Automatic Resource Allocation in Business Processes: A Systematic Literature Survey

Luise Pufahl[1*], Sven Ihde[2], Fabian Stiehle[1], Mathias Weske[2], Ingo Weber[1]

[1]School of CIT, Technical University of Munich
[2]HPI, University of Potsdam

*Corresponding author: Luise Pufahl, E-mail: luise.pufahl@tum.de

Organizations execute various business processes to operate their business and serve their clients. During execution, upcoming process tasks must be allocated to internal resources, such as humans or machines. This is a complex decision-making problem with a high impact on the effectiveness and efficiency of business processes. A wide range of system-initiated (largely automated) resource allocation approaches were developed during the last decades. In this study, we present a comprehensive overview of this field, by discussing the results of 61 primary studies. We identified the primary studies through a rigorous structured literature review, covering publications from 1995 to 2023. We report on which allocation capabilities and goals are supported, the use of process models, execution data, task and resource attributes, the type of algorithmic solution, and evaluation methods. In the review, we mainly observed approaches supporting 1-to-1 allocation, process-oriented goals, the use of process models, and rule-based approaches. Based on our results, we envision future research to explore data-driven and context-adaptable solutions and contribute to a better understanding of approaches' performance impact.

Keywords - *business process, resource management, resource allocation, optimization*

Acknowledgements:

1 Introduction

Business Process Management (BPM) is applied by organizations to run their operations effectively and efficiently. It is a management paradigm that positions business processes, which consist of a set of connected activities carried out to reach a specific business goal (Weske, 2019), in the center of its efforts to facilitate operational excellence and continuous improvement (Dumas, La Rosa, Mendling, Reijers, et al., 2018). For the successful execution of business processes, organizations need a rich set of internal resources, such as human resources, machines, vehicles, materials etc. (Cabanillas, 2016; Huang, van der Aalst, Lu, & Duan, 2011). Often, resources are valuable assets, frequently costly and limited (Arias, Saavedra, Marques, Munoz-Gama, & Sepúlveda, 2018). Thus, the success of a business process is closely intertwined with the efficient use of resources. *Resource allocation* aims at ensuring that each activity of a particular process case (i.e., a task) is executed at the right time and with the right resources (Cabanillas, 2016; Kumar, Van Der Aalst, & Verbeek, 2002). A specific characteristic of business processes is that their focus is typically not on a single resource or activity but on coordinating various activities to reach a business goal: constraints regarding the order of process activities need to be reflected during allocation (Xu, Liu, Zhao, & Ding, 2013).

However, in general resources are not only involved or required in one single process case of a single business process, but in several cases of several different business processes, which may run concurrently (Zhao, Liu, Dai, & Ma, 2016). Consider the case of a physician who has to diagnose and treat different patients during the day. The physician is not only involved in the treatment processes but also in the reimbursement processes of the hospital or clinic, in which patients or their respective health insurers get invoiced. By not only taking into account the resource needs for one individual task but for the set of tasks to be executed, *resource allocation* is a multi-objective decision-making problem (Zhao, Zeng, Zheng, & Yang, 2017) that has to handle "the allocation of limited resources among competing [tasks]" (Luss, 2012). Additionally, each business process has certain time, cost, and quality goals that must be considered (Xu et al., 2013).

Resource allocation can be done manually by a human expert or by an IT system that proposes or enforces a resource allocation, which we call system-initiated resource allocation. Traditionally, in the BPM domain, requirements for the needed resources are specified per activity as soon as a process model is planned to be executed (Dumas et al., 2018). Business Process Management Systems (BPMSs) often support the allocation of tasks to resources with allocation rules, which were structurally studied by Russell, van der Aalst, ter Hofstede, and Edmond (2005). Additionally, operations research offers a broad range of optimization approaches for resource allocation in different application areas (Kamrani, Ayani, & Moradi, 2012). Numerous studies have emerged that have built on those, adapting them to the needs of business processes and integrating them into BPMSs (Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Xie, Chien, & Tang, 2016). Additionally, increased computational power allows the use of meta-heuristics, like genetic algorithms (e.g., (Huang, Lu, & Duan, 2012b)) and machine learning approaches, leveraging increasing amounts of process execution data. Among others, these developments have led to a broad spectrum of studies exploring system-initiated (largely automated) resource allocation approaches for business processes. These approaches might not necessarily use an executable process model as an artifact.

The variety of approaches and techniques for resource allocation in business processes renders a systematic survey valuable for researchers and practitioners alike. Existing literature studies in the area of resource allocations in business processes focus on resource management more generally and have studied fewer works (Cabanillas, 2016) or did not discuss in detail the targeted resource allocation problems and solution approaches (Arias, Saavedra, et al., 2018).

In this study, we aim to fill this gap. We present a comprehensive overview of the field, discussing 61 identified studies and reporting which allocation capabilities and goals are supported. We discuss their use of process models, execution data, task and resource attributes, the type of algorithmic solution, and evaluation methods. Our various classifications can help to gain insights into the design of such approaches and help to identify trends and research gaps. For practitioners, they can guide the design of future practical implementations.

We have followed a methodology for rigorous and replicable systematic literature reviews (SLRs) (Kitchenham, 2004; Okoli & Schabram, 2010). We found that most of the identified resource allocation approaches assign one task to precisely one resource; approaches supporting many-to-1 and 1-to-many were less prevalent. The identified approaches often have process-oriented resource allocation goals, such as minimizing process costs. The SLR offers insights into the different applied solution techniques and finds that mainly rules, heuristics and machine learning approaches are used. Lastly, the review reveals that many studies lack replicability.

In Section 2, we first establish the background on resource allocation in business processes. Then, the related work is presented in Section 3. Subsequently, the applied research method—the SLR protocol—is explained in detail in Section 4. The identified studies of the SLR are summarized, analyzed, and discussed in Section 5. In Section 6, the implications of our findings are discussed, along with open research fields and limitations of the research. Finally, Section 7 concludes the paper.

2 Background

To achieve its business goals, an organization runs a set of business processes that are executed with the help of different internal resources, such as humans, IT services, or machines. The following section presents the essential concepts of business processes, resources, and their allocation.

2.1 Business Process

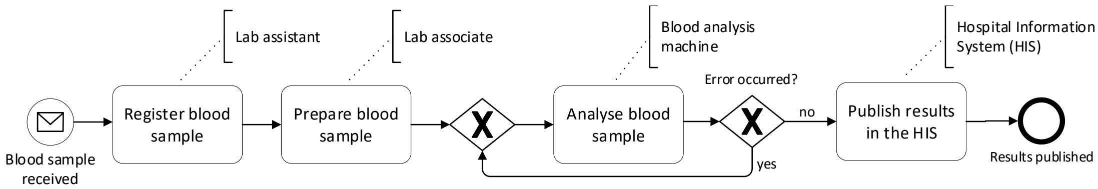


Figure 1: Laboratory process of a hospital given as BPMN process diagram, with resource information in activity annotations.

Process model. "A business process consists of a set of activities that are executed in coordination in an organizational and technical environment to reach a specific business goal" (Weske, 2019, Chapt. 1). For managing the documentation, redesign, execution, monitoring, and analysis of a business process, often process models are used as a formal representation of a business process. An example of a business process model from healthcare is shown in Figure 1, represented in the industry standard Business Process Model and Notation (BPMN) (OMG, 2011), specifically as a "process diagram". The shown laboratory process describes handling a blood sample: once received, the sample is registered, prepared, and then analyzed by the blood analysis machine. The results are published in the hospital information system if no errors occur. Apart from BPMN, many process modeling languages have been proposed over time. Prominent examples include Event-driven Process Chains (EPCs) (Scheer, Thomas, & Adam, 2005), Petri net (Van der Aalst, 1998)—as a rather formal specification language—or Declare (van Der Aalst, Pesic, & Schonenberg, 2009) as a declarative modeling language.

Process and activity instances. A business process model acts as a blueprint for a set of business process instances. A process instance represents the concrete execution of a business process case. For a concrete process instance, activity instances are initiated during runtime, which need time and resource(s) to be executed. In this paper, we call them tasks[1]. Each task follows a specific *lifecycle* (Weske, 2019,

[1]Please note that this definition of a *task* is different from the BPMN standard (OMG, 2011), in which it is defined as a single unit of work in a process diagram. In contrast, as does part of the literature, we use *task* as a shorthand for *activity instance*.

Chapt. 3)[2]: After its initialization, it is ready for execution as soon as its incoming control flow edges have been triggered. Then, it is allocated to one or more resources. At a certain point, the allocated resource(s) start(s) it, and after the work has been done, the task is completed. Furthermore, each task has specific characteristics, so-called task data (e.g., due date), based on which the resource allocation decision can be made.

Process performance. Reaching its business goals is typically essential for an organization, and as per the above definition, business processes are the activities that the organization undertakes to achieve its goals. Thus, organizational success depends strongly on effective processes. In addition, a process should be efficient in its execution. According to Dumas et al. (2018, Chapt. 2), four main process performance dimensions can be distinguished: time, cost, quality, and flexibility. In the following paragraph, we sketch sample measures for each dimension and refer the interested reader to (Dumas et al., 2018, Chapt. 2,7,8) for more details on process performance measures.

A relevant measure for the *time* dimension is the cycle time, which measures the time between the start and end of a process instance (i.e., processing times plus waiting times between activities). Regarding the *cost* dimension, the goal is usually to minimize the costs. The operational process costs can be calculated by measuring the time used to execute a task and multiplying it by the cost per time unit of the resource. Resource utilization is the time a resource needs to execute process tasks divided by its available working time. This measure can be used to evaluate whether process costs can be further decreased by increasing resource utilization. Process *quality* can be identified by calculating, for instance, the rate of successfully executed process instances. Process *flexibility* is the capability to react to changes in the process execution, e.g., the capability to react to missing resources.

Process execution and data. In the BPM domain, Business Process Management Systems (BPMSs) have been designed to "control the execution of business processes instances, according to the logic defined in the respective process model." (Durán, Rocha, & Salaün, 2019). As shown in the architectural model in Figure 2, a BPMS consists of a process modeling and a process engine component connected to several services (Dumas et al., 2018; Weske, 2019). Business processes can be modeled with the process modeling component and stored in the process model repository. Process models can be deployed on the

[2]Please be aware that this is an abstracted version of the lifecycle of a task. A more detailed lifecycle is provided by Russell et al. (2005), where it is also considered that tasks can be *offered* to resources, who then can decide on the allocation and start on their own. We will discuss later in Sect. 2.2 how this is realized.

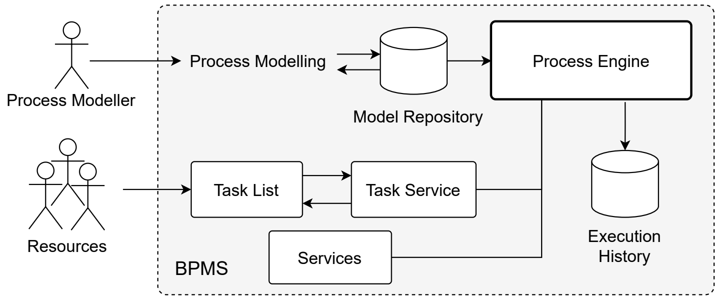


Figure 2: Architecture of a BPMS (based on (Weske, 2019, Chapt.8)), including its task service and task list of resources.

process engine. Then, process instances can be started in the process engine and are executed according to the behavior specified in the deployed process model. When business processes are supported in their execution by an information system or a BPMS, then process execution data is stored, from which event logs can be generated (Remy, Pufahl, Sachs, Böttinger, & Weske, 2020). Event logs are useful for evidence-based process analysis with the help of process mining techniques, such as the automated discovery of process models (van der Aalst, 2016). An event log consists of a series of events for different process instances reflecting, e.g., the start or completion of a task. An excerpt of the event log from the laboratory process shown in Figure 1 is represented in Table 1.

Each event has a case id (e.g., `9845`) as a reference to a process instance, a timestamp (e.g., `14/01/2021 11:22`) to denote when the event has occurred, the executed activity (e.g., `Register blood sample`) and its lifecycle state (e.g., `start`). Additional other data can be given, such as resource information (e.g., `Lab assistant`) that can be useful also for the analysis of resource behavior.

2.2 Resources and their Allocation in Business Processes

Resources are the fundamental basis for organizations to execute the tasks necessary to reach the goals of business processes. In the literature, the definitions for concepts like *resource* vary according to the

Table 1: Illustration of the event log structure

case id	timestamp	task	lifecycle state	resource
...	...	...	...	...
9845	14/01/2021 11:22	Register blood sample	start	Lab assistant
9845	14/01/2021 11:25	Register blood sample	complete	Lab assistant
9852	14/01/2021 11:26	Register blood sample	start	Lab assistant
9852	14/01/2021 11:30	Register blood sample	complete	Lab assistant
9845	14/01/2021 11:26	Prepare blood sample	start	Lab associate
...	...	...	...	...

background and goals of the research. The semi-formal definitions below provide clarity regarding the meaning of these terms for the context of this paper. To start, we define[3] a resource as anything necessary to execute tasks (e.g., human, vehicle, software, tools) in the context of a business process:

Definition 1 (**Resource**). *A resource r is an entity required to execute a process task. It has a set of attributes describing its capabilities, capacities, and availability at a given point in time t; those values might change during runtime. R is the set of all resources.*

Ouyang, Wynn, Fidge, ter Hofstede, and Kuhr (2010) distinguish *human* and *non-human* resources. Non-human resources are subdivided into *application* and *non-application* resources. An application resource describes anything that can execute tasks, such as a software service. In contrast, non-application resources, e.g., a transport vehicle, can be used by human or application resources to execute a task. Non-application resources can be further sub-divided into *consumable* (e.g., machine oil) or *durable* (e.g., surgical instruments) resources.

Resource allocation. Resource allocation[4] is a key part of resource management in business processes (Cabanillas, 2016). Its goal is assigning a process task to the most appropriate resource(s) among the available resources at runtime or planned for runtime (Kamrani et al., 2012), as shown in Figure 3. Resource allocation[5] in processes is a complex decision problem that should consider the following aspects:

- Multiple business processes and process cases may run concurrently, such that conflicting requests for the same resource have to be resolved and imbalances between the workload of resources should be handled (Zhao et al., 2016).
- For a rational resource allocation, the performance goals of business processes, such as minimizing time or costs, should be considered in the resource allocation (Zhao et al., 2016).

