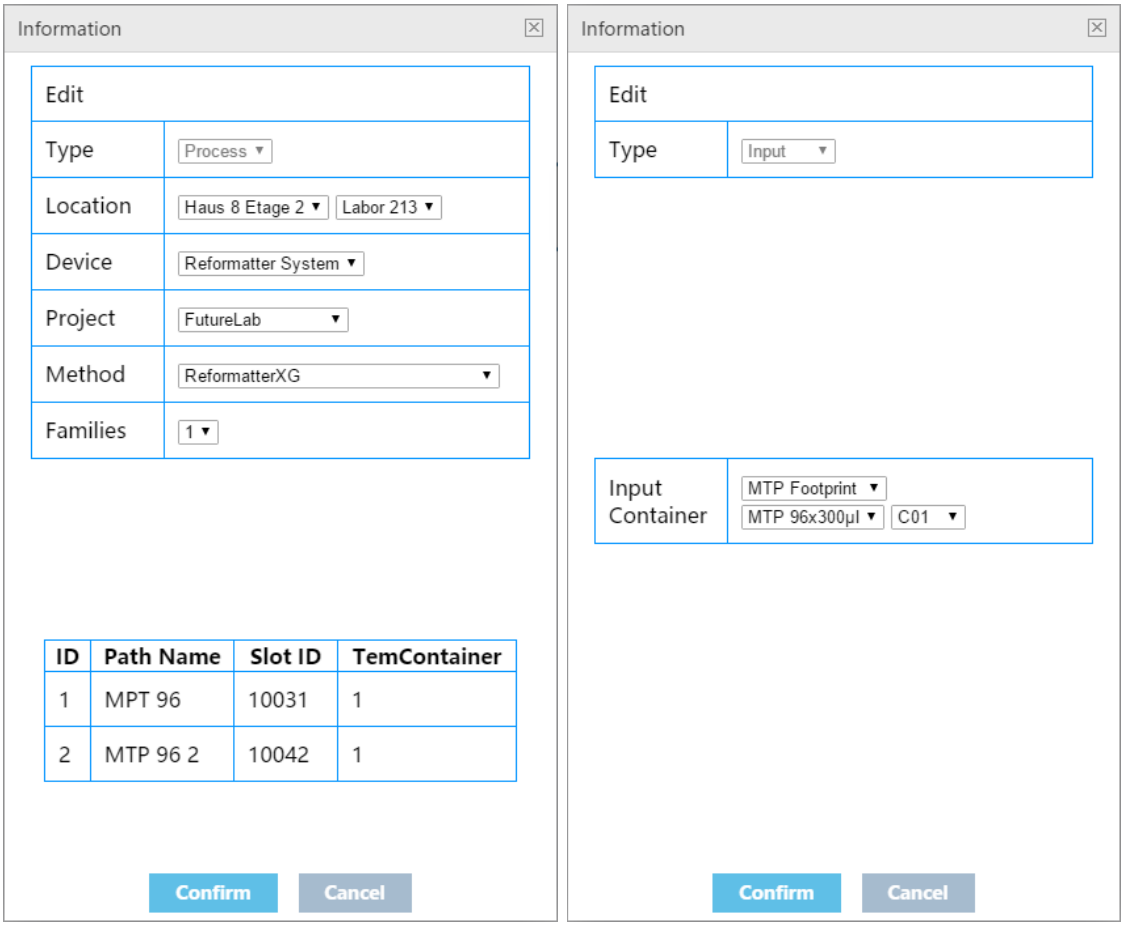


Figure 36 *UI for process edition*

Similar to the creation, the edition UI has the same type in structure. The only different part is the locked combo box of process type. The type of process cannot be changed in edition. For the input process, there is just container information and for output process, there is only slot information.

4.2.3 Data Synchronization

As explained in chapter 3, the activity of the automated system is controlled by the PCS or ICS in the local controller. The activity is created and modified via the local control system. On the other hand, the activity information is also stored in the database of HWMS. Once there is a change on the local control system, the changing should be reacted in HWMS.

For the data synchronization, there are two kinds of solutions: periodic and event-driven. The advantage of periodic solution is to realize the procedure of data synchronization automatically. The frequency for synchronization should not be low. On the other hand, most of the activities will not be changed after the creation and even addition of new activities is not happened every day. The periodic synchronization is no necessary and wastes the system resource.

Furthermore, the hierarchical data structure is suitable for the local update with event-driven. In other words, it is not necessary to update the whole data structure when there is only local stricture will be used. There are three levels in the system to synchronize data: device, project and families. Here is an example to explain how to complete data synchronization of the device. Once the device has been selected, the Update button can be clicked to synchronize the project information. When the PCS is SAMI, the synchronization procedure is shown in Figure 37.

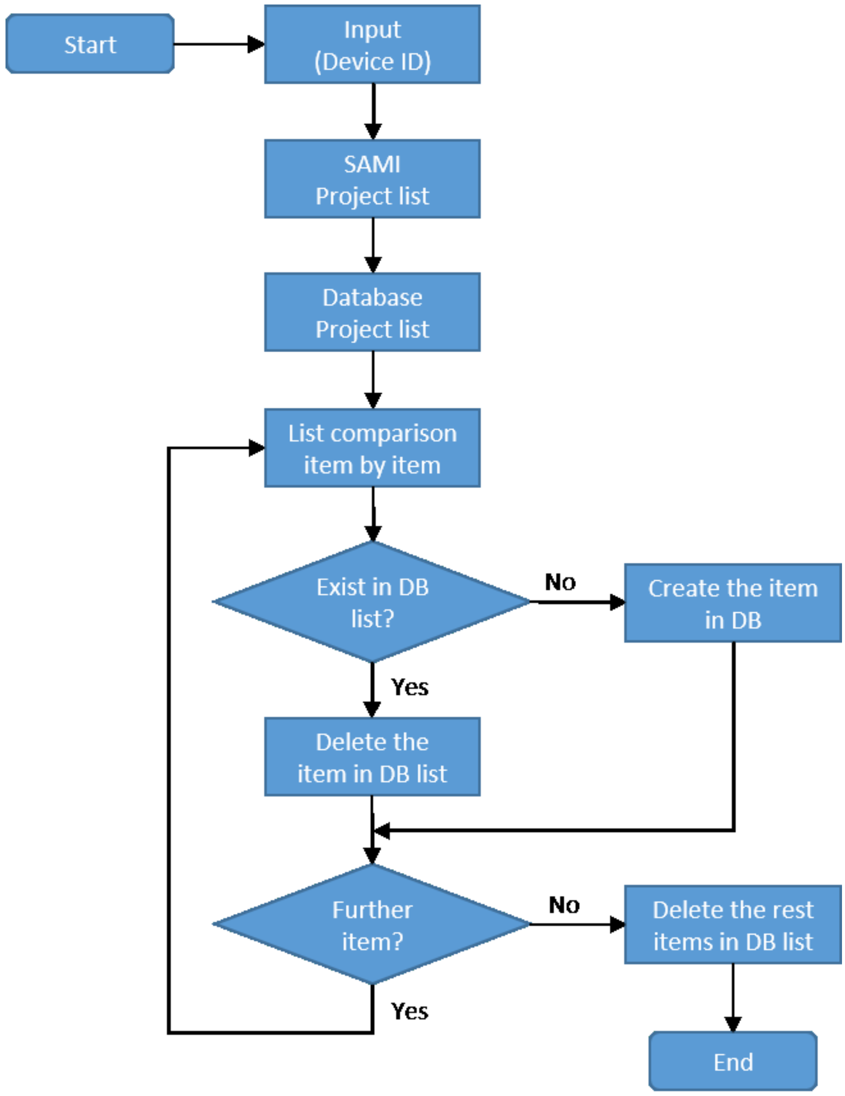


Figure 37 *Flowchart for data synchronization*

When the device has been selected, the device ID can be used to get the SAMI project list (Slist) form the PCS and the database project list (Dlist) form the WPS database. Then, each item of Slist will be checked, whether it exists in the Dlist. If yes, this item will be deleted in the Dlist. If no, this item will be added into the Dlist. When all the items have been checked, the remaining items in the Dlist are the redundant items, which are no longer useable. These items should be deleted from the database.

Because of the hierarchical data structure, removing anything in the database requires extra care. Some data is not only related to the LMMS, but also to the WPS. When an item will be deleted, there is a special procedure for the deletion. Here is an example to show the procedure, when a method needs to be deleted as shown in Figure 38.

When a method needs to be deleted, the items related to this method should be deleted earlier in LMMS and WPS. For LMMS, the "DeviceMethodSchedule" (MS) list is found. Each MS item is used to search for the "*DeviceMethodScheduldeDetails*" (MSD) list. Then the list of MSD and this MS item will be deleted in order. All items in the MS list should repeat the search-delete procedure until the list is empty. At the same time, the process list, which relates this method is created. For each process item, the transportation list and workflow list are selected and deleted with this process item. When all process items are deleted, the data of WPS is ready for the deletion of this method. After the both procedure of LMMS and WPS, the method can be removed safely.

This example presents the deletion of the method. For other parts such as device, MS, etc. each of them has a special procedure for its deletion. As the principle, the lower level data should be operated before the higher level and both LMMS and WPS should be searched and implemented.

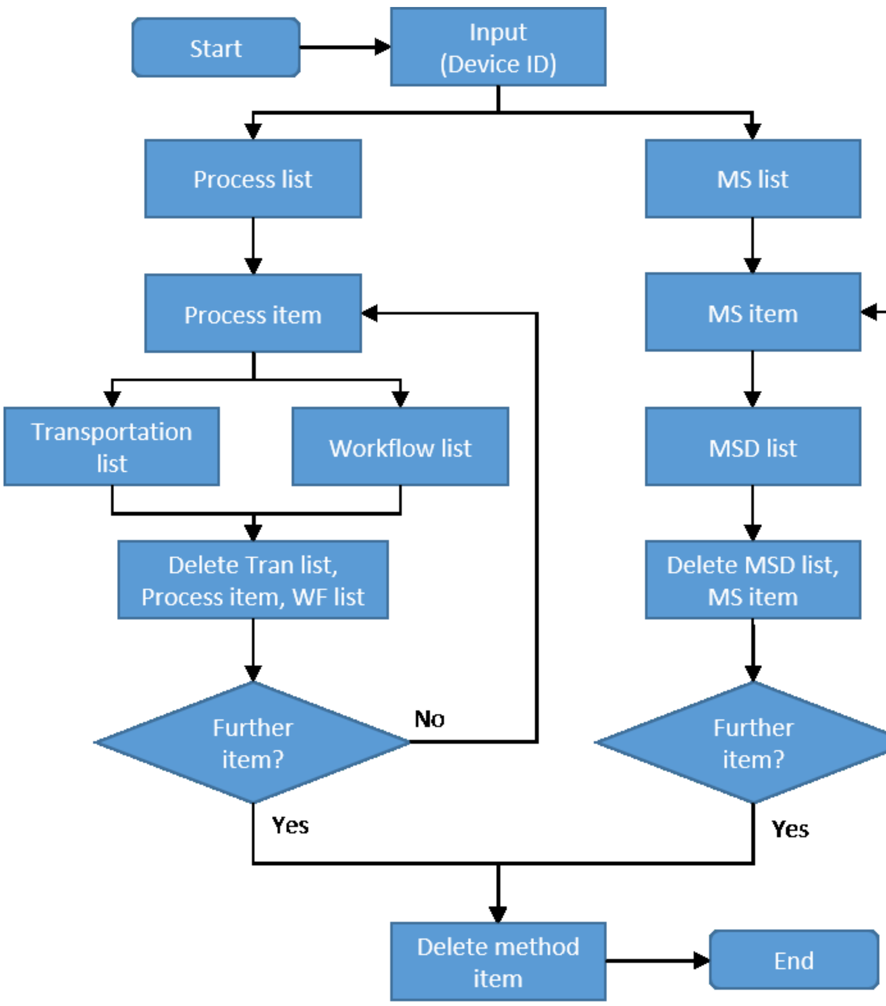


Figure 38 *Flowchart for Method delete*

4.2.4 Workflow Management

The process and transportation information are saved in the form of workflow in HWMS. After the designing of the workflow, it will be saved in the UI as shown below.

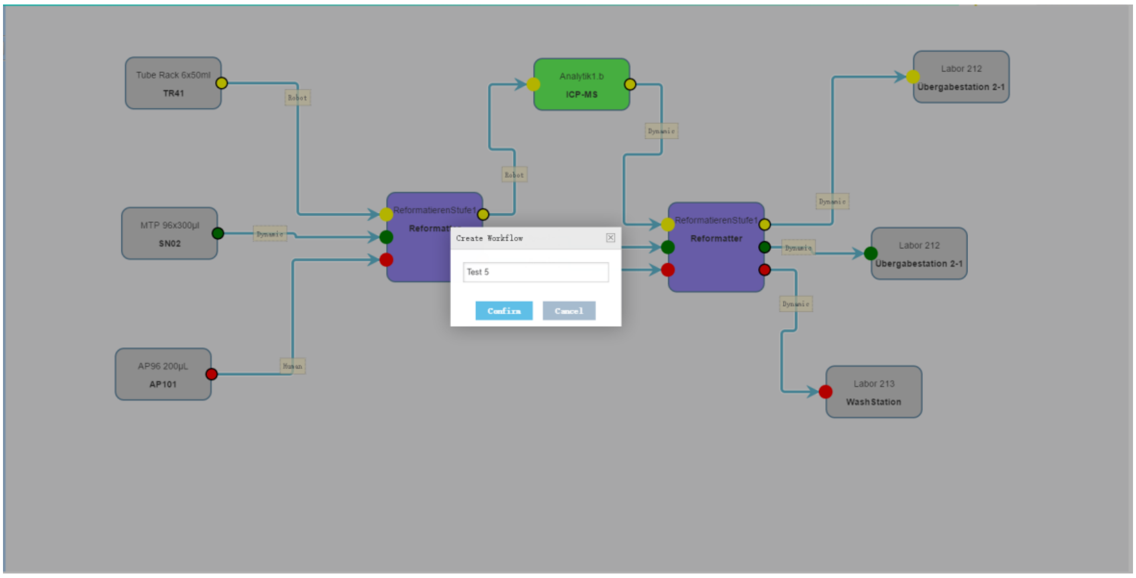


Figure 39 *UI for saving Workflow*

The workflow name can be typed in here and later this name is used to search for this workflow not only for modification but also for execution.

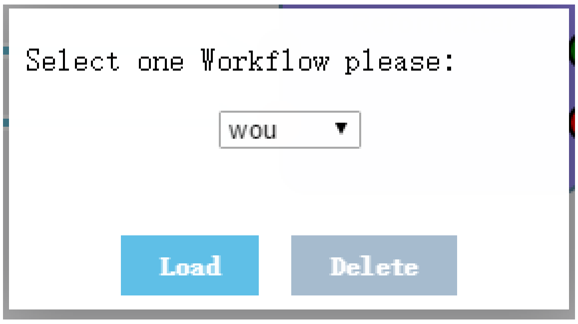


Figure 40 *UI for loading a Workflow*

The dialog as shown in Figure 40 helps user to select one workflow which can be deleted or loaded to the work space. Then, this workflow can be checked and edited.

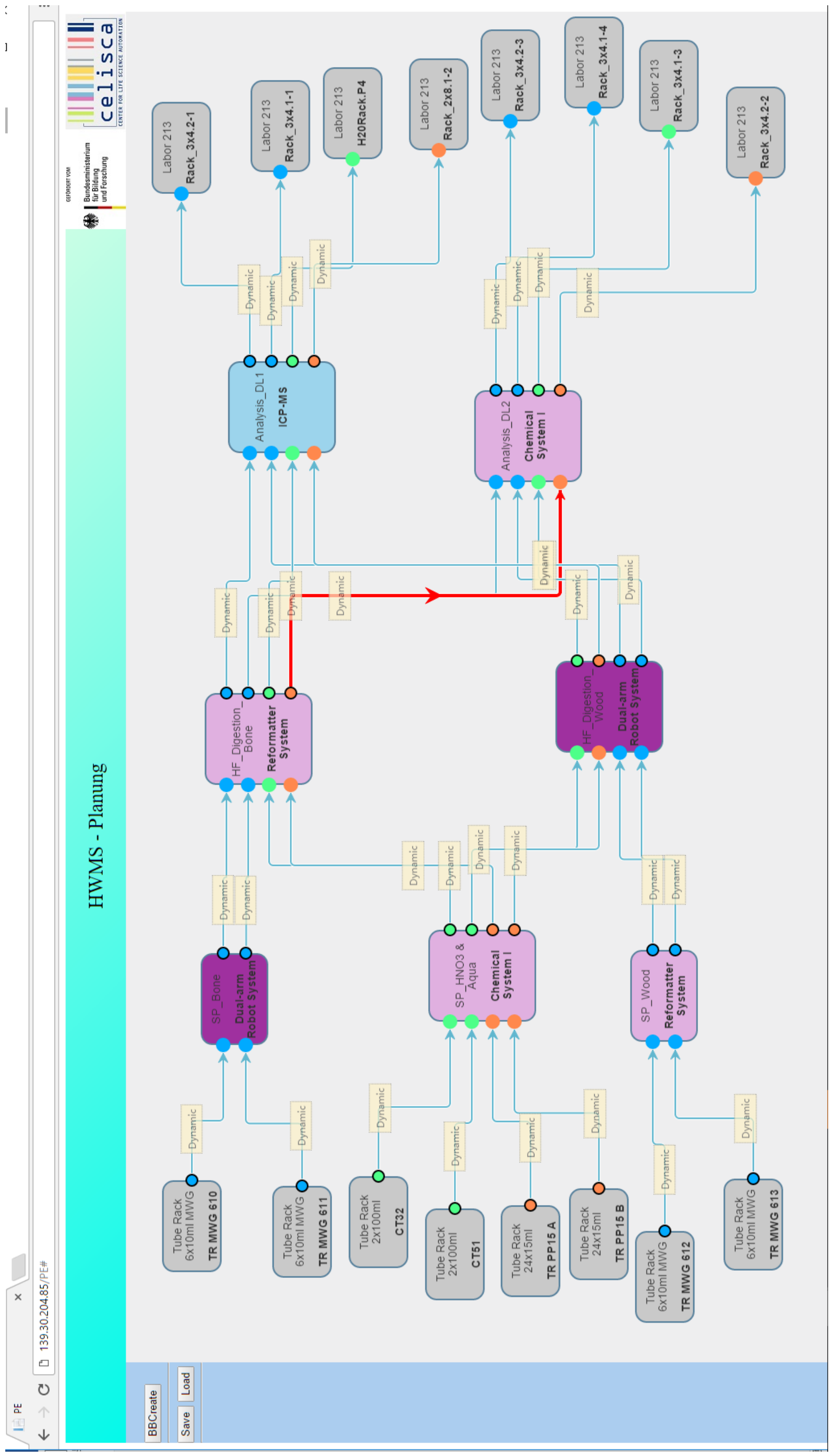


Figure 41 *An example of a workflow*

In the workflow as shown in Figure 41, each box represents a process. The process name and device name are in the box. Various box colors present different type of the device. The green box is the analytical device for example. The solid circles in the left of the box are the input of the process and on the right side with black border are the output. Each color of the input and output represents various container templates. When the input and output are the same color, the connection is valid. If not, the planning editor will ignore it. The connection between two processes is one transportation. The transportation has not only the information of source and target, but also the transportation type. There are three types of transportation: human assistant, mobile robot and dynamic. Human assistant can operate all the transportation tasks but the human resource in laboratories is quite limited because they have other manual operation tasks at the same time. Mobile robots are the new transporter in laboratories. It can work in 7/24 without any low level mistakes such as put the labware on a wrong slot. On the other hand, the special environment of the laboratory and the walking mechanisms of the mobile robot determine that, not all the slots can be reached by mobile robots. However, there are some transportation task can be operated by both human and robot. When the user does not want to choose one for the transportation, it is also possible to select the type by HWMS. At that time, the transportation type is dynamic as shown in the yellow label on the transportation line.

4.3 Workflow Execution System (WES)

This section explains the data structure of WES. In WES, there are two tables – Task and TaskStatus. The 'Task table' integrates the processes and transportations, which are the parts of the Workflow. Based on the result of the scheduler, these tasks will be executed one by one.

4.3.1 Data Structure

The data structure of WES is shown as Figure 42. The 'Task table' connects directly to the 'Process table' and the 'Transportation table'. The processes and transportations are transformed to tasks including P task and T task before execution. With the help of the initial task list, the static scheduler will generate a time schedule for the execution. This time schedule is saved in the task table. When the item of tasks is operated, the data is linked to the process or transportation table to get more detailed information.

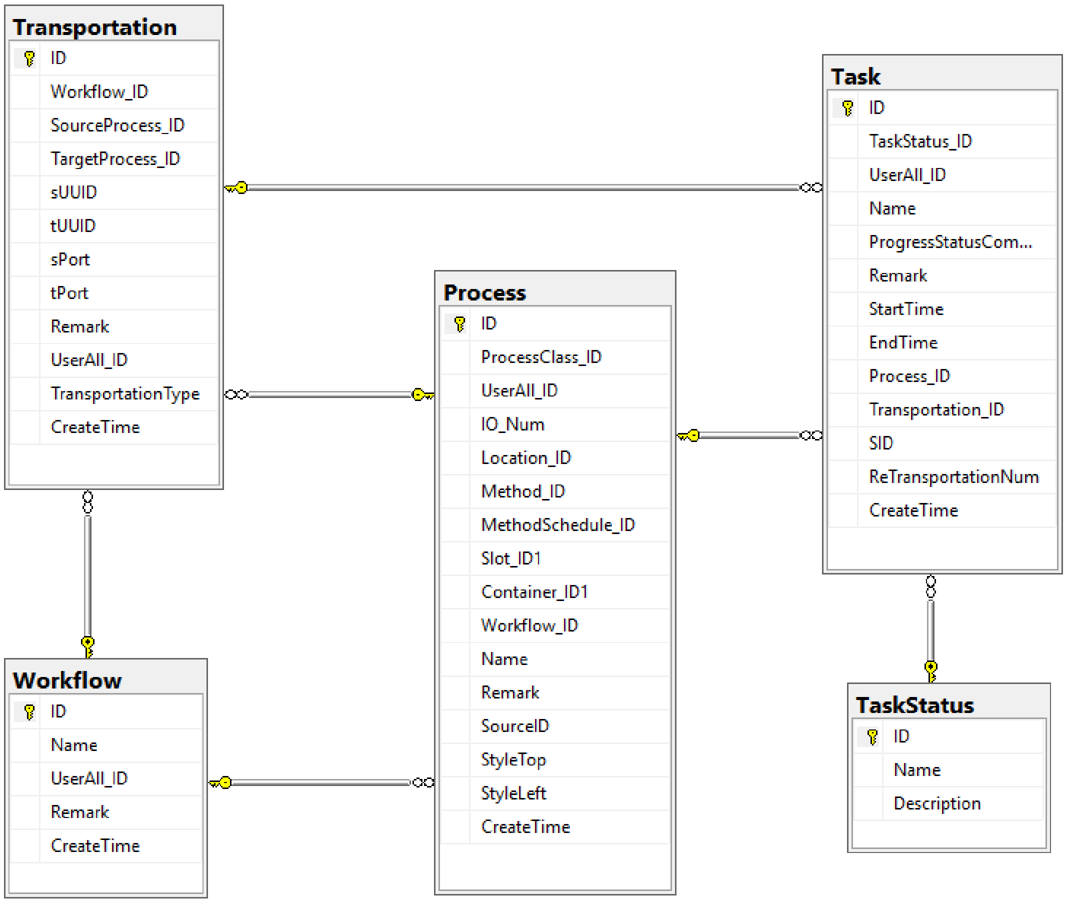


Figure 42 *Data structure of WES*

The TaskStatus is used to record the status of the Task's. There are four statuses for task – Running, Finish, Block and Wait. The status helps the user to track the procedure of the execution, and it is the event to drive the procedure of the execution as well.

4.3.2 Execution Management

The execution management is used to handle workflows and monitor the procedure of the execution. With the help of the UI as shown in Figure 43, the workflows can be loaded into the WES.

There are four statuses of the task in WES: ready, running, complete and failed. The table without background color is the status ready. That means this task is waiting for execution. During the execution, the background color of the task is yellow. When the task is completed successfully, the task is green, otherwise, is red.

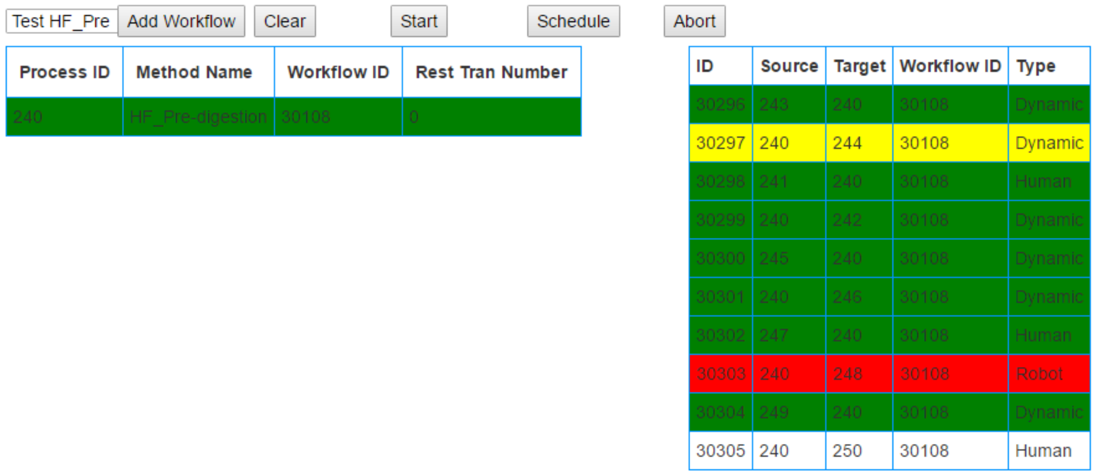


Figure 43 *Status monitor for WES*

Figure 44(a) shows the task list for the selected workflow. Compare with planning editor, in WES user can loads more than one workflows. In other words, the WES can handle more than one workflows at the same time. After the loading of workflows, the user needs to click the "Schedule" button to create a time schedule for the execution. Details of scheduling will be explained in next chapter. After the scheduling, the execution can be stated as shown in Figure 44(b). When all transportation tasks, which relate to the process have been completed, the process is able to be started (Figure 44(c)). After all tasks are completed, the workflow is done as shown in Figure 44(e).

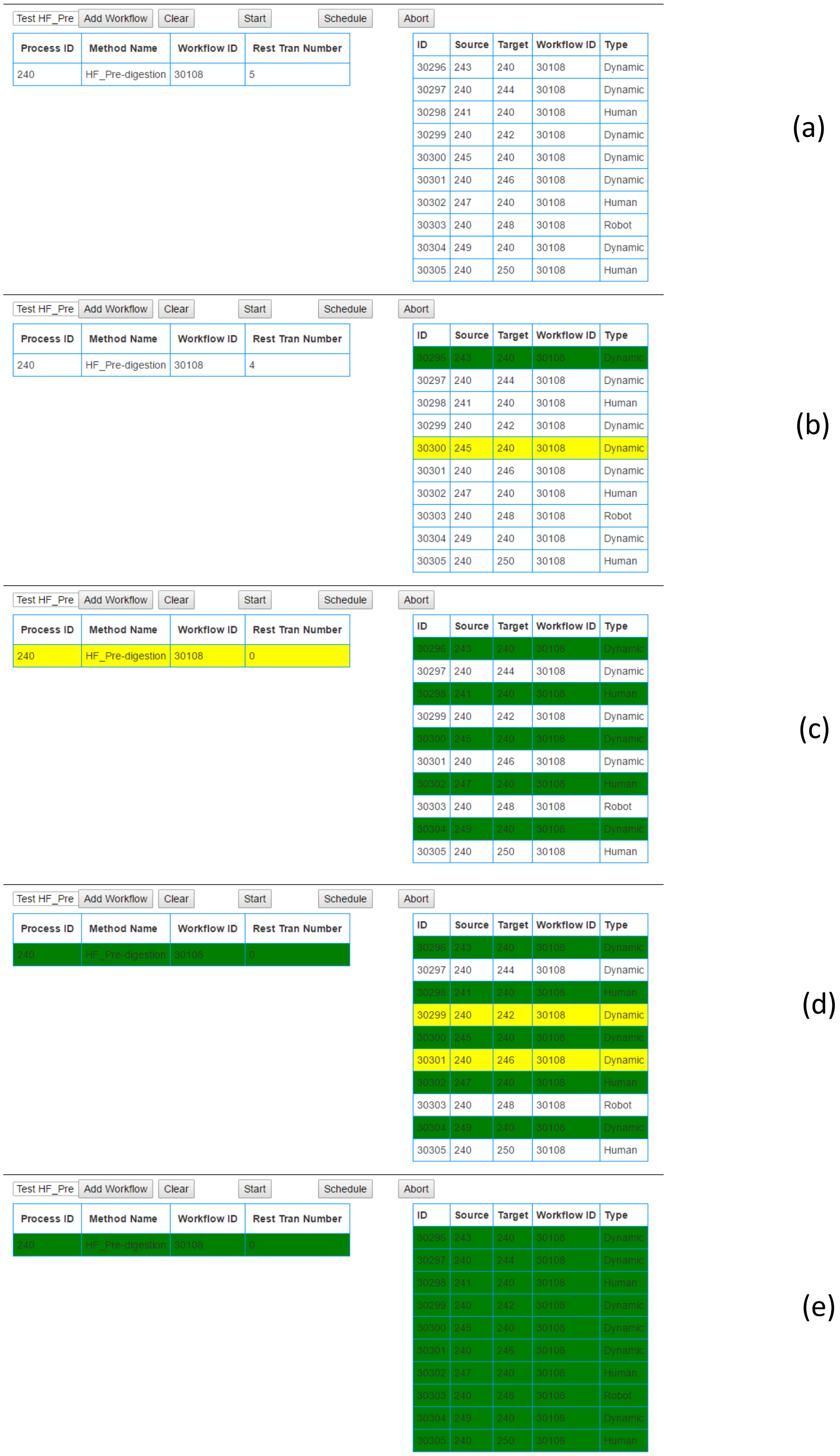


Figure 44 *UI for workflow execution system*

4.3.3 Execution Strategy

Before execution, the time schedule is generated to guide the operation procedure of the tasks. In order to guarantee the stability and safety of the laboratory, an event-driven strategy is adopted. The laboratory environment is dynamic and complex and it is distinguished from single system, which has several functions with constant operation duration. Some automated systems, which integrate various devices cannot guarantee the constant operation durations. Although the difference for the same activity is less than three minutes, it is not suitable to drive the procedure with time. On the other hand, the transportation time is more unstable because of the unexpected situation.

