

Article

On the Problem of Forming Sustainable Production Schedules in the Context of Conflicting Objective Functions of Management Agents

Zhanna V. Burlutskaya ^{*}, Irina V. Vatamaniuk , Aleksei M. Gintciak , Daria A. Ablavatskaia 
and Kapiton N. Pospelov 

Laboratory of Digital Modeling of Industrial Systems, Peter the Great St. Petersburg Polytechnic University, St Petersburg 195251, Russia; aleksei.gintciak@spbpu.com (A.M.G.); daria.zubkova@spbpu.com (D.A.A.); kapiton.pospelov@spbpu.com (K.N.P.)

* Correspondence: zhanna.burlutskaya@spbpu.com

Abstract

This study addresses the foundational step of developing a classification and taxonomy of agent objective functions as a prerequisite for analyzing stability and forming robust production schedules in distributed manufacturing systems. The research is based on the premise that instability or insufficient robustness in scheduling solutions often arises from the neglect of the inherent multi-agent nature of real-world distributed production systems. These systems are characterized by the presence of multiple decision-making entities, each pursuing its own objectives or performance indicators. Since strategic management in such systems is typically oriented toward achieving global system-level goals, it often overlooks the interests of individual agents. As a result, the implemented decisions may encounter resistance from specific agents and lead to deterioration in the performance of their individual objective functions. These features underline the need to develop tools for identifying robust solutions, in which both the system as a whole and its constituent agents can achieve sustainably high performance across their respective objectives. The aim of this study is to analyze the divergent objective functions of management agents in distributed manufacturing systems in the context of forming robust production schedules. The research explores typical objective functions of structural units within the production system and presents their classification in terms of constraints, nature, granularity, behavioral orientation, and inter-agent dependency. The outcomes of the study include a comprehensive taxonomy of agent objective functions, along with the selection of relevant game-theoretic models for each pair of agents based on their interaction strategies. The findings contribute to the development of methodological and technological tools for decision support in sustainable manufacturing, extending current research on intelligent agent modeling and coordination in complex production environments.

Keywords: sustainable manufacturing systems; decision intelligence in manufacturing; sustainable production schedules; objective functions of management agents; distributed production systems



Received: 13 November 2025

Revised: 25 January 2026

Accepted: 3 February 2026

Published: 5 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The effectiveness of production management in industrial enterprises largely depends on the robustness of the managerial decisions made [1–3]. In real-world manufacturing environments, the management structure is typically distributed, which leads to a diversity of local objectives and production plans across various structural units. This diversity

significantly complicates the process of finding optimal decisions that satisfy the goals and constraints of all stakeholders [4–7]. To address this challenge, decision support systems (DSS) have been developed based on multi-criteria optimization tools [8–12]. However, such stochastic algorithms are often limited in their ability to account for local constraints of individual units within a production system. More critically, they often fail to handle conflicts arising from divergent objective functions.

The existence of numerous relatively autonomous structural units, each possessing its own goals, knowledge, and potential behavior strategies, implies the inherently multi-agent nature of production systems [13]. Therefore, considering this multi-agent nature is essential for achieving stable solutions in the management of distributed manufacturing systems. At the same time, accounting for multi-agent interactions complicates the decision-making process due to the exponential growth of possible management scenarios.

It is important to emphasize that distributed production systems belong to the class of socio-technical systems, which combine both technical and social agents. Unlike broader socio-economic systems, socio-technical systems consider social agents not as an abstract society but as individual decision-making entities with distinct goals and unique strategies, shaped by bounded rationality in information processing and decision-making. Consequently, neither emergent artificial intelligence algorithms (such as particle swarm optimization, bee colony algorithms, or need-opportunity networks)—though effective in modeling complex multi-agent systems—nor traditional simulation paradigms like system dynamics (widely used in demographic modeling or epidemiological simulations), adequately satisfy the requirements of complex manufacturing systems. These approaches often reduce social agents to homogeneous collectives, albeit capable of self-organization and evolution, but still treated as groups of rational actors rather than heterogeneous and individually motivated agents.