[3]The definition is based on a broad range of different definitions provided (Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Bellaaj, Sellami, Bhiri, & Maamar, 2017; Bussler & Jablonski, 1995; Cabanillas et al., 2013; Doerner, Gutjahr, Kotsis, Polaschek, & Strauss, 2006; Erasmus et al., 2018; Ihde, Pufahl, Völker, Goel, & Weske, 2022; Rhee, Cho, & Bae, 2010; Xie, Chen, Ni, & Wu, 2019; Xu, Liu, & Zhao, 2009; Xu et al., 2013; Yaghoibi & Zahedi, 2017; Zhao et al., 2016; Zhou & Chen, 2008).

[4]In operations research, the term *resource assignment* is also used for *resource allocation* (Kamrani et al., 2012). In contrast, in business process management, *resource assignment* defines the specification of resource requirements for process activities at design time (Cabanillas, 2016). In this work, we refer to this as resource allocation constraints.

[5]We talk about *(task) scheduling* (Slack & Brandon-Jones, 2018) when the solution of a resource allocation is a concrete schedule, including the information on which task is worked by which resource at which time. Usually, the resource allocation happens before the task execution, see the above discussion of the lifecycle of a task. If the resource allocation is some time ahead (e.g., a day before the task execution), it is often called *planning*. A resource allocation plan might require an adaptation later in contrast to a real-time resource allocation.

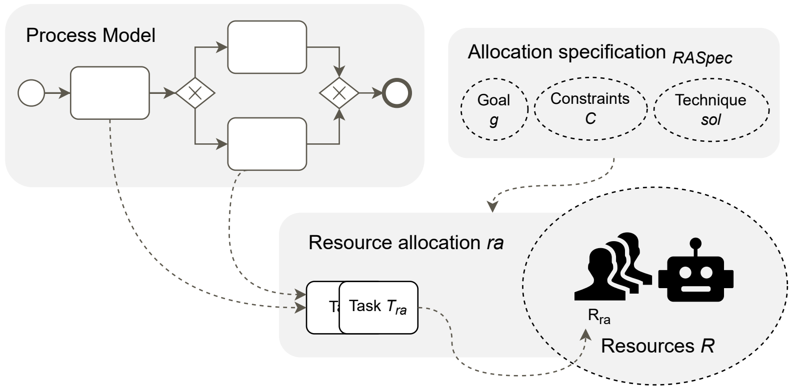


Figure 3: An organization with its business processes and resources where tasks resulting from process executions need to be allocated to one or more resources. Note that goals, constraints, etc. may be left unspecified in a given setting.

- Often, process activities require the allocated resources to possess certain competencies, capabilities, or rights, which should be considered as constraints during resource allocation (Arias, Saavedra, et al., 2018) (in many cases referred to as resource assignment (Cabanillas, 2016) in the BPM domain).
- In addition, resource allocation in business processes is not only about optimizing a single activity but should take a set of activities and the process into account (Xu et al., 2013).

Definition 2 (**Resource Allocation**). *More formally, let R be a set of resources and let T be a set of tasks that are ready for execution, which may span different processes and instances. Furthermore, let $R_{ra} \in 2^R$ be the set of available resources and $t \in T$. Then, resource allocation is a function*

$$ra : 2^R \times T \mapsto R$$

such that $ra(R_{ra}, t) = r$ indicates that a resource r is allocated for task t in a state where resources in R_{ra} are available. To allocate several resources to a given task, the function ra' is defined as follows:

$$ra' : 2^R \times T \mapsto \mathscr{P}(R)$$

such that $ra'(R_{ra}, t) = R_a$ indicates that a set of resources $R_a \in \mathscr{P}(R)$ is allocated for task t in a state where resources in $R_{ra} \in 2^R$ are available. To allocate a resource to a set of tasks, ra'' is defined as follows:

$$ra'' : 2^R \times 2^T \mapsto R$$

such that $ra''(R_{ra}, T_r) = r$ indicates that a resource r is allocated to a set T_r of tasks in the ready state. Again, resources in R_{ra} are available.

According to Definition 2, a task can be assigned to several resources, and several tasks to a resource. Traditionally, in BPM, 1-to-1 mappings are often supported. In this literature review, we want to analyze how frequently (if at all) the more complex resource allocation capabilities are supported.

A resource allocation decision has to be made in a dynamic system where new tasks arrive, resources complete tasks, and the availability of resources is changing. Thus, in addition to the process model, the system dynamics must also be considered during resource allocation. Resource allocation approaches often capture these dynamics with distributions, e.g., the inter-arrival time of tasks, the duration of the business process activities, probabilities of executing specific process paths (i.e., branching probabilities), and a model to capture the availability of resources. In this work, we want to analyze which approaches use which kind of information regarding the system dynamics in addition to the process model.

Realization of a resource allocation. Resource allocation can be a manual effort in an organization, where a human being assigns tasks to qualified resources or the staff members select tasks independently from a shared task list. Supported by a BPMS, its task service handles its resource allocation if an activity is ready for execution (Weske, 2019). The task service adds the task to at least one task list of a resource from where the resource can start and complete its execution (cf., Figure 2) (Weske, 2019). One simple but often used task allocation pattern in existing BPMSs is the *role-based distribution* (Russell et al., 2005), where the task is added to the task lists of all workers with a specific role. A role is a grouping mechanism for resources with similar capabilities and responsibilities, such as *lab assistant*. A resource can be assigned to one or multiple roles. Analyzing a comprehensive range of BPMSs, Russell et al. (2005) differentiate between pull and push patterns. Pull patterns describe situations where the resource has been made aware of a task by the system but initiates the execution itself, such as the *resource-initiated execution* patterns. In contrast, the push patterns describe situations where the system actively offers or allocates a task to a resource (e.g., *round robin allocation*, allocating tasks to available resources in a cyclic order).

In operations research, the problem of allocating a resource to tasks has a long tradition and is known as the *Assignment Problem*; it has been discussed in different versions (Kamrani et al., 2012), for which also several solution techniques have been proposed *to provide a more optimized allocation*. In contrast to BPM, the focus is often on a single activity or workstation. The overall business process with control flow relations between the process activities is rarely considered. Over the last decades, several approaches have been developed in addition to the existing resource patterns to support the task service of a BPMS or any IT system supporting the process execution and to allow a system-initiated resource allocation approach for business processes, using also insights from operations research. In this paper, we want to investigate the research studies in this area structurally. As a system has to know the determining factors for an allocation, we introduce the resource allocation specification in the next paragraph.

Resource allocation specification. As mentioned before, resource allocation is subject to performance measures and goals—like minimizing cost—that it should optimize for. Hence, a concrete resource allocation specification can guide a system-initiated resource allocation.

In particular, a resource allocation specification consists of parameters that influence a resource allocation by providing an optimization goal, concrete limitations that should be fulfilled, and a technique for solving the allocation.

Definition 3 (**Resource Allocation Specification**). *More formally, a resource allocation specification* $RA_{Spec} = (g, C, sol)$ *consists of*

- *a weighted allocation goal* $g = \{g_1 * w_1, g_2 * w_2, ..., g_n * w_n\}$ *with sub-goals* g_i *and their respective weight* $w_i \geq 0$ *that the resource allocation should optimize for; notice that* $\sum_{i=1}^{n} w_i = 1$,
- *a set of constraints C of the resources and tasks, where we can distinguish here between hard constraints that need to be upheld and soft constraints which may be associated with a penalty (Kumar et al., 2002), and*
- *a concrete solution technique sol to make allocation decisions; this may also be manual.*

Solution Quality. Multiple feasible solutions might exist for a particular business process scenario. We call a solution *feasible* if it satisfies all constraints of the resource allocation specification. Given a goal evaluation function e and all available information at one point in time t, optimization methods typically aim to find the best solution –the *global optimum*– among the feasible ones, i.e., the one that maximizes (or minimizes) e. Relevant information is here the resource allocation specification RA_{Spec}, the available resources R_{ra} including their attributes, and the tasks T being ready for execution including their attributes, which can also be from different processes and instances. Naturally, finding a global optimum is always in the best interest. Nonetheless, in reality, a balance between the effort and time to find a resource allocation solution and the quality of the solution has to be found. If the solution quality is high, but the time to produce it exceeds the limits of the underlying business process, then the high solution quality might not be relevant anymore. Thus, in practice, it is often the case that a limited number of solutions are checked. In situations where only a limited subset of solutions is explored, there is a possibility that the best solution found is not the global optimum but rather a *local optimum*. A local optimum is a solution that is the best within a specific neighborhood or region of the solution space but is not necessarily the absolute best solution across the entire solution space. More formally, we define the solution quality as follows:

Definition 4 (**Solution and Solution Quality**). *Let S be the set of feasible solutions, where a solution $s \in S$ comprises a set of resource allocations that jointly cover all enabled tasks: $s = \{ra(R_{ra}, t) \mid t \in T\}$, where T is the set of enabled tasks and R_{ra} the set of available resources as above. Let $s_{max} \in S$ be an optimal solution. Let $e : S \to \mathbb{R} \subseteq [0, 1]$ be a function that evaluates a given solution, such that $\forall s_i \in S : e(s_i) \leq e(s_{max})$. The solution quality is a function $q : S \to \mathbb{R} \subseteq [0, 1]$ that quantifies how close a specifically feasible solution $s \in S$ is to the optimal solution s_{max}, given the evaluation function e for the optimization.*

$$q(s) = \frac{e(s)}{e(s_{max})} = \begin{cases} 1, & \textit{if } s \textit{ is a global optimum} \\ < 1, & \textit{otherwise} \end{cases}$$

For $s' \notin S$ we define $q(s) = 0$, in other words, infeasible solutions are assigned minimal quality.

One approach to reducing the effort in finding the optimal solution is focusing on a single task for the allocation decision (Xu, Liu, Zhao, Yongchareon, & Ding, 2016). As an organization's current set of to-be-executed tasks carries different levels of importance, focusing on a single task can lead to a local optimum solution. If an allocation approach considers all to-be-executed tasks, then the approach tries to reach a global optimum. Naturally, this refers to the decision at that point in time t. This SLR wants to analyze how different approaches target this challenge.

3 Related Work

This section presents and discusses existing BPM surveys and surveys on resource allocation in business processes. A first comprehensive study on the use and representation of (primarily human) resources in existing BPMSs was given by Russell et al. (2005). Studying different existing process modeling languages and BPMSs, the authors provide a resource meta-model, a task lifecycle, and a collection of patterns to create, pull, push, and detour tasks to resources that help to set up a task service in a BPMS. However, the resource patterns do not focus on the optimization problem associated with resource allocation[6]. An extension of the resource patterns for teams collaboratively working on tasks is presented by Van der Aalst and Kumar (2001). It discusses aspects that must be considered when teams work on tasks and presents an architecture to integrate groupware, such as a shared calendar, with BPMSs. The focus of the resource patterns is mainly on what is supported by modeling languages and BPMSs, whereas the subject of this study is the state-of-the-art in research.

In a comprehensive survey, Van der Aalst (2013) provides typical BPM research use cases. Resources and their allocation do not play a central role in the presented use cases. Still, it is mentioned that resource information can be discovered from event logs and enhanced process models, that BPMSs have to handle resources, and that studies that deal with optimal resource allocations were observed.

Cabanillas (2016) examines research works on resource handling in process- and resource-oriented systems. They categorize the works into: 1. resource assignment (i.e., the definition of resource requirements for process activities at design time), 2. resource allocation (i.e., designation of concrete resources to a specific task during runtime) and 3. resource analysis (i.e., evaluation of process execution with a focus on resources). The author did not intend to present an exhaustive literature review but a framework with representative works, so only a select set of studies were explored.

Arias, Saavedra, et al. (2018) provide a systematic mapping study on human resource allocation in business process management and process mining. The focus of the study is on human resources and reporting on research methodology (e.g., type of study) and where and when the studies were published. It does not discuss the identified solutions in detail.

A survey on automated planning and BPM (Marrella, 2019) shows that artificial intelligence can leverage particular challenges in business process management, such as the automatic generation of process models, allowing process adaptations, or enabling conformance checking. It gives a concrete method for building planning problems. Although resources play an indirect role, resource allocation is not discussed as an application area.

In their structured literature review, Yari Eili and Rezaeenour (2022) examine process-aware recom-

[6]The resource patterns do not have an inherent allocation goal association. Therefore, it falls upon the process designer to evaluate whether a chosen set of patterns aligns with the allocation goal for a given business process. Additionally, the resource patterns predominantly adopt a localized perspective, typically focusing on allocating an individual task. An exception is here the pattern *system-determined work queue content* as it provides the possibility that the system sorts the tasks, for example, based on their priority in the working list.

mender systems that utilize event logs as their input. The authors categorize these systems based on whether they recommend the next activity or the next resource. They delve into the used recommendation approaches (e.g., data mining, process minings, statistical or hybrid approaches), the implementation environments, and the evaluation method. However, the primary focus of their review centers on the recommendation systems themselves, with limited coverage of the mechanisms for resource allocation. The authors do not investigate the optimization objectives of these approaches, the specifics of their input data, or the roles of process models and the execution data.

In contrast to prior literature reviews, we aim to deliver a comprehensive analysis of research studies enabling a system-initiated resource allocation in the business process through an exhaustive literature review. We want to provide a structured comparison of the studies' proposed approaches concerning their optimization objectives, capabilities, utilization of process models and execution data, input data sources, and applied solution techniques.

4 Research Method

A Systematic Literature Review (SLR) (Kitchenham, 2004; Okoli & Schabram, 2010) allows the identification and analysis of relevant and existing literature related to specific research questions while aiming to minimize bias and maximize reproducibility. This section describes the search protocol followed in the SLR. First, the addressed research questions are presented in Section 4.1. The data sources and the search strings for the primary search, the backward and forward search, the relevance checks, and the data extraction scheme are described in Section 4.2.

4.1 Research Questions

This study aims to answer the overarching question *"What is the state-of-the-art of system-initiated resource allocation approaches for business processes?"*. We divided this general question into four more concrete sub-questions to support more fine-grained analysis:

RQ1 *What are the targeted resource allocation goals and capabilities?* When an automatic approach is responsible for the resource allocation in business processes, it usually needs an allocation goal as per Definition 3. Furthermore, approaches can support traditional 1-to-1 allocations of tasks to resources but also other allocation capabilities (see Definition 2). To help practitioners and researchers in identifying allocation approaches for a particular allocation problem with a specific target or an allocation capability, an overview of the approaches' different allocation goals and their capabilities is needed.

RQ2 *What are the respective roles of process models and process execution data in the resource allocation approach?* Resource allocation in business processes is unique because the ordering constraints between process activities needs to be considered (Zhao et al., 2017) in the resource allocation specification (cf. Definition 3). With this question, we wanted to understand if and how information from process models or execution data is encoded and considered in the identified approaches.