The workflow has a "transportation-process-transportation" structure. The event, which is used to drive the execution is the status change of the transportation and process. The execution strategy is shown in Figure 45.

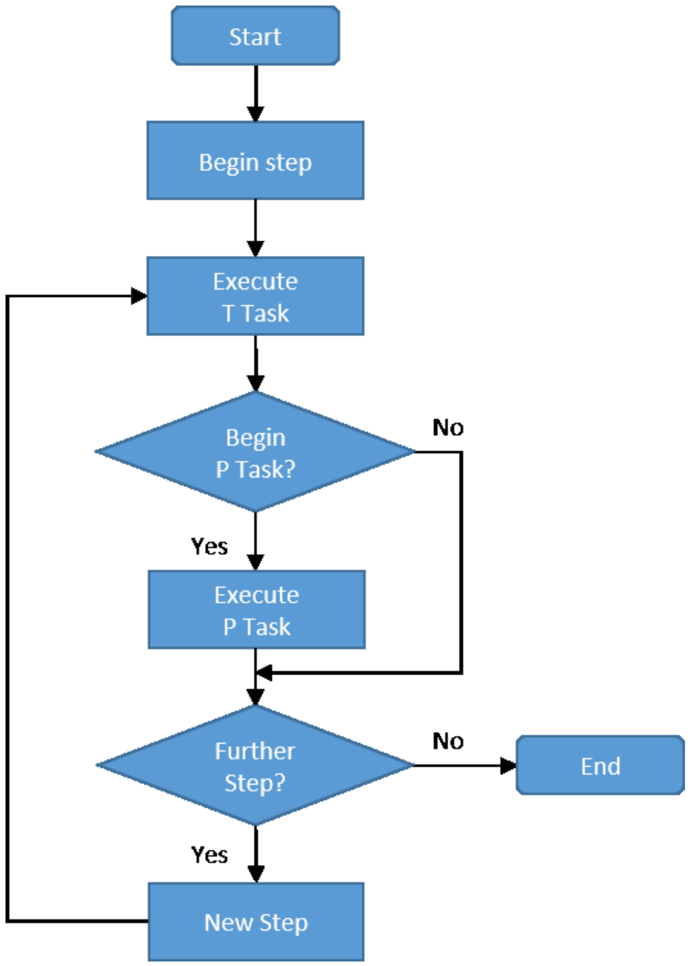


Figure 45 *Flowchart for execution strategy*

The steps represent the sequence of the transportation tasks. This sequence comes from the scheduler (see chapter 5) and is used to guide the execution procedure. The strategy details are explained as following:

- Based on the result of the scheduler, the execution procedure starts with begin step and the transportation task, which is defined as the begin step will be executed. Even the labware is already in the start slot of the process, there is a zero-duration transportation to transmit the container information for the execution.
- When the first transportation task is completed, there are two checks: whether there is further step and whether the target process can be started. From here on, there are two independent operation loops. One is for transportation tasks and the other controls the process execution.
- If there is further step for the transportation task, this step will be selected and operated later. If there is no further step, there are only two possibilities: the workflow is completed or the next step should wait for the process, which is still running.
- For the process loop, the system checks whether the process can be start or not based on the completeness of transportation tasks. After the execution, the system will trigger the step, which is waiting for this process and operate the transportation task. Then, these two loops will be activated again.
- The finish check is just searching, whether there are running transportation or process tasks. It is used to determine the completion of the execution.

The execution strategy is absolutely based on the sequence of the transportation, because the execution procedure is driven by events. However, that not means the time schedule is useless. It is the reference for the later dynamic scheduling to face the unexpected situation.

4.4 Error Handling

There are two major sources of the errors existing in HWMS: Multiple Robot Management System (MRMS) and process control adapter system (PCAS). MRMS reports errors regarding the mobile robot transportation tasks and PCAS collects errors from various automated systems. When unexpected situations are arising during the workflow execution, the error informations are collected and handled by HWMS in the process control layer. The errors trigger the procedure of rescheduling as explained in Chapter 6. Different from the delay, all the errors need the human assistance. Therefore, the error message and assistance requirement will be sent immediately to HACS.

The error message from MRMS is generated following the flowchart as shown in Figure 46. This message is used to guide the assistant to find the mobile robot and solve the problem based on the location information. Then the assistant needs to complete the transportation task instead of robot. When the task is finished, the assistant confirms the requirement and HWMS can continue the execution of the workflow. Some errors require more assistances. For example, when the robot cannot find the labware, it is possible that the Kinect sensor is not online. It makes sense to remind the human assistance to check the battery and connection status of the Kinect sensor. When the error message is generated, it will be sent to HACS for human assistance.

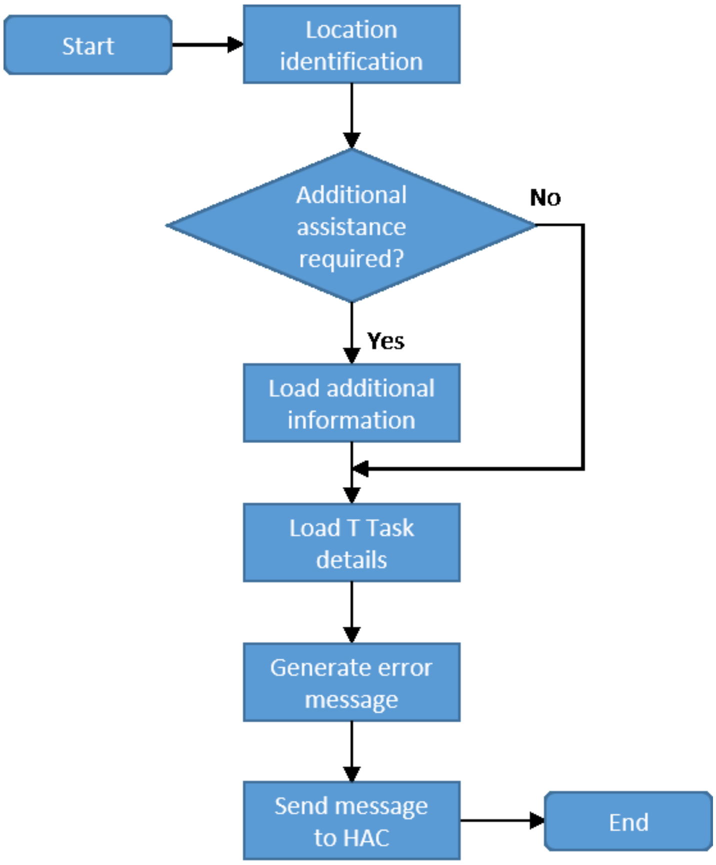


Figure 46 *Flowchart for MRMS error message generation*

The error from PCAS can be directly sent to HACS. The dynamic scheduler needs to consider the suspended time of this automated system. When the error is solved, the process task is resumed and the workflow continues to be executed.

Chapter 5 Intelligent Scheduler

Based on the discussion in chapter two and three, an intelligent scheduler is required for the resource allocation and the task distribution. In this chapter, the schedule model will be defined to transform the laboratory environment information to a specific model. Then, the scheduling strategy will be presented, which includes the basic scheduling structure and constraints. Based on the framework of the scheduler, the genetic algorithm and two modified genetic algorithms are implemented for the optimal solution searching. In the end, an experiment will be used to assess and compare the performance features of each algorithm and select the suitable algorithms.

5.1 Schedule Model Definition

The scheduling model is the bridge to connect the laboratory and the scheduler. On the one hand, the model offers parameters and constrains of the planned workflow for scheduling. On the other hand, the result of the scheduler can be translated to the form, which guides the operations of automated systems and the transportation tasks between them. Therefore, the scheduling model is a particular but simplified description of the laboratory. The particular details are the necessary information for executions and constrains. However, not all the details are helpful for scheduling and the simplification raises efficiency of the scheduler.

There are two key foundation elements in the laboratory to be scheduled: resource and activity. They are defined respectively in the following part.

5.1.1 Resource Definition

Resource in the schedule model includes various labware and workstations. As the object of activity, labware contains samples and materials for execution. The set for labware, LW is described by the following definition.

$$LW = \{LW_i \mid 1 \leq i \leq I\}$$

$$LW_i = \{ID_{LWi}, Barcode, LWtype, LWslot, LWStatus\}$$

where,

- I is the quantity of labware and LW_i is a single labware.
- ID_{LWi} and $Barcode$ are used for the identification of the labware, which help users and the system to track samples.

- $LWtype$ is a specific parameter, which includes several properties of the labware such as length, width, height, capacity, footprint etc.
- $LWslot$ is the slot, where this labware located.
- $LWStatus$ records the current status of the labware, whether it is empty or full.

As the actor in the execution, the workstation set, WS is defined as below,

$$WS = \{WS_j \mid 1 \le j \le J\}$$

$$WS_j = \{ID_{WSj}, Location, P_{j1}, P_{j2}, \dots, P_{jk}\}$$

where,

- J is the quantity of the automated workstations and WS_j is a specific workstation.
- ID_{WSj} is the identification of the automated workstations.
- $Location$ is used to guide the transportation system to locate the workstation.
- $P_{j1}, P_{j2}, \dots, P_{jk}$ are representations of the processes that will be executed on this workstation.

5.1.2 Activity Definition

The second key foundation element for scheduling is activity. There are two kinds of activity in automated laboratory: process and transportation. Process is the specific activity of automated workstation to implement one or several steps of the experiment. For the process set P,

$$P = \{P_k \mid 1 \le k \le K\}$$

$$P_k = \{ID_{Pk}, ID_{WSj}, Duration_{Pk}, sT_{Pk}, eT_{Pk}\}$$

where,

- K is the quantity of processes in the workflow and P_k is a process.
- ID_{Pk} is the identification of a process.
- ID_{WSj} is the identification of the workstation, which is going to implement this process.
- $Duration_k$ is the duration time of this process.

- sT_{Pk} and eT_{Pk} record the start time and end time of this process. They are generated by the scheduler.

Between two processes, there are at least one transportation to connect them. Transportation combines two stand-alone processes to enable a combined operation. Transportation also includes the dependent relationship between the source and target processes. This relationship will be discussed later.

For the transportation set T,

$$T = \{T_m \mid 1 \leq m \leq M\}$$

$$T_m = \{ID_{Tm}, ID_{PSou}, ID_{PTar}, TranType, Duration_{Tm}, sT_{Tm}, eT_{Tm}\}$$

where,

- M is the quantity of transportations in the workflow and T_m is a single transportation task.
- ID_{Tm} is the identification of a transportation task.
- ID_{PSou} and ID_{PTar} are the identification parameters of the source and target process.
- $TranType$ and $Duration_{Tm}$ present the type of transportation task and based on this type, which time is required for this task.
- sT_{Tm} and eT_{Tm} record the start time and end time of this transportation task. They are generated by the scheduler.

5.1.3 Workflow Definition

It is necessary to combine resources and activities together as a complete model package for the scheduling. This package is defined as Workflow. The Workflow WF

$$WF = \{LW, WS, P, T\}$$

is defined as the workflow for the scheduling and execution.

One workflow includes the resources and activities, which are required by the investigation. It not only offers detailed information of the laboratory environments and planned experiments to the scheduler, but also accepts and records the results of the scheduling.

5.2 Scheduling Strategy

Based on the scheduling model, a scheduling strategy is used to guarantee that the resource is allocated reasonably and the processes are executed safely and stable with high efficiency and low cost.

5.2.1 Scheduling Structure

According the presentation of the scheduler in the concept part, the scheduler should be used in two different scenarios: static and dynamic scheduling. The former generates the initial schedule before the execution of the workflow and the latter handles the unexpected situations and resource changing during the execution. Both static and dynamic scheduling are utilizing the same data structure to accept workflow details and generate the time schedule.

Following this strategy, the structure of the hybrid scheduler in this study is shown in Figure 47.

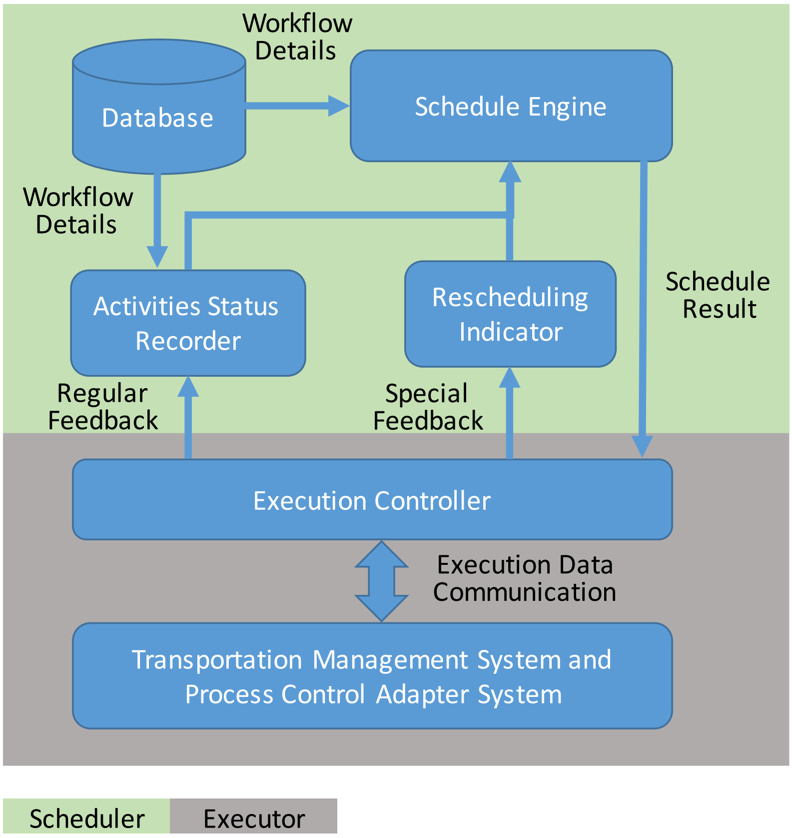


Figure 47 *Hybrid scheduling structure*

As the common component for static and dynamic scheduling, the *Schedule Engine* has two operation modes. For static scheduling mode, the *Schedule Engine* receives the workflow details, which include the detailed information and parameters of the activities that will be executed in the laboratory. At the same time, this information is sent to the *Activity Status Recorder* to build the storage structure. Based on the scheduling model, the *Schedule Engine* calculates the time schedule for the workflow and sends it to the *Execution Controller* to guide the execution by the *Transportation Management System* and the *Process Control Adapter System*. The *Activity Status Recorder* requires the status of the execution as the regular feedback every second. When there is an unexpected event happening such as process task failure or transportation task delay, the *Execution Controller* will send a special feedback to the *Rescheduling Indicator*, which decides whether a rescheduling is required. If no, this unexpected event will be ignored. If yes, the *Schedule Engine* can calculate again with dynamic scheduling mode based on the workflow details and the current status of the execution. The result of the *Schedule Engine* is used to update the time schedule of the workflow for execution to finish the dynamic scheduling procedure.

5.2.2 Scheduling Principle and Constraints

In essence, the inputs of the scheduler are preprocessed attributes and constraints of activities, which are encapsulated into a schedule model. The goal of the scheduler is to create a timetable to handle activities of the workflow. This timetable is also presenting the sequence of activities. In other words, the time schedule problem can be transformed to the activity-sequence problem. The activity sequence determines both the schedule timetable and the resource allocation. In this thesis, there are two sub-sequences in the activity sequence: a process- and a transportation sequence. One process sequence can exist in more than one activity sequence. However, one transportation sequences with specific transportation type set correspond to a unique activity sequence based on the constraints, which are explained in the following part. Therefore, the scheduling problem is transformed to a finite transportation sequence searching problem.

The role of constraints is particularly important for scheduling. Thus, two types of constraints are explained to improve the efficiency and accuracy of the scheduling approach.

Process constraints:

- One process is not allowed to begin until the processes and transportation tasks, which have higher priority are completed successfully.
- One automation system can handle only one process of the same or a different workflow simultaneously. The process is allowed to be started unless the previous processes on the same automated workstation/integration system are completed.
- The process should begin immediately if it satisfies the first condition.

Transportation constraints: transportation is not allowed to begin until the process, which is the source of the transportation, is completed.

These constraints are not only constraining the scheduling result, but also guiding the execution procedures. They are the fundamental that provide the security guarantee for automated workflow executions.

5.2.3 Scheduling Requirements Analysis

Because of the different scenario, the requirements of static and dynamic scheduling are quite different as shown in Table 10. Following the progress of the workflow execution, the complexity of the workflow decreases due to the decreasing number of remaining unfinished activities. Therefore, the Complexity for static scheduling is higher than dynamic. Dynamic scheduling requires fast reaction speed and short computation time to minimize system latency and avoid further error. However, the optimal solution rate for static scheduling should be high to increase the performance of the scheduler. For dynamic scheduling, the optimal solution rate does not take the highest priority. Therefore, static scheduling requires high optimization solution but dynamic scheduling focusses on the computation time. This is the most important standard and gist for schedule algorithm selection.

Table 10 *Difference between static and dynamic scheduling*

Difference	Static Scheduling	Dynamic Scheduling
Complexity	High	Depend on current state
Reaction speed	Non-limited	Fast
Computation time	Non-limited	As short as possible
Optimal solution rate	High	Acceptable

It is possible to use just one algorithm with different parameter sets or two algorithms to meet the requirements. If there are two algorithms, the performance and efficiency should be considered because of the expense for modeling and data transform. One of the best solution is to use two algorithms based on the same data structure to save both coding and computational efforts.

5.3 Scheduling Algorithm – Genetic Algorithm

Comparing with HNN and SA, GA is the best choice for scheduling in the laboratory because of the balance of performance and stability. HNN has higher performance than GA, but there is risk to get a non-feasible solution, which can bring danger to the laboratory. SA is more stable than GA to search for optimal solution, but it requires much higher computation effort.

Moreover, GA has high flexibility and compatibility. There are several steps: encoding, population generation, evaluation, crossover, mutation etc. Each of them has more than one method to realize the function and it is open to integrate other algorithms to bring some new features to meet the requirements of the scheduling.

The pseudocode of GA is shown below, and the major detailed operation will be explained step by step in the following sections and the flowchart for GA is shown in Figure 49.

```
// Genetic Algorithm
Pn ← population_scale                       // Initialize population scale
Pop ← initial_population(Pn)                // Initialize population
Cr ← crossover_rate                         // Initialize crossover rate
Mr ← mutation_rate                          // Initialize mutation rate
i ← 0                                       // Initialize generation count
MX ← maximum_generations                    // Initialize maximum generations
// Compute a new population and objective function value
O ← fitness(Pop)                            // Evaluate initial population
BestIndividual ← cal(Pop, O)                // Search the best individual in the population

while i < MX:
    SelPop ← select(Pop)                    // Selection operation
    SelPop ← across(SelPop, Cr)             // Crossover operation
    SelPop ← aberrance(SelPop, Mr)          // Mutation operation
    SelPop ← cab(SelPop)                    // Adjustment operation
    Pop ← replace(SelPop)                   // Replace the selected individuals
    O ← fitness(Pop)                        // Evaluate the new population
    BI_i ← cal(Pop, O)                      // Search the best solution in the new population
    if BI_i < BestIndivivual
        BestIndivivual = BI_i               // Best individual storage
    end
    i ← i + 1                               // Increment trial count
end
```

Figure 48 *Pseudocode of Genetic Algorithm*

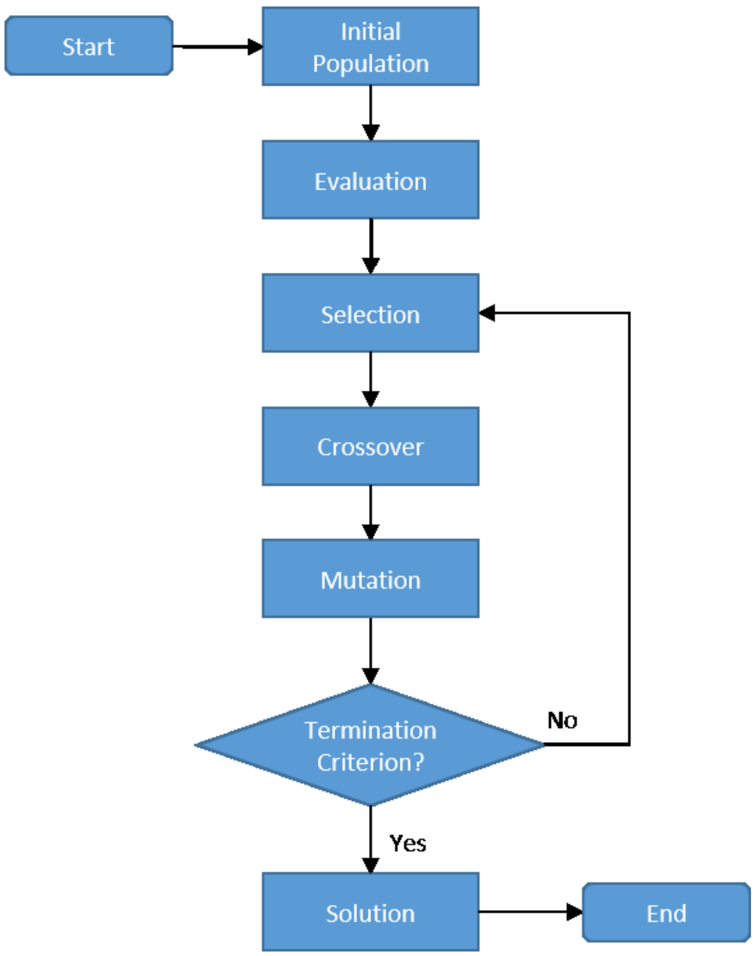


Figure 49 *GA flowchart*

The initial population is consisted of a group of individuals, which are generated randomly and based on the solution domain. According the result of evaluation, several individuals are selected to generate new individuals. Crossover and mutation are the operations for the new chromosomes' generation. Then the termination criterion will be checked. If it is satisfied, the operation can be stopped and the solution is found. Otherwise, the operation will go back to evaluation and repeat again.

5.3.1 Chromosome Encoding

As explained in the scheduling strategy, the solution of the scheduler should include two parts: transportation sequence and transportation type. Multiple-layer-coding is used to create the chromosome. There are two layers in the chromosome: sequence layer and type layer.

As the sequence, the chromosome members should be consecutive integers starting with one ending with the count of transportation tasks. There are two types of transportations: human assistant and mobile robot. Therefore, the type layer can be consisted of binary numbers and the quantity of the type layer is the same as sequence layer.

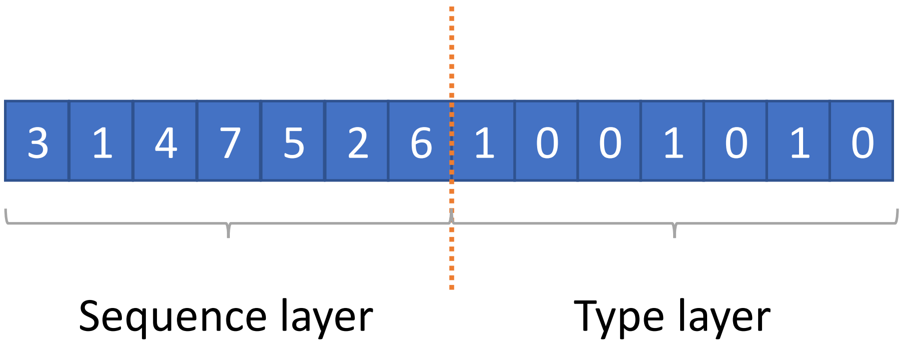


Figure 50 *An example chromosome with 7 transportation tasks*

As shown in Figure 50, there are seven transportation tasks for this chromosome. There are fourteen numbers: seven integers in the left for sequence layer and seven binary numbers in the right for type layer. In type layer, zero represents human assistant and one means mobile robot.

5.3.2 Chromosome Adjustment for Solution Domain

As the searching space of the algorithms, the solution domain is limited by the scheduling constraints, which are defined in the scheduling strategy. Based on the scheduling constraints, adjustment is built to adjust the random chromosome. In order to keep the raw information, the adjustment changes the chromosome as little as possible. The chromosome diversity is also an important aspect for the modification. These problems are considered during the designing of the adjustment.

Moreover, the modification is not only in the sequence layer, but also in the type layer. When the user confirms the transportation type for the task, the corresponding type member is fixed. On the other hand, not all the slots are suitable for the mobile robot. The transportation type must be fixed to the human assistant, when the transportation task involves one or two slots, which cannot be handled by mobile robots.

This adjustment is not only used to modify the initial population, but also for the change of population such as crossover and mutation. That means this adjustment will be operated frequently. Therefore, the performance and efficiency of the adjustment should be paid abundant attention. The flowchart for adjustment of sequence layer is shown in Figure 51.