The presence of a diverse set of autonomous social and technical agents interacting across multiple hierarchical levels necessitates a detailed modeling of agent interactions, which in turn restricts the choice of applicable system modeling tools. Accurately modeling such a system as a set of interconnected intelligent agents can only be achieved through the application of multi-agent systems (MAS). However, this introduces the challenge of designing both the intelligent agents themselves and the organizational structure of the production system, especially when the objective functions of structural units are divergent [4,14–18]. Therefore, to assess the stability and controllability of schedules in a distributed production system, and to identify potential conflicts of interest among its agents, it becomes necessary to introduce a classification of agent objective functions. These functions are viewed through the MAS paradigm, where structural units are treated as intelligent agents [4,19,20].

The goal of this study is to analyze the divergent objective functions of decision-making entities within distributed production systems in the context of forming robust production schedules. The work explores typical objective functions of structural units and proposes a classification based on constraints, nature, parameterization, behavioral focus, and inter-agent dependency. It presents a taxonomy of agent relationships, their impact on the production schedule, and suggests appropriate game-theoretic models for each agent pair according to their interaction strategy. The results of this study lay the conceptual foundation for the development of intelligent decision support systems, enabling the transition toward intelligent, adaptive, and resilient manufacturing [21].

2. Materials and Methods

The current problem with scheduling in distributed production systems is the complexity of managing such systems. Because it requires consideration of the interaction of

heterogeneous agents, both technical and human. And it requires the use of adequate and sustainable optimization methods. Traditional methods of multi-criteria optimization are unable to fully take into account local constraints and resolve conflicts of goals between autonomous agents of the system. Therefore, when designing a model, it is necessary to take into account the diversity of agents and their goals, which can only be achieved using a multi-agent system.

At the core of any multi-agent system (MAS) operating within a distributed manufacturing environment lies a clear definition of the agents' objective functions. An agent's objective function determines its behavior (desires and intentions in terms of the BDI paradigm [22]), its ability to coordinate with other agents, and its contribution to achieving the global production goals. In manufacturing systems, selecting these functions requires considering both the internal characteristics of agents and the external environment in which they operate.

First, the objective function should reflect the agent's role in the system, including its hierarchical level (e.g., strategic–enterprise, tactical–department, operational–worker), functional designation (production, maintenance, supply), and degree of autonomy [23]. Second, the nature of the agent's interactions with others must be considered: cooperation, competition, or a combination of both [24]. These interactions are modeled using game theory, where the objective functions indicate whether the agent participates in a cooperative game with shared benefits, a non-cooperative game with individual interests, or a hybrid (coopetitive) setting [5,25–27].

A third critical principle is the alignment between local and global objectives. Individual objective functions must be compatible with the overall strategy of the production system. This alignment is achieved through goal-setting, coordination, and redistribution mechanisms embedded within the system's architecture.

Finally, the objective functions must be resilient to environmental dynamics, endowing agents with adaptability and robustness. Proper selection of parameters and their weights in the agent's objective function enables mechanisms of self-organization, continual learning, and system restructuring without compromising its integrity. Thus, appropriate selection and formalization of agent objective functions are essential for building an effective, flexible, and adaptive multi-agent manufacturing system.

For the identified types of agents, we consider possible typical objective function patterns based on the introduced ontology (see Table 1). The objective function patterns specify the dependencies of the variables and the required extremum. These patterns are not final computational formulas, but semantic patterns.

Production Unit and Maintenance Unit were separated into separate structural units, since their objective function patterns are different, but affect the global function of the entire enterprise.

The Worker and Employee have the same objective function pattern throughout the system, since the Employee is the supervisor of the Worker. However, they differ fundamentally in terms of interactions, level of responsibility and degree of influence on other agents.

These types of agents are a typical and sufficient set for studying the task of creating a production schedule in distributed production systems. However, depending on the specifics of the system, the importance and degree of influence of the stakeholders, the list of agents can be expanded and reduced, which are special cases of the model being developed.