RQ3 *Which input data regarding resources and tasks do the different resource allocation approaches use?* As outlined in Definition 3, a resource allocation specification encompasses the incorporation of constraints. In defining these constraints, one typically leverages resource and task-related information additionally to the ordering constraints of business processes (see RQ2) to identify a useful resource allocation. To understand the broad spectrum of possibilities, we want to extract the input information that is used by the approaches for constraining the resource problem. Arias, Munoz-Gama, and Sepúlveda (2018) derived a taxonomy of resource attributes applied by resource allocation approaches in BPM. When addressing this research question, we make use of this taxonomy to understand and classify the input data used in the identified studies and extend it by task attributes.

RQ4 *Which solution strategies are used?* Additional to the goals and constraints, a solution technique is applied for solving a resource allocation problem (cf. Definition 3). With this research question, we want to understand the types of solution approaches used for resource allocation for business processes, such as rules, heuristics, and machine learning. Furthermore, we want to investigate the targeted solution quality of the approaches (cf. Definition 4) and want to focus here on whether approaches follow a single-task optimization in the context of individual process cases or a multi-task optimization, where the whole set of to-be-executed process tasks is considered.

RQ5 *How applicable are the proposed resource allocation approaches, given the availability of evaluations and prototypes?* The applicability of an allocation approach in real-world settings depends on whether it has been evaluated and whether a corresponding prototype has been made available. A rigorous evaluation increases the trust of researchers and practitioners that a resource allocation approach leads to the intended results. Based on Zelkowitz and Wallace (1998), we investigated which approaches have no evaluation, an argumentation on a toy example, a case study, or a controlled experiment. A research prototype helps enable the implementation of an approach in a real-world setting and allows benchmarking different approaches and comparing them with each other for a certain business scenario. We intend to distinguish whether a prototype is available, not provided, not accessible, or only as pseudocode.

4.2 Search for studies and data extraction

Our search for studies was split into two phases: a primary search and a secondary search. Research databases were queried with a predefined set of search terms concerning the abstract, title, and keywords in the primary search. The resulting studies were reviewed to identify a core set of relevant studies. This core set was then used to conduct a backward-forward search. By doing this, we could find additional relevant studies and maximize the completeness of our search. The complete search process, with the resulting number of studies, is illustrated as a BPMN process diagram in Figure 4.

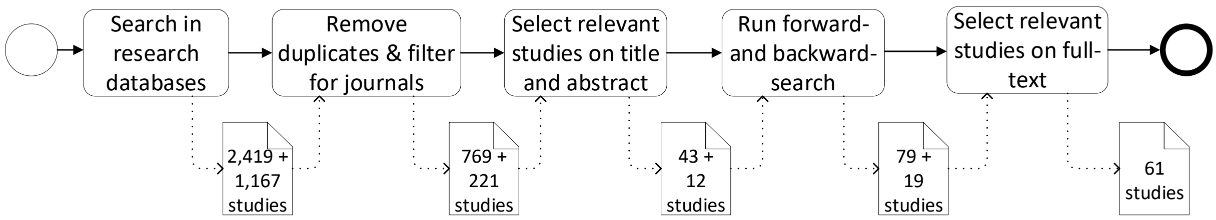


Figure 4: Search process and the number of studies as result of the difference steps, including the searches in 2019 and 2023 indicated by the $+$.

In the primary search, we queried several research databases: *ACM Digital Library*, *IEEE Xplore*, *SciVerse Scopus*, and *Web of Science*, with the following general search term: (RESOURCE | STAFF | TASK) x (ALLOCATION | ASSIGNMENT | SCHEDULING | OPTIMIZATION | PLANNING) x (BUSINESS PROCESS | PROCESS MINING). The search was conducted in June 2019 and repeated in September 2023 in order to cover more recent publications. We observed that the defined search led to an enormous set of resulting studies in some preceding searches. This set included a high proportion of short conference papers. Thus, we decided to focus on journal papers, which are usually more detailed and on average of higher quality, and filtered for such articles in those databases (where possible). We were aware that high-quality works are also published at conferences, which we planned to include in the backward-forward search. The exact search queries per database, including synonyms, and the resulting number of studies are shown in Table 2.

Table 2: Database, search queries, and resulting number of studies in the primary search done in 2019 and the repeated search in 2023 for 2019-2023. TS of Web of Science can be used for searches for topic terms within a record, such as search in abstract, author keywords, etc.

Database	Search queries	# 2019	# 2023
ACM Digital Library	`recordAbstract:(resource staff task) + (allocation assignment scheduling optimization planning) + (“business processes” “process mining”)`	535	27
IEEEXplore	`((task OR staff OR resource) AND (allocation OR assignment OR scheduling OR optimization OR planning) AND (“process mining” OR “business processes”))`	1086	836
Science Direct	`(resource OR staff OR task) AND (allocation OR assignment OR scheduling OR optimization OR planning) AND (“process mining” OR “business process management”)`	61	36
Web of Science	`(TS=(task OR staff OR resource) AND (allocation OR assignment OR scheduling OR optimization OR planning) AND (“process mining” OR “business processes”)) AND LANGUAGE: (English)`	722	268

Table 3: Inclusion and exclusion criteria

#	Criteria
IN1	The study describes an algorithm or technique to support resource allocation in the context of business processes.
EX1	The study provides exclusively a survey on topics related to resources or business processes.
EX2	The study is not written in English.
EX3	The study is not published in a peer-reviewed journal.
EX4	The study focuses only on modeling resources or resource analysis.
EX5	The study focuses only on the planning of activities only, not their allocation to resources.
EX6	The study focuses on the design, configuration, or application of an ERP system.
EX7	The study only describes a resource allocation approach for one specific use case, such as manufacturing processes (Howard et al., 1999), and not a generalized approach for business processes.
EX8	The study describes an approach for allocating complete business process instances for their execution in an execution environment. The process instance is considered as an entire block and not as a set of related tasks (for instance, some studies on executing process instances in the cloud, like (Wei & Blake, 2016)).

The resulting 2,419 studies from 2019 and the 1,167 additional studies from 2023 were, with the help of the literature management tool *Citavi*, checked for duplicates, and non-journal publications were filtered out. The remaining 769 studies and 221 studies from the more recent search were checked independently by two authors of this paper for their relevance, based on the title and abstract and with the help of the inclusion and exclusion criteria given in Table 3. Thereby, also some duplicates were identified manually and removed.

All studies that both authors categorized as relevant were accepted. Studies for which a disagreement existed were discussed, and a final decision was jointly made. The resulting core set of relevant studies, comprising 43 papers and 12 papers of the more recent search, was then used for the backward-forward search to find additional studies. We also considered conference or workshop papers in the backward search because we assumed that these studies referenced by a journal article have a high implication for the research field. For this step, *Web of Science* was used. Identified papers were immediately checked for their relevance. Any uncertainties regarding their relevance were discussed among co-authors and jointly resolved. Relevant papers were added to the core set of studies, which resulted in 79 studies plus 19 of the recent search.

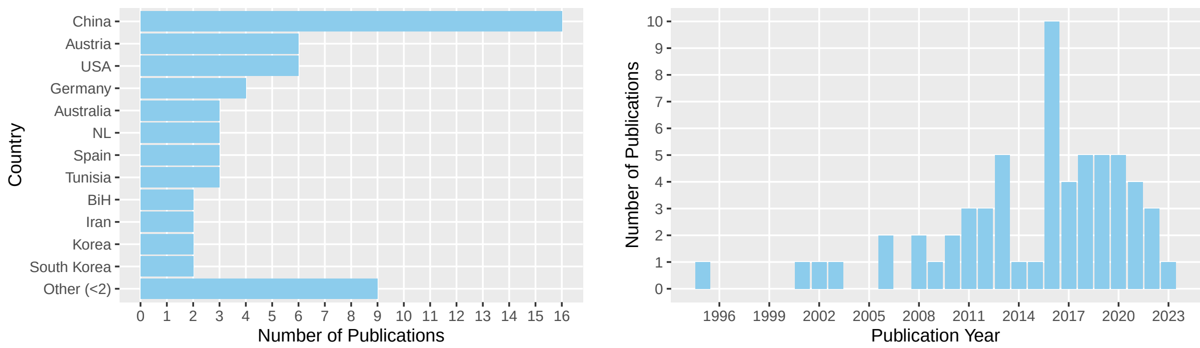
Next, we read the full text of the 98 studies and excluded another 37 because they did not fulfill the inclusion and exclusion criteria or were duplicates. Each exclusion decision was discussed with the group of co-authors. Reasons were often that the focus was on process modeling or resource analysis, the presented resource allocation approach was too domain-specific, or that the work had been published in different types of papers, whereby the most extensive versions had been kept. One study (R. Liu, Agarwal, Sindhgatta, & Lee, 2013) was added based on an expert's suggestion. The final set of 61 studies was read thoroughly, and relevant information was extracted according to the following predefined data schema : 1. year and country, 2. type of resources, 3. allocation capability (1-to-1, 1-to-m, m-to-1, m-to-n), 4. allocation goal, 5. usage of process model and execution data, 6. task and resource attributes, 7. solution technique, 8. single- vs. multi-task optimization, 9. type of evaluation (none, simulation experiments, experiments with real-world data, case study etc.), and 10. prototype (available, only pseudocode, etc.).

5 Resource Allocation Approaches in Business Processes

In this section, we discuss the results of the SLR, structured by our research questions. Before we do so, we report on basic statistics about the final set of 61 primary studies. We extracted the originating country based on the affiliation of the corresponding author. In rare cases where the corresponding author could not be identified, the author's affiliation first listed was chosen. Figure 5a shows the results. The first work of our study collection was published in 1995, followed by few studies published around 2002 (see Figure 5b). After 2010, more published studies on resource allocation in processes can be observed, with a peak in 2016. Most papers studied in this review were published between 2016 and 2021.

Additionally, we observed that most of the studies focused on human and application resources based on the classification by Ouyang et al. (2010). Only one study is here an exception: Doerner et al. (2006) considers all kinds of resources, including consumable ones.

Tables 4 and 5 show an overview of all studies sorted by year of publication and the categorization according to the data extraction scheme. The following subsections present details of these results. The resource allocation types and the targeted optimization goals of the approaches are presented in Section 5.1, addressing RQ1. Next, the role of process models and process execution data in the resource allocation approaches is discussed in Section 5.2, addressing RQ2. The considered input data, that is, the attributes of tasks and resources, are presented and classified in Section 5.3, addressing RQ3. Solution



(a) Originating country

(b) Publishing year

Figure 5: Resulting studies categorized by the originating country (corresponding author and the published year). Countries with one publication each are shown as "Other". Numbers for 2023 are not comparable to earlier years due to the timing of the search.

approaches are discussed in Section 5.4, addressing RQ4. Finally, the evaluation techniques and the usage of research prototypes are presented in Section 5.5, addressing RQ5.

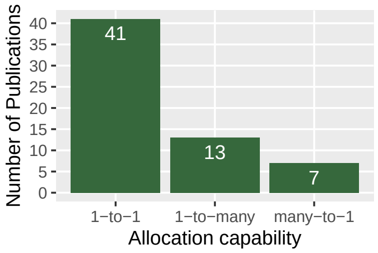
5.1 RQ1: Resource allocation capabilities and optimization goals

In this subsection, *RQ1: What are the targeted resource allocation goals and capabilities?* is explored. To do so, we present the results regarding the resource allocation capabilities supported in the literature, followed by targeted optimization goals.

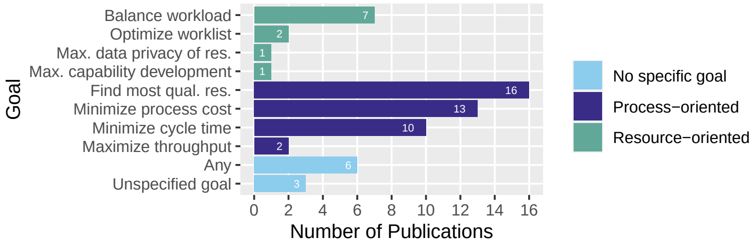
Figure 6a shows how many studies support the different resource allocation capabilities based on Definition 2 . Thereby, three resource allocation capabilities can be differentiated (relations as |tasks|-to-|resources|):

- 1-to-1 allocation, where one resource is assigned to exactly one task;
- 1-to-many allocation, where a team of resources is assigned to one task; and
- many-to-1 allocation, where the capacity of a resource is greater than one, and thus, several tasks can be assigned to it.

The concrete studies supporting a certain capability are given in Table 4. Most studies support an assignment of one resource to one task. For example, Doerner et al. (2006) represent the business process as a Petri net and resources as tokens on resource-places in the net. Depending on whether there is an arc from a resource-place to a transition and/or vice versa, a firing transition (representing an activity execution) consumes and/or produces *at most one* resource token. Hence, this approach offers 1-to-1 allocation.



(a) Supported allocation capabilities (|tasks|-to-|resources|)



(b) Optimization goals for allocation

Figure 6: Support allocation capabilities and optimization goals of the studies.

13 studies support 1-to-many allocations. Some of these studies consider the possibility of having one or more resources assigned to a task. An example is Kamrani et al. (2012) that offer the possibility to define a constraint in the resource allocation specification, which encodes the number of resources needed for an activity. Bessai and Charoy (2016) provide an approach for assigning tasks to crowd workers. If a certain task needs multiple skills that can only be fulfilled by several crowd workers, their approach can also assign a team to a task.

Furthermore, seven studies consider that resources can have capacities greater than one (e.g., a nurse can transport one or several blood tubes at once) or that resources should work on similar or related tasks in sequence. In either case, these studies support many-to-1 allocations. For instance, Xu et al. (2016) allow resources to have several slots available for a particular time. Thus, the resource can process several tasks in parallel. Pflug and Rinderle-Ma (2016) provide an approach that regularly classifies the existing tasks into sets of similar tasks; then, a set of similar tasks is assigned to a resource that can work on them one after another.

Many studies support a specific resource allocation goal (c.f. Definition 3) in their approach. We can group these goals based on the two entities involved in a resource allocation: the business process with its tasks and the resource responsible for executing the work. It can be observed that a majority of studies focus on goals concerning the improvement of the performance of the *process* (e.g., minimizing the process costs or cycle time). A smaller portion of studies focuses on improving the organization of the resources' workload and a resource orientation in their goal definition. Few studies have no (specific) optimization goal. An overview of the different optimization goals—and how many studies consider them—can be found in Figure 6b. In the following, we discuss these categories in more depth.