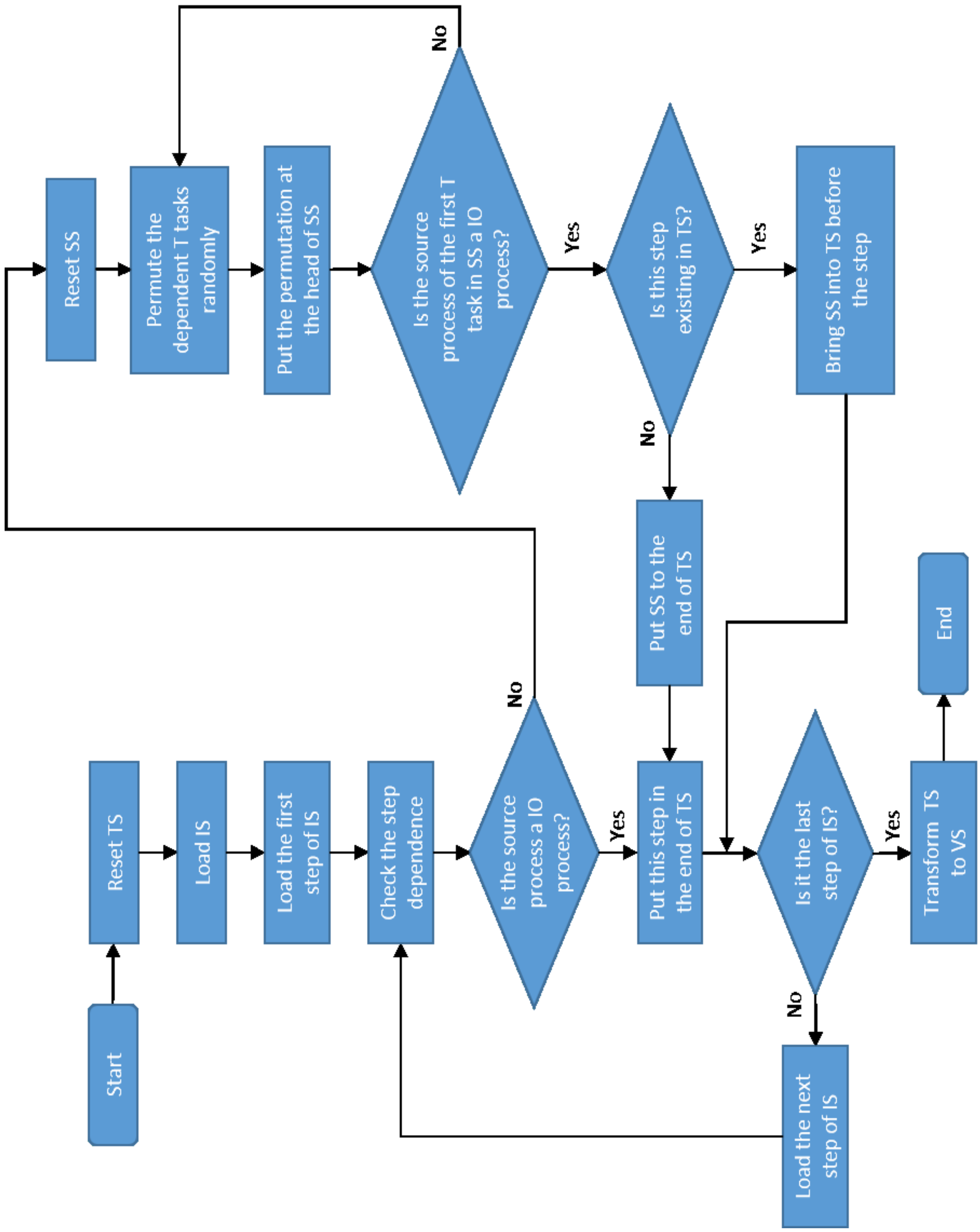


Figure 51 *Flowchart for adjustment*

There are four sequences in the adjustment procedure: initial sequence (IS), valid sequence (VS), sub sequence (SS) and temporal sequence (TS). The initial sequence is the random chromosome, which is the input of adjustment and the valid sequence is the result of the adjustment. The sub sequence is used to operate the dependence sequence of one specific step. When there is new step loaded, the sub sequence needs to be reset. The temporal sequence is used to record the temporal data for the procedure of the adjustment.

Table 11 *An example dependence table*

T Task ID	Source P Task ID	Dependent T Task ID
0	0	-
1	0	-
2	0	-
3	1	0, 1, 2
4	1	0, 1, 2
5	1	0, 1, 2
6	2	3
7	3	4

In order to differentiate the transportation tasks in different sequences, the tasks in the initial sequence will be called as steps. Based on the workflow details, the dependence of the transportation tasks are confirmable and saved in the dependence table as shown in Table 11. This table presents the dependence of the transportation and process tasks for the workflow. The source process is unique for each transportation task. If the source process is an IO process, there is no dependent transportation task. However, it is possible that there are more than one dependent transportation tasks for the source process. In order to keep the variety of the chromosomes, the dependent transportation tasks will be permuted randomly and put them at the head of sub sequence. Then the first task of the sub sequence needs to be check, whether its source process is an IO process. The permutation and check operation need to be repeated until the first task has no dependent transportation task. Then the sub sequence can be put into the temporal sequence. If the step is already in the temporal sequence, the sub sequence is put just before this step. When this step appears more than one time in the temporal sequence, the subsequence should locate before the first one. If this step is not the last one, the next step can be loaded and repeat the procedures above. After the last step completes the operation, the temporal sequence has to transform to a valid sequence. Because of the additional sub sequences, some of tasks appear more than one time in the temporal sequence. The transform operation is to keep the task, which appears as the first time, and to delete the other same tasks. After the transforming, the adjustment for sequence layer is completed.

The additional sub sequence strategy is the kernel of the sequence adjustment. The most important point is to ensure the validity of the result. Moreover, the random permutation brings the variety of the chromosomes. Although these operations change the original sequence, this strategy keeps maximal feature information of the initial chromosome.

Moreover, the type layer of the chromosome also need to be corrected. In a workflow, the type of some transportation tasks are defined by user during the workflow designing. Because of the limitation of the slots and locations, some transportation tasks cannot be completed by mobile robot. These two situations fix the transportation type. The adjustment in type layer is to ensure the type limitation for each transportation task.

5.3.3 Fitness Function

The evaluation function, which is used to assess a specific solution, estimates the fitness of the candidates for the selection. In this approach, the evaluation function reflects a comprehensive optimization goal: shortest execution duration and best balance of human and robot for transportation. The human assistance and mobile robot resources are limited and roving. Human resources are required by both, manual operation steps and transportation tasks in laboratories. The high work stress will decrease the efficiency. On the other hand, when there are additional human resource or more available robots, they should be utilized reasonable in time. Therefore, the balance score is also essential to evaluate the solution.

$$Fitness = TF + T_{Bal} \tag{1}$$

Equation (1) defines the fitness function. TF is defined as the duration time of the workflow and T_{Bal} is the balance score of the available human and robot resources. Therefore, the fitness is the score, which combines both execution time and the balance score of a workflow. Equations (2)-(6) describe the calculation of T.

$$TF = max\left\{ \begin{matrix} max\,\{eT_{Pk}\} \mid 1 \leq k \leq K \\ max\,\{eT_{Tm}\} \mid 1 \leq m \leq M \end{matrix} \right\} \tag{2}$$

$$eT_{Pk} = sT_{Pk} + Duration_{Pk} \tag{3}$$

$$eT_{Tm} = sT_{Tm} + Duration_{Tm} \tag{4}$$

$$sT_{Pk} = max\{eT_{Pi} + Duration_{Ti}\} \mid i \in SP_k \tag{5}$$

$$sT_{Tm} = max\begin{Bmatrix} max\{eT_{Ti} + Duration_{Ti}{}'\} \mid i \in ET_m \\ eT_{Pj} \mid j \in EP_m \end{Bmatrix} \tag{6}$$

As the definition before, eT_{Pk} and eT_{Tm} are the finish time of process P_k and transportation T_m. They depend on their start time sT_{Pk} , eT_{Tm} and the duration of the activity $Duration_{Pk}$, $Duration_{Tm}$. The last finish time is just the complete time of the workflow. SP_k is the collection of processes, which have higher priority and are executed in another workstation. The process cannot be executed until all of its dependent activities are completed. The transportation is not allowed to start until a process, which is the source of the transportation is completed. $Duration_{Ti}{}'$ is defined as the extra transportation time posed by the difference between the last target slot and current source slot.

$$T_{Bal} = \alpha T_H + (1-\alpha)T_R \mid \alpha \in [0,1] \tag{7}$$

$$T_H = \sum_{i \in SS_H}(Duration_{Ti} + Duration_{Ti}{}') \tag{8}$$

$$T_R = \sum_{i \in SS_R}(Duration_{Ti} + Duration_{Ti}{}') \tag{9}$$

Equations (7)-(9) explain the calculation of balance score T_{Bal}. T_H and T_R represent the total time for human transportation and robot transportation. SS_H and SS_R are the collections of human transportation and robot transportation. α is the balance factor in equation (7). As the critical determinant of the balance score, the balance factor can be set by both users and the rescheduling indicator, which adapts the change of the transportation resources dynamically in real time.

5.3.4 Selection

Based on the result of the evaluation of the population, some individuals will be selected for the further operation. Usually, stronger individual, which have better fitness are preferred. Roulette wheel selection(RWS) and stochastic tournament(ST) are always used for selection.

As a classic proportional selection method, RWS is widely used in GA. The probability of being selected is just related to the fitness as explained in Equation (10).

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j} \tag{10}$$

p_i is the probability of being selected and f_i is the fitness of individual i. When the individual has better fitness, the selected rate is higher.

Figure 52 shows a roulette wheel with 10 individuals. The individual 7 and 10 have higher fitness and should be selected easily. On the other hand, the individual 1 has lowest possibility to be selected.

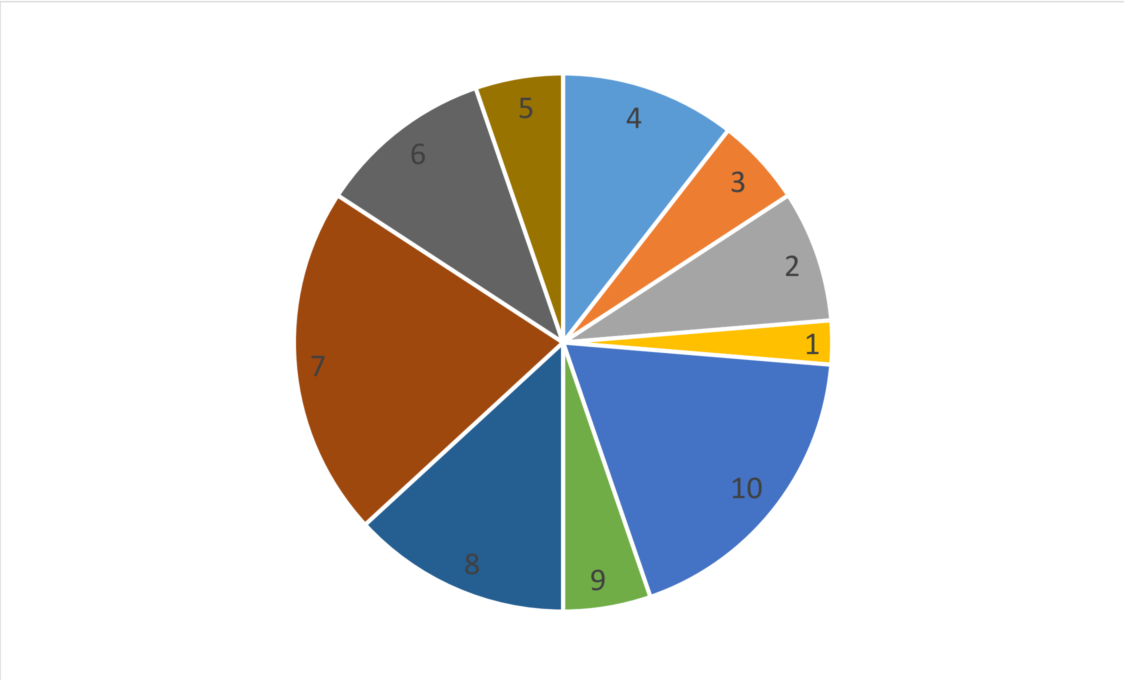


Figure 52 *Roulette wheel with 10 individuals*

ST is based on the RWS to select two individuals, and then selects the better one. It is more complex than RWS, but has better result for the selection.

5.3.5 Crossover

Crossover is an operation to generate new chromosomes based on the selected individuals. Two individuals are required for crossover. They will exchange part of their gene values to generate new individuals.

In this approach, One-point crossover is used to recombine two individuals. In order to increase the performance, only the sequence layer will be operated for crossover. The example is shown in Figure 53.

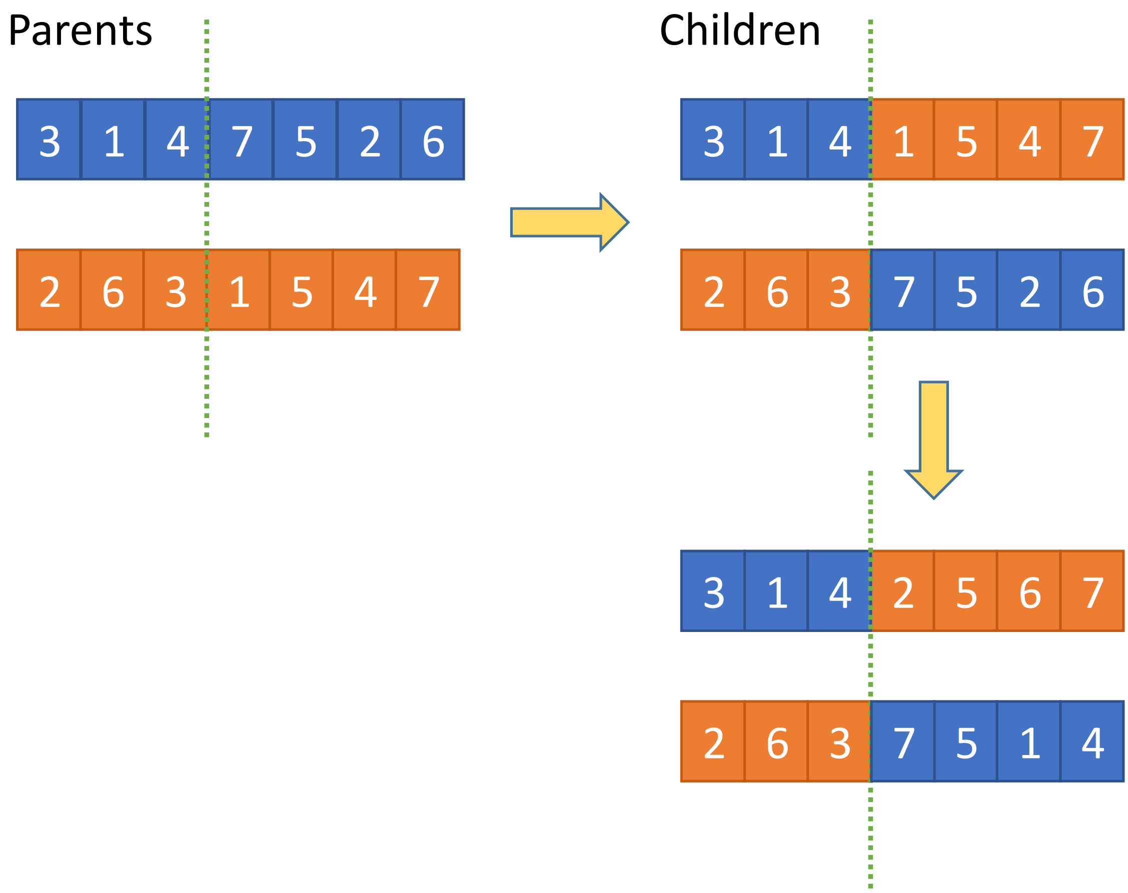


Figure 53 *Crossover procedure for sequence layer*

The first step of the crossover is to select the crossover point for the parents' chromosomes by random. In this example, the point is in the third position. Then the data beyond that point in either chromosome is swapped between the two parents' chromosomes. The results are individuals of children. However, these chromosomes are not valid because of the repetitive and missing steps. They need to be adjusted by adjustment, which has been explained before. After that, two new individuals are generated.

5.3.6 Mutation

In GA, mutation, which is analogous to biological mutation, alters one or more gene values in one chromosome from its initial state. In mutation, the chromosome may change entirely from the previous chromosome.

As the same as crossover, the point for mutation is selected by random. The mutation can be happened for the whole, so there are two situations as shown in Figure 54. Figure 54A explains the procedure of mutation, when the mutation point is in the sequence layer. A random step for example step 4 takes the position of the mutation point and the original step (step 5) is moved to the original position of step 4. The step number in sequence is unique. Therefore, the mutation operation is also an exchanging operation. When the mutation point is in the type layer as shown in Figure 54B, the type is inverted the bit, because the data in type layer are binary numbers.

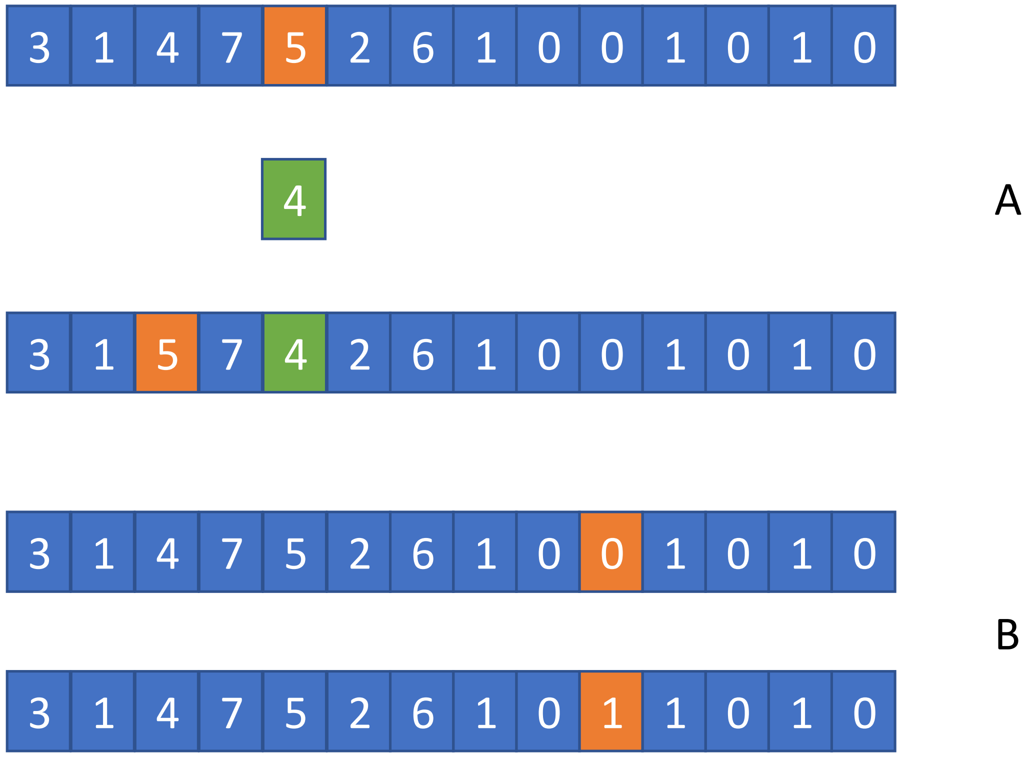


Figure 54 *Two situations for mutation*

After the mutation, the chromosome is maybe not valid anymore. Therefore, the adjustment is required again to make sure the new individual is available.

5.4 Scheduling Algorithm – Modified Genetic Algorithm

In order to obtain better performance, some other algorithms are used to make evaluation and comparison to select the suitable algorithms. As presented before, the algorithms should base on the same or similar data structure to reduce the effort for encoding and adjustment. Because of the high flexibility of genetic algorithm, it is possible to combine it with some other algorithms to create modified genetic algorithms. In this study, two modified genetic algorithms: GASA and GAPSO are designed and evacuated.

5.4.1 GASA

GASA is the genetic algorithm mixed simulated annealing. As explained in chapter 2, simulated annealing is also a good algorithm for optimization. However, it requires high computational resource and not suitable for the real time operation environment.

As the kernel part of simulated annealing, Metropolis-Hastings (MH) algorithm is used to select the individuals during the evolution process of GA. On the one hand, it improves the quality of the population and reduces the iteration number of GA operation. On the other hand, it brings huge number of computation effort. It is a challenge to balance the performance and requirements during the parameters selection. The flowchart of the GASA operation is shown in Figure 55 and the pseudocode is shown in Figure 56.

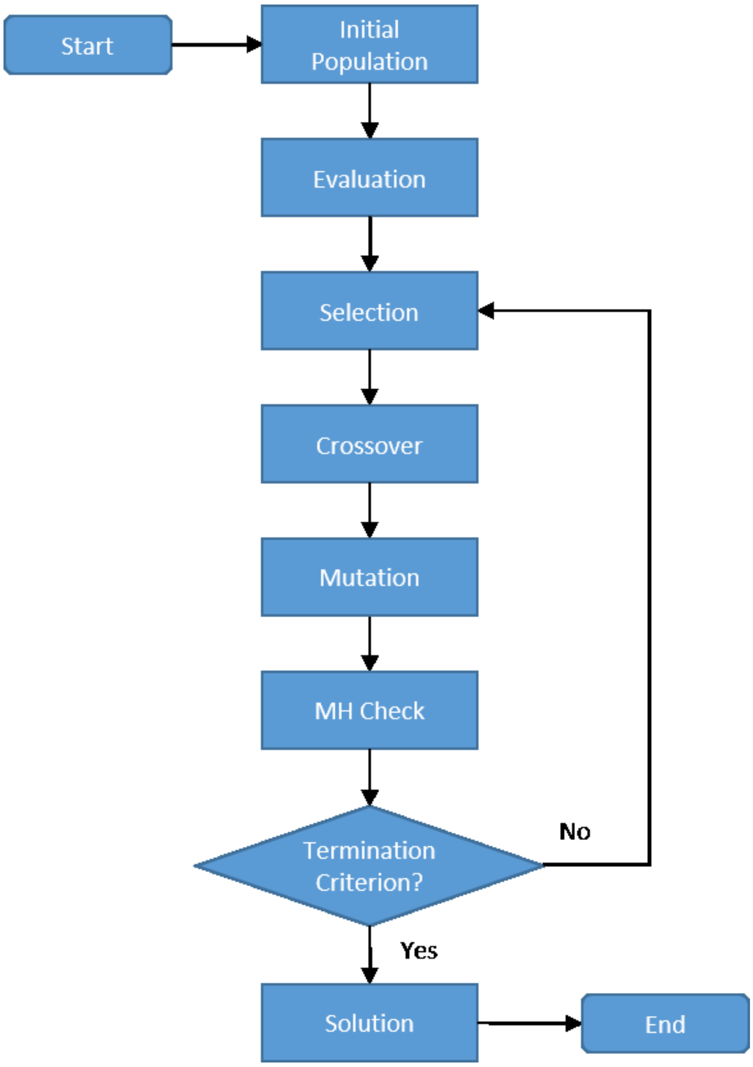


Figure 55 *GASA Flowchart*

```
// Genetic Algorithm mixed Simulated Annealing with geometric temperature cooling
Pn ← population_scale                        // Initialize population scale
Pop ← initial_population(Pn)                 // Initialize population
Cr ← crossover_rate                          // Initialize crossover rate
Mr ← mutation_rate                           // Initialize mutation rate
MX ← maximum_generations                     // Initialize maximum generations
T ← T_0                                      // Initialize temperature
α ← α_0                                      // Initialize cooling factor
T_end ← Tend                                 // Initialize end temperature
// Compute a new population and objective function value
O ← fitness(Pop)                             // Evaluate initial population
BestIndividual ← cal(Pop, O)                 // Search the best individual in the population

while T > T_end:
    i ← 0                                    // Initialize generation count
    while i < MX:
        SelPop ← select(Pop)                 // Selection operation
        SelPop ← across(SelPop, Cr)          // Crossover operation
        SelPop ← aberrance(SelPop, Mr)       // Mutation operation
        SelPop ← cab(SelPop)                 // Adjustment operation
        Pop' ← replace(SelPop)               // Replace the selected individuals
        O' ← fitness(Pop')                   // Evaluate the new population
        for j ← 1: Pn
            delta = O(j) − O'(j)             // Difference between two individuals
            if delta ≤ 0
                Pop(j) = Pop'(j) , O(j) = O'(j)  // Accept new individual
            else
                if random(0,1) ≤ e^(−delta/T) // Metropolis criterion checking
                    Pop(j) = Pop'(j) , O(j) = O'(j)
                end
            end
        end
        BI_i ← cal(Pop, O)                   // Search the best solution in the new population
        If BI_i < BestIndivivual
            BestIndivivual = BI_i            // Best individual storage
        end
        i ← i + 1                            // Increment trial count
    end
    T ← αT                                   // Cooling process
end
```

Figure 56 *Pseudocode of Genetic Algorithm mixed Simulated Annealing*

After the GA operation, the MH check helps to filter out the worse new individuals comparing with the old individuals. It can increase the successful rate for the optimization with less generations. However, it also increases the complexity and computation effort.

5.4.2 GAPSO

Particle swarm optimization (PSO) is a population based stochastic optimization technique, which is attributed to Kennedy and Eberhart [105], [106]. It simulates the behaviors of bird flocking and solves a problem by generating a population of candidate solutions, which are called particles. These particles are moved around in the search space according to the mathematical formulae over the particle's position and velocity. The movement of particle is influenced by its local best known position, but it is also guided toward the global best known positions in the search space, which are updated as better positions are found by other particles. The purpose of this procedure is to move the swarm toward the best solutions [107], [108].

PSO is initialized with a group of random particles (solutions) and then

The core operation of PSO is to search for optima by updating generations. After initialization with a group of random particles (solutions), the loop for updating generations is started. In every iteration, all particles are updated by following two "best" values. One is the best solution it has achieved so far, which is called Pbest. Another "best" value, which is tracked by the particle swarm optimizer is the best solution, obtained so far by all particles in the population. This best value is a global best and called Gbest.

Based on the two best values, the particle updates its velocity and positions with following Equation (11) and (12).

$$v[\,] = v[\,] + c1 * rand(\,) * (Pbest[\,] - present[\,]) + c2 * rand(\,) * (Gbest[\,] - present[\,]) \tag{11}$$

$$present[\,] = present[\,] + v[\,] \tag{12}$$

$v[\,]$ is the particle velocity, $present[\,]$ is the current particle (solution). $rand(\,)$ is a random number between 0 and 1. $c1$, $c2$ are learning factors and they are constant. The pseudocode of the procedure is as shown in Figure below.

```
For each particle
    Initialize particle
END

Do
    For each particle
        Calculate fitness value
        If the fitness value is better than the best fitness value (Pbest) in history
            set current value as the new Pbest
    End

    Choose the particle with the best fitness value of all the particles as the Gbest
    For each particle
        Calculate particle velocity according Equation (11)
        Update particle position according Equation (12)
    End
While maximum iterations or minimum error criteria is not attained
```

Figure 57 *Pseudocode of particle swarm optimization*

Based on the analysis of PSO, the advantages and disadvantages are shown clearly[109], [110]. For advantages, PSO is based on the intelligence and it can be applied into both scientific research and engineering domain. Then PSO has no overlapping and mutation calculation. The search is carried out by the speed of the particle. Based on the development of several generations, only the optimal particle can transmit information to the other particles. Therefore, the speed of the researching is very fast. Compared with the other algorithms, it has better optimization ability and can be completed easily. On the other hands, disadvantages of PSO are the method easily suffering from the partial optimism, which causes the less exact for the management of its speed and direction. Therefore, the method cannot solve the problems of scattering and optimization and the method cannot work for non-coordinate systems, such as the energy field solution.

Based on the original PSO, a modified genetic algorithm is designed. The detailed flowchart is shown in Figure 58 and the pseudocode of GAPSO is as shown in Figure 59.

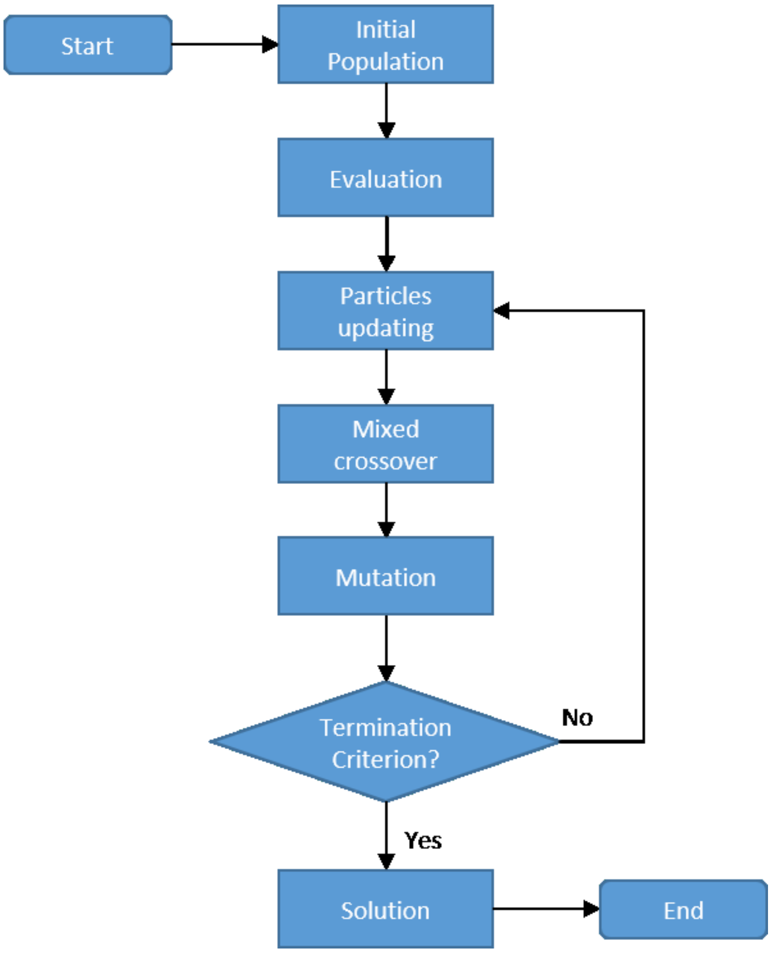


Figure 58 *GAPSO flowchart*

```
// Genetic Algorithm mixed Particle Swarm Optimization
Pn ← population_scale                       // Initialize population scale
Pop ← initial_population(Pn)                // Initialize population
Cr ← crossover_rate                         // Initialize crossover rate
Mr ← mutation_rate                          // Initialize mutation rate
i ← 0                                       // Initialize generation count
MX ← maximum_generations                    // Initialize maximum generations
// Compute a new population and objective function value
O ← fitness(Pop)                            // Evaluate initial population (Pbest)
BestIndividual ← cal(Pop, O)                // Search the best individual in the population
BestFitness ← min(O)                        // Search the best Fitness (Gbest)
BestPop ← Pop                               // Initialize the best population

while i < MX:
    // Mixed crossover operation
    Pop ← mixacross(Pop, BestPop, BestIndividual, BestFitness, Cr)
    Pop ← aberrance(Pop, Mr)                // Mutation operation
    Pop ← cab(Pop)                          // Adjustment operation
    O ← fitness(Pop)                        // Evaluate the new population
    // Update best elements
    BestPop ← updateP(Pop, O)
    BestFitness ← updateF(O)
    BestIndividual ← updateI(Pop, O)

    i ← i + 1                               // Increment trial count
end
```

Figure 59 *Pseudocode of Genetic Algorithm mixed Particle Swarm Optimization*

In GAPSO, the updating for particle velocity and position is implemented by a mixed crossover and mutation operation. They are guided by their own best known position in the entire solution domain. The benefit of this algorithm is to avoid the local optimum and keep the efficient feature of PSO at the same time.