Table 1. Examples of Typical Objective Functions for Agents in a Production System.

Agent	Objective Function	Variable Definitions	Function Explanation
Customer	$F = \max(\frac{1}{T_{\text{exec}}}, Q_{\text{product}}, \frac{1}{P_{\text{product}}})$	T_{exec} —actual order execution time; Q_{product} —product quality rating; P_{product} —product cost	The key parameters for the customer are execution speed, cost, and product quality. This depends on production specifics, market competition, and supplier capabilities. It influences supplier choice and repeat orders.
Supplier	$F = \max(\frac{1}{T_{\text{delivery}}}, \frac{1}{C_{\text{logistics}}}, D_{\text{contract}})$	T_{delivery} —total delivery time; $C_{\text{logistics}}$ —logistics costs; D_{contract} —proportion of fulfilled contractual terms (from 0 to 1)	Considers delivery timelines and the complexity of the logistics infrastructure. Affects contractual relationships between the enterprise and the supplier.
Enterprise	$F = \max(Q_{\text{perf}}, D_{\text{deadline}}, \frac{1}{C_{\text{total}}}, \frac{1}{R_{\text{risk}}}, D_{\text{contract}})$	Q_{perf} —average quality of order fulfillment (in points or % compliance); D_{deadline} —proportion of orders completed on time (0 to 1); C_{total} —unit cost of order fulfillment (per unit or % of budget); R_{risk} —aggregate production risk indicator (failures, defects, disruptions); D_{contract} —proportion of fulfilled contractual terms (0 to 1)	This function integrates key production performance dimensions: quality, deadlines, cost, load balancing, risk mitigation, reliability, and contract compliance. Weighting coefficients allow tailoring priorities to the enterprise's strategy. The goal is to balance external demands (quality, deadlines, contracts) with internal objectives (resources, risks, incidents), ensuring sustainable system performance.
Production Unit	$F = \max(\frac{P_{\text{actual}}}{P_{\text{plan}}}, \frac{1}{N_{\text{errors}}})$	P_{actual} —actual output; P_{plan} —planned output; N_{errors} —defect rate	Focuses on meeting the production plan and minimizing defects. Depends on the type of production unit. Influences output stability and quality.
Maintenance Unit	$F = \min(T_{\text{downtime}})$	T_{downtime} —equipment downtime managed by the unit	The main task is to ensure uninterrupted processes with minimal cost. This function depends on the responsiveness and scheduling of maintenance operations and affects the overall production cycle efficiency.
Worker, Employee	$F = \max(n, \sum w_i, \frac{1}{T_{\text{work}}}, \frac{1}{N_{\text{errors}}})$	n —number of completed tasks; w_i —priority (weight) of the i -th completed task; T_{work} —total working time spent; N_{errors} —defects, errors	Evaluates the productivity of the worker/employee by considering the number and priority of completed tasks, time spent, and mistakes made. Task weights allow differentiating effectiveness: completing higher-priority tasks contributes more to overall efficiency.

To assess the stability and controllability of production schedules and processes in a distributed manufacturing system, as well as to identify potential conflicts of interest among agents, it is useful to introduce a classification of agent objective functions. This will provide the basis for selecting an appropriate optimization model and game-theoretic representation of agents based on the structure of their objective functions. The following set of criteria can be used to classify agent objective functions in distributed manufacturing systems, allowing for the systematization and comparison of agent goals under conditions of cooperation, competition, and cooptation (Table 2) [28–30].

Table 2. Classification Criteria for Agent Objective Functions.

Criteria	Description	Example
Goal Orientation	Individual (selfish), global (altruistic), or mixed (shared utility)	Minimizing task time/Maximizing overall quality
Nature of the Objective Function	Cost-related (expenses, profit), time-related (deadlines), quality-related (stability, conformance), or multi-criteria	

Table 2. *Cont.*

Criteria	Description	Example
Constraints	Hard (explicit boundaries) or soft (penalties, barrier functions)	Resource, schedule, or standard constraints
Parameter Detail	Aggregated or parameterized (considering environmental conditions, competencies