Reference	Year	Country	Capability	Optimization Goal	PM/PD	Role	Type	Solution technique	Evaluation	Prototype
Bussler and Jablonski (1995)	1995	Germany	many-to-1	Unspecified goal	PM	Prep.	Single	Rule	No evaluation	not acc., pseudoc.
Van Hee, Reijers, Verbeek, and Zerguini (2001)	2001	NL	1-to-many	Minimize cycle time	PM	Prep.	Single	Heuristic	Toy example	no implementation
Kumar et al. (2002)	2002	USA	1-to-1	Find most qual. res.	PM	Input	Single	Rule	Toy example	no implementation
Eder, Pichler, Gruber, and Ninaus (2003)	2003	Austria	1-to-1	Balance workload	PM	Prep.	Multiple	Heuristic	No evaluation	no implementation
Doerner et al. (2006)	2006	Austria	1-to-1	Minimize process cost	PM	Prep.	Multiple	Metaheuristic	Case study	pseudocode
Ha, Bae, Park, and Kang (2006)	2006	Korea	1-to-1	Balance workload	PM	Prep.	Multiple	Rule	(Comp.) sim. experiments	pseudocode
(Rhee et al., 2010)	2008	USA	many-to-1	Balance workload	PM	Input	Multiple	Heuristic	(Comp.) sim. experiments	not acc., pseudoc.
Zhou and Chen (2008)	2008	China	1-to-1	Minimize process cost	PM	Prep.	Multiple	Metaheuristic	Toy example	no implementation
Xu et al. (2009)	2009	Australia	1-to-many	Minimize process cost	PM	Input	Multiple	Rule	(Comp.) sim. experiments	pseudocode
Huang, van der Aalst, Lu, and Duan (2010)	2010	NL	1-to-1	Any	PM	Input	Single	Machine Learning	(Comp.) experiments	available
Delias, Doulamis, Doulamis, and Matsatsinis (2010)	2010	Greece	1-to-1	Balance workload	PM	Prep.	Multiple	Exact Algorithm	(Comp.) sim. experiments	not accessible
Huang, Lu, and Duan (2011)	2011	China	1-to-1	Unspecified goal	PD	Input	Single	Mined Rule	Experiments + Case study	not acc., pseudoc.
Huang, van der Aalst, et al. (2011)	2011	China	1-to-1	Minimize process cost	PM	Prep.	Multiple	Machine Learning	(Comp.) experiments	available
Kamrani et al. (2012)	2011	Sweden	1-to-many	Minimize process cost	PM	Input	Multiple	Heuristic	(Comp.) sim. experiments	pseudocode
Huang et al. (2012b)	2012	China	1-to-1	Minimize process cost	PM	Prep.	Multiple	Metaheuristic	(Comp.) sim. experiments	not acc., pseudoc.
Huang, Lu, and Duan (2012a)	2012	China	1-to-1	Find most qual. res.	PD	Prep.	Single	Rule	Case study	available
T. Liu, Cheng, and Ni (2012)	2012	China	1-to-1	Find most qual. res.	PD	Input	Multiple	Mined Rule	(Comp.) experiments	not acc., pseudoc.
Kumar, Dijkman, and Song (2013)	2013	USA	1-to-1	Any	PD	Prep.	Multiple	Heuristic	Case study	not acc., pseudoc.
Barba, Weber, Del Valle, and Jiménez-Ramírez (2013)	2013	Spain	1-to-1	Optimize worklist	PM	Prep.	Single	Logic Programming	(Comp.) sim. experiments	not acc., pseudoc.
Cabanillas et al. (2013)	2013	Austria	1-to-1	Find most qual. res.	PM	Input	Single	Heuristic	No evaluation	available
R. Liu et al. (2013)	2013	USA	1-to-many	Find most qual. res.	PD	Prep.	Single	Rule	Toy example	no implementation
Xu et al. (2013)	2013	Australia	1-to-1	Minimize cycle time	PM	Input	Multiple	Rule	(Comp.) sim. experiments	not acc., pseudoc.
Schall, Satzger, and Psaier (2014)	2014	Austria	1-to-many	Find most qual. res.	PM	Prep.	Single	Rule	(Comp.) sim. experiments	not acc., pseudoc.
Zhao, Yang, Liu, and Wu (2015)	2015	China	1-to-1	Minimize cycle time	PD	Prep.	Single	Rule	(Comp.) experiments	not accessible
Djedović, Žunić, Avdagić, and Karabegović (2016)	2016	BiH	1-to-many	Minimize process cost	PM	Prep.	Multiple	Metaheuristic	Case study	not accessible
Havur, Cabanillas, Mendling, and Polleres (2016)	2016	Austria	1-to-1	Minimize cycle time	PM	Input	Multiple	Logic Programming	(Comp.) sim. experiments	no implementation
Bessai and Charoy (2016)	2016	France	1-to-many	Minimize cycle time	PM	Input	Multiple	Heuristic	(Comp.) sim. experiments	not acc., pseudoc.
Pflug and Rinderle-Ma (2016)	2016	Austria	many-to-1	Maximize throughput	PD	Input	Multiple	Machine Learning	Case study	not acc., pseudoc.
Schönig, Cabanillas, Jablonski, and Mendling (2016)	2016	Germany	1-to-1	Unspecified goal	PD	Input	Multiple	Mined Rule	(Comp.) sim. experiments	not accessible
Wibisono, Nisafani, Hyerim Bae, and You-Jin Park (2016)	2016	Indonesia	1-to-1	Minimize cycle time	PM	Input	Single	Rule	(Comp.) sim. experiments	pseudocode
Xie et al. (2016)	2016	China	1-to-many	Minimize cycle time	PM	Prep.	Multiple	Exact Algorithm	Experiments + Case study	available

Table 4: Studies and their categorizations ordered by publication year, Part 1 (PD = Process data, PM = Process model).

Reference	Year	Country	Capability	Optimization Goal	PM/PD	Role	Type	Solution technique	Evaluation	Prototype
Xu et al. (2016)	2016	China	many-to-1	Balance workload	PM	Input	Multiple	Metaheuristic	(Comp.) sim. experiments	not acc., pseudoc.
Yaghoubi and Zahedi (2016)	2016	Iran	1-to-1	Optimize worklist	PD	Prep.	Multiple	Machine Learning	(Comp.) experiments	not accessible
Zhao et al. (2016)	2016	China	1-to-1	Find most qual. res.	PD	Prep.	Multiple	Mined Heuristic	(Comp.) experiments	pseudocode
Bellaaj et al. (2017)	2017	Tunisia	1-to-1	Find most qual. res.	PD	Prep.	Single	Rule	No evaluation	not acc., pseudoc.
Hirsch and Ortiz-Peña (2017)	2017	USA	1-to-1	Minimize cycle time	PM	Input	Multiple	Exact Algorithm	(Comp.) sim. experiments	not acc., pseudoc.
Zhao et al. (2017)	2017	China	1-to-1	Minimize cycle time	PM	Input	Multiple	Metaheuristic	(Comp.) experiments	not accessible
Yaghoibi and Zahedi (2017)	2017	Iran	many-to-1	Balance workload	PM	Prep.	Multiple	Rule	(Comp.) experiments	pseudocode
Arias, Munoz-Gama, Sepúlveda, and Miranda (2018)	2018	Chile	1-to-1	Find most qual. res.	PD	Prep.	Multiple	Exact Algorithm	Experiments + Case study	available
Djedovic, Karabegovic, Avdagic, and Omanovic (2018)	2018	BiH	1-to-many	Minimize process cost	PD	Prep.	Multiple	Metaheuristic	(Comp.) sim. experiments	not accessible
Erasmus et al. (2018)	2018	NL	1-to-1	Find most qual. res.	PM	Input	Multiple	Rule	Case study	not acc., pseudoc.
Abdulhameed, Helal, Awad, and Ezat (2018)	2018	Egypt	1-to-1	Find most qual. res.	PD	Prep.	Single	Mined Rule	Case study	no implementation
Si, Chan, Dumas, and Zhang (2018)	2018	Macau	1-to-1	Minimize process cost	PM	Input	Single	Metaheuristic	Case study	not acc., pseudoc.
Durán et al. (2019)	2019	Colombia	1-to-many	Any	PM	Prep.	Multiple	Heuristic	(Comp.) sim. experiments	available
Lee, Lee, Kim, and Choi (2019)	2019	Korea	1-to-1	Find most qual. res.	PD	Prep.	Single	Rule	Experiments + Case study	no implementation
Luo, Liu, Yin, Li, and Wu (2019)	2019	China	1-to-1	Find most qual. res.	PD	Prep.	Single	Machine Learning	No evaluation	not acc., pseudoc.
Soeffker, Ulmer, and Mattfeld (2019)	2019	Germany	1-to-1	Maximize throughput	PM	Prep.	Multiple	Heuristic	(Comp.) sim. experiments	not acc., pseudoc.
Xie et al. (2019)	2019	China	1-to-1	Minimize process cost	PM	Input	Multiple	Metaheuristic	Experiments + Case study	not acc., pseudoc.
Bellaaj Elloumi, Sellami, and Bhiri (2020)	2020	Tunisia	1-to-1	Minimize process cost	PD	Prep.	Single	Machine Learning	(Comp.) sim. experiments	not acc., pseudoc.
Jemel, Ben Azzouna, and Ghedira (2020)	2020	Tunisia	1-to-many	Max. data privacy of res.	PM	Prep.	Multiple	Logic Programming	(Comp.) sim. experiments	not accessible
Pereira, Varajão, and Uahi (2020)	2020	Portugal	1-to-1	Find most qual. res.	PM	Prep.	Single	Rule	No evaluation	no implementation
Yu, Jia, Liu, and Ma (2020)	2020	China	1-to-1	Minimize cycle time	PM	Prep.	Single	Rule	Toy example	no implementation
Zhao, Pu, and Jiang (2020)	2020	China	1-to-1	Minimize process cost	PD	Prep.	Single	Machine Learning	(Comp.) sim. experiments	not acc., pseudoc.
Barba, Jimenez-Ramirez, Reichert, Del Valle, and Weber (2021)	2021	Spain	1-to-1	Any	PM	Prep.	Multiple	Logic Programming	Case study	not acc., pseudoc.
Durán, Rocha, and Salaün (2021)	2021	Spain	1-to-many	Minimize process cost	PM	Prep.	Multiple	Rule	Experiments + Case study	not accessible
Hou et al. (2021)	2021	China	many-to-1	Balance workload	PM	Input	Multiple	Exact Algorithm	(Comp.) sim. experiments	not acc., pseudoc.
Pika and Wynn (2021)	2021	Australia	1-to-1	Max. capability development	PD	Prep.	Single	Machine Learning	Experiments + Case study	available
Ihde et al. (2022)	2022	Germany	many-to-1	Any	PM	Prep.	Multiple	Any	Case study	available
R. Liu, Kumar, and Lee (2022)	2022	USA	1-to-many	Find most qual. res.	PD	Prep.	Multiple	Rule	Experiments + Case study	not accessible
Yeon, Lee, Pham, and Kim (2022)	2022	South Korea	1-to-1	Find most qual. res.	PD	Prep.	Single	Mined Rule	Case study	no implementation
Park and Song (2023)	2023	South Korea	1-to-1	Any	PD	Input	Single	Machine Learning	Experiments + Case study	available

Table 5: Studies and their categorizations ordered by publication year, Part 2 (PD = Process data, PM = Process model).

Process-oriented optimization. A majority of the identified studies aim at *finding the most qualified resource*. A resource allocation tries to identify the most qualified resource for a specific process task based on the task's needs. The focus is here to optimize the process. As shown in Figure 6b, the second-most frequent goal is to *minimize the process costs*. Followed by the goal to *minimize the cycle time* of process executions and by a smaller portion of studies that aims at *maximizing the throughput*. The latter two goals are both time-oriented. But whereas *minimizing the cycle time* tries to ensure that each instance has the shortest possible time between start and end, *maximizing the throughput* aims at increasing the total output of a process within a given timeframe. This can also lead to longer cycle times for certain process instances.

Resource-oriented optimization. Most studies focusing on a resource-oriented optimization try to *balance the workload* between resources, i.e., tasks are distributed equally to the available resources. In contrast to finding the most qualified resource, tasks might also be allocated to less appropriate resources to equalize workloads. Two studies aim at *optimizing the worklist*: one study focuses on prioritizing tasks (Barba et al., 2013), and the other one on minimizing the entropy of worklists (Yaghoubi & Zahedi, 2016). Pika and Wynn (2021) want to maximize the capability development of the workers involved in business processes. To increase workers capabilities, their approach assigns tasks with which resources are less familiar, based on historical information. Finally, Jemel et al. (2020) aim at maximizing the data privacy of resources. They are providing an allocation approach for inter-organizational business processes and want to reduce the amount of information shared of involved resources.

No explicit optimization goal. Six studies support the idea that the optimization goal can be individually defined when applying and implementing the resource allocation approach. Huang et al. (2010) maximize the allocation reward, and the user can specify the calculation of the reward. Durán et al. (2019) describe that a multi-objective optimization problem has to be defined for a resource allocation, and Ihde et al. (2022) provide a way to define the optimization goal individually for each process activity at design-time. Barba et al. (2021); Kumar et al. (2013); Park and Song (2023) all propose approaches where the objective function can be individually defined. These studies provide more generalized approaches that are usable in more application scenarios. Practitioners are enabled to define their resource allocation goals individually. Three studies working with resource allocation rules (Bussler & Jablonski, 1995; Huang, Lu, & Duan, 2011; Schönig et al., 2016) are not associated with a specific resource

	Process Model	Process Execution Data
Used as preparation for an input artifact for the allocation technique	**Enhanced Process Models** (Barba et al., 2021, 2013; Bussler & Jablonski, 1995; Delias et al., 2010; Djedović et al., 2016; Doerner et al., 2006; Eder et al., 2003; Huang et al., 2012b; Ihde et al., 2022; Jemel et al., 2020; Pereira et al., 2020; Schall et al., 2014; Xie et al., 2016; Yu et al., 2020; Zhou & Chen, 2008) **Stochastic model** (Durán et al., 2019, 2021; Huang, van der Aalst, et al., 2011; Soeffker et al., 2019; Van Hee et al., 2001) **Queue Model** (Ha et al., 2006; Yaghoibi & Zahedi, 2017)	**Characteristics of resources** (Abdulhameed et al., 2018; Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Bellaaj et al., 2017; Huang et al., 2012a; Kumar et al., 2013; Lee et al., 2019; R. Liu et al., 2013, 2022; Luo et al., 2019; Pika & Wynn, 2021; Zhao et al., 2016, 2020, 2015) **Characteristics of tasks** (Bellaaj Elloumi et al., 2020; Yaghoubi & Zahedi, 2016) **Discovered enhanced process model and simulation model** (Djedovic et al., 2018; Yeon et al., 2022)
Used as direct input for the allocation technique and their usage	**To create a heuristic or metaheuristic** (Bessai & Charoy, 2016; Cabanillas et al., 2013; Kamrani et al., 2012; Rhee et al., 2010; Xie et al., 2019; Xu et al., 2016; Zhao et al., 2015) **To create rule or logical programming** (Erasmus et al., 2018; Havur et al., 2016; Kumar et al., 2002; Wibisono et al., 2016; Xu et al., 2009, 2013) **To create a linear programming model** (Hirsch & Ortiz-Peña, 2017; Hou et al., 2021) **To create a ML model** (Huang et al., 2010)	**To mine allocation rules** (Huang, Lu, & Duan, 2011; T. Liu et al., 2012; Schönig et al., 2016) **To cluster tasks** (Pflug & Rinderle-Ma, 2016) **To train a machine learning model** (Park & Song, 2023)

Table 6: Role of process models and data in resource allocation approaches.

allocation goal. The process expert has to evaluate whether the rules are supporting the process goals.