Without selection operation, all the individuals will be sent to the further steps. This is the biggest difference to the original GA. Instead of selection, the Pbest and Gbest of particles will be updated in every iteration. In the mixed crossover step, the individuals will crossover with Pbest and Gbest respectively to improve their genes with “best” chromosomes. After the mutation operation, the termination criterion needs to be checked. If it is necessary, the procedure will jump back to evaluation and repeat again.

5.5 Algorithms Evaluation and Comparison

The purpose of algorithms evaluation and comparison is to select the suitable algorithms for both static and dynamic scheduling. The first and important step is to design the experiment based on the algorithms’ characteristic and the optimization requirements. A typical workflow is used to evaluate various algorithms. The result will present the algorithms’ performance and it is the ground for the algorithm selection.

5.5.1 Experiment Design

The different requirements of these two scheduling can be ascribed to the limitation of computational time and the quality of the results. For static scheduling, it requires high quality of the result and it is acceptable if the computational time is less than one minute. In contrast, dynamic scheduling should be completed in ten seconds. However, the requirement of result quality is not as strict as the static scheduling.

As iterative optimization algorithms, the most important parameters for both computational time and the quality of the results are the generation scale and population scale. Normally, when the number of iteration and population are bigger, there is higher probability to reach the optimum solution. However, it brings higher computational requirement. That means the algorithm with more iteration and population needs more computational time and has possibly better result. In contrast, less iteration and population requires less computational time and the quality of the result is not so good. The other parameters such as crossover and mutation rate can influence the rate of convergence. These parameters are decided by experience formulas and algorithm optimization.

Based on the different requirements and the characteristic of the algorithms, two groups of parameters are prepared for each algorithm to simulate the static and dynamic scheduling scenarios. The characteristics and theory determine that it is desired but not guaranteed, that the global optimum will eventually discovered all the time. In order to estimate the results, every algorithm with one group of parameters is processed for 100 times. The distribution of the results is used to evaluate the performance of this algorithm in this scenario.

A typical workflow is designed for scheduling simulation to evaluate the algorithm performance in two scenarios. The workflow is shown in Figure 60. In this workflow, the gray blocks, which are located in the left and right sides, represent the input contains and output slots. Among them, there are seven processes, which are distributed on four automated workstations in three different laboratories. Different process colors represent different operation types including biology, chemistry and etc. The connections between processes are transportations with the transportation type labels. The detailed information of the transportations and automated processes is shown in Table 12 and 13.

Table 12 *Transportation Duration Time (min)*

ID \ ID	0	1	2	3	4
0	[0, 0]	[2, 3]	[3, 5]	[5, 7]	[8, 12]
1	[2, 3]	[0, 0]	[2, 4]	[4, 6]	[7, 10]
2	[3, 5]	[2, 4]	[0, 0]	[2, 4]	[5, 8]
3	[5, 7]	[4, 6]	[2, 4]	[0, 0]	[4, 7]
4	[8, 12]	[7, 10]	[5, 8]	[4, 7]	[0, 0]

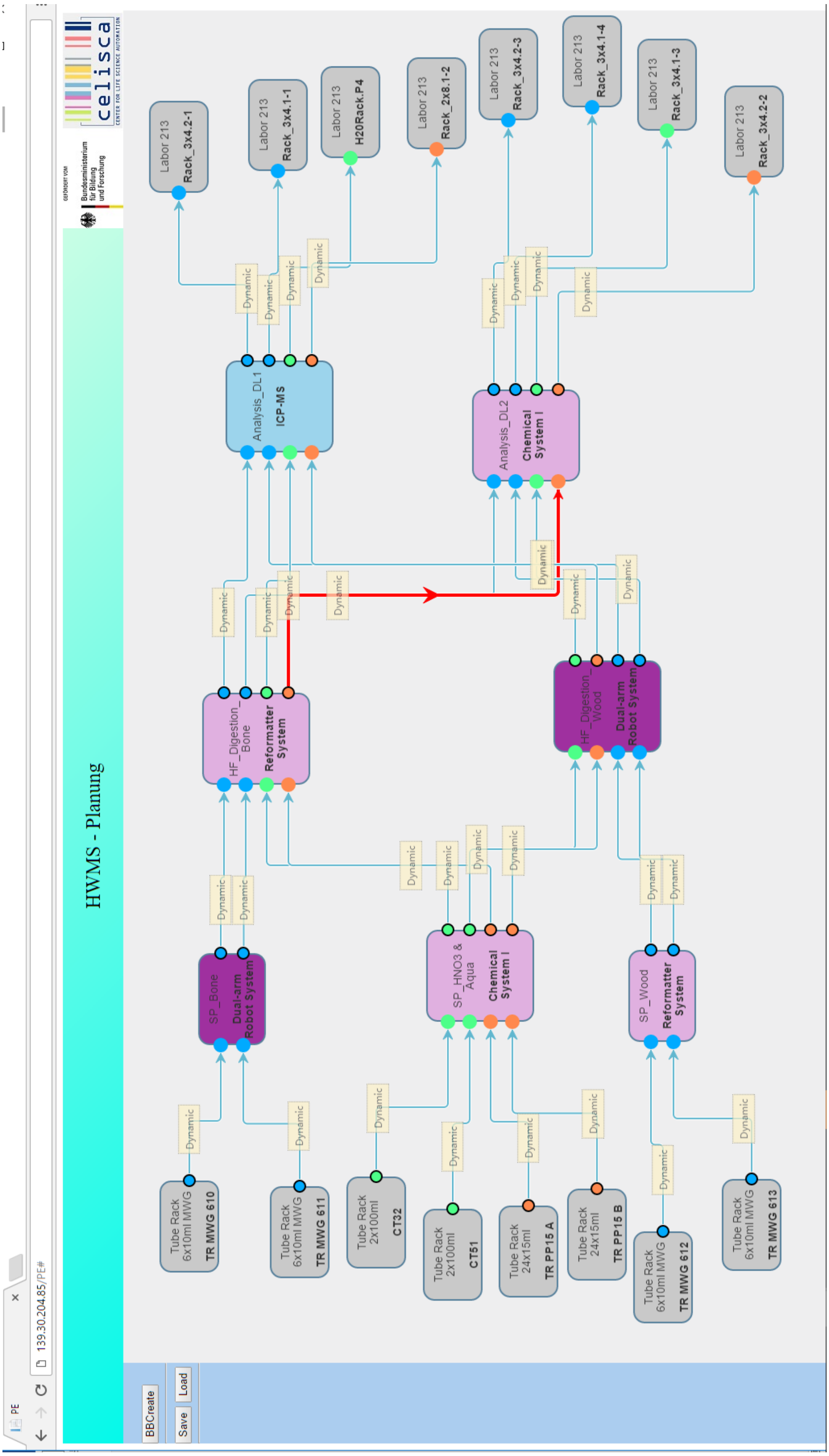


Figure 60 *Example Workflow*

Table 13 *Detailed Information of Processes*

Process		Workstation		
ID	Name	ID	Name	Duration
1	Start Point	0	Manual Lab	0 min
2	SP_Bone	1	Reformatter	6 min
3	SP_HNO3 & Aqua	2	Chemical System I	20 min
4	SP_Wood	4	Dual-arm Robot System	9 min
5	HF_Digestion_Bone	4	Dual-arm Robot System	24 min
6	HF_Digestion_Wood	1	Reformatter	18 min
7	Analysis_DL1	3	ICP-MS	11 min
8	Analysis_DL2	2	Chemical System I	16 min
9	End Point	0	Manual Lab	0 min

Each transportation has two endpoints, the source endpoint and target endpoint. An endpoint color represents a specific container type. The transportation cannot be created when the user tries to connect two endpoints with different colors to avoid low-level mistakes during the workflow designing.

Table 12 describes the transportation time between two workstations. The *ID* in both horizontal and vertical titles represents the ID number of each workstation as described in Table 13. Each cell has two numbers, which respectively stand for human transportation and mobile robot transportation duration. Table 13 shows the detailed information of each process. It includes the workstation, which will be used and the duration time for the processes.

As the example, the parameters of GA for static scheduling are listed in Table 14. The population scale and generation scale are reduced to 40 and 100 in the dynamic scenario to decrease the computational time. There is no specific solution or formulas to generate the GGAP, crossover and mutation rate. The parameters in this study are based on the experience formulas and algorithm optimization.

Table 14 *Genetic Algorithm Parameters for Static Scheduling*

Settings and Parameters	Value
GA population scale	60
GA GGAP	0.3
GA generation scale	200
GA crossover rate	0.4
GA mutation rate	0.5

In this workflow, there are seven processes based on four automated workstations. Including the start and end position, this workflow involves five laboratories/rooms located in two floors. The processes "SP_Bone", "SP_HNO3 & Aqua" and "SP_Wood" prepare the samples and materials for the reaction. After the processes "HF_Digestion_Bone" and "HF_Digestion_Wood", the samples will be analyzed with the help of "Analysis_DL1" and "Analysis_DL2". When the analysis processes are completed, all the labware will be transported back to the end position. Inevitably, the parallel operation and resource contention are existed all the time during the execution of the workflow. The complexity of the transportation tasks have reached the highest level including both human and robot tasks. Therefore, this workflow is a suitable instance to evaluate scheduling algorithms.

5.5.2 Scheduling Solution

The Figure 61 shows the results of the optimization procedure by the GA. Along with the growth of the generation, the fitness and mean value are improved dramatically. Since 125th generation, the fitness is invariant. That means the optimized solution is presented.

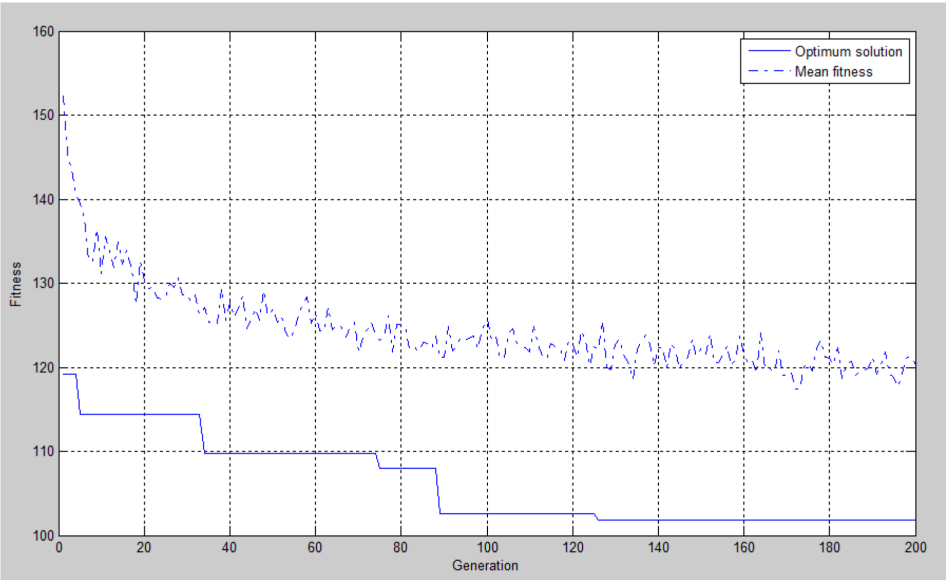
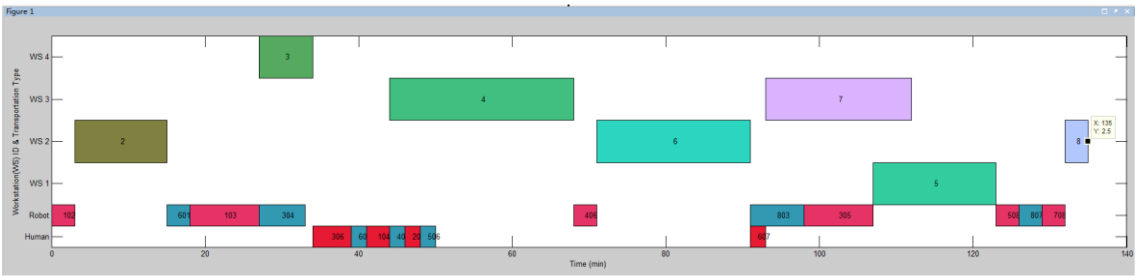
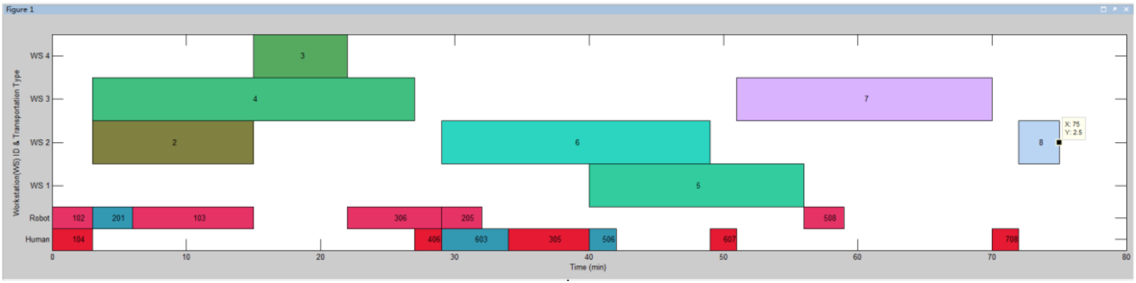


Figure 61 *Evolution procedure of GA*



(a)



(b)

Figure 62 *Scheduling result in Gant chart. Numbers on the process tasks located in WS 1-4 are the Process ID: 1: StartPoint, 2: SP_Bone, 3: SP_HNO3 & Aqua, 4: SP_Wood, 5: HF_Digestion_Bone, 6: HF_Digestion_Wood, 7: Analysis_DL1, 8: Analysis_DL2. The name of transportation task for Robot and Human is formed as "SourceID+0+TargetID". For example, the transportation "306" is the transportation task from PorcessID 3 (SP_HNO3 & Aqua) to ProcessID 6 (HF_Digestion_Wood).*

The result of the scheduler is presented in a Gantt chart to describe the realistic work list in laboratories. As shown in Figure 62A it is the Gantt chart of a random chromosome in the initial population. This Gantt chart is one of the valid solutions for the workflow and it must satisfy all conditions. The process tasks are distributed in the four automated stations without error. The number of each task is the process ID as presented in Table 13 and this ID is also used to explain the transportation task.

In the transportation part, the red blocks present the normal transportation tasks. The blue blocks are additional transportation time, which is used to describe the time cost because of the difference of the start position and the end position of the former task. Saving the transportation resource is the purpose of the scheduler. The numbers on each transportation task is the name of the task. It contains the IDs of the source process and target process.

According to the random solution, this workflow needs 135 minutes to be completed and the transportation time is 67 minutes. The Figure 62B shows the optimized solution. Compared with the random one, it takes less operation time (75 minutes) and less transportation time (51 minutes). Not only GA, both GASA and GAPSO can generate this optimized solution as well. Therefore, it is not necessary to present the solutions of GASA and GAPSO again.

5.5.3 Scheduling Results Comparison

In order to evaluate the quality of the results for these three algorithms in two scenarios, each algorithm with one group of parameters will be operated for 100 times as describing in the experiment design. The result is presented with the score of the fitness, which combines both workflow duration time and the balance score of the transportation task allocation.

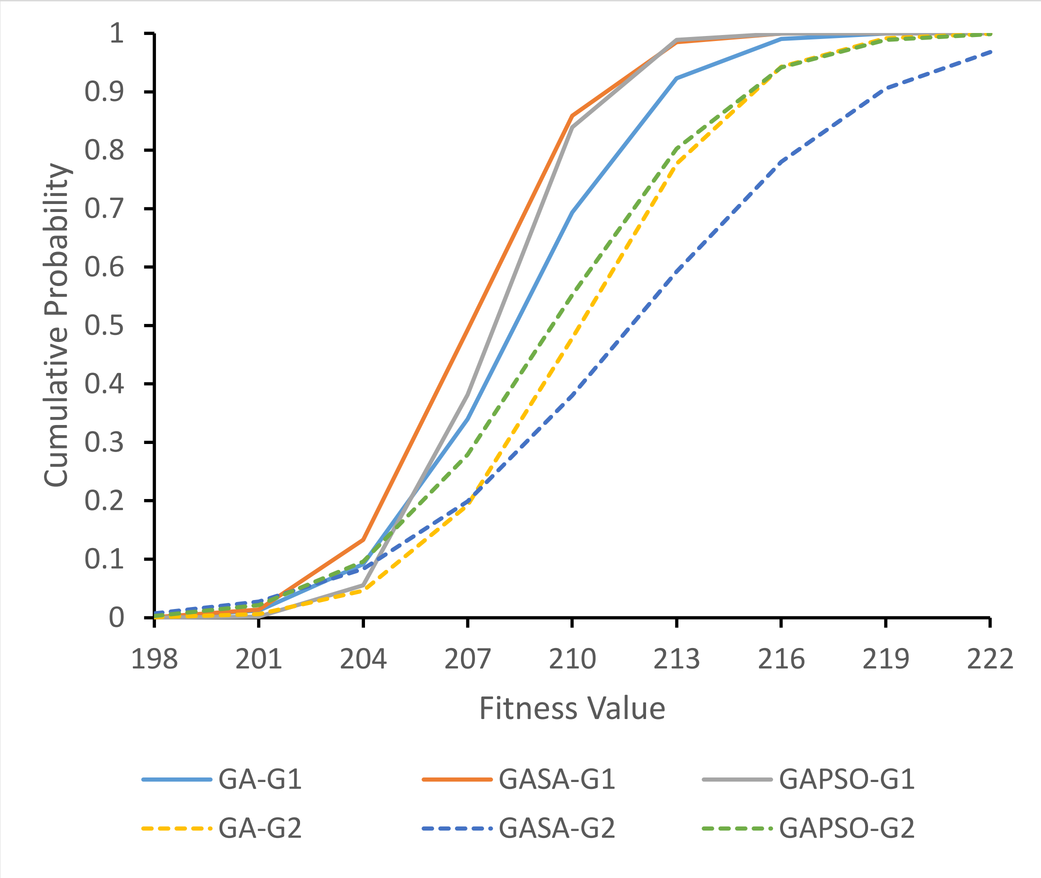


Figure 63 *Cumulative normal distribution of three algorithms with G1 and G2*

Figure 63 presents the cumulative normal distribution of these three algorithms with two groups of parameters. Solid lines and dotted lines represent the scheduling results with G1 and G2 parameter respectively. It can be seen from Table 15 that G1 is used to simulate the static scheduling and G2 is for dynamic scheduling. The mean of the fitness values represents the performance of the algorithm. Two generally adopted deviation estimating indexes, the MAE (Mean Absolute Deviation) and the MAPE (Mean Absolute Percentage Error) reflect the difference between the result and optimum. They reflect the stability and the robustness of the combination of algorithms and parameter groups. When the values of MAE and MAPE are smaller than others, the fitness values of this combination do not fluctuate as much as other combination. That means, it is robust against the random poputation initialisation.

Table 15 *Evaluation results*

	Duration (s)	Success rate (%)	Mean	MAE	MAPE (%)
GA-G1	24.81	100	208.37	2.34	0.90
GASA-G1	24.66	100	207.04	1.05	0.51
GAPSO-G1	24.75	100	207.72	1.77	0.86
GA-G2	4.98	84	210.27	4.27	2.07
GASA-G2	4.92	66	211.74	5.78	2.81
GAPSO-G2	4.84	93	209.95	3.45	1.67

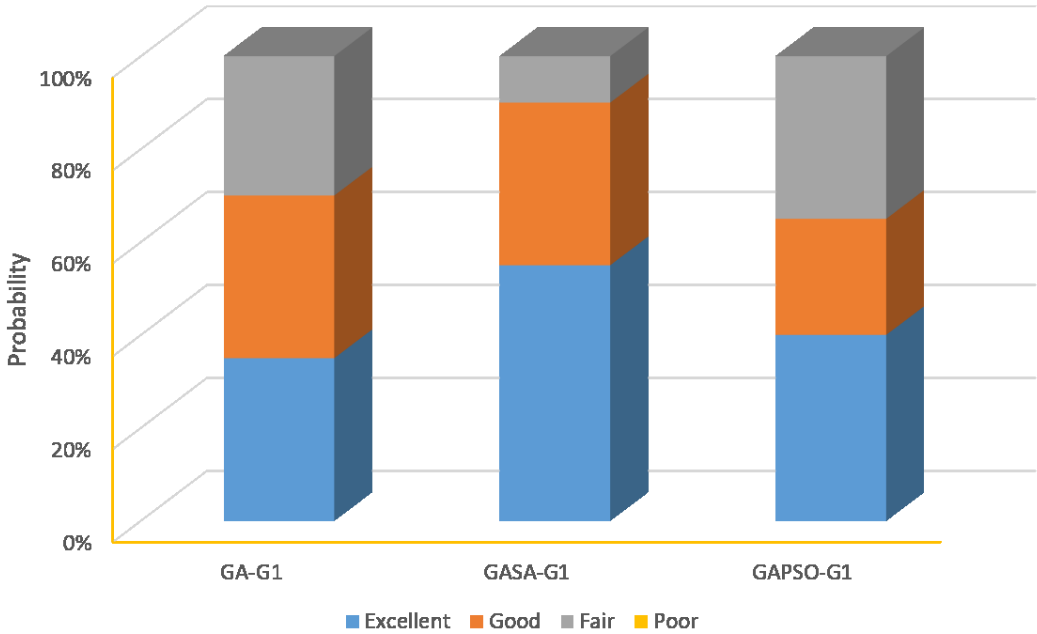


Figure 64 *Scheduling results distribution with G1*

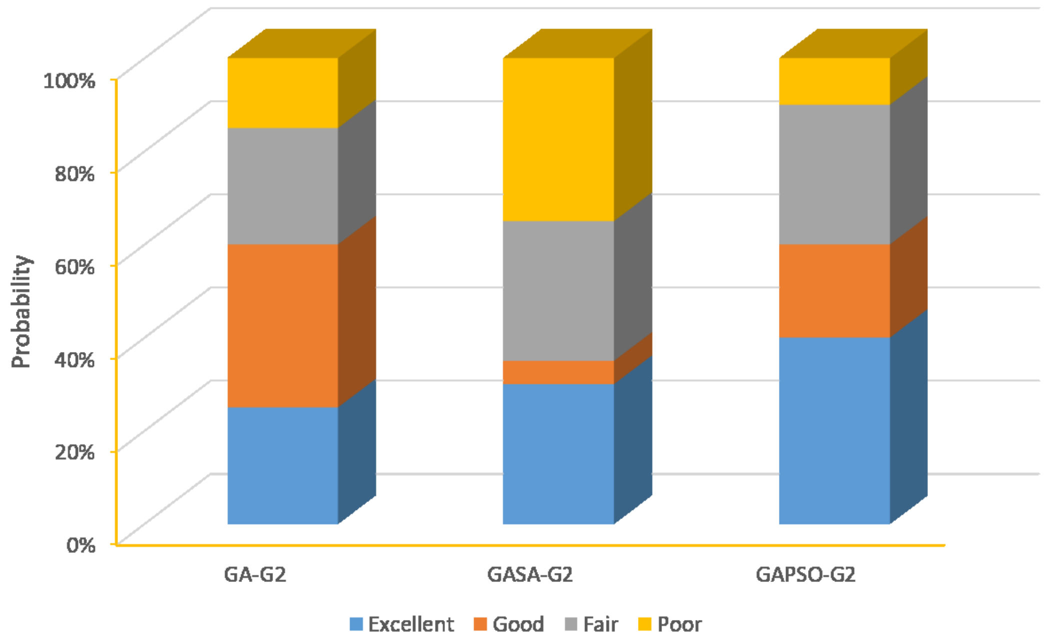


Figure 65 *Scheduling results distribution with G2*

When the fitness values are no more than 103%, 105% and 108%, they are divided as *Excellent*, *Good* and *Fair*. When the result is more than 108% of the optimum, it is evaluated as poor, which will not be calculated as a successful result. With the parameter of G1, the success rate is all 100% as shown in Figure 64. Moreover, the results of GASA have the best distribution with the lowest Mean, MAE and MAPE. Therefore, GASA-G1 has the best performance and stability. For the parameter G2, the mean of GA and GAPSO are almost the same. That means, their performance is similar. In contrast, GAPSO has lower MAE and higher successful rate. Moreover, Figure 65 shows that, GAPSO-G2 has the best cumulative normal distribution comparing with the other two algorithms with parameter G2. This implies that GAPSO-G2 is more stable and reliable than the other two algorithms.

5.5.4 Algorithm Selection

For the static scheduling scenario, algorithms with parameter group G1 required around 25 seconds' computation time. All algorithms achieved 100% success rate. In contrast, GASA has the highest performance with lowest MAE and MAEP. For the dynamic scheduling scenario, these three algorithms used less than 5 seconds to finish the processing with parameter group G2. The results were not as good as the G1 cause of the time limitation. GAPSO presents best performance and stability in this group. However, GASA got an unacceptable result with 66% success rate and it proved the validity and necessity to adopt two different algorithms for static and dynamic scheduling. Based on this result it is clear that (a) GASA is suitable for static scheduling due to the highest performance and stability; (b) GAPSO is appropriate for dynamic scheduling due to its high efficiency to obtain acceptable in several seconds.

Chapter 6 Rescheduling Indicator

As explained before, dynamic scheduling adopts a predictive-reactive strategy. That means it is implemented by rescheduling. In the last section, the rescheduling algorithm has been selected. In this chapter, the problems, when and how to reschedule should be solved. As the base for trigger, the triggering policy will be discussed at first. Triggering conditions and strategy will guild the indicator, under which situation to start reschedule and how to reschedule. Of cause, the unexpected situations are dynamic and unpredictable. How to make a classification for correct rescheduling is also significate.

6.1 Triggering Policy

For predictive-reactive scheduling three policies can be used to trigger the rescheduling process: periodic, event-driven and hybrid [101]. ***Vieira et al.*** presented analytical models to predict the performance of dynamic workshops [103], [111]. ***Church and Uzsoy*** analyzed both periodic and event-driven triggers in single and parallel-machine models [102]. From these literature, it can be seen that: (a) hybrid policy has the best performance for the response of the unexpected situation, but it requires much more rescheduling than others. It is suitable for the open and flexible environments; (b) periodic policy can offer both high and low dynamic response. It requires quite simple triggering conditions: timer. It is an ideal policy, if the unexpected situations happen periodic. In contrast, this policy is adopted with completely reactive scheduling normally; (c) event-driven policy obtains high-quality schedule with less rescheduling than periodic policy. It is suitable for predictive-reactive scheduling to handle the close and stable environments.

In the life science laboratory, there are chemical and biological experiments operating, which requires clean and safe environments. As a result, the laboratory is relative close and the unexpected situations appear randomly and infrequently. Therefore, event-driven policy is selected for rescheduling.

6.2 Triggering Condition

The execution controller collects the unexpected situations from the transportation management system and the automated systems and offers special feedback to the rescheduling indicator. The unexpected events, which are accounted as triggering conditions in laboratories, are proposed in Table 16.

Table 16 *Unexpected situations*

Transportation (T task)	Process (P task)
Transportation Delay	Initialization Delay
Transportation Failure	Initialization Error
Resource Status Change	Execution Failure

The delay means that a task requires more time to be complete. Errors and failures represent that the task is interrupted and waits for the assistance by human. There are two approaches for the execution of transportation orders in HWMS: mobile robots and human assistants. Mostly, the delay of mobile robot transportations appears with unexpected collision avoidance, which requests more time to guarantee the security of transportation (T tasks). Human assistants have to perform several different tasks in laboratories and cannot ensure to complete all T tasks in time. Beside the number of available mobile robots and human assistants the transportation resource status also includes the battery status and current position of each robot and the distribution of the transportation delays.