5.2 RQ2: Role of Process Models and Execution Data

This section provides the results to answer *RQ2: What are the respective roles of process models and process execution data in the resource allocation approach?* When analyzing this question, it can be observed that most of the primary studies (39 studies) use process models for resource allocation, and fewer (22 studies) use process execution data in the form of event logs. We summarize our results in Table 6. We further categorized the studies into those that use process model or data to prepare specific artifacts (e.g., a process simulation model), which is then used as input for the resource allocation approach, and those that directly apply it as input.

5.2.1 Usage of process models

Process models capture the activities, events, and their control flow relations. They are used to prepare a specific artifact as input (22 studies) or as direct input (17 studies) to the resource allocation technique (see Table 6). For these studies, the process expert has to feed the proposed approaches by the studies with the process model and additional information, leading to considerable manual work.

A process model can capture further information as attributes of its process elements. The first category of studies (15 studies) enhances process models with additional information. On the one hand, requirements of the activities for the resource allocation are captured, such as required resource types, policies, time constraints, or data. For example, Ihde et al. (2022) require for each process activity the definition of needed resource types, the allocation problem, and the solution technique. On the other hand, business process dynamics are also captured, such as estimates or distributions of the activity execution times, inter-arrival times of process cases, or branching probabilities.

Other studies use process models to create a stochastic model (4 studies) for the resource allocation technique. A stochastic model represents the system's dynamics of the process in which resource allocation occurs. In Huang, van der Aalst, et al. (2011); Soeffker et al. (2019), the process model is used to create a Markov decision process capturing the process states, possible actions, and reward functions. Van Hee et al. (2001) use a stochastic workflow net and Durán et al. (2019, 2021) transform a given BPMN diagram into a Maude simulation model. Both models reflect the order of the process nodes, the distribution of the activities' duration, the branching probabilities, etc.

A third category creates queueing models (2 studies) as a basis for their resource allocation approach. Ha et al. (2006); Yaghoibi and Zahedi (2017) create queueing networks where each resource type used for a process activity presents a node with a queue. Based on the process model, the flows and the probabilities between these resource types are deduced. This network is later used to define rules for balancing the workload of the resources.

As direct input (17 studies), traditional process models are used to create a heuristic or metaheuristic (8 studies) for resource allocation, rules, or logic programming (6 studies). We discuss the concrete solution techniques in Section 5.4 in detail. Three studies use the model as input for linear programming. Finally, one study uses the process model as input to create a machine learning (ML) model.

5.2.2 Usage of process execution data

Process execution data contains information about the past process executions. This data can be leveraged to extract insights and apply these insights to resource allocations in the future. Process execution data is used for resource allocation in 23 studies. Most studies use event logs (cf. Section 2.1), also including resource information. Only Luo et al. (2019) use a simplified log, an activity-employee log where the executed activities, the employee, and needed duration are included.

17 primary studies use data to prepare an artifact for the resource allocation technique. A majority of these studies (13 studies) employ an event log to identify some insights on resources of the process (i.e.,

Resource Attribute	Description
Previous Performance	All performance attributes that are based on the execution history of the resource. The performance of resources can be, for example, the cost, quality, or execution time of previous executions of the resource.
Workload	Attributes that are based on the schedule of a resource. The workload includes attributes such as availability, backlog of allocated tasks, or idle level.
Role	Attributes pertaining to the role of a resource, such as authorizations, organizational position, or responsibilities.
Expertise	Attributes encoding a resource's capabilities, skills, and knowledge. The expertise includes functional attributes associated with the resource directly, such as adaptability, and non-functional attributes, which may include environmental factors and employed aids. It also includes attributes based on work variety, i.e., the analysis of similar and dissimilar tasks in the execution history of a resource.
Resource Amount	Attributes encoding the number of resources that exist. For non-application resources, this may encode the number of resources in stock. For human and application resources, this may encode the number of resources of a specific type.
Social Context	Attributes based on the social network of a resource. These attributes may measure the ability to collaborate or the overall compatibility of resources, but also more abstract social constructs, such as the social position or influence of a resource within its network.
Trustworthiness	Degree of trust to execute a task.
Experience	Attributes such as years of service or other quantifiable attributes based on the experience of a resource. Note the difference to expertise that assesses the actual ability of a resource.
Preference	Attributes expressing the preference for a resource executing certain types of tasks.

Table 7: Brief description of resource attributes, based on Arias, Munoz-Gama, and Sepúlveda (2018).

characteristics of resources), e.g., their previous performance, their expertise, their workload, their team compatibility, or their social context. Other studies identify characteristics of tasks during preparation for resource allocation, such as the similarity between tasks (Yaghoubi & Zahedi, 2016) or misallocations of tasks in the past (Bellaaj Elloumi et al., 2020). Finally, Yeon et al. (2022) discover an enhanced process model with the information on how often a performer has executed an activity from the event log, and Djedovic et al. (2018) learn the process dynamics from the event log and discover a process simulation model with the distribution of the activities' duration, etc.

Six studies use process data as direct input. Of these, three studies use data to mine rules for the allocation. Additionally, Pflug and Rinderle-Ma (2016) use an event log to cluster similar tasks so that they can jointly be allocated and Park and Song (2023) train a machine learning models from event logs.

5.3 RQ3: Resource and Task Attributes

In this subsection, we investigate *RQ3: Which input data are used for resource allocation in business processes?* In a resource allocation, characteristics of both the tasks and resources can be considered to identify a fitting match. These characteristics are often encoded as task and resource attributes and then used as constraints in the resource allocation specification (cf. Definition 3).

5.3.1 Resource attributes

Arias, Munoz-Gama, and Sepúlveda (2018) provide a taxonomy for human resource allocation criteria based on a previously conducted mapping study. We utilized this taxonomy to classify the resource and task attributes discussed in the identified primary studies. The different resource attributes are briefly de-

Resource Attribute	Corresponding Studies
Previous Performance (32)	(Abdulhameed et al., 2018; Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Bellaaj et al., 2017; Bellaaj Elloumi et al., 2020; Bessai & Charoy, 2016; Djedovic et al., 2018; Durán et al., 2019, 2021; Havur et al., 2016; Huang, Lu, & Duan, 2011; Huang et al., 2012a, 2012b, 2010; Huang, van der Aalst, et al., 2011; Kumar et al., 2013; Lee et al., 2019; R. Liu et al., 2022; T. Liu et al., 2012; Park & Song, 2023; Pika & Wynn, 2021; Rhee et al., 2010; Van Hee et al., 2001; Wibisono et al., 2016; Xie et al., 2019; Xu et al., 2013; Yaghoubi & Zahedi, 2016; Yeon et al., 2022; Zhao et al., 2016, 2020, 2015, 2017; Zhou & Chen, 2008)
Workload (23)	(Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Barba et al., 2021, 2013; Bellaaj et al., 2017; Durán et al., 2019, 2021; Eder et al., 2003; Ha et al., 2006; Hou et al., 2021; Huang et al., 2012a, 2012b; Kumar et al., 2013, 2002; Pflug & Rinderle-Ma, 2016; Rhee et al., 2010; Soeffker et al., 2019; Xu et al., 2009, 2016; Yaghoibi & Zahedi, 2017; Yaghoubi & Zahedi, 2016; Zhao et al., 2016, 2015; Zhou & Chen, 2008)
Role (17)	(Bellaaj et al., 2017; Bussler & Jablonski, 1995; Delias et al., 2010; Djedović et al., 2016; Havur et al., 2016; Hirsch & Ortiz-Peña, 2017; Hou et al., 2021; Jemel et al., 2020; Kumar et al., 2013, 2002; Schönig et al., 2016; Wibisono et al., 2016; Xu et al., 2009, 2013; Yaghoibi & Zahedi, 2017; Yu et al., 2020; Zhao et al., 2020)
Expertise (13)	(Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Bessai & Charoy, 2016; Djedović et al., 2016; Huang et al., 2010; Kamrani et al., 2012; R. Liu et al., 2013; Luo et al., 2019; Pereira et al., 2020; Schall et al., 2014; Xu et al., 2013; Yu et al., 2020; Zhao et al., 2016, 2020)
Social Context (8)	(Abdulhameed et al., 2018; Kumar et al., 2013; Lee et al., 2019; R. Liu et al., 2013, 2022; Schall et al., 2014; Yeon et al., 2022; Zhao et al., 2020)
Resource Amount (5)	(Doerner et al., 2006; Durán et al., 2021; Si et al., 2018; Xie et al., 2019, 2016)
Trustworthiness (1)	(Jemel et al., 2020)
Experience (1)	(Zhao et al., 2020)
Preference (1)	(Huang et al., 2010)
Any (5)	(Bussler & Jablonski, 1995; Cabanillas et al., 2013; Doerner et al., 2006; Erasmus et al., 2018; Ihde et al., 2022)

Table 8: Classification of considered resource attributes, based on (Arias, Munoz-Gama, & Sepúlveda, 2018).

scribed in Table 7. Despite their focus on human resources, we were still able to categorize the attributes identified from our set of primary studies. Indeed, we found that most studies (37 studies) focus on human resources exclusively; 24 studies talk about resources in a more general sense. Of these, 15 mention machines or software systems. Notably, only Doerner et al. (2006) considers human, application, and non-application resources; all other studies do not consider non-application resources only.

The result of our resource attribute classification is presented in Table 8, showing the different categories of attributes and the corresponding studies considering them in their allocation.

As shown in Table 8, we found that most studies (32 studies) consider attributes from the category previous performance of a resource. In this category, most studies consider cost (14 studies) as performance criteria. 23 studies consider workload. For workload, availability of a resource is considered most often (13 studies). 17 studies consider the role attribute and 13 studies consider attributes belonging to the expertise category. We found that social context and resource amount are considered less often (8 and 5 studies). Trustworthiness is considered only by Jemel et al. (2020), experience only by Zhao et al. (2020), and preference only by Huang et al. (2010). Five studies take a more flexible approach and can consider any quantifiable attribute. This approach, for example, can be achieved as in Doerner et al. (2006), by defining a custom cost function, or, as in Ihde et al. (2022), by defining configurable input for the selected allocation algorithms.

Task Attributes	Corresponding Studies
Req. Role (20)	(Barba et al., 2021; Bellaaj et al., 2017; Bellaaj Elloumi et al., 2020; Cabanillas et al., 2013; Djedović et al., 2016; Durán et al., 2021; Ha et al., 2006; Hirsch & Ortiz-Peña, 2017; Hou et al., 2021; Huang et al., 2012b; Jemel et al., 2020; Kumar et al., 2002; R. Liu et al., 2022; Pereira et al., 2020; Rhee et al., 2010; Schall et al., 2014; Xu et al., 2009, 2013; Yu et al., 2020; Zhao et al., 2020)
Req. Expertise (8)	(Bessai & Charoy, 2016; Erasmus et al., 2018; Kamrani et al., 2012; Kumar et al., 2013; R. Liu et al., 2022; Luo et al., 2019; Pereira et al., 2020; Schall et al., 2014)
Req. Workload (6)	(Eder et al., 2003; Hou et al., 2021; Ihde et al., 2022; Rhee et al., 2010; Wibisono et al., 2016; Yaghoibi & Zahedi, 2017)
Req. Resource Amount (3)	(Djedovic et al., 2018; Djedović et al., 2016; Kamrani et al., 2012)
Req. Social Context (3)	(Abdulhameed et al., 2018; Yaghoubi & Zahedi, 2016; Yeon et al., 2022)
Req. Performance (1)	(Schall et al., 2014)
Req. Trustworthiness (1)	Jemel et al. (2020)
Estimated Performance (27)	(Barba et al., 2013; Bellaaj Elloumi et al., 2020; Delias et al., 2010; Doerner et al., 2006; Durán et al., 2019; Ha et al., 2006; Havur et al., 2016; Huang, Lu, & Duan, 2011; Huang et al., 2012a; Kamrani et al., 2012; Kumar et al., 2013; T. Liu et al., 2012; Rhee et al., 2010; Schönig et al., 2016; Si et al., 2018; Van Hee et al., 2001; Wibisono et al., 2016; Xie et al., 2016; Xu et al., 2013; Yaghoibi & Zahedi, 2017; Zhao et al., 2016, 2015; Zhou & Chen, 2008)
Priority (5)	(Djedovic et al., 2018; Ihde et al., 2022; Kumar et al., 2002; Zhao et al., 2016, 2017)
Any (5)	(Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Bussler & Jablonski, 1995; Huang et al., 2010; Ihde et al., 2022; Pflug & Rinderle-Ma, 2016)

Table 9: Classification of the considered task attributes.

5.3.2 Task attributes

Contrary to Arias, Munoz-Gama, and Sepúlveda (2018), we also considered task attributes. Task attributes can encode the requirements of what kind of resource is required; thus, they become part of the assignment constraints. These requirements can be mapped onto our previously identified attributes. For example, a task may require a resource with a certain role or past performance (requiring a certain level of quality or time in which the task must be performed). Additionally, we identified attributes influencing allocation decisions more indirectly, i.e., they are do not impose requirements on the resource:

- **Estimated performance** describes performance attributes, such as duration or cost, which are considered for the allocation. Contrary to the *previous performance* attribute of a resource, these are estimated from a task perspective.

- **Priority** contains attributes assessing the priority of a task. Priority may be encoded via a deadline or a simple scale indicating the importance of a task.

Table 9 shows the result of our classification. In total, 34 studies encode some requirements as task attributes. Most studies require the resource to have a specific role (20 studies), certain expertise (8 studies), or a certain workload (6 studies). Three studies consider resource amount and social context. Only Schall et al. (2014) require a performance attribute and Yaghoubi and Zahedi (2016) require a social context attribute. 27 studies consider the estimated performance of a task. Of these, most studies (23) consider time-related attributes (e.g., duration, service time) of a task. Only five studies consider the priority of a task. Jemel et al. (2020) require a certain degree of trustworthiness. While we found one study that considered experience (Zhao et al., 2020) and one that considered the preference (Huang et al., 2010) of a resource, we did not find any study in which a task would in turn require an experience or preference attribute for allocation. Five studies provide flexible means to encode arbitrary quantifiable task attributes to be considered for allocation.

5.4 RQ4: Solution Techniques

Resource allocation is typically viewed as an optimization problem that needs to be solved (Kamrani et al., 2012; Park & Song, 2023; Zhao et al., 2016). In the following, we investigate *RQ4: Which solution strategies are used?* In the primary studies, we could identify different types of solution techniques, namely, the categories *rules or logic programming, heuristics and metaheuristics, exact algorithms, and mining and machine learning*. A special case that does not fit the given categorization is Ihde et al. (2022), categorized as *any*. The approach given by (Ihde et al., 2022) allows the process designer to define any solution technique and its goals for each activity individually.

When deciding on a solution strategy, it is crucial to balance the effort and time needed to generate a fitting solution with the quality[7] of the resulting solution (Cormen, Leiserson, Rivest, & Stein, 2022). In this work, we want to provide an indicative categorization of solution techniques regarding their execution cost (i.e., their computational effort) and solution quality, as typically observed in the average case. Figure 7 provides an intuitive sketch of our categorization. Exceptions might exist but are not shown here. Currently, a quantitative benchmark of each individual technique from the 61 studies is not feasible. This limitation arises not only from the sheer number but also because most approaches lack publicly available prototypes; a re-implementation of the studies for comprehensive benchmarking is hampered by the replicability of many of the studies, as discussed in Section 5.5.

Solution techniques that focus primarily on the quality of the result tend to have much higher execution costs (i.e., effort and time). The most prominent representatives of these techniques are exact algorithms solving linear or non-linear programs (Nickel, Rebennack, Stein, & Waldmann, 2022). On the other end of the spectrum are solution techniques such as rules and logic programming, which minimize the execution cost in exchange for not finding the best solution in all cases (Havur et al., 2016).

[7] In general, the solution quality can be measured on the basis of a solution quality function, as captured in Definition 4, that calculates how close the solution is to the most optimized solution possible.

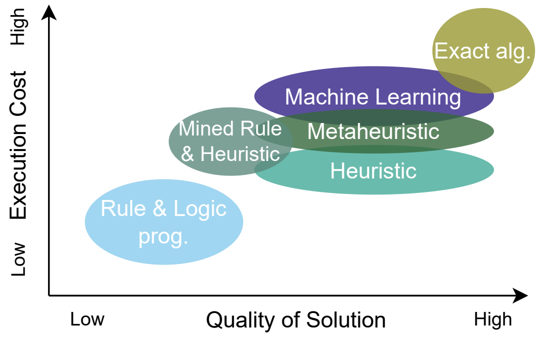


Figure 7: Identified solution strategies for resource allocation in business processes and an indicative categorization regarding their execution cost and solution quality in the average case.

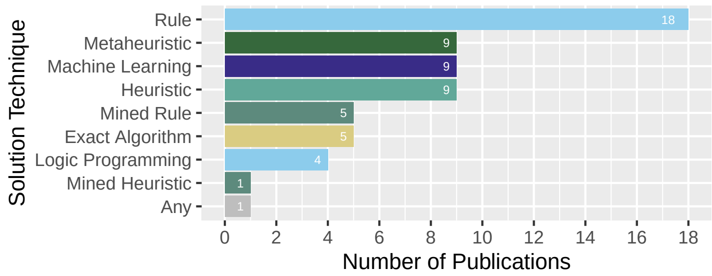


Figure 8: Found solution techniques assigned to algorithm categories.

Heuristics and metaheuristics can be positioned in the middle of this spectrum. They tend to result in a higher quality of the solution compared to rule-based approaches (Havur et al., 2016; Nickel et al., 2022), while exhibiting only slightly higher execution cost. Usually, metaheuristics posses a higher execution cost but are also often able to provide higher solution quality than heuristics (Xu et al., 2016). Some approaches are trained based on historical data: mined rules/heuristics and machine learning approaches. Rulesheuristics are mined from historical process execution data. As a result, solution quality tends to be higher than the non-mined version (Zhao et al., 2016). Still, mined rules also only consider a constraint solution space. The execution costs are, on average, higher if the mining activity is also taken into account—for example, by apportioning it as cost to the execution costs per allocation. Compared to mined rules, machine learning approaches tend to find, on average, higher quality solutions. Machine learning acts problem-specific due to the knowledge deduced from history. However, the effort is higher due to the more involved training phase.

In the following, we discuss the solution approaches in more detail, starting with techniques which tend towards lower execution cost and quality: rules and logic programming $\rightarrow$ heuristics and metaheuristics $\rightarrow$ approaches using exact algorithms$\rightarrow$ approaches that use historical data: mining and machine learning approaches. At the end of this section, we compare the solution techniques regarding their support concerning single- or multi-task optimization.

5.4.1 Rules & Logic Programming

A large share of the studies (22 studies) follow a rule-based solution technique. Rules minimize the effort and time to find a suitable solution but tend to provide lower quality results compared to other types of solution techniques (Cormen et al., 2022). There could be cases where a rule-based approach returns a high-quality solution. However, in the average case, this cannot be expected (Havur et al., 2016). These techniques are especially useful in scenarios where a solution needs to be found in a relatively short timeframe. For example, Ha et al. (2006) propose a dynamic rebalancing rule executed as soon as a resource is idle. It assigns a task to the resource's task list by a selected dispatching rule, such as the earliest due date. Some studies use information about the resources previously identified from the process execution data. For instance, Bellaaj et al. (2017) deduce so-called obstacle-aware resource indicators. These indicators help assess the performance of a resource in terms of time, cost, and quality under a specific workload. These indicators are then considered in a rule in combination with the current workload to identify an ordered list of candidate resources.

We also decided to group logic programming approaches with rules, as logic programming defines constraints for machine execution, which are later used for BPMSs to execute processes. Therefore, logic programming approaches strongly resemble rule-based approaches. However, with only four (Barba et al., 2021, 2013; Havur et al., 2016; Jemel et al., 2020) studies out of the 22, they represent a minority.

5.4.2 Heuristics and Metaheuristics

The second biggest identified group comprises heuristics and meta-heuristics (18 studies).

Heuristics are specialized algorithms tailored to the unique characteristics of a specific optimization problem (Stork, Eiben, & Bartz-Beielstein, 2022). They try to provide fast numerical solutions, but aim to minimize the chance of ending up with only a locally best solution. Therefore, they tend to generate, on average, higher-quality solutions than rule-based approaches. However, they tend to require more computational effort and time to produce a solution (Nickel et al., 2022). In sum, nine studies use a heuristic. For instance, Kumar et al. (2013) formulate an assignment problem with the goal to maximize the resource compatibility. Based on this, they develop a greedy heuristic where the compatibility is optimized for running cases sequentially but not globally. By comparing the heuristic with the exact solution of the optimization algorithms, the authors find that the solution quality of the heuristic is, on average, worse by 17–19%.

Metaheuristics are general-purpose optimization algorithms that are problem-independent. They are versatile and can be applied to a broad spectrum of different problems and their specific instances Stork et al. (2022). Six of the nine studies using a metaheuristic apply a genetic algorithm. For example, Xu et al. (2016) use a genetic algorithm to assign the tasks of a set of running process cases to available resources, maximizing the process throughput. It is compared to a set of heuristics which it outperforms. Other works apply stochastic branch-and-bound (Doerner et al., 2006), ant colony optimization (Huang et al., 2012b), and particle swarm optimization (Zhao et al., 2017)

5.4.3 Exact algorithms

Five studies have employed exact algorithms, also referred to as non-heuristic or complete algorithms (Stork et al., 2022). These studies formulate the optimization problem as either a linear or a non-linear program. Alternatively, Delias et al. (2010) formulates a continuous optimization problem transferred into a discrete one.

An exact solution approach, such as the branch-and-bound algorithm, evaluates each possible solution and compares it to find the best one. This guarantees an optimal solution in contrast to the approaches presented before. The drawback is that evaluating every possible solution requires high computational effort. An example of this group is (Hirsch & Ortiz-Peña, 2017): they formulate a nonlinear mixed-integer programming problem and produce a linearized version. The authors suggest solving it with a professional tool and, additionally, propose heuristics for the cases where no solution can be found in an acceptable time.

5.4.4 Mining and Machine Learning

The last group of approaches uses solution techniques that are individually adapted to a particular business process scenario by learning from historical data. Therefore, they tend to result in higher quality allocations. An additional effort is the mining or training on the historical data that could be apportioned to the execution costs per allocation. The cost highly depends on how often the mining or training is executed. Further points to consider are: (i) the dependence on the quality of the historical data, and (ii) whenever a change occurs in the setting (i.e., a new resource gets added, processes change, etc.), the algorithm has to be retrained. 11 studies support individualized resource allocation techniques for certain business process scenarios:

- *Mined rules and heuristics*: Six papers used their adaptive approach to learn allocation rules or heuristics from past process executions. Most studies use an association rule mining approach to identify relevant rules. For example, Schönig et al. (2016) mine the resource patterns (Russell et al., 2005), such as the *Retain Familiar* pattern, from an event log.

- *Machine learning*: The remaining nine studies use supervised machine learning approaches (3 studies, e.g., decision tree, bayesian neural network), unsupervised machine learning approaches (3 studies, e.g., clustering, generative probabilistic learning) and reinforcement learning (3 studies). An example of the latter is Huang, van der Aalst, et al. (2011): The authors apply Q-learning to make allocation decisions similar to a human decision-maker as soon as a new task is enabled. Based on a predefined goal, the new task is assigned to a qualified resource. The reinforcement agent continuously learns from the dynamics of the past process execution to make better decisions in the future. Comparing the approach to simple dispatching rules, such as FIFO, the authors show that it results in higher solution quality. Yaghoubi and Zahedi (2016) reuse this approach to minimize the entropy of a task list and allocate similar tasks to a resource.

Comparing single vs. multi-task optimization. Generally, a resource allocation decision is made at a specific time, meaning the currently available information about tasks and resources is then used to create a solution. We categorized the approaches that focus on a narrow scope, such as finding the most qualified resource for a specific process task as *single-task* optimization approaches (24 studies). Approaches that consider all available tasks of a process or an organization and their different level of importance are considered as *multi-task* optimization (37 studies).

Resource allocation, in general, is primarily an NP-hard problem. Therefore, a common way to handle these problems is to reduce the complexity as much as possible while still attaining meaningful results. Single-task optimization achieves this by reducing the complexity by limiting the problem to select the best-matching resource for a specific task. This might be useful in business processes where resources are not limited or the importance between tasks is the same. Even though a single-task approach reduces the complexity of the problem, most studies further limit their approach by only allowing a 1-to-1 allocation of tasks to resources (except four papers (Bussler & Jablonski, 1995; R. Liu et al., 2013; Schall et al., 2014; Van Hee et al., 2001; Zhao et al., 2020)).

Most studies (37) support a multi-task optimization approach. Despite the higher complexity of

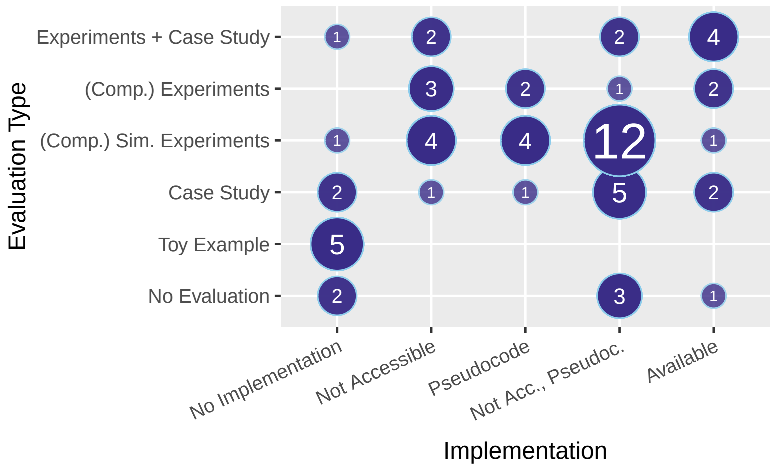


Figure 9: Evaluation methods and the associated research prototypes per method.

multi-task approaches, only 21 papers reduce the complexity by limiting the allocation to a 1-to-1 allocation of tasks to resources.

5.5 RQ5: Evaluation and Research Prototypes

In this subsection, we discuss the methods applied by the studies to evaluate their approach and the availability of research prototypes. Thus, in this subjection, we investigate *RQ5: How applicable are the proposed resource allocation approaches, given the availability of evaluations and prototypes?*

Based on work by Zelkowitz and Wallace (1998), we have categorized the approaches into those having no evaluation, argumentation on a toy example (i.e., an assertion with regards to Zelkowitz and Wallace (1998)), case study, controlled experiments, whereby we differentiated between simulation experiments with synthetic data, and experiments with real-world data, and a combination of methods (*Experiments + Case study*). Furthermore, we distinguish between works having no implementation, a not accessible prototype, only pseudocode available, or a prototype that is not (publicly) but has accessible pseudocode, and finally, a prototype available. Figure 9 shows the number per evaluation category and implementation type as a bubble diagram.

Surprisingly, we can observe that five studies provide no evaluation at all. Five discuss their approach on a toy example, which is rather an assertion about the functionality and usefulness of their approach. By summing up these two categories, it can be observed that 18% of the studies have not adequately evaluated their approaches. Cabanillas et al. (2013) is an interesting case in this category: the authors provide a Java implementation where the approach can be tested and used but did not include an evaluation in their paper. Three other works with no evaluation discuss a prototype, but they are not publicly accessible; only pseudocode is provided.

Eleven studies employ case studies to evaluate their approach. The advantages of a case study are that the implications of the resource allocation approach can be studied in detail, and interesting insights can be found. However, it is challenging to see which results are generalizable and which are not (Zelkowitz & Wallace, 1998). Most of these approaches have a prototype in place to apply the approach to the selected case. However, only two prototypes are publicly accessible: Huang et al. (2012a) provide a plugin for the process mining toolkit ProM (Kalenkova, De Leoni, & van der Aalst, 2014), and Ihde et al. (2022) provide a stand-alone system implementation.

A majority of studies evaluate their approaches with controlled experiments (see the *[Comp.] sim. experiments* and *[Comp.] experiments* in Figure 9), whereby most of the studies use synthetic data from simulations (23 studies). Many of these simulation experiments evaluate different parameters of their approaches with regard to different settings. Six studies (Djedovic et al., 2018; Hou et al., 2021; Huang et al., 2012b; Soeffker et al., 2019; Wibisono et al., 2016; Xu et al., 2016) also conduct comparative simulation experiments, in which their solution is compared to other approaches. Only Durán et al. (2019) provide a prototype publicly available on a website. For the other 22 *[Comp.] sim. experiments*-studies in this category, no prototype is accessible, but most of them provide pseudocode (16 studies).