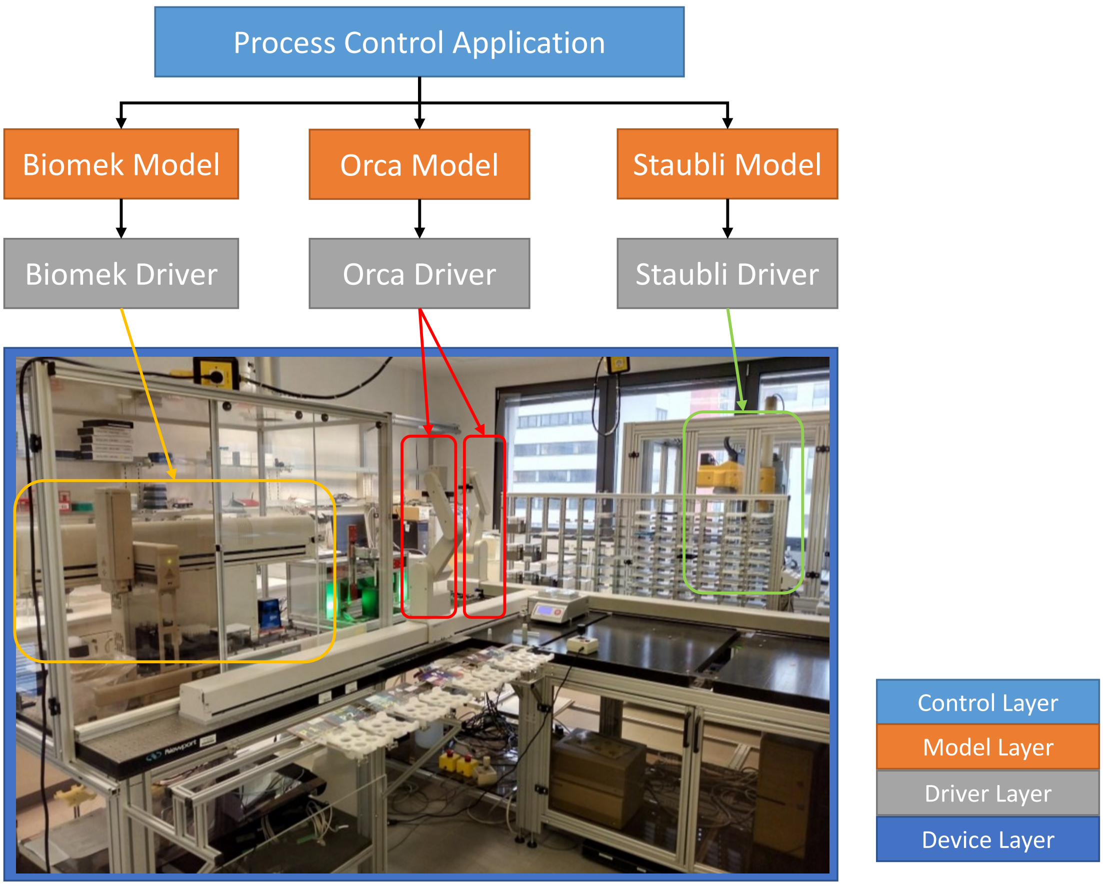


Figure 66 *Controlling Structure of automated system Reformatter*

In order to describe the process (P task) concerning unexpected events visually, the controlling structure for the automated system *Reformatter* [112] is given in Figure 66. There are four layers in the process control system. On the top layer is the process control application (PCA), which is used to handle the whole automated workstation. This application integrates various subsystems with the help of the model layer. Each model is a function package, which can be recognized and used by the PCA for specific task scheduling and distributions. Driver Layer is the interface between Model Layer and Device Layer. The driver is generated by the provider of the device and enables higher layer to access hardware functions. A one-on-one relationship exists between one model and its related driver. Moreover, one driver can control more than one device. As shown in this automated system *Reformatter*, the orca driver is designed to handle two orca robots.

Based on the execution history of various automated systems, most of the automated systems' problems appear during the initialization. Beside the hardware issues, the initial failures occur also in the Model Layer or the Driver Layer. The initial failures from the Model Layer can be solved by repeating the procedure of initialization but the failures from the Driver Layer requires assistance. Therefore, the model issues generate initialization delays and the driver issues become initialization errors. If there is an error occurring during the execution, the P task will be blocked and produces the execution failure.

6.3 Triggering Strategy

The major purpose of the triggering strategy is to answer two questions: when and how to trigger the rescheduling procedure. Errors and failures will block the T and P task and they require rescheduling without doubt. The delay handling is more complex and it is a challenge for the triggering strategy.

Different from the error and failure situation, delays will not stop the process and in the reality, the delay is calculated after the task is completed. Of cause, when the task is not finished in the maximal duration, the task will be set as failure. It is not necessary to reschedule for short time delays because these are also quite normal during the execution of workflows. In this study, when the delay accumulation is over the threshold, the rescheduling procedure will be triggered. Furthermore, the distribution of the delays is the most important parameter to guide the rescheduling.

In order to balance the task distribution between mobile robots and human assistants, the balance factor is required in the schedule engine of HWMS. The obvious benefit is to adapt schedule result to the exchange of the transportation resources. This balance factor should be generated by the rescheduling indicator based on the delay distribution and transportation resource status change.

As the most important parameter for the T tasks distribution, the balance factor is given by the administrator as the initial setting before the workflow execution. Following the executing, the balance factor is dynamic because of the changing of the transportation resource. One of the challenges for the rescheduling indicator is to generate a suitable balance factor.

6.4 Balance factor Generation

There is no formula or equation for the balance factor. It is the reflection of the status of the transportation resource. The value of the balance factor is defined by administer, who has rich experience to handle the executions in laboratories. It is possible to generate an initial dataset based on various status of the mobile robots and human assistants.

There is no standard to descript of the transportation status. In this study, some core parameters form MRMS, HACS and the record of the T task execution are used to assess the status. Both MRMS and HACS can offer the available count of the mobile robots and human. Furthermore, MRMS provides the battery voltage of each robot. The execution record can offer the delay information of the previous T tasks. These parameters are the decision fundaments for the balance factor.

Therefore, there are five inputs to generate the balance factor and the number of combinations with these five parameters is more than 200 million. It is no possible to generate the balance factor for all these combinations by human. A forecasting solution is required to solve the problem.

There are two methods suitable for the forecasting as the mainstream solutions: expert system and artificial neural network. Both of them are based on the limited existing information to train a forecast model. The expert system is one of the most popular method for the dynamic system [113], [114]. However, it requires too much resource to build knowledge base and has the limitation to face new situations, which are out of the knowledge base. Another major method is artificial neural network (ANN) and it is used widely in various field [115], [116]. It has more flexibility and lower training cost than expert system. Therefore, it is suitable for the dynamic laboratory environment.

6.4.1 Establishment of Artificial Neural Networks

The computational scheme of the Artificial Neural Network (ANN) is presented in Figure 67. The ANN procedure has two steps: the model training and forecasting. In the former step, historical real operation data and their corresponding transportation balance factors constitute the training samples, which are used to teach the network regarding the potential relationship between the operation status data and the balance factor. In the latter step, the trained ANN will forecast the balance factor based on the real-time system status data.

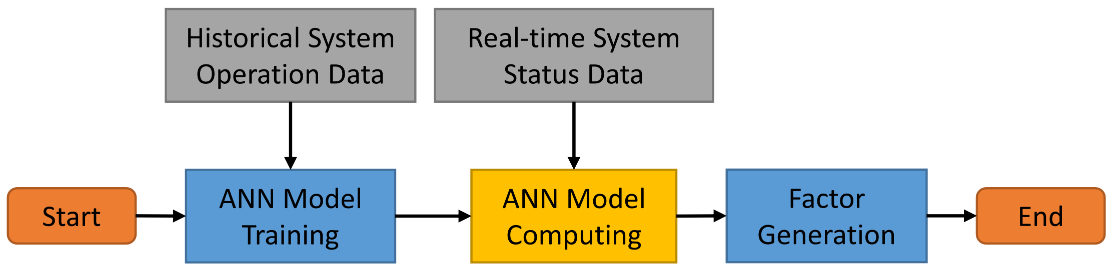


Figure 67 *The computational schema for ANN model*

In ANN theory, the multi-layer perception (MLP) network is popular because of its striking nonlinear fitting capacity. It has been proved that the three-layer MLP network is able to cope with any kinds of nonlinear problems [117], [118].

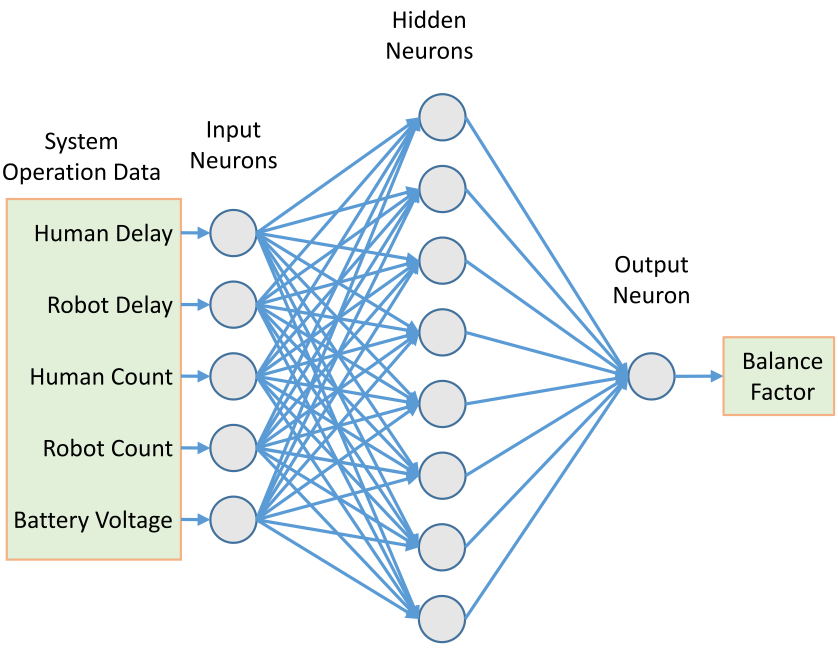


Figure 68 *The MLP network structure with 8 hidden neurons*

Moreover, by considering the real-time performances and the computing requirements of the system, a three-layer MLP network is selected and designed as shown in Figure 68. The input of the MLP network is the current operation status data, which consist of five parts: human assistant delay, mobile robot delay, mobile robot battery voltage and the number of available assistants and robots as explained before. These data reflect the current status and capacity for the transportation tasks. On the other side, the output of the MLP network is the balance factor, which is used to guide the distribution of the transportation tasks in the schedule engine.

6.4.2 Training Algorithm Selection

There are several algorithms to train the MLP network, including Back Propagation (BP), Resilient Back Propagation (RPROP), Quasi-Newton Back Propagation (BFGS) and Stochastic Gradient Descent Back Propagation (SGD) [117], [118]. In order to select the best one among them, a set of trial simulations is provided here and, in these simulations, four MLP networks are generated and trained with different algorithms respectively via neural network toolbox in Matlab using the same training procedure as given in Figure 69. The mean absolute error (MAE) of the balance factor and the mean absolute percentage error (MAPE) are adopted to compare the mapping performance of these four algorithms completely. As shown in Figure 68-71, the RPROP, BFGS and BFGS algorithms cannot reach the requested performance threshold before 150 iterative steps while the BP algorithm use 86 steps to satisfy the train goal. That means the training solution with BP algorithm has highest performance comparing with other three algorithms. Therefore, the BP is selected for the MLP network training.

```
%% Data Loading and Normalization
[inputn,inputps]=mapminmax(input_train);
[outputn,outputps]=mapminmax(output_train);
%% Training Parameters of MLP Network
tic
net=newff(inputn,outputn,10);
net.trainFcn='trainlm';
net.trainParam.show=20;
net.trainParam.epochs=150;
net.trainParam.lr=0.1;
net.trainParam.goal=0.0004;
net.trainParam.min_grad=1e-15;
net=train(net,inputn,outputn);
toc
%% Training End and Ready for Testing
```

Figure 69 *Training procedure of MLP networks*

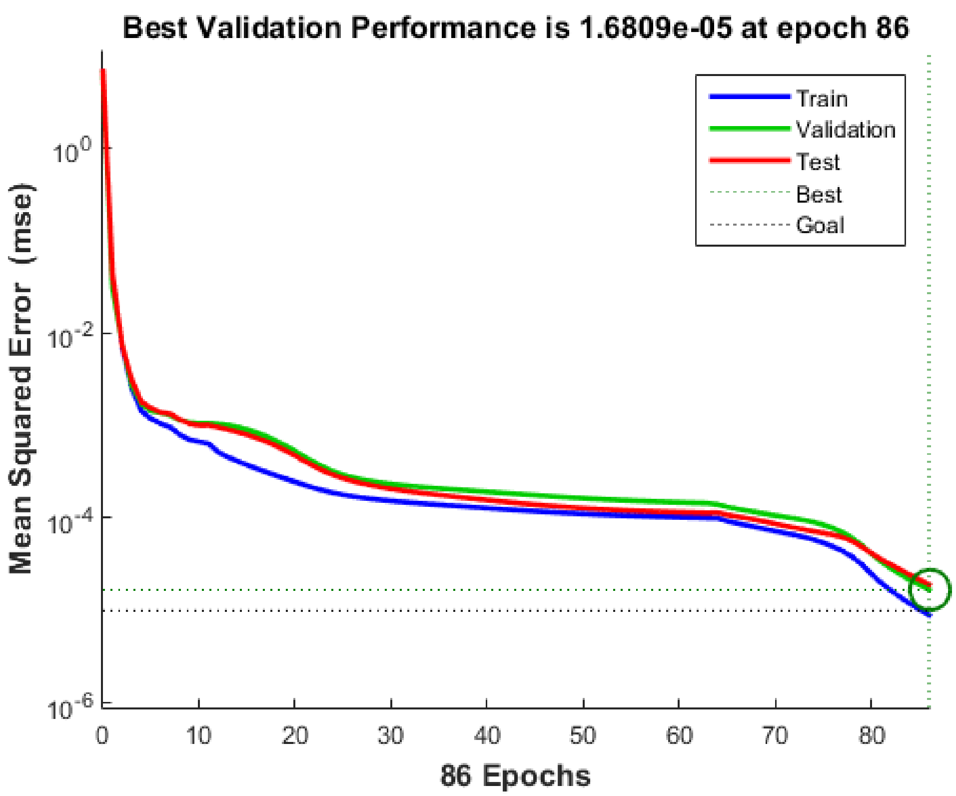


Figure 70 *The performance of training with BP algorithm*

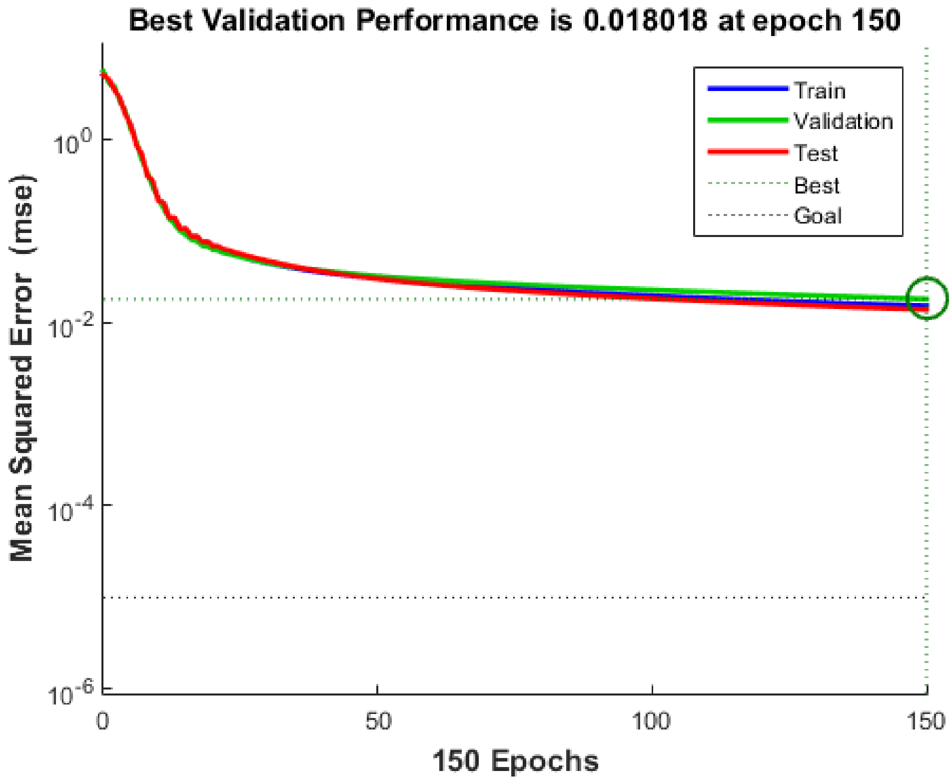


Figure 71 *The performance of training with RPROP algorithm*

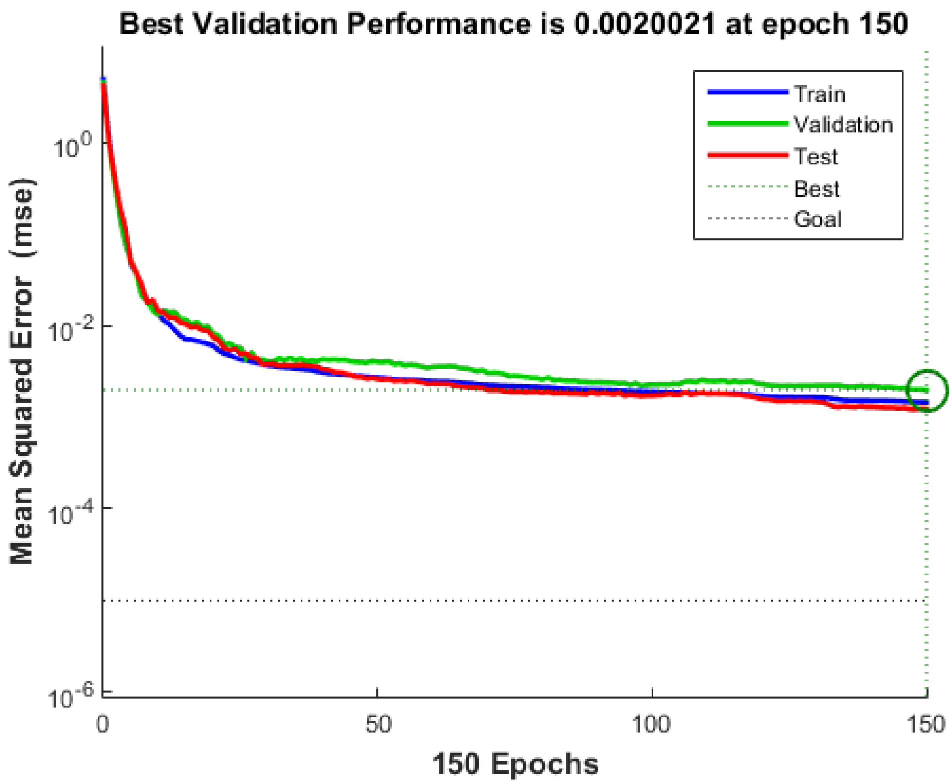


Figure 72 *The performance of training with BFGS algorithm*

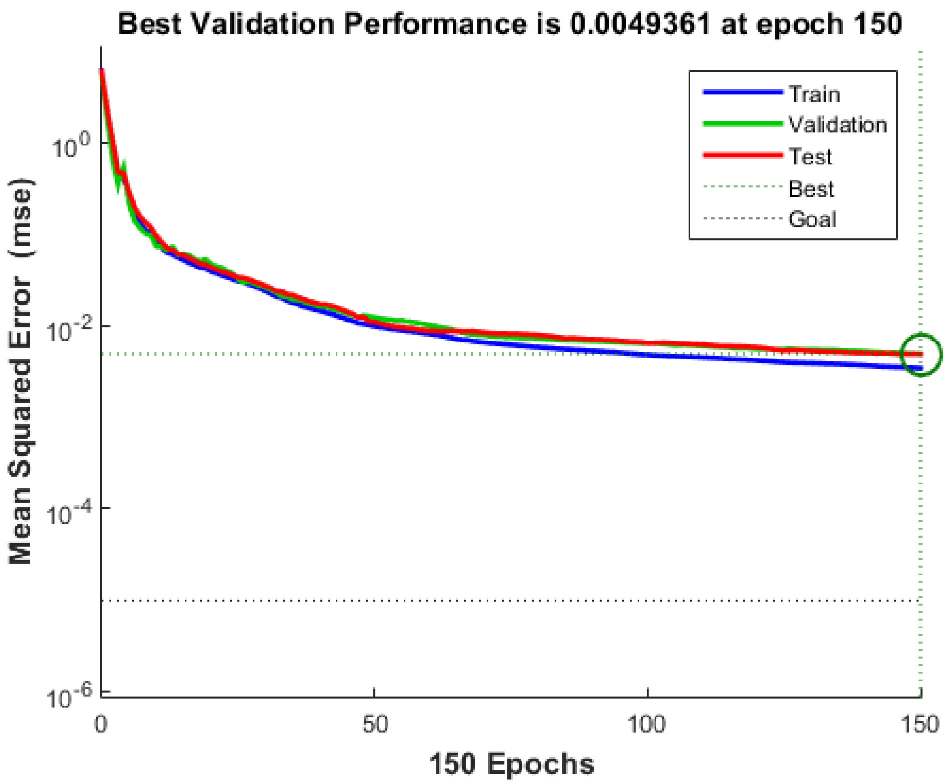


Figure 73 *The performance of training with SGD algorithm*

Moreover, the training performance depends not only the algorithm, but also the number of hidden neurons. In Figure 68, the MLP structure contains 8 neurons in the hidden layer. The number of the hidden neurons influences the performance and efficiency of the training as well. Based on the comparison of the training algorithms, BP algorithm is used to evaluate the impact with different number of hidden neurons.

Table 17 *Simulation results with different number of hidden neurons*

Hidden Neurons	Estimation Indexes		
	MAE	MAPE (%)	Time (S)
10	0.3259	9.33	5.3218
11	0.3176	8.65	5.4073
12	0.3186	7.31	5.7965
13	0.2989	5.96	5.9838
14	0.3041	6.02	6.0130
15	0.3137	6.94	6.1667
16	0.2773	3.92	6.3245
17	0.2651	2.34	6.5769
18	0.2468	1.97	6.4925
19	0.2342	1.89	4.9757
20	0.1353	1.14	3.7933
21	0.1097	1.05	3.2997
22	0.1151	1.47	3.6074
23	0.1246	1.44	4.0320
24	0.2299	1.75	4.1476
25	0.2331	1.94	4.2309

As shown in Table 17, different number of hidden neurons have different performance and duration time. When the number is twenty-one, both of the model accuracy and the operation time performance are the best.

6.4.3 Validity Check

Before the implementing the MLP in the rescheduling indicator, a validity check is necessary to evaluate the performance and accuracy of the MLP network with BP training algorithm and twenty-one hidden neurons.

There are 320 groups of historical data in the database, which are already confirmed with suitable balance factors by administrator. 300 groups of them, which are selected randomly are used for training and the rest 20 groups validate the trained ANN model. Four training algorithms train four ANN models with the same data (300 groups' data) and these models are validated with the same data (20 groups' data) as well.

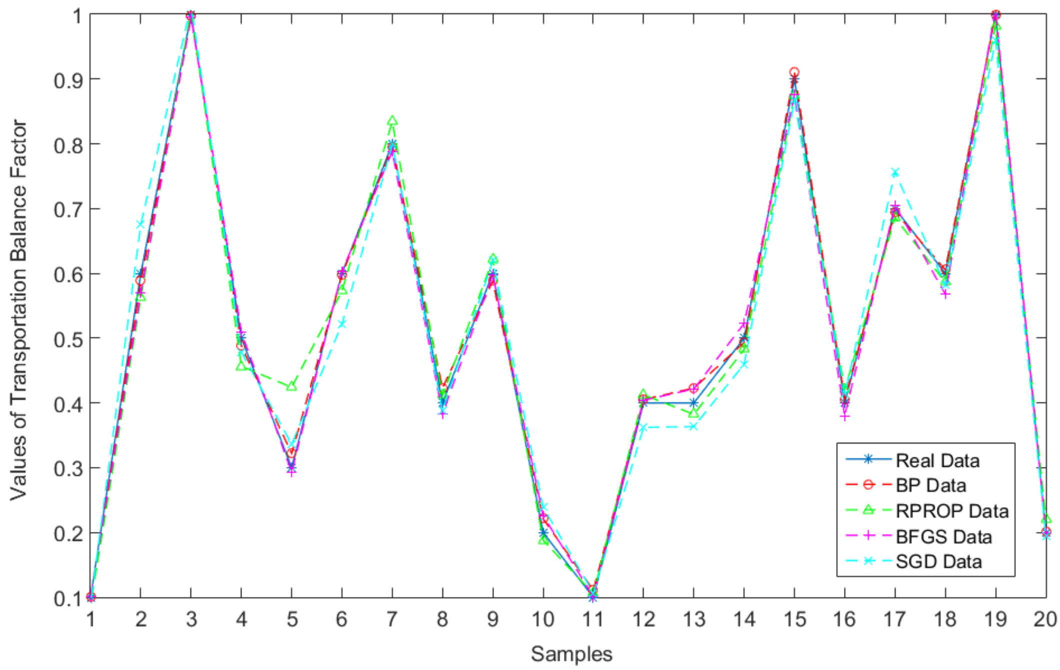


Figure 74 *Validation result of the trained ANN model*

Table 18 *Estimated Results of the Validation*

Training algorithm	MAPE (%)	Time (S)
BP	1.42	4.0265
RPROP	6.49	3.1687
BFGS	3.87	5.2344
SGD	6.91	4.6428

The result of the ANN model validation is given in Figure 74 and Table 18. The MAPE represents the difference between the forecast results and real data. And the training time of each algorithm is also shown in Table 18. The forecast result of the BP data is quite nearby the real data. That means the ANN model forecasts the balance factor effectively. Comparing with other algorithms, BP has the best performance (1.42% MAPE) and second short computational time (4.02 seconds). This performance can satisfy the requirement of the indicator.

According the result of validation, the ANN model can generate a suitable balance factor based on the status of the executing procedure. A correct balance factor is essential to guide the scheduler responding unexpected situations and this is proved in the chapter 7.

Chapter 7 System Test and Application

The system test for HWMS includes two parts: workflow design and workflow execution. The communication between HWMS and sub systems are the critical areas for testing. The reactions of various situations, which includes normal and unexpected events, will be presented in the workflow execution test. Finally, an application for life science analysis is applied to try out HWMS.

7.1 Workflow Design

In order to create a workflow, there are four steps: execution environment data preparing, processes definition, transportations definition and saving the workflow in database. These steps are demonstrate in the following part of this section. As shown in Figure 75, the home page of HWMS contains three parts: Planning Tool, Execution and LMMS. The Planning Tool and LMMS will be test to create a new workflow.

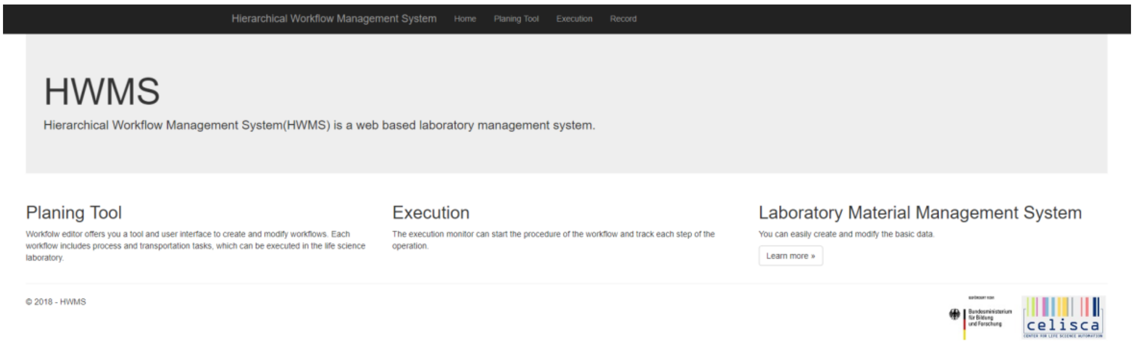


Figure 75 *Homepage of HWMS*

The Figure 31 and 30 in Chapter 4 show the user interface of LMMS and explain how to create and modify a location or container. However, it is no possible to edit the project, method, method schedule and method schedule detail in LMMS. These information should be updated based on the PCSs of automated stations and instruments. Furthermore, the automated systems can be integrated into HWMS, if they have the web service interface. There is no specific limitation for the automated systems to be adapted to HWMS.