Eight studies have conducted experiments with real-world data. This type of evaluation can act on more realistic data and typically provide observations and insights with higher confidence that these would hold in practice. Five conducted comparative experiments (Huang, van der Aalst, et al., 2011; T. Liu et al., 2012; Yaghoibi & Zahedi, 2017; Zhao et al., 2016, 2017). ProM plugins as prototypes are provided by Huang et al. (2010); Huang, van der Aalst, et al. (2011). The other studies of this category provide mainly pseudocode. However, several make neither a prototype nor pseudocode available.

Nine studies use a combination of evaluation methods with controlled experiments and case studies (see the *Experiments + Case studies* in Figure 9) to strengthen the evaluation of their approaches. Arias, Munoz-Gama, Sepúlveda, and Miranda (2018) also provide a ProM plugin and Xie et al. (2016) a Math-Lab implementation as research prototypes. Pika and Wynn (2021) and Park and Song (2023) provide stand-alone implementations.

In summary, we can observe that many studies provide no publicly available prototype, but mainly pseudocode. Pseudocode heavily depends on how detailed the concepts of an approach are described within the study. Some studies with no accessible prototype have linked to a stand-alone solution, which is not accessible anymore. Of the ten studies providing a prototype, five have used public platforms, i.e., the process mining platform ProM (Arias, Munoz-Gama, Sepúlveda, & Miranda, 2018; Huang et al., 2012a, 2010; Huang, van der Aalst, et al., 2011) or the mathematical toolbox MathLab (Xie et al., 2016), to implement their resource allocation approaches. In contrast, the other five implemented stand-alone prototypes.

6 Discussion

The literature survey presented above shows a strong global research interest in automatic support for resource allocation in business processes and a variety of approaches based on different techniques. Based on our results, we observe several open research challenges (see Section 6.1). Furthermore, we discuss implications for practitioners in Section 6.2 and threats to validity in Section 6.3.

6.1 Observations and Future Directions

Our various classification results can be leveraged to spot gaps and neglected research directions. In this Section, we report on our main observations and envision future research to: leverage process execution data, explore additional aspects of resource and task characteristics, increase adaptability, and conduct comprehensive performance studies.

6.1.1 Leverage Process Execution Data

Most approaches use a process model with estimations of the process dynamics (e.g., activity duration, arrival rates of new process cases) as input for the resource allocation algorithm (c.f. Section 5.2). However, this leaves aside information on past executions of processes, which is available in historic process execution data (e.g., event logs). Such datasets have become more available and accessible in recent years. They can be leveraged to replace or fine-tune estimates of process dynamics by investigating the past performance of processes. Several approaches already use this data to gain insights into a resource's behavior and preferences. However, characteristics of tasks (e.g., similarity of tasks) are less investigated. Also, only two studies use process execution data to learn about the overall dynamics of a business process. Yeon et al. (2022) discover an enhanced process model and Djedovic et al. (2018) create a process simulation model from execution data. We expect that future research will increasingly make use of historic execution data. Here, we see a line for future research in making use of aspects of execution data previously not explored and using execution data as input for solution techniques.

6.1.2 Explore Additional Aspects of Resource and Task Characteristics

While many approaches consider attributes like cost or availability of a resource, only few published studies consider resources' preferences (Huang et al., 2010), their experience (Zhao et al., 2020) or trustworthiness (Jemel et al., 2020). Task characteristics, such as the priority of tasks, are also considered less frequently. However, these may play a similarly relevant role in finding a suitable resource allocation. Our classification of resource and task attributes in Section 5.3 can be leveraged to identify such gaps and investigate them in future research. Similarly, only Doerner et al. (2006) considers non-application resources. Such research can be important to process cases where non-application resources constitute a real bottleneck; for example, when highly specialized and costly equipment is required to complete a task. We envision a future line of research investigating such processes.

6.1.3 Increase Adaptability

Most approaches focus on human resource allocation and specific allocation goals (e.g., minimizing the process cost). Few offer the possibility to customize the resource allocation goal, e.g., (Huang, van der Aalst, et al., 2011). Many approaches published in BPM outlets follow the idea that, to reach the overall optimization goal, the resource allocation approach should be the same for all process activities. In contrast, in operations research, techniques are proposed and developed that match the specifics of a particular activity, e.g., which type of resources are needed. Ihde et al. (2022) propose an approach to select a resource allocation technique for specific process activities individually. This follows the idea of the resource patterns (Russell et al., 2005) in BPMSs that can be selected per activity. However, these kind of approaches give no decision support on how an allocation approach applied to a process activity influences the overall process goal.

Thus, we envision future research exploring approaches adaptable to different process settings, and providing decision support on when a given approach should take preference over another, contributing to the overall allocation goal.

6.1.4 Conduct Comprehensive Performance Studies

In Section 5.5, we found that most studies use synthetic data in simulations to demonstrate the effectiveness of their approach. So far, no large scale benchmark studies have been conducted comparing different resource allocation approaches, replicating and contextualizing their results. We believe this is a pressing research topic. Such research can reveal new insights and future research directions and support practitioners when transferring different approaches into real-world applications. Such a study should go beyond runtime performance. As we discussed in Section 5.4, usually a trade-off between quality of the allocation and time to find an allocation exists. For example, rules may be fast but lead to solutions which might not always be optimal, whereas linear programming usually provides high-quality solutions but may take longer to compute. However, conducting such a study will be challenging. Only a minority of studies provide a replicable evaluation of their approach and a publicly available prototype. In the future, researchers should emphasize making their approaches, prototypes, and data sets available, ideally following open science principles enabling replicability and comparability.

For studies utilizing comparative experiments to compare the effectiveness to other approaches we could observe that the selection of approaches for the comparison was often not done in a transparent and replicable manner. The results of this SLR can help researchers identify related approaches for evaluation in a more structured manner in the future.

6.2 Insights for Practitioners

Classifications like ours can provide insights into the design process of artifacts (Williams, Chatterjee, & Rossi, 2008). In this section, we propose a decision flow to illustrate how our various classifications can be used to guide the design of future practical implementations. In particular, our decision flow—depicted in Figure 10—can be used to select relevant studies given specific requirements. This flow can be used to identify applicable studies, guide practical implementations, and use these techniques in resource allocation implementations of BPMSs. First, we suggest identifying the required allocation capability. If a 1-to-1 allocation of one task to one resource is needed, all studies can be considered because studies supporting 1-to-many or many-to-1 also support the basic variant of 1-to-1. If 1-to-many or many-to-1, the related studies should be selected (cf. Section 5.1). Subsequently, it should be decided whether multiple to-be-allocated tasks of a process and their different levels of importance should be considered at any allocation. If not, again, all studies can be considered. Otherwise, a multi-task optimization approach is required (cf. Section 5.4).

The presented studies support different goals in their resource allocation approach. It is helpful to select studies that support the same goal as in the intended business scenario or studies that support *Any* goal (cf. Section 5.1). An alternative is to adopt and adapt an approach in a way that follows the needed optimization goal.

After the goal of the resource allocation has been defined; also the constraints need to be specified for which different resource and task attributes need be considered (c.f. 5.3). Based on these, studies can be selected that also apply similar attributes. Otherwise, the selected approaches could be extended by the required constraints.

A fifth relevant selection criterion is the available process information. If process execution data is available, a process model can be discovered with the help of process discovery techniques (Dumas et al., 2018); then, all remaining studies can be considered. If not, studies that support a process model as input should be selected. Finally, the trade-offs between solution quality an execution time/cost have to be considered. As discussed in Section 5.4, some classes of solution techniques tend to exhibit higher solution quality but may result in higher execution time/cost, such as linear approaches. Others tend to exhibit low execution times/costs but may lead to lower solution qualities, such as rules.

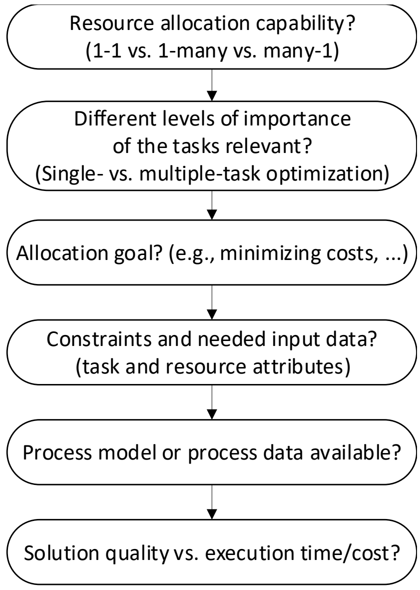


Figure 10: Decision flow for selecting relevant studies for an application process.

6.3 Threats to Validity

In an SLR, biases in the selection of the studies and the data extraction process can be threats to the validity of its results (Cooper, 2015). To avoid selection bias in the study search, we followed a specific search protocol described in Section 4.2. We further conducted the relevance check based on defined inclusion and exclusion criteria with at least two co-authors, with discussions between them for all cases of disagreement. The primary search needed to be limited by the search terms and was additionally limited by focusing on journal articles. We have observed a broad range of short conference/workshop articles on this research topic, presenting idea sketches. So, we decided to focus on more mature work expected to be found in journals. We complemented our search by considering conference and workshop papers in the secondary search, the backward/forward search. The study search was initially conducted in mid-2019; to consider more recent publications, we repeated the search at the beginning of 2021 and again at the end of 2023. In the full-text reading, we excluded duplicates in content. We only kept the more mature version of the papers, usually the journal article, to avoid double counting one approach.

The group of co-authors discussed each exclusion. In sum, we selected 61 primary studies which we believe represent the research field well. Still, the risk exists that relevant studies might not have been included since they did not meet our search criteria.

Additionally, biases can exist in the data extraction process, which we mitigated as follows. Data coding was done first individually for each paper by different co-authors. Issues and ambiguities were discussed with the co-authors. After having finished the data extraction from all papers, data categories were distributed among the authors, and the data extraction per category was validated and standardized. Nevertheless, studies sometimes do not provide the information directly on a particular aspect, so the authors need to form interpretations.

7 Conclusion

This survey provide a structured analysis of system-initiated approaches for resource allocation in business processes. The complexity of the problem of assigning tasks to available resources with different capabilities has led to various approaches with different strengths and weaknesses. The structured literature search identified 61 studies providing resource allocation approaches published mainly in the last two decades. In this survey, for the first time, the approaches have been analyzed in terms of their goals, capabilities, input data, techniques used, and evaluation methods. With this SLR, we studied five research questions and come to the following conclusions.

Regarding research question RQ1 (the targeted resource allocation capabilities and goals), we found that mainly 1-to-1 allocations between tasks and resources are supported, but also studies could be identified that support many-to-1 and 1-to-many allocations. Several optimization goals are pursued, such as minimizing process costs, whereby process-oriented goals are mainly supported. Regarding RQ2 (the role of process models and process execution data), we found that process models and estimations on the process dynamics are often used as inputs to the resource allocation approaches, and process execution data taken from IT systems increasingly plays a role. To answer RQ3, we classified resource and task attributes and the relation between them. This can be used in future research to identify possible attributes that were neglected and should be considered in the future.

Allocation rules have the disadvantage of not always leading to the best solution. Nevertheless, they are utilized in many approaches, as observed in the context of RQ4 (solution strategies). Because such rules provide solutions in a short timeframe, they are still relevant for many business processes. Additionally, the primary studies used heuristics, metaheuristics, and exact algorithms as solution techniques, and trained rules and machine-learning approaches. The latter two provide more context-sensitive solutions if the training data is of good quality and representative of future cases. Regarding RQ5 (evaluation strength and prototype availability), we showed that many studies evaluate their approaches with simulation experiments, but only a few works provide publicly available prototypical implementations.

This survey offers researchers an overview of existing approaches, open research challenges, and the possibility of identifying structurally related approaches for comparison. Based on our observations, we have outlined possible future research opportunities: leverage process execution data, explore additional aspects of resource and task characteristics, increase adaptability, and conduct comprehensive performance studies. Our various classifications can be used to guide the design of future research or practical implementations. One major limitation of many existing works is their replicability. This makes comprehensive performance studies difficult. Thus, in the future, we plan to establish a benchmarking framework for resource allocation approaches to allow better comparisons of the functionality and complexity of the approaches.

References

Abdulhameed, N. M., Helal, I. M., Awad, A., & Ezat, E. (2018). A resource recommendation approach based on co-working history. *International Journal of Advanced Computer Science and Applications*, *9*(7).

Arias, M., Munoz-Gama, J., & Sepúlveda, M. (2018). Towards a taxonomy of human resource allocation criteria. In *Business process management workshops* (pp. 475–483). Springer International Publishing.

Arias, M., Munoz-Gama, J., Sepúlveda, M., & Miranda, J. C. (2018). Human resource allocation or recommendation based on multi-factor criteria in on-demand and batch scenarios. *European Journal of Industrial Engineering*, *12*(3), 364–404.

Arias, M., Saavedra, R., Marques, M. R., Munoz-Gama, J., & Sepúlveda, M. (2018). Human resource allocation in business process management and process mining: A systematic mapping study. *Management Decision*, *56*(2), 376–405.

Barba, I., Jimenez-Ramirez, A., Reichert, M., Del Valle, C., & Weber, B. (2021). Flexible runtime support of business processes under rolling planning horizons. *Expert Systems with Applications*, *177*, 114857.

Barba, I., Weber, B., Del Valle, C., & Jiménez-Ramírez, A. (2013). User recommendations for the optimized execution of business processes. *Data & knowledge engineering*, *86*, 61–84.

Bellaaj, F., Sellami, M., Bhiri, S., & Maamar, Z. (2017). Obstacle-aware resource allocation in business processes. In *International conference on business information systems* (pp. 207–219).

Bellaaj Elloumi, F., Sellami, M., & Bhiri, S. (2020). Avoiding resource misallocations in business processes. *Concurrency and Computation: Practice and Experience*, *32*(15), e4888.

Bessai, K., & Charoy, F. (2016). Business process tasks-assignment and resource allocation in crowd-sourcing context. In *2016 ieee 2nd international conference on collaboration and internet computing (cic)* (pp. 11–18).

Bussler, C., & Jablonski, S. (1995). Policy resolution for workflow management systems. In *Proceedings of the twenty-eighth annual hawaii international conference on system sciences* (Vol. 4, pp. 831–840).