In contrast, LMMS is compatible with all kinds of labwares. A container template represents a specific labware type, which is defined with dimension features, capacity, etc. as shown in Figure 76. Each labware has a unique barcode and the barcode information is stored in LMMS. It is helpful to identify and track the labware. Furthermore, the dimension details are also used to check the compatibility with the slot to avoid errors.

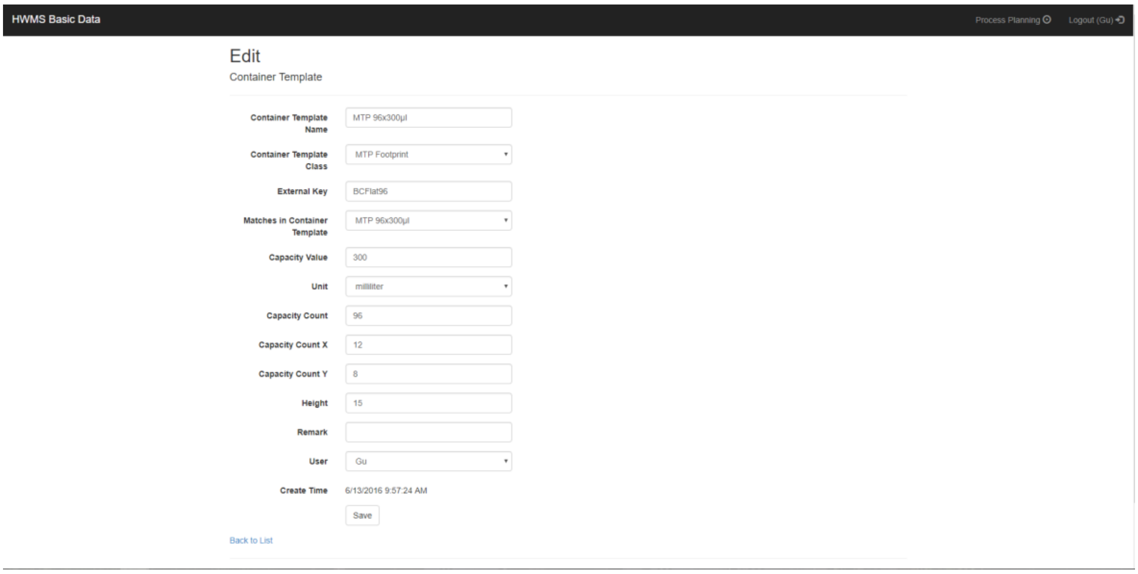


Figure 76 *Container template definition*

With the help of LMMS, all related environment data can be imported into HWMS. When the preparation is completed, the definition of process and transportation can be started via "Planning Tool".

In Figure 34 and 33, the two types of processes can be defined. Figure 77 shows the methods list with project named "FurtherLab" in the "Reformatter System". These methods are created in the PCS of this automated system and they are updated to HWMS by the data synchronization function as explained in chapter 4.

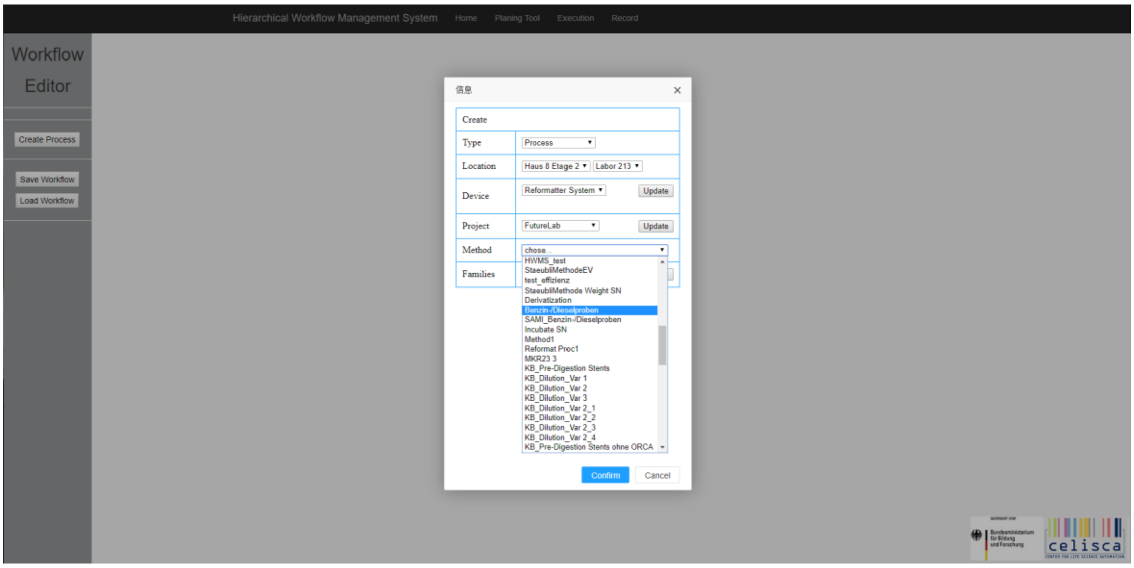


Figure 77 *Updated method list*

After the definition of processes, the transportations can be created by connecting the endpoints of processes. The various colours of the endpoints represent different types of containers. That means the endpoint with the same colour can be connected as a transportation as shown in Figure 78. During the transportation definition, the transportation type needs to be selected. If the "Dynamic" is chosen, the type will be decided by scheduler.

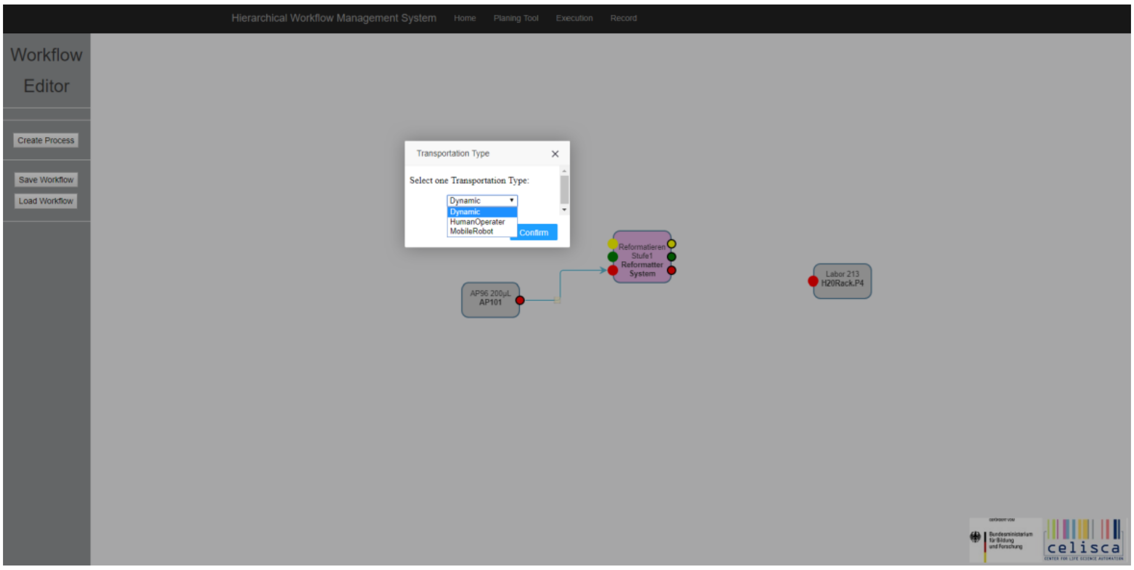


Figure 78 *Transportation definition*

When the design of a workflow is complete, it should be saved in HWMS with a name as shown in Figure 39. Then this workflow is ready for execution. Certainly, this workflow can be loaded into "Planning Tool" in anytime to make a correction or modification. After changing, the new workflow needs to be saved again.

7.2 Workflow Execution

As shown in Figure 38, the workflow needs to be loaded into WES. The execution procedure without errors and unexpected situations has been presented in Figure 44 in Chapter 4. The execution is operated in the integrated systems: PACS, MRMS and HACS.

In PACS, when the PCS of the automated system accepts the process task, it will start to execute the process as shown in Figure 79.Based on this system, the labwares and executing status can be tracked in real time. Furthermore, it offers the error information to HWMS when the process task is blocked or failed.

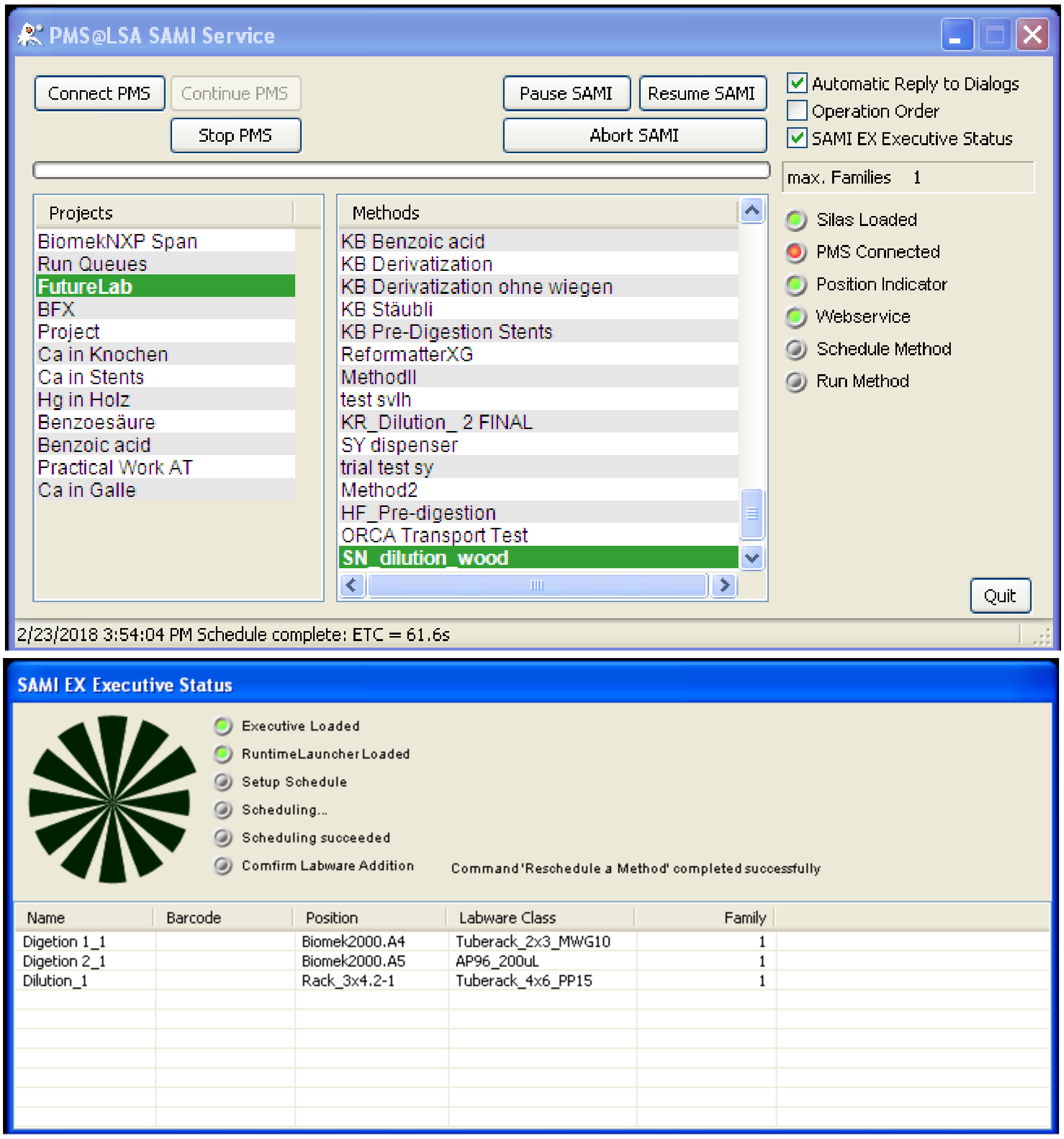
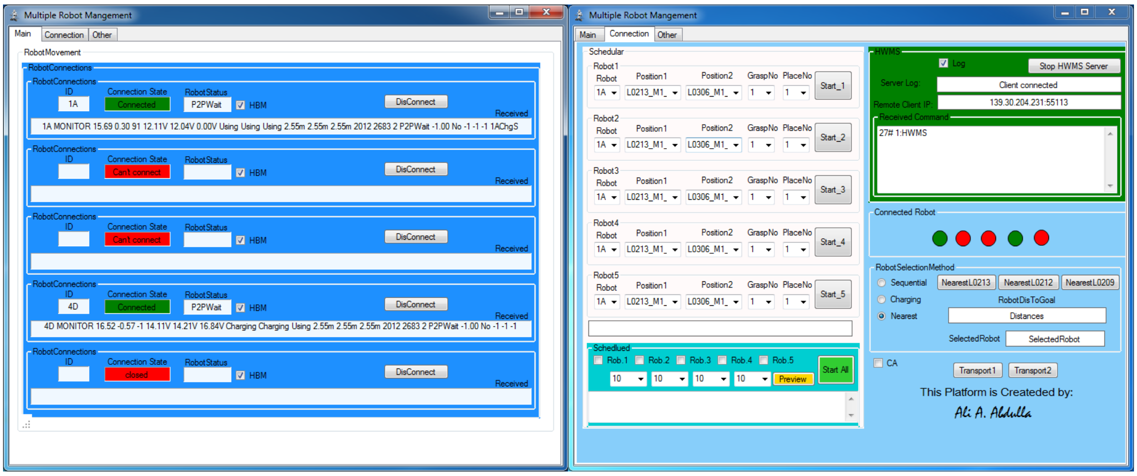


Figure 79 *Interface of the PCS in Reformatter*

As an effective robot management system, MRMS contains more than one mobile robots as shown in Figure 80(a). It collects the status information of multiple robots. Moreover, MRMS connects to the HWMS to offer the status information of the mobile robots as shown in Figure 80(b). On the other hand, the MRMS accepts the transportation tasks and allocates them to the mobile robot as well.



(a) (b)

Figure 80 *User interface of MRMS*

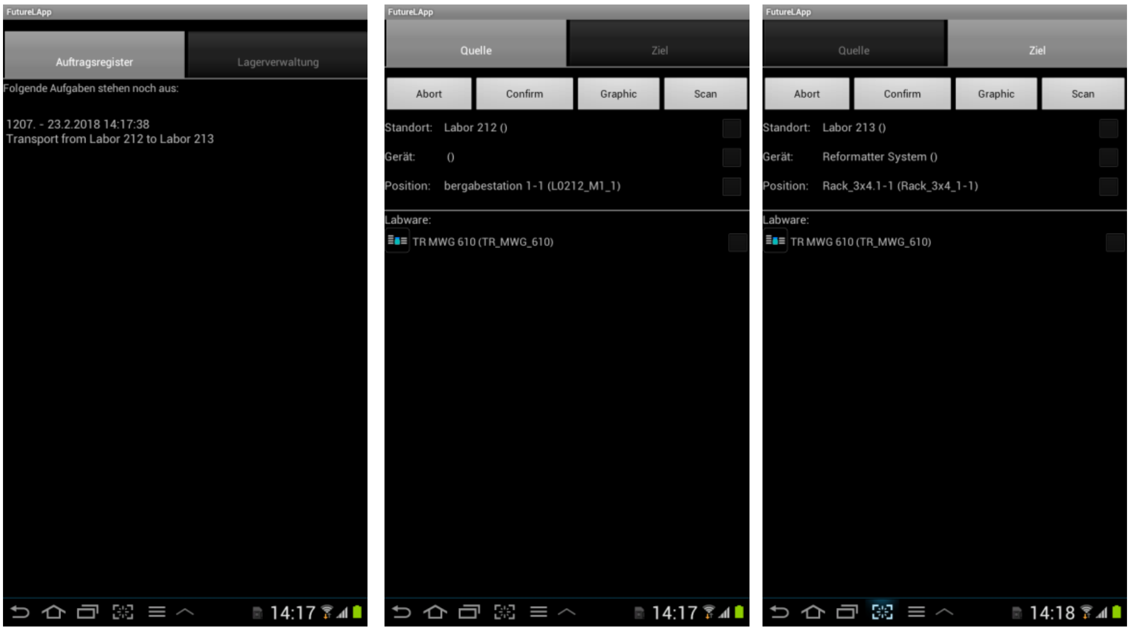
In the HWMS side, it is not necessary to design a user interface to control MRMS, because the administrator must not intervene the procedure of the robot transportation. The information from MRMS is stored in the database as shown in Figure 81.

ID	Name	Status	TaskID	Battery_1	Battery_2	Position_X	Position_Y	Position_Z	Time		.	Reconnection	ErrorMessage
1	Robot 1A	P2PWait	NULL	12.00	12.00	15.69	0.30	2	2018-02-26 1...	..	N	0	No
2	Robot 2B	NULL	NULL	NULL	NULL	14.81	-0.62	2	2017-12-18 1...	..	N	0	No
3	Robot 3C	NULL	NULL	NULL	NULL	NULL	NULL	4	2017-11-03 1...	..	N	0	NULL
4	Robot 4D	P2PWait	NULL	13.60	14.20	16.52	-0.57	2	2018-02-26 1...	..	N	0	No
5	Robot 5E	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	..	N	0	NULL

Figure 81 *Robot status in HWMS*

When the robot is online, its battery and position information is updated every 10 second to HWMS. This information is used to guide the distribution transportation tasks and the error handling.

When HWMS sends a transportation task to the HACS, the mobile device, which is carried by the assistant, receives the task as shown in Figure 82. The details of the source and target position, which are shown on the screen, guide the assistant to complete the task. Furthermore, the assistant can scan the barcode of the slots and labware to guarantee the correctness of the transportation.



(a) (b) (c)

Figure 82 *User interface of the mobile device in HACS (T task). (a): The Task list of a mobile device shows a Transportation task from Labor 212 to Labor 213 (b): Detailed source information of the transportation task (c): Detailed target information of the transportation task*

7.3 Workflow Scheduling

In HWMS, the result of scheduling is not visible. However, it is stored in the database as shown in Figure 83 and 82. The task table includes the process and transportation tasks of the workflows. It offers the task information to the scheduler and controls the execution procedure as well. The task status records the status of each task. If the task is a process task, it has a Process_ID. For the transportation task, it contains a Transportation_ID and a TaskProcedureStep_ID, which is used to link to the TaskProcedureStep table. The TaskProcedureStep table saves the scheduling result. The start time and end time explains the timetable of the execution. As explained before, the WES is an event-drive system. The time table is used to check the validity and evaluate the solution. The sequence row records the execution sequence of the transportation tasks. Combining with the transportation type information, the workflow can be executed completely. An additional element TaskSequence contains the ID of the dependent process. That means, when this process is finished, this transportation task should start immediately. In another word, this transportation task is waiting for the completion of its dependent process task.

ID	TaskStatus_...	UserAll_ID	Name	StartTime	EndTime	Process_ID	Transportation_ID	SourceSlot	TargetSlot	TransportContainer	WaitingSequence	TaskProcedureSteps_ID	CreateTime
1	1	2	NULL	NULL	NULL	240	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
2	1	2	NULL	NULL	NULL	241	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
3	1	2	NULL	NULL	NULL	242	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
4	1	2	NULL	NULL	NULL	243	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
5	1	2	NULL	NULL	NULL	244	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
6	1	2	NULL	NULL	NULL	245	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
7	1	2	NULL	NULL	NULL	246	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
8	1	2	NULL	NULL	NULL	247	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
9	1	2	NULL	NULL	NULL	248	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
10	1	2	NULL	NULL	NULL	249	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
11	1	2	NULL	NULL	NULL	250	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
12	1	2	243_240	NULL	NULL	NULL	30296	10040	10040	8	NULL	1	2018-02-23 ...
13	1	2	240_244	NULL	NULL	NULL	30297	10040	10040	8	NULL	2	2018-02-23 ...
14	1	2	241_240	NULL	NULL	NULL	30298	14	10030	28	NULL	3	2018-02-23 ...
15	1	2	240_242	NULL	NULL	NULL	30299	10070	10070	28	NULL	4	2018-02-23 ...
16	1	2	245_240	NULL	NULL	NULL	30300	10031	10031	30	NULL	5	2018-02-23 ...
17	1	2	240_246	NULL	NULL	NULL	30301	10069	10069	30	NULL	6	2018-02-23 ...
18	1	2	247_240	NULL	NULL	NULL	30302	10041	10041	31	NULL	7	2018-02-23 ...
19	1	2	240_248	NULL	NULL	NULL	30303	13	15	31	NULL	8	2018-02-23 ...
20	1	2	249_240	NULL	NULL	NULL	30304	10042	10042	32	NULL	9	2018-02-23 ...
21	1	2	240_250	NULL	NULL	NULL	30305	10066	10066	32	NULL	10	2018-02-23 ...
22	1	2	NULL	NULL	NULL	303	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
23	1	2	NULL	NULL	NULL	304	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
24	1	2	NULL	NULL	NULL	305	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
25	1	2	NULL	NULL	NULL	306	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
26	1	2	NULL	NULL	NULL	307	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
27	1	2	NULL	NULL	NULL	308	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
28	1	2	NULL	NULL	NULL	309	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
29	1	2	NULL	NULL	NULL	310	NULL	NULL	NULL	NULL	NULL	NULL	2018-02-23 ...
30	1	2	307_304	NULL	NULL	NULL	30312	10030	12	16	NULL	11	2018-02-23 ...
31	1	2	309_304	NULL	NULL	NULL	30313	10053	13	4	NULL	12	2018-02-23 ...
32	1	2	305_304	NULL	NULL	NULL	30314	10043	10027	23	NULL	13	2018-02-23 ...
33	1	2	304_303	NULL	NULL	NULL	30315	13	16	16	NULL	14	2018-02-23 ...
34	1	2	303_308	NULL	NULL	NULL	30316	16	10067	16	NULL	15	2018-02-23 ...
35	1	2	304_310	NULL	NULL	NULL	30317	12	10070	4	NULL	16	2018-02-23 ...
36	1	2	304_306	NULL	NULL	NULL	30318	10027	10030	23	NULL	17	2018-02-23 ...

Figure 83 *Task table for execution*

I...	StartTime	EndTime	Sequence	TaskSequence	StepSequen...	Name	Times	Source...	Target...	TaskType
1	2018-02-23 14:42:52.727	2018-02-23 14:44:52.727	9	NULL	5	243_240	1	243	240	0
2	2018-02-23 15:06:52.727	2018-02-23 15:08:52.727	12	NULL	4	240_244	1	240	244	0
3	2018-02-23 14:26:52.727	2018-02-23 14:32:52.727	6	NULL	9	241_240	1	241	240	0
4	2018-02-23 15:02:52.727	2018-02-23 15:04:52.727	11	NULL	6	240_242	1	240	242	0
5	2018-02-23 14:38:52.727	2018-02-23 14:40:52.727	8	NULL	7	245_240	1	245	240	0
6	2018-02-23 14:58:52.727	2018-02-23 15:00:52.727	10	240	NULL	240_246	1	240	246	0
7	2018-02-23 14:34:52.727	2018-02-23 14:36:52.727	7	NULL	3	247_240	1	247	240	0
8	2018-02-23 15:27:52.727	2018-02-23 15:34:52.727	13	NULL	15	240_248	1	240	248	1
9	2018-02-23 14:18:52.727	2018-02-23 14:20:52.727	5	NULL	13	249_240	1	249	240	0
10	2018-02-23 15:10:52.727	2018-02-23 15:12:52.727	14	NULL	2	240_250	1	240	250	0
11	2018-02-23 14:06:52.727	2018-02-23 14:08:52.727	0	NULL	NULL	307_304	1	307	304	0
12	2018-02-23 14:10:52.727	2018-02-23 14:12:52.727	1	NULL	11	309_304	1	309	304	0
13	2018-02-23 14:14:52.727	2018-02-23 14:16:52.727	2	NULL	12	305_304	1	305	304	0
14	2018-02-23 14:31:52.727	2018-02-23 14:47:52.727	3	304	NULL	304_303	1	304	303	1
15	2018-02-23 15:09:52.727	2018-02-23 15:25:52.727	4	303	NULL	303_308	1	303	308	1
16	2018-02-23 15:14:52.727	2018-02-23 15:16:52.727	15	NULL	10	304_310	1	304	310	0
17	2018-02-23 15:18:52.727	2018-02-23 15:20:52.727	16	NULL	16	304_306	1	304	306	0

Figure 84 *Task procedure step table for execution*

In this case, there are 36 tasks in the WES belongs to two workflows. Before the scheduling, each transportation task creates a task step in the TaskProcedureStep table, which is used to generate the executing sequence and conditions.

Because of the data structure, all tasks are independent and not involve the workflow level. It means the execution of each task is not related to the workflow information. The relationship of the processes and transportations in the workflow level are already considered in the scheduler. The execution schedule sequence satisfies the conditions and limitation.

As shown in Figure 85, there are two workflows in the WES and the schedule result proves that the scheduler is suitable for multiple workflows. In the automated workstation area, the green processes belong to one workflow and the blue processes come from another workflow. In WES, each process task is allocated an executing-taskID, which is shown as the number on the process block in various automated stations. All these processes are executed using the same workstations and resource of the laboratories. There is no collision or error for scheduling and executing.

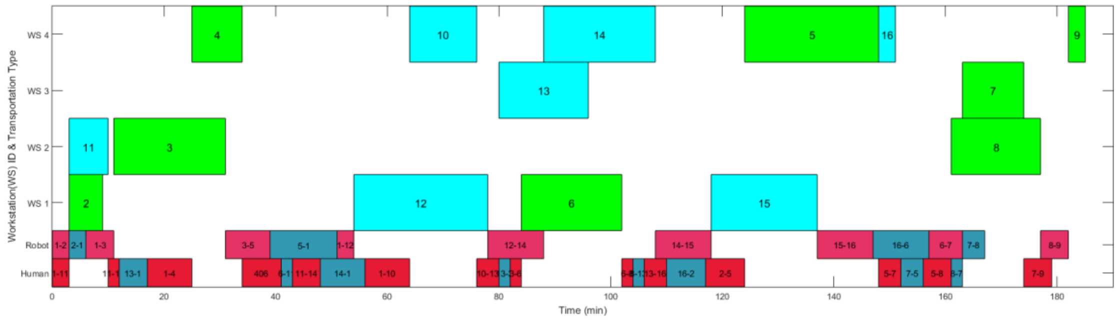


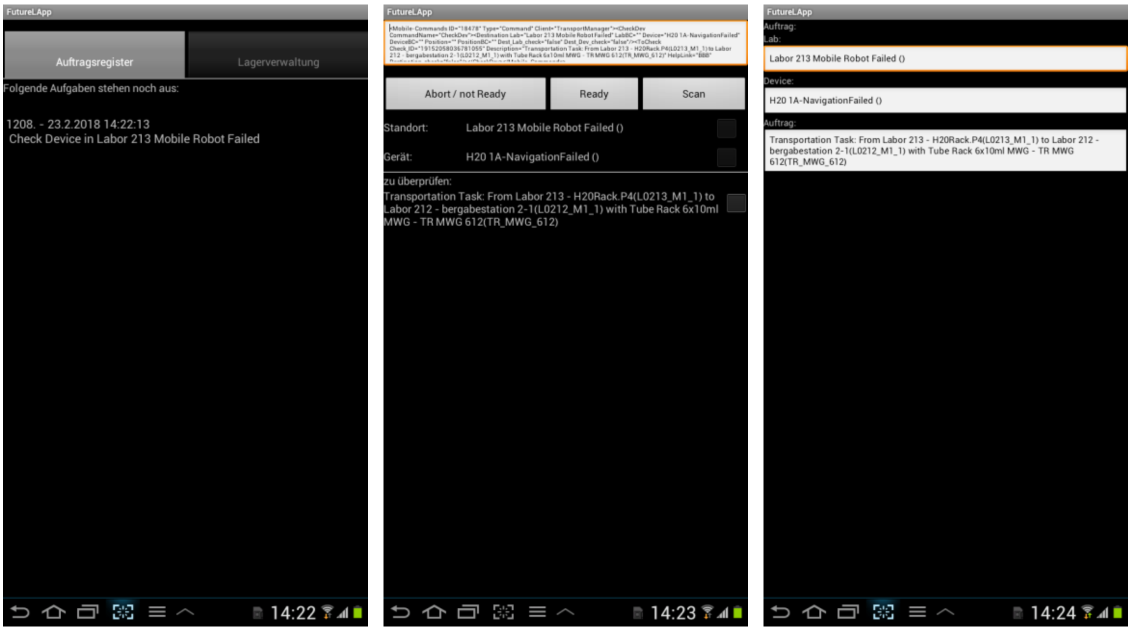
Figure 85 *Schedule result for dual workflows. Numbers on the process tasks located in WS 1-4 are the Process ID from 2 to 16. Process 2 – 9 belong to the workflow" Calcium in bone 4-2" and Process 10 – 16 belong to the workflow" Calcium in bone 2-2". The name of transportation task for Robot and Human is formed as "SourceID+ - +TargetID". For example, the transportation "3-5" is the transportation task from PorcessID 3 (SP_HNO3_Bone) to ProcessID 5 (HF_Digestion_Bone).*

7.4 Unexpected Situations Handling

When there are unexpected situations, HWMS has to work out them with two steps: a) requiring the human assistance to solve the problem; b) dynamic scheduling, if it is necessary. These two steps are explained respectively.

7.4.1 Human Resource for Errors

When there are errors in MRMS or PACS, HWMS can detect them immediately. The error message will be generated by HWMS and sent to HACS for the human assistance. Here is an example of the MRMS error during the operating for transportation task. As shown in Figure 86, the error message including the device name (Robot H20 1A), position (Labor213) and the error details. The details of the error show the error type and the task details. The assistant, who accepts the requirement, has to complete the transportation task instead of this robot. Then the confirm button can be clicked to notice HWMS that, the error is solved and the task is completed as well.



(a) (b) (c)

Figure 86 *User interface of the mobile device in HACS (error handling)*

Furthermore, there is a difference to generate the position information between the MRMS error and PACS error. The automated systems are located in the laboratories and the position information is the name of the laboratories. However, the raw position information of the mobile robot includes the two dimensional coordinate values and the floor number. This information needs to be transfer as a readable message. As shown in Figure 87, this floor is segmented into eight areas and corresponded to the coordinate values including the corridor. It helps the assistant to find the robot conveniently.

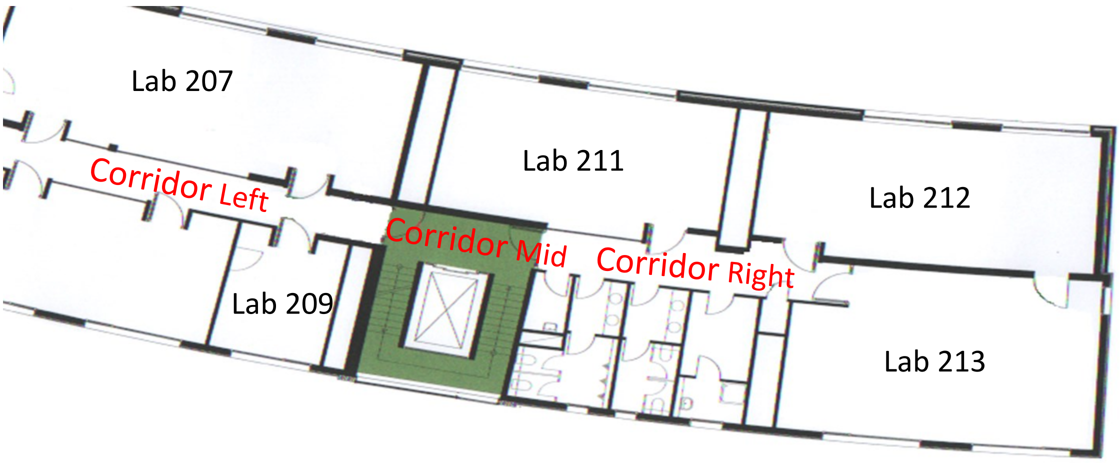


Figure 87 *Layout of the building in 2ed floor*

7.4.2 Dynamic Scheduling with Unexpected Situations

Dynamic scheduling updates the time schedule based on both the status of the transportation resource and the unexpected situations. The combinations of these two factors is almost unlimited as explained in Chapter 6. Four common scenarios are used to test the performance of the dynamic scheduling. The scheduling result is shown with a pair of Gantt charts to make a comparison between the initial and new solutions.

As shown in Figure 88(a), the result of the static scheduling is generated before executing. At 20^{th} minute of the execution, a mobile robot is going to charge station because of the low battery voltage. This resource changing triggers the rescheduling process and the trained ANN model generates a new balance factor 0.6 instead of the old one 0.4. That means, more transportation tasks will be distributed to the human. The new balance factor is used to generate a new time schedule as shown in Figure 88 (b). Compared to the original solution, the dynamic time schedule allocates the transportation task *3-5* to human assistants instead of mobile robots. The effect of this task shifting is explained as follows: (a) the process task *Analytik 1.d (5)* will be executed 10 minutes earlier than the old schedule. However, it will not influence other tasks; (b) there is no increase of the complete time; (c) the operation time of mobile robots decreases from 46 minutes to 28 minutes, and at the same time, the operation time of human assistances decreases from 9 minutes to 22 minutes. The complete transportation time decreases by 5 minutes, the purpose of the rescheduling to shift transportation task from human to robot is realized.

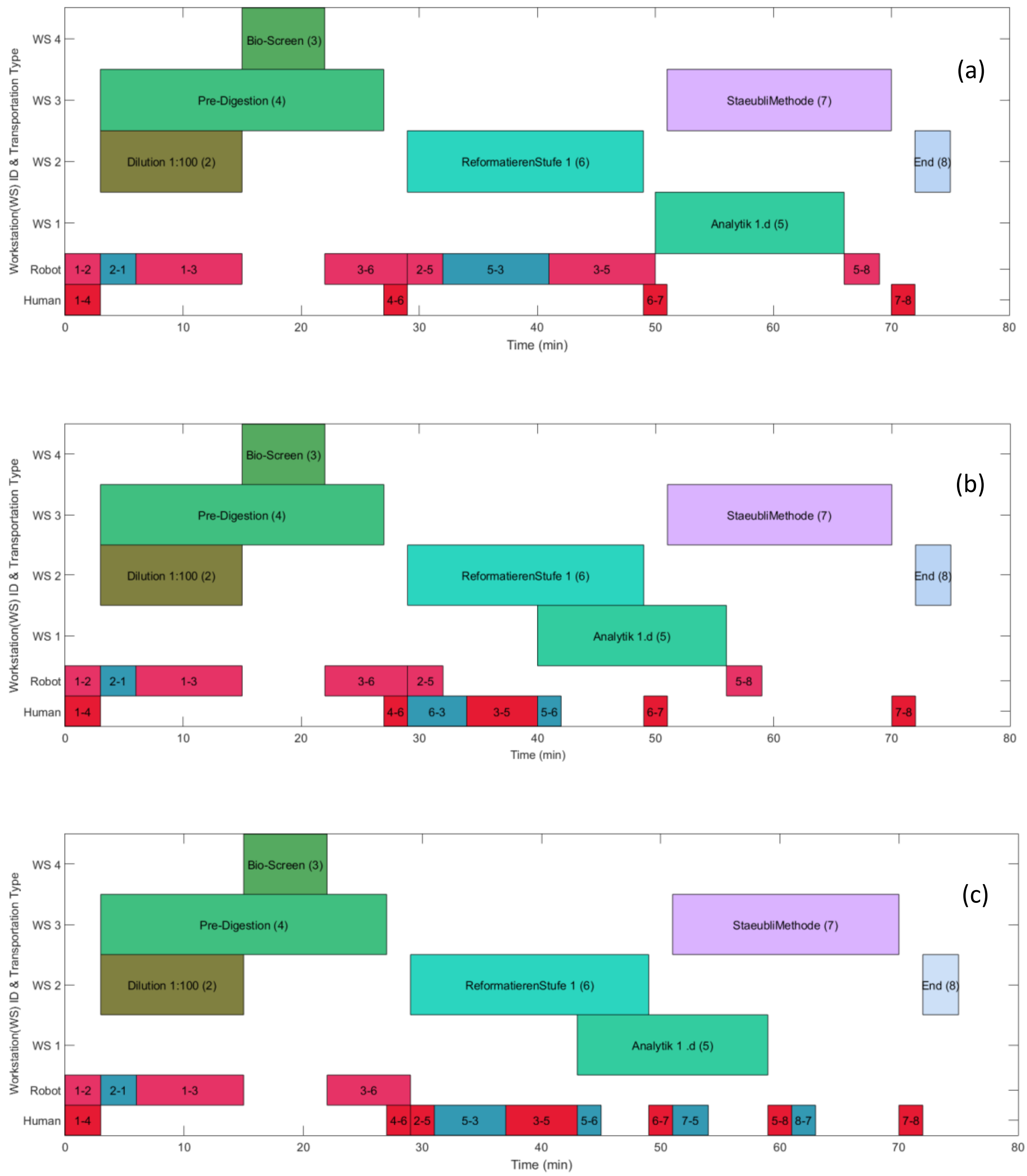


Figure 88 *Comparison of dynamic scheduling results 1. Process name and ID are shown on the Process block located in WS 1-4. The name of transportation task for Robot and Human is formed as "SourceID+ - +TargetID". For example, the transportation "3-5" is the transportation task from PorcessID 3 to ProcessID 6. (a): Static scheduling result (b) Dynamic scheduling result at 20th minute (c): Dynamic scheduling result at 28th minute*

When the transportation task 3-6 is completed at 28th minute, HWMS losses the connection to MRMS. The balance factor is changed to one. All transportation tasks are switch to human as shown in Figure 88(c).

As shown in Figure 89(a), the static scheduling result is generated for another workflow with the balance factor 0.5. At 10th minute, a new human assistance is online. At the same time, the human delay is lower than before. The rescheduling indicator changes the balance factor to 0.7. The transportation task 2-5 is switched to human as shown in Figure 89(b).

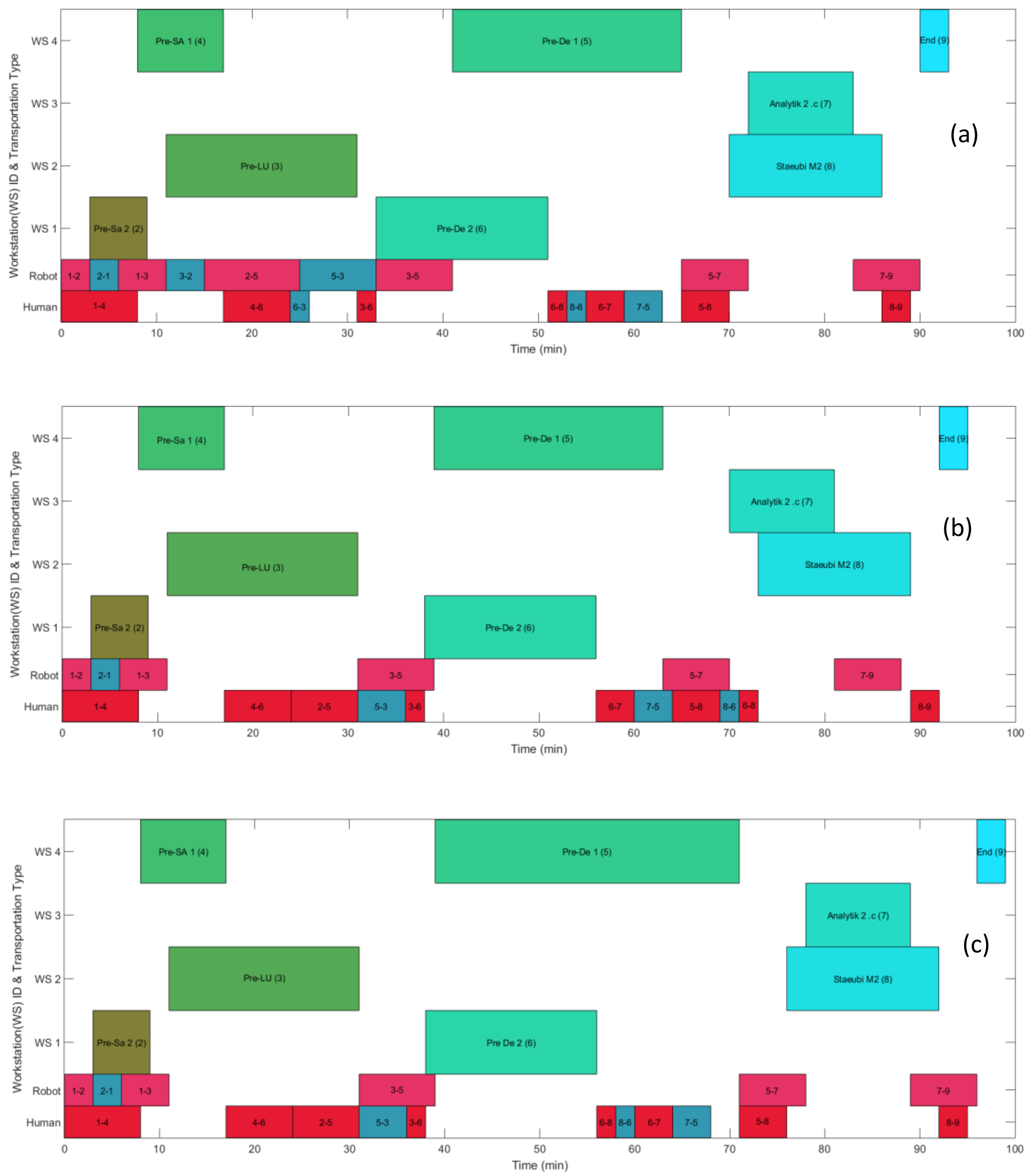


Figure 89 *Comparison of dynamic scheduling results 2. Process name and ID are shown on the Process block located in WS 1-4. The name of transportation task for Robot and Human is formed as "SourceID+ - +TargetID". For example, the transportation "3-5" is the transportation task from PorcessID 3 to ProcessID 5. (a): Static scheduling result (b) Dynamic scheduling result at 10th minute (c): Dynamic scheduling result at 45th minute*

At 45th minute, the workstation "Reformatter" has an error. HWMS creates and sends the assistance requirement to HACS. At the same time, the dynamic scheduling updates the time schedule as shown in Figure 89(c). It increases 8 minutes for the process Pre-De 1 with id 5. Based on the history, the errors can be solved in around 8 minutes usually. The balance factor is not changed and the tasks are allocated as before.

7.5 Application

Osteoporosis is the most common reason for a broken bone among the elderly. It may be due to the low bone mass and great bone loss [2], [119], [120]. As an effective method for a diagnosis of osteoporosis, the determination of the bone elemental composition is used to examine the stage and cause of this disease. Furthermore, there is a correlation between the calcium concentrations and osteoporosis [121]–[123].

There are various solutions for determining the elemental compositions of bone. As a multi-element technique, ICP-MS is powerful to determine elements both in the trace range and at higher concentrations [124], [125]. It plays the most important role in this application. The other operations and steps are used to solve and support the ICP-MS. The flowchart of the automated sample preparation and analysis is shown in Figure 90 . The “Transportation I” is the transport using the central system integrator and the “Transportation II” is the transport between workstations, which can be completed by human or mobile robots. The other steps are operated in three automated workstations/devices and can be integrated into four processes tasks. The workflow of this experiment is designed as shown in Figure 91. It has been executed using HWMS and the results are as expected. During the execution, there are several unexpected situation. HWMS can handle the errors and modify the time schedule as presented in the system test part.

With the help of HWMS, less human effort is required during the execution. The human resource is used only because of the errors and human transportation. The administrator can track the procedure of the executing in anytime via the execution monitor. The integrated LMMS handles various samples and labware efficiently without error. Thanks to the connection between HWMS and ADE, the user can also search for the results of the analysis easily in the execution recorder of HWMS.

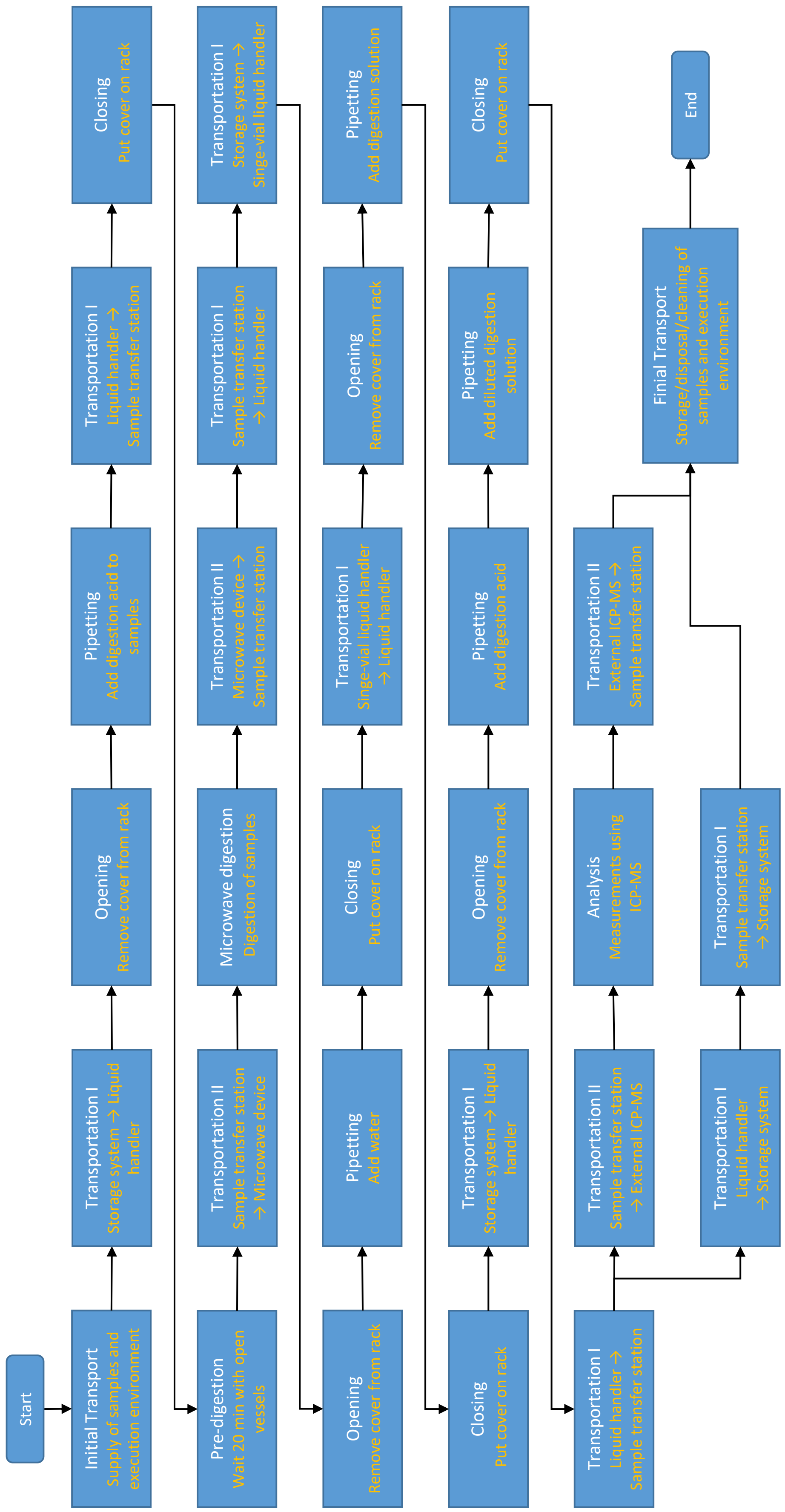


Figure 90 *Flowchart of the automated sample preparation and analysis (Redrawn from [2])*

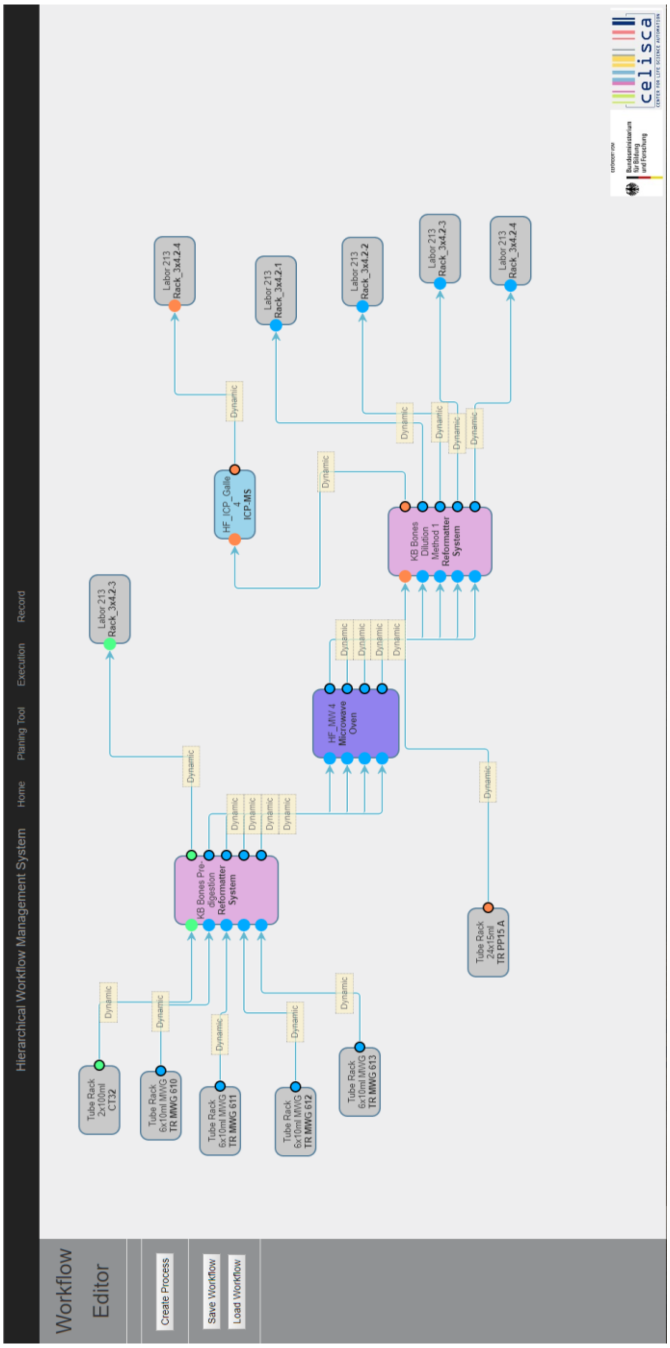


Figure 91 *Workflow designing in HWMS*

This workflow has been executed for 34 times. 20 of them are based on the real samples and others use water to simulate the execution procedures. There is one error appearing during the execution on the workstation “Reformatter”. HWMS detects the error immediately and sends message to request the human assistance. When the error is solved, the execution procedures are continued as normal.

Over 50 workflows are designed and executed to test the functionality and stability of HWMS and various sub-systems including HACS, MRMS, PCS, etc. Over 1000 simulations are used to check the validation and performance of the scheduler in both static and dynamic scenarios

Moreover, there are various workflows are designed and executed based on HWMS such as determination of mercury in wood materials and determination of the gall elemental composition, etc. Based on these applications, over 100 experiments are executed via HWMS in laboratories. Both human assistants and mobile robots are taking part in the executions. Two third of them arise unexpected situation: 2% process initialization error, 3% process execution failure, 16% transportation error and 79% transportation resources changes. In almost 50% experiments, the dynamic scheduling is triggered at least one time to reallocate the transportation tasks between human and robot. The rescheduling result responds correctly to the error and the change of transportation resources. In the meanwhile, the computation time for each rescheduling is no more than 20 seconds and it requires around 10 seconds on average. Furthermore, the errors of automated systems are also handled carefully. After the error is solved, the experiment continues automatically. Therefore, the HWMS enhances the efficiency and stability of the laboratories observably.

Chapter 8 Conclusion and Outlook

8.1 Conclusion

In this study, a new laboratory management system HWMS has been presented to schedule and handle the execution procedure for life science investigates. Comparing with the existed solution, HWMS focus on the laboratory environment description, automated system integration, execution resource distribution and unexpected situation handling.

In order to realize a full automatic experiment, a four layer system structure is designed for life science laboratory. With the help of PACS, all kinds of automated systems from various companies with different controllers can be integrated into HWMS. The robot and human mixed transportation solution is the bridge to connect two standalone automated systems. Furthermore, the human resource is used to handle errors during the execution as well. This structure is the base of HWMS. As the combination of processes and transportations, a workflow can represent a complex experiment, which is available to be executed in the laboratory.

As the resource of the experiment, the laboratory environment is modeled with container, location, slot and device. The LMMS is used to update and monitor the resource changing manually. These resource details are the base to define the processes and transportations for a complete workflow.

The WSP offers the planning tool for workflow definition and modification. The web based graphic user interface with ergonomic design can avoid human errors maximally. With the help of the data synchronization, the current activities' information of the automated systems can be updated in the HWMS on time and without any error.

The WES controls the automated systems, MRMS and HACS to execute the workflow following the time schedule, which is generated by scheduler. Based on the execution strategy, the workflow is completed safely and efficiently. Furthermore, the WES has the ability to handle errors during the execution. It improves the system stability and robustness against the unexpected situations.

The scheduling model, which includes the laboratory resources, activities and workflows, is defined to generate the execution sequence to guide the operation procedure of the workflows. Based on the strategy of scheduling and execution, the sequence of transportation tasks and the corresponding transportation type can represent a unique solution for the workflow execution. A time schedule of this solution is also confirmed at the same time. Moreover, the distribution of the transportation tasks is considered, when the transportation type is set as dynamic during the workflow designing. Based on the various status of MRMS and HACS, the task distribution can influence the execution efficiency conspicuous. The balance factor is the reflection of the status of both MRMS and HACS. It is used to generate the balance score. The sum of the workflow duration time and the balance score is presented as the fitness function to evaluate the solutions for execution.

In order to handle the unexpected situation during the execution, the dynamic scheduling is necessary in HWMS. Comparing with other solutions, the predictive-reactive scheduling is suitable for the laboratory environment. The dynamic scheduling result is used to update the time schedule to handle the resource changing and unexpected errors. Because of the different requirements of the scheduling before and during the execution, the scheduling engine should contain two mode: static and dynamic scheduling. The genetic algorithm and two modified genetic algorithm- GASA and GAPSO are used to search the optimal solution for the workflow based on the current status of the transportation resource. In order to evaluate the performance and features of the algorithms, an experiment is designed to simulate the scenarios of static and dynamic scheduling with two groups of parameters. Based on the comparing of the result quality, GASA is suitable for the static scheduling and GAPSO shows the best performance in the dynamic scheduling scenarios.

The rescheduling indicator is designed to trigger and guide the procedure of rescheduling to realize the dynamic scheduling. The trigger conditions are the delays and errors, which are brought by unexpected situations. The balance factor is related the transportation resource changing. There are 5 parameters describing the current status of the transportation. The ANN with MLP network is proved its performance for the balance factor forecasting. With the help of a series of simulation, the training method BP with 21 hidden neurons is adopted to train the MLP networks. The result of the validation shows that the ANN solution is suitable to generate the balance factor.

Finally, a series of system tests check the major functions of HWMS. The performance and efficiency are proved. Several unexpected situations are appearing to test the performance of the dynamic scheduling and the functionality of the error handling.

8.2 Outlook

Following the development of the robot technology, new mobile robot with high performance and flexibility can be adopted in the laboratory to complete all transportation tasks instead of human. Of cause, there is challenge to use robot to solve errors. However, the fully automated laboratory without human can be realized in the near further. It requires more operation details and higher performance of the system management system.

Furthermore, the HWMS needs more connection to the result of investigations. The major purpose of investigations in the laboratory is to develop and optimize the experiment procedures. If the HWMS has more ability to analysis the result of the execution, it is helpful for the development and optimization.

The HWMS is compatible with various automated systems and transportation management systems. When there are new systems or robots appearing in the laboratory, they can be integrated into HWMS without any obstacle. In contrast, the HWMS is also open for the new types of labware and slots. That means this system can be implemented in other laboratories.

Moreover, when the laboratories are not in the same building, the transportation management systems, which contain both indoor and outdoor transfer solutions, require more complex handling and task distribution. A more powerful and intelligent scheduler is necessary to allocate the transportation resource and handle more unexpected situation with completely reactive scheduling.

Bibliography

[1] "Definition of LIFE SCIENCE." [Online]. Available: http://www.merriam-webster.com/dictionary/life+science. [Accessed: 15-Sep-2016].

[2] H. Fleischer and K. Thurow, *Automation solutions for analytical measurements: concepts, and applications*. pp. 2-4, Wiley-VCH, Germany, 2017.

[3] E. Afgan *et al.*, "The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update," *Nucleic Acids Res.*, vol. 44, no. W1, pp. W3–W10, 2016.