Cabanillas, C. (2016). Process-and resource-aware information systems. In *Edoc, 2016 ieee 20th international* (pp. 1–10).

Cabanillas, C., García, J. M., Resinas, M., Ruiz, D., Mendling, J., & Ruiz-Cortés, A. (2013). Priority-based human resource allocation in business processes. In *International conference on service-oriented computing* (pp. 374–388).

Cooper, H. (2015). *Research synthesis and meta-analysis: A step-by-step approach* (Vol. 2). Sage publications.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms*. MIT press.

Delias, P., Doulamis, A., Doulamis, N., & Matsatsinis, N. (2010). Optimizing resource conflicts in workflow management systems. *IEEE Transactions on Knowledge and Data Engineering*, *23*(3), 417–432.

Djedovic, A., Karabegovic, A., Avdagic, Z., & Omanovic, S. (2018). Innovative approach in modeling business processes with a focus on improving the allocation of human resources. *Mathematical Problems in Engineering*, *2018*.

Djedović, A., Žunić, E., Avdagić, Z., & Karabegović, A. (2016). Optimization of business processes by automatic reallocation of resources using the genetic algorithm. In *2016 xi international symposium on telecommunications (bihtel)* (pp. 1–7).

Doerner, K., Gutjahr, W. J., Kotsis, G., Polaschek, M., & Strauss, C. (2006). Enriched workflow modelling and stochastic branch-and-bound. *European journal of operational research*, *175*(3), 1798–1817.

Dumas, M., La Rosa, M., Mendling, J., Reijers, H. A., et al. (2018). *Fundamentals of business process management* (Vol. 2). Springer.

Durán, F., Rocha, C., & Salaün, G. (2019). A rewriting logic approach to resource allocation analysis in business process models. *Science of Computer Programming*, *183*, 102303.

Durán, F., Rocha, C., & Salaün, G. (2021). Resource provisioning strategies for bpmn processes: Specification and analysis using maude. *Journal of Logical and Algebraic Methods in Programming*, *123*, 100711.

Eder, J., Pichler, H., Gruber, W., & Ninaus, M. (2003). Personal schedules for workflow systems. In *International conference on business process management* (pp. 216–231).

Erasmus, J., Vanderfeesten, I., Traganos, K., Jie-A-Looi, X., Kleingeld, A., & Grefen, P. (2018). A method to enable ability-based human resource allocation in business process management systems. In *Ifip working conference on the practice of enterprise modeling* (pp. 37–52).

Ha, B.-H., Bae, J., Park, Y. T., & Kang, S.-H. (2006). Development of process execution rules for workload balancing on agents. *Data & Knowledge Engineering*, *56*(1), 64–84.

Havur, G., Cabanillas, C., Mendling, J., & Polleres, A. (2016). Resource allocation with dependencies in business process management systems. In *Bpm* (pp. 3–19).

Hirsch, M. J., & Ortiz-Peña, H. (2017). Information supply chain optimization with bandwidth limitations. *International Transactions in Operational Research*, *24*(5), 993–1022.

Hou, S., Ni, W., Wang, M., Liu, X., Tong, Q., & Chen, S. (2021). Bottleneck-aware resource allocation for service processes: A new max-min approach. *International Journal of Web Services Research (IJWSR)*, *18*(3), 1–21.

Howard, A., Kochhar, A., & Dilworth, J. (1999). Application of a generic manufacturing planning and control system reference architecture to different manufacturing environments. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *213*(4), 381–396.

Huang, Z., Lu, X., & Duan, H. (2011). Mining association rules to support resource allocation in business process management. *Expert Systems with Applications*, *38*(8), 9483–9490.

Huang, Z., Lu, X., & Duan, H. (2012a). Resource behavior measure and application in business process management. *Expert Systems with Applications*, *39*(7), 6458–6468.

Huang, Z., Lu, X., & Duan, H. (2012b). A task operation model for resource allocation optimization in business process management. *IEEE Transactions on Systems, man, and cybernetics-part a: systems and humans*, *42*(5), 1256–1270.

Huang, Z., van der Aalst, W. M., Lu, X., & Duan, H. (2010). An adaptive work distribution mechanism based on reinforcement learning. *Expert Systems with Applications*, *37*(12), 7533–7541.

Huang, Z., van der Aalst, W. M., Lu, X., & Duan, H. (2011). Reinforcement learning based resource allocation in business process management. *Data & Knowledge Engineering*, *70*(1), 127–145.

Ihde, S., Pufahl, L., Völker, M., Goel, A., & Weske, M. (2022). A framework for modeling and executing task-specific resource allocations in business processes. *Computing*, *104*(11), 2405–2429.

Jemel, M., Ben Azzouna, N., & Ghedira, K. (2020). Rpminter-work: a multi-agent approach for planning the task-role assignments in inter-organisational workflow. *Enterprise Information Systems*, *14*(5), 611–640.

Kalenkova, A. A., De Leoni, M., & van der Aalst, W. M. (2014). Discovering, analyzing and enhancing bpmn models using prom. In *Bpm (demos)* (p. 36).

Kamrani, F., Ayani, R., & Moradi, F. (2012). A framework for simulation-based optimization of business process models. *Simulation*, *88*(7), 852–869.

Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, *33*(2004), 1–26.

Kumar, A., Dijkman, R., & Song, M. (2013). Optimal resource assignment in workflows for maximizing cooperation. In *Business process management* (pp. 235–250). Springer.

Kumar, A., Van Der Aalst, W. M., & Verbeek, E. M. (2002). Dynamic work distribution in workflow management systems: How to balance quality and performance. *Journal of Management Information Systems*, *18*(3), 157–193.

Lee, J., Lee, S., Kim, J., & Choi, I. (2019). Dynamic human resource selection for business process exceptions. *Knowledge and Process Management*, *26*(1), 23–31.

Liu, R., Agarwal, S., Sindhgatta, R. R., & Lee, J. (2013). Accelerating collaboration in task assignment using a socially enhanced resource model. In *Business process management* (pp. 251–258). Springer.

Liu, R., Kumar, A., & Lee, J. (2022). Multi-level team assignment in social business processes: An algorithm and simulation study. *Information Systems Frontiers*, 1–21.

Liu, T., Cheng, Y., & Ni, Z. (2012). Mining event logs to support workflow resource allocation. *Knowledge-Based Systems*, *35*, 320–331.

Luo, Z., Liu, L., Yin, J., Li, Y., & Wu, Z. (2019). Latent ability model: A generative probabilistic learning framework for workforce analytics. *IEEE Transactions on Knowledge and Data Engineering*, *31*(5), 923-937. doi: 10.1109/TKDE.2018.2848658

Luss, H. (2012). *Equitable resource allocation: models, algorithms and applications*. John Wiley & Sons.

Marrella, A. (2019). Automated planning for business process management. *Journal on data semantics*, *8*(2), 79–98.

Nickel, S., Rebennack, S., Stein, O., & Waldmann, K.-H. (2022). *Operations research*. Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from https://doi.org/10.1007/978-3-662-65346-3_5 doi: 10.1007/978-3-662-65346-3_5

Okoli, C., & Schabram, K. (2010). A guide to conducting a systematic literature review of information systems research. *Sprouts: Working Papers on Information Systems*, *10*(26), http://sprouts.aisnet.org/10-26.

OMG. (2011). Notation BPMN version 2.0. *OMG Specification, Object Management Group*, 22–31.

Ouyang, C., Wynn, M. T., Fidge, C., ter Hofstede, A. H., & Kuhr, J.-C. (2010). Modelling complex resource requirements in business process management systems. *ACIS 2010 Proceedings*.

Park, G., & Song, M. (2023). Optimizing resource allocation based on predictive process monitoring. *IEEE Access*.

Pereira, J.-L., Varajão, J., & Uahi, R. (2020). A new approach for improving work distribution in business processes supported by bpms. *Business Process Management Journal*.

Pflug, J., & Rinderle-Ma, S. (2016). Application of dynamic instance queuing to activity sequences in cooperative business process scenarios. *International Journal of Cooperative Information Systems*, *25*(01), 1650002.

Pika, A., & Wynn, M. T. (2021). A machine learning based approach for recommending unfamiliar process activities. *IEEE Access*, *9*, 104969–104979.

Remy, S., Pufahl, L., Sachs, J. P., Böttinger, E., & Weske, M. (2020). Event log generation in a health system: A case study. In *International conference on business process management* (pp. 505–522).

Rhee, S.-H., Cho, N. W., & Bae, H. (2010). Increasing the efficiency of business processes using a theory of constraints. *Information Systems Frontiers*, *12*(4), 443–455.

Russell, N., van der Aalst, W. M. P., ter Hofstede, A. H., & Edmond, D. (2005). Workflow resource patterns: Identification, representation and tool support. In *Caise* (pp. 216–232).

Schall, D., Satzger, B., & Psaier, H. (2014). Crowdsourcing tasks to social networks in bpel4people.

World Wide Web, *17*(1), 1–32.

Scheer, A.-W., Thomas, O., & Adam, O. (2005). Process modeling using event-driven process chains. *Process-aware information systems*, *119*.

Schönig, S., Cabanillas, C., Jablonski, S., & Mendling, J. (2016). A framework for efficiently mining the organisational perspective of business processes. *Decision Support Systems*, *89*, 87–97.

Si, Y.-W., Chan, V.-I., Dumas, M., & Zhang, D. (2018). A petri nets based generic genetic algorithm framework for resource optimization in business processes. *Simulation Modelling Practice and Theory*, *86*, 72–101.

Slack, N., & Brandon-Jones, A. (2018). *Operations and process management: principles and practice for strategic impact*. Pearson UK.

Soeffker, N., Ulmer, M. W., & Mattfeld, D. C. (2019). Adaptive state space partitioning for dynamic decision processes. *Business & Information Systems Engineering*, *61*(3), 261–275.

Stork, J., Eiben, A. E., & Bartz-Beielstein, T. (2022). A new taxonomy of global optimization algorithms. *Natural Computing*, *21*(2), 219–242.

Van der Aalst, W. M. (1998). The application of petri nets to workflow management. *Journal of circuits, systems, and computers*, *8*(01), 21–66.

Van der Aalst, W. M. (2013). Business process management: a comprehensive survey. *International Scholarly Research Notices*, *2013*.

Van der Aalst, W. M., & Kumar, A. (2001). A reference model for team-enabled workflow management systems. *Data & Knowledge Engineering*, *38*(3), 335–363.

van Der Aalst, W. M., Pesic, M., & Schonenberg, H. (2009). Declarative workflows: Balancing between flexibility and support. *Computer Science-Research and Development*, *23*(2), 99–113.

van der Aalst, W. M. P. (2016). *Process mining: data science in action*. Heidelberg: Springer.

Van Hee, K., Reijers, H., Verbeek, H., & Zerguini, L. (2001). On the optimal allocation of resources in stochastic workflow nets. In *Proceedings of the seventeenth uk performance engineering workshop* (pp. 23–34).

Wei, Y., & Blake, M. B. (2016). Proactive virtualized resource management for service workflows in the cloud. *Computing*, *98*(5), 523–538.

Weske, M. (2019). *Business process management - concepts, languages, architectures, third edition*. Springer.

Wibisono, A., Nisafani, A. S., Hyerim Bae, H. B., & You-Jin Park, Y.-J. P. (2016). A dynamic and human-centric resource allocation for managing business process execution. *International Journal of Industrial Engineering: Theory, Applications and Practice, 23 (4), 2016*.

Williams, K., Chatterjee, S., & Rossi, M. (2008). Design of emerging digital services: A taxonomy. *EJIS*, *17*(5), 505–517. doi: 10.1057/ejis.2008.38

Xie, Y., Chen, S., Ni, Q., & Wu, H. (2019). Integration of resource allocation and task assignment for optimizing the cost and maximum throughput of business processes. *Journal of Intelligent Manufacturing*, *30*(3), 1351–1369.

Xie, Y., Chien, C.-F., & Tang, R.-Z. (2016). A dynamic task assignment approach based on individual worklists for minimizing the cycle time of business processes. *Computers & Industrial Engineering*, *99*, 401–414.

Xu, J., Liu, C., & Zhao, X. (2009). Resource planning for massive number of process instances. In *Otm confederated international conferences" on the move to meaningful internet systems"* (pp. 219–236).

Xu, J., Liu, C., Zhao, X., & Ding, Z. (2013). Incorporating structural improvement into resource allocation for business process execution planning. *Concurrency and Computation: Practice and Experience*, *25*(3), 427–442.

Xu, J., Liu, C., Zhao, X., Yongchareon, S., & Ding, Z. (2016). Resource management for business process scheduling in the presence of availability constraints. *ACM Transactions on Management Information Systems (TMIS)*, *7*(3), 1–26.

Yaghoibi, M., & Zahedi, M. (2017). Cycle time reduction and runtime rebalancing by reallocating dependent tasks. *International Journal of Engineering*, *30*(12), 1831–1839.

Yaghoubi, M., & Zahedi, M. (2016). Resource allocation using task similarity distance in business process management systems. In *2016 2nd international conference of signal processing and intelligent systems (icspis)* (pp. 1–5).

Yari Eili, M., & Rezaeenour, J. (2022). A survey on recommendation in process mining. *Concurrency and Computation: Practice and Experience*, *34*(26), e7304.

Yeon, M.-S., Lee, Y.-K., Pham, D.-L., & Kim, K. P. (2022). Experimental verification on human-centric network-based resource allocation approaches for process-aware information systems. *IEEE Access*, *10*, 23342–23354.

Yu, W., Jia, M., Liu, C., & Ma, Z. (2020). Task preemption based on petri nets. *IEEE Access*, *8*, 11512–11519.

Zelkowitz, M. V., & Wallace, D. R. (1998). Experimental models for validating technology. *Computer*, *31*(5), 23–31.

Zhao, W., Liu, H., Dai, W., & Ma, J. (2016). An entropy-based clustering ensemble method to support resource allocation in business process management. *Knowledge and Information Systems*, *48*(2), 305–330.

Zhao, W., Pu, S., & Jiang, D. (2020). A human resource allocation method for business processes using team faultlines. *Applied Intelligence*, *50*(9), 2887–2900.

Zhao, W., Yang, L., Liu, H., & Wu, R. (2015). The optimization of resource allocation based on process mining. In *International conference on intelligent computing* (pp. 341–353).

Zhao, W., Zeng, Q., Zheng, G., & Yang, L. (2017). The resource allocation model for multi-process instances based on particle swarm optimization. *Information Systems Frontiers*, *19*(5), 1057–1066.

Zhou, Y., & Chen, Y. (2008). Project-oriented resource assignment: from business process modelling to business process instantiation with operational performance consideration. *International Journal of Computer Integrated Manufacturing*, *21*(1), 97–110.