[4] V. Simonyan, J. Goecks, and R. Mazumder, "Biocompute Objects-A Step towards Evaluation and Validation of Biomedical Scientific Computations," *PDA J. Pharm. Sci. Technol.*, vol. 71, no. 2, pp. 136–146, 2017.

[5] A. Budd *et al.*, "A quick guide for building a successful bioinformatics community," *PLoS Comput. Biol.*, vol. 11, no. 2, p. e1003972, 2015.

[6] J. Goecks, C. Eberhard, T. Too, Galaxy Team, A. Nekrutenko, and J. Taylor, "Web-based visual analysis for high-throughput genomics," *BMC Genomics*, vol. 14, pp. 397–407, 2013.

[7] R. Seethala and P. Fernandes, *Handbook of Drug Screening*. pp. 6-20, Boca Raton, United States: CRC Press, 2001.

[8] S. A. Haney, *High Content Screening: Science, Techniques and Applications*. pp. 25-42, Hoboken, United States: John Wiley & Sons, 2008.

[9] M. Zhang, B. J. Nelson, and R. Felder, *Life Science Automation Fundamentals and Applications*. pp. 37-52, Boston, United States: Artech House, 2007.

[10] K. W. Moore *et al.*, "Implementation of a High Specification Dual-Arm Robotic Platform to Meet Flexible Screening Needs," *J. Assoc. Lab. Autom.*, vol. 12, no. 2, pp. 115–123, 2007.

[11] S. Holzmüller-Laue, B. Göde, and K. Thurow, "Model-driven complex workflow automation for laboratories," in *2013 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 758–763, 2013.

[12] W. Bühl, J. Daniel, M. Höynck, W. Jänicke, and S. Szarowski, "Laboratory resource planning for quality control of pharmaceuticals," *Pharm. Ind.*, vol. 66, no. 11 A, pp. 1430–1434, 2004.

[13] S. Holzmüller-Laue and B. Göde, "Agile Business Process Management in Research Projects of Life Sciences," in *Perspectives in Business Informatics Research*, J. Grabis and M. Kirikova, Eds. Springer Berlin Heidelberg, pp. 336–344, 2011.

[14] H. Liu, N. Stoll, S. Junginger, and K. Thurow, "Mobile robotic transportation in laboratory automation: Multi-robot control, robot-door integration and robot-human interaction," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pp. 1033–1038, 2014.

[15] A. A. Abdulla, H. Liu, N. Stoll, and K. Thurow, "A robust method for elevator operation in semi-outdoor environment for mobile robot transportation system in life science laboratories," in *2016 IEEE 20th Jubilee International Conference on Intelligent Engineering Systems (INES)*, pp. 45–50, 2016.

[16] M. M. Ali, H. Liu, N. Stoll, and K. Thurow, "Multiple lab ware manipulation in life science laboratories using mobile robots," in *2016 17th International Conference on Mechatronics - Mechatronika (ME)*, pp. 1–7, 2016.

[17] G. Grüne, S. Lockemann, V. Kluy, and S. Meinhardt, *Business Process Management within Chemical and Pharmaceutical Industries: Markets, BPM*

Methodology and Process Examples. Springer Science & Business Media, Berlin, Germany, pp. 33-54, 2013.
[18] A. Costigliola, F. A. P. Ataíde, S. M. Vieira, and J. M. C. Sousa, “Simulation Model of a Quality Control Laboratory in Pharmaceutical Industry” *IFAC-Pap.*, vol. 50, no. 1, pp. 9014–9019, 2017.
[19] B. Gode, S. Holzmuller-Laue, K. Rimane, M. Y. Chow, and N. Stoll, “Laboratory Information Management Systems - An Approach as an Integration Platform within Flexible Laboratory Automation for Application in Life Sciences,” in *2007 IEEE International Conference on Automation Science and Engineering*, pp. 841–845, 2007.
[20] G. A. Gibbon, “A brief history of LIMS,” *Lab. Autom. Inf. Manag.*, vol. 32, no. 1, pp. 1–5, 1996.
[21] C. Paszko and E. Turner, *Laboratory Information Management Systems, Second Edition,*. CRC Press, Boca Raton, United States, pp.6-22, 2001.
[22] K. Thurow, B. Göde, U. Dingerdissen, and N. Stoll, “Laboratory Information Management Systems for Life Science Applications,” *Org. Process Res. Dev.*, vol. 8, no. 6, pp. 970–982, 2004.
[23] C. Stephan, M. Kohl, M. Turewicz, K. Podwojski, H. E. Meyer, and M. Eisenacher, “Using Laboratory Information Management Systems as central part of a proteomics data workflow,” *PROTEOMICS*, vol. 10, no. 6, pp. 1230–1249, 2010.
[24] A. Beteva *et al.*, “High-throughput sample handling and data collection at synchrotrons: embedding the ESRF into the high-throughput gene-to-structure pipeline,” *Acta Crystallogr. D Biol. Crystallogr.*, vol. 62, no. 10, pp. 1162–1169, 2006.
[25] O. Mathon *et al.*, “The time-resolved and extreme conditions XAS (TEXAS) facility at the European Synchrotron Radiation Facility: the general-purpose EXAFS bending-magnet beamline BM23,” *J. Synchrotron Radiat.*, vol. 22, no. 6, pp. 1548–1554, 2015.
[26] M. W. Bowler *et al.*, “MASSIF-1: a beamline dedicated to the fully automatic characterization and data collection from crystals of biological macromolecules,” *J. Synchrotron Radiat.*, vol. 22, no. 6, pp. 1540–1547, 2015.
[27] S. Delagenière *et al.*, “ISPyB: an information management system for synchrotron macromolecular crystallography,” *Bioinforma. Oxf. Engl.*, vol. 27, no. 22, pp. 3186–3192, 2011.
[28] A. De Maria Antolinos *et al.*, “ISPyB for BioSAXS, the gateway to user autonomy in solution scattering experiments,” *Acta Crystallogr. D Biol. Crystallogr.*, vol. 71, no. Pt 1, pp. 76–85, 2015.
[29] M. List *et al.*, “Efficient Sample Tracking With OpenLabFramework,” *Sci. Rep.*, vol. 4, 2014.
[30] Agaram Technologies, “QuaLIS LIMS.” [Online]. Available: https://www.agaramtech.com/. [Accessed: 15-Sep-2016].
[31] Autoscibe, “Matrix Gemini LIMS.” [Online]. Available: http://www.autoscribeinformatics.com/. [Accessed: 15-Sep-2016].
[32] Common Cents Systems, “ApolloLIMS.” [Online]. Available: http://www.apollolims.com. [Accessed: 15-Sep-2016].
[33] AgileBio SARL, “LabCollector LIMS,” *LabCollector*. [Online]. Available: http://labcollector.com/. [Accessed: 10-Apr-2018].
[33] W. M. P. van der Aalst, A. H. M. ter Hofstede, and M. Weske, “Business Process Management: A Survey,” in *Business Process Management*, W. M. P. van der Aalst and M. Weske, Eds. Springer Berlin Heidelberg, pp. 1–12, 2003.

[34] Ryan K.L. Ko, Stephen S.G. Lee, and Eng Wah Lee, “Business process management (BPM) standards: a survey,” *Bus. Process Manag. J.*, vol. 15, no. 5, pp. 744–791, 2009.
[35] A. Lux, J. Hess, and R. Herterich, “Business Process Management as Basis for Enterprise Management Systems,” in *2013 IEEE 15th Conference on Business Informatics*, pp. 350–355, 2013.
[37] L. Yang, “Customizing uEngine: An Open Source Business Process Management,” Master Thesis, Inje University, Kimhae, Korea, 2008.
[37] L. Yang, Y. Choi, and J. Jang, “API and Component Based Customization of an Open Source Business Process Management System: uEngine,” in *Fourth International Conference on Networked Computing and Advanced Information Management*, vol. 2, pp. 226–231, 2008.
[39] Advantys, “Independent community for business process management professionals,” *WorkflowGen*. [Online]. Available: https://www.workflowgen.com/. [Accessed: 27-Jun-2016].
[39] S. Holzmüller-Laue, B. Göde, H. Fleischer, and K. Thurow, “Improved Compliance by BPM-Driven Workflow Automation,” *J. Lab. Autom.*, p. 2211068214549626, 2014.
[41] D. Georgakopoulos, M. Hornick, and A. Sheth, “An overview of workflow management: From process modeling to workflow automation infrastructure,” *Distrib. Parallel Databases*, vol. 3, no. 2, pp. 119–153, 1995.
[41] J. Becker and M. zur Mühlen, “Towards a Classification Framework for Application Granularity in Workflow Management Systems,” in *Advanced Information Systems Engineering*, M. Jarke and A. Oberweis, Eds. Springer Berlin Heidelberg, pp. 411–416, 1999.
[42] T. McPhillips, S. Bowers, D. Zinn, and B. Ludäscher, “Scientific workflow design for mere mortals,” *Future Gener. Comput. Syst.*, vol. 25, no. 5, pp. 541–551, 2009.
[44] B. Ludäscher *et al.*, “Scientific process automation and workflow management,” *Sci. Data Manag. Chall. Exist. Technol. Deploy. Comput. Sci. Ser.*, vol. 230, pp. 476–508, 2009.
[44] W. van der Aalst and K. M. van Hee, *Workflow Management: Models, Methods, and Systems*. MIT Press, Cambridge, United States, 2004.
[45] K. Kochut *et al.*, “IntelliGEN: A Distributed Workflow System for Discovering Protein-Protein Interactions,” *Distrib. Parallel Databases*, vol. 13, no. 1, pp. 43–72, 2003.
[46] K. G. Achilleos, C. C. Kannas, C. A. Nicolaou, C. S. Pattichis, and V. J. Promponas, “Open source workflow systems in life sciences informatics,” in *2012 IEEE 12th International Conference on Bioinformatics Bioengineering (BIBE)*, pp. 552–558, 2012.
[47] B. Gabor and B. Kemme, “Exp-WF:Workflow Support for Laboratory Information Systems,” in *22nd International Conference on Data Engineering Workshops (ICDEW’06)*, pp. 69–69, 2006.
[48] A. Y. Ng and M. Jordan, “PEGASUS: A Policy Search Method for Large MDPs and POMDPs,” in *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, San Francisco, CA, USA, pp. 406–415, 2000.
[49] U. Kang, C. E. Tsourakakis, and C. Faloutsos, “PEGASUS: A Peta-Scale Graph Mining System Implementation and Observations,” in *2009 Ninth IEEE International Conference on Data Mining*, pp. 229–238, 2009.
[50] E. Deelman *et al.*, “Mapping Abstract Complex Workflows onto Grid Environments,” *J. Grid Comput.*, vol. 1, no. 1, pp. 25–39, 2013.

[51] E. Deelman *et al.*, "GriPhyN and LIGO, building a virtual data Grid for gravitational wave scientists," in *11th IEEE International Symposium on High Performance Distributed Computing, 2002. HPDC-11 2002. Proceedings*, pp. 225–234, 2002.

[52] E. Deelman *et al.*, "Pegasus, a workflow management system for science automation," *Future Gener. Comput. Syst.*, vol. 46, pp. 17–35, 2015.

[54] "Momentum Integration Software." [Online]. Available: https://www.thermofisher.com/order/catalog/product/MOMENTUM. [Accessed: 14-Mar-2018].

[55] "Orbitor^{TM} RS Microplate Mover." [Online]. Available: https://www.thermofisher.com/order/catalog/product/F01743. [Accessed: 14-Mar-2018].

[56] S. Neubert, B. Göde, X. Gu, N. Stoll, and K. Thurow, "Potential of Laboratory Execution Systems (LESs) to Simplify the Application of Business Process Management Systems (BPMSs) in Laboratory Automation," *SLAS Technol. Transl. Life Sci. Innov.*, vol. 22, no. 2, pp. 206–216, Apr. 2017.

[57] X. Gu, S. Neubert, N. Stoll, and K. Thurow, "Intelligent scheduling method for life science automation systems," in *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2016, pp. 156–161.

[53] S. Sathasivam and W. A. T. W. Abdullah, "Logic Learning in Hopfield Networks," *ArXiv08044075 Cs*, 2008.

[54] K. Gurney, *An introduction to neural networks*. London, England: UCL Press, pp. 8-24, 1997.

[55] R. Rojas, *Neural Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg, Berlin, Germany, pp. 21-46, 1996.

[56] T. M. Willems and L. E. M. W. Brandts, "Implementing heuristics as an optimization criterion in neural networks for job-shop scheduling," *J. Intell. Manuf.*, vol. 6, no. 6, pp. 377–387, 1995.

[57] N. Fnaiech, H. Hammami, A. Yahyaoui, C. Varnier, F. Fnaiech, and N. Zerhouni, "New Hopfield Neural Network for joint Job Shop Scheduling of production and maintenance," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, pp. 5535–5541, 2012.

[58] A. Yahyaoui, N. Fnaiech, and F. Fnaiech, "A Suitable Initialization Procedure for Speeding a Neural Network Job-Shop Scheduling," *IEEE Trans. Ind. Electron.*, vol. 58, no. 3, pp. 1052–1060, 2011.

[59] D. Bertsimas and J. Tsitsiklis, "Simulated Annealing," *Stat. Sci.*, vol. 8, no. 1, pp. 10–15, 1993.

[60] C.-R. Hwang, "Simulated annealing: Theory and applications," *Acta Appl. Math.*, vol. 12, no. 1, pp. 108–111, 1987.

[66] S. P. Brooks and B. J. T. Morgan, "Optimization Using Simulated Annealing," *J. R. Stat. Soc. Ser. Stat.*, vol. 44, no. 2, pp. 241–257, 1995.

[62] I. O. Bohachevsky, M. E. Johnson, and M. L. Stein, "Generalized Simulated Annealing for Function Optimization," *Technometrics*, vol. 28, no. 3, pp. 209–217, 1986.

[63] J. Zhang, "Simulated annealing: in mathematical global optimization computation, hybrid with local or global search, and practical applications in crystallography and molecular modelling," *ArXiv13086220 Phys.*, 2013.

[64] X. Yu, D. Zhan, L. Nie, and X. Xu, "A Novel Genetic Simulated Annealing Algorithm for the Resource-Constrained Project Scheduling Problem," in

International Workshop on Intelligent Systems and Applications, 2009. ISA 2009, pp. 1–4, 2009.

[65] P. P. Das and S. Acharyya, "Simulated annealing variants for solving resource Constrained Project Scheduling Problem: A comparative study," in *2011 14th International Conference on Computer and Information Technology (ICCIT)*, pp. 469–474, 2011.

[66] A. Sprecher, R. Kolisch, and A. Drexl, "Semi-active, active, and non-delay schedules for the resource-constrained project scheduling problem," *Eur. J. Oper. Res.*, vol. 80, no. 1, pp. 94–102, 1995.

[67] E. Demeulemeester and W. Herroelen, "A Branch-and-Bound Procedure for the Multiple Resource-Constrained Project Scheduling Problem," *Manag. Sci.*, vol. 38, no. 12, pp. 1803–1818, 1992.

[68] R. Kolisch, "Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation," *Eur. J. Oper. Res.*, vol. 90, no. 2, pp. 320–333, 1996.

[69] P. Brucker, A. Drexl, R. Möhring, K. Neumann, and E. Pesch, "Resource-constrained project scheduling: Notation, classification, models, and methods," *Eur. J. Oper. Res.*, vol. 112, no. 1, pp. 3–41, 1999.

[70] K. Bouleimen and H. Lecocq, "A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version," *Eur. J. Oper. Res.*, vol. 149, no. 2, pp. 268–281, 2003.

[71] M. Mitchell, *An introduction to genetic algorithms*, 7. print. Cambridge, United States, pp. 4-28, 2001.

[72] D. E. Goldberg and J. H. Holland, "Genetic Algorithms and Machine Learning," *Mach. Learn.*, vol. 3, no. 2–3, pp. 95–99, 1988.

[73] Z. Michalewicz, "GAs: What Are They?," in *Genetic Algorithms + Data Structures = Evolution Programs*, Springer Berlin Heidelberg, pp. 13–30, 1994.

[74] T. Back, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, Oxford, England, pp. 15-42, 1996.

[75] L. M. Schmitt, "Theory of Genetic Algorithms II: models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling," *Theor. Comput. Sci.*, vol. 310, no. 1, pp. 181–231, 2004.

[76] L. M. Schmitt, "Theory of genetic algorithms," *Theor. Comput. Sci.*, vol. 259, no. 1, pp. 1–61, 2001.

[77] S. Hartmann, "A competitive genetic algorithm for resource-constrained project scheduling," *Nav. Res. Logist. NRL*, vol. 45, no. 7, pp. 733–750, 1998.

[83] S. Hartmann, *Scheduling Medical Research Experiments – An Application of Project Scheduling Methods*. 1997.

[79] G. Zhang, L. Gao, and Y. Shi, "An effective genetic algorithm for the flexible job-shop scheduling problem," *Expert Syst. Appl.*, vol. 38, no. 4, pp. 3563–3573, 2011.

[80] F. Pezzella, G. Morganti, and G. Ciaschetti, "A genetic algorithm for the Flexible Job-shop Scheduling Problem," *Comput. Oper. Res.*, vol. 35, no. 10, pp. 3202–3212, 2008.

[81] R. Cheng, M. Gen, and Y. Tsujimura, "A tutorial survey of job-shop scheduling problems using genetic algorithms—I. representation," *Comput. Ind. Eng.*, vol. 30, no. 4, pp. 983–997, 1996.

[82] L.-N. Xing, Y.-W. Chen, P. Wang, Q.-S. Zhao, and J. Xiong, "A Knowledge-Based Ant Colony Optimization for Flexible Job Shop Scheduling Problems," *Appl. Soft Comput.*, vol. 10, no. 3, pp. 888–896, 2010.

[83] P. Fattahi, M. S. Mehrabad, and F. Jolai, "Mathematical modeling and heuristic approaches to flexible job shop scheduling problems," *J. Intell. Manuf.*, vol. 18, no. 3, pp. 331–342, 2007.
[84] W. Xia and Z. Wu, "An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems," *Comput. Ind. Eng.*, vol. 48, no. 2, pp. 409–425, 2005.
[85] P. Brandimarte, "Routing and scheduling in a flexible job shop by tabu search," *Ann. Oper. Res.*, vol. 41, no. 3, pp. 157–183, 1993.
[86] L. De Giovanni and F. Pezzella, "An Improved Genetic Algorithm for the Distributed and Flexible Job-shop Scheduling problem," *Eur. J. Oper. Res.*, vol. 200, no. 2, pp. 395–408, 2010.
[87] M. Gen, J. Gao, and L. Lin, "Multistage-Based Genetic Algorithm for Flexible Job-Shop Scheduling Problem," in *Intelligent and Evolutionary Systems*, M. Gen, D. Green, O. Katai, B. McKay, A. Namatame, R. A. Sarker, and B.-T. Zhang, Eds. Springer Berlin Heidelberg, pp. 183–196, 2009.
[93] S. Neubert *et al.*, "Architecture for a Combined Mobile Robot and Human Operator Transportation Solution for the Hierarchical Life Science Automation," presented at the 14th International Conference on Informatics in Control, Automation and Robotics, 2018, pp. 41–49.
[94] A. A. Abdulla, H. Liu, N. Stoll, and K. Thurow, "A Secure Automated Elevator Management System and Pressure Sensor based Floor Estimation for Indoor Mobile Robot Transportation," *Adv. Sci. Technol. Eng. Syst. J.*, vol. 2, no. 3, pp. 1599–1608, Aug. 2017.
[95] S. Neubert *et al.*, "Architecture for a Combined Mobile Robot and Human Operator Transportation Solution for the Hierarchical Life Science Automation," presented at the 14th International Conference on Informatics in Control, Automation and Robotics, 2018, pp. 41–49.
[96] Microsoft, "Windows Forms." [Online]. Available: https://docs.microsoft.com/en-us/dotnet/framework/winforms/. [Accessed: 14-Mar-2018].
[88] C. Cabrera *et al.*, "Dynamically Optimizing Experiment Schedules of a Laboratory Robot System with Simulated Annealing," *J. Lab. Autom.*, vol. 19, no. 6, pp. 517–527, 2014.
[98] J. Xiao, L. J. Osterweil, Q. Wang, and M. Li, "Dynamic Scheduling in Systems with Complex Resource Allocation Requirements."
[90] D. Ouelhadj and S. Petrovic, "A survey of dynamic scheduling in manufacturing systems," *J. Sched.*, vol. 12, no. 4, p. 417, 2009.
[91] D. M. Markowitz and L. M. Wein, "Heavy Traffic Analysis of Dynamic Cyclic Policies: A Unified Treatment of the Single Machine Scheduling Problem," *Oper Res*, vol. 49, no. 2, pp. 246–270, 2001.
[92] G. E. Vieira, J. W. Herrmann, and E. Lin, "Rescheduling Manufacturing Systems: A Framework of Strategies, Policies, and Methods," *J Sched.*, vol. 6, no. 1, pp. 39–62, 2003.
[93] L. Church and Re. Uzsoy, "Analysis of periodic and event-driven rescheduling policies in dynamic shops," *Int. J. Comput. Integr. Manuf.*, vol. 5, no. 3, pp. 153–163, 1992.
[94] G. E. Vieira, J. W. Herrmann, and E. Lin, "Predicting the performance of rescheduling strategies for parallel machine systems," *J. Manuf. Syst.*, vol. 19, no. 4, pp. 256–266, 2000.
[95] I. Sabuncuoglu and S. Karabuk, "Rescheduling frequency in an FMS with uncertain processing times and unreliable machines," *J. Manuf. Syst.*, vol. 18, no. 4, pp. 268–283, 1999.

[96] J. F. Kennedy, J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*. Morgan Kaufmann, San Francisco, United States, pp. 12-36, 2001.

[97] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *, IEEE International Conference on Neural Networks, 1995. Proceedings*, vol. 4, pp. 1942–1948 vol.4, 1995.

[98] P. E Konstantinos, *Particle Swarm Optimization and Intelligence: Advances and Applications: Advances and Applications*. IGI Global, Hershey, United State, pp. 25-41, 2010.

[99] A. E. Olsson, *Particle Swarm Optimization: Theory, Techniques and Applications*. Nova Science Publishers, New York, United States, pp.3-18, 2011.

[100] Q. Bai, “Analysis of Particle Swarm Optimization Algorithm,” *Comput. Inf. Sci.*, vol. 3, no. 1, p. 180, 2010.

[110] D. P. Rini, S. M. Shamsuddin, and S. S. Yuhaniz, “Particle swarm optimization: technique, system and challenges,” *Int. J. Comput. Appl.*, vol. 14, no. 1, pp. 19–26, 2011.

[111] G. E. Vieira, J. W. Herrmann, and E. Lin, “Analytical models to predict the performance of a single-machine system under periodic and event-driven rescheduling strategies,” *Int. J. Prod. Res.*, vol. 38, no. 8, pp. 1899–1915, May 2000.

[112] E. Vorberg, H. Fleischer, S. Junginger, N. Stoll, and K. Thurow, “Automated sample preparation for mercury analysis in wood materials,” *IET Sci. Meas. Technol.*, vol. 10, no. 5, pp. 398–404, 2016.

[113] W. Wen, “A dynamic and automatic traffic light control expert system for solving the road congestion problem,” *Expert Syst. Appl.*, vol. 34, no. 4, pp. 2370–2381, May 2008.

[114] N. P. Padhy, “Unit commitment using hybrid models: a comparative study for dynamic programming, expert system, fuzzy system and genetic algorithms,” *Int. J. Electr. Power Energy Syst.*, vol. 23, no. 8, pp. 827–836, Nov. 2001.

[115] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence/Book and Disk*, Har/Dskt edition. Englewood Ciffs, NJ: Prentice Hall, 1991.

[116] R. Hecht-Nielsen, “Theory of the backpropagation neural network,” in *International 1989 Joint Conference on Neural Networks*, 1989, pp. 593–605 vol.1.

[117] A. Gibson and J. Patterson, *Deep Learning: The Definitive Guide: A Practitioner’s Approach*, 1st ed. Beijing Boston Farnham Sebastopol Tokyo: O’Reilly UK Ltd., 2017.

[118] J. Heaton, *Artificial Intelligence for Humans, Volume 3: Deep Learning and Neural Networks*. St. Louis, MO: CreateSpace Independent Publishing Platform, 2015.

[119] T. R. Helliwell *et al.*, “Elemental analysis of femoral bone from patients with fractured neck of femur or osteoarthrosis,” *Bone*, vol. 18, no. 2, pp. 151–157, Feb. 1996.

[120] D. Faibish, S. M. Ott, and A. L. Boskey, “Mineral Changes in Osteoporosis A Review,” *Clin. Orthop.*, vol. 443, pp. 28–38, Feb. 2006.

[121] N. B. Roberts, H. P. J. Walsh, L. Klenerman, S. A. Kelly, and T. R. Helliwell, “Determination of elements in human femoral bone using inductively coupled plasma atomic emission spectrometry and inductively coupled plasma mass spectrometry,” *J. Anal. At. Spectrom.*, vol. 11, no. 2, pp. 133–138, Jan. 1996.

[122] J. M. Burnell, D. J. Baylink, C. H. Chestnut, M. W. Mathews, and E. J. Teubner, “Bone matrix and mineral abnormalities in postmenopausal osteoporosis,” *Metabolism.*, vol. 31, no. 11, pp. 1113–1120, Nov. 1982.

[123] R. Lappalainen, M. Knuuttila, S. Lammi, and E. M. Alhava, “Fluoride content related to the elemental composition, mineral density and strength of bone in healthy and chronically diseased persons,” *J. Chronic Dis.*, vol. 36, no. 10, pp. 707–713, 1983.
[124] S. Ashoka, B. M. Peake, G. Bremner, K. J. Hageman, and M. R. Reid, “Comparison of digestion methods for ICP-MS determination of trace elements in fish tissues,” *Anal. Chim. Acta*, vol. 653, no. 2, pp. 191–199, Oct. 2009.
[125] J. S. Becker, S. F. Boulyga, C. Pickhardt, J. Becker, S. Buddrus, and M. Przybylski, “Determination of phosphorus in small amounts of protein samples by ICP-MS,” *Anal. Bioanal. Chem.*, vol. 375, no. 4, pp. 561–566, Feb. 2003.

Declaration

This dissertation 'Hierarchical Workflow Management System for Life Science Applications' is a presentation of my original research work. Wherever contributions of others are involved, every effort is made to indicate this clearly, with due reference to the literature, and acknowledgement of collaborative research and discussions. The work of this dissertation has been done by me under the guidance of Prof. Dr.-Ing. habil. Kerstin Thurow and Prof. Dr. -Ing. Norbert Stoll, at the University of Rostock, Germany. Also the dissertation has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Xiangyu Gu

Rostock, 20.04.2018

Curriculum Vitae

Personal Details

Name:	Xiangyu Gu
Date of birth:	25.01.1987
Email address:	gxyu125@gmail.com
Nationality:	China
Address:	Erich-Weinert-Str. 12 18059, Rostock

Education

2015- 2018	Ph.D student at the University of Rostock Research group "LSA-Information Technologies" under the supervisory of Prof. Dr.-Ing. habil. Kerstin Thurow and Prof. Dr.-Ing. Norbert Stoll. Research topic " Hierarchical Workflow Management System for Life Science Applications"
2011- 2014	M.Sc. Electronic Engineering at the University of Rostock Master thesis was implemented under the supervisory of Prof. Dr.-Ing. habil. Kerstin Thurow with the topic "Real-Time Data Transformation for the 4D Simulation"
2005- 2009	B.Sc. Automation at University of Electronic Science and Technology of China

Work Experience

2014	Quality Engineer Volkswagen Automatic Transmission (Tianjin) Co., Ltd, China

List of Publications

- X. Gu, S. Neubert, N. Stoll, and K. Thurow, "Intelligent scheduling method for life science automation systems," in *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2016, pp. 156–161
- X. Gu, S. Neubert, N. Stoll, and K. Thurow, "A new method for the indicator of dynamic scheduling in life science laboratory using artificial neuro networks," in *2018 International Instrumentation and Measurement Technology Conference (I2MTC)*, 2018 (Accepted)
- S. Neubert, B. Göde, X. Gu, N. Stoll, and K. Thurow, "Potential of Laboratory Execution Systems (LESs) to Simplify the Application of Business Process Management Systems

(BPMSs) in Laboratory Automation," *SLAS Technol. Transl. Life Sci. Innov.*, vol. 22, no. 2, pp. 206–216, Apr. 2017

- S. Neubert, T. Roddelkopf, X. Gu, B. Göde, S. Junginger, N. Stoll and K. Thurow, "Architecture for a Combined Mobile Robot and Human Operator Transportation Solution for the Hierarchical Life Science Automation," presented at the 14th International Conference on Informatics in Control, Automation and Robotics, 2018, pp. 41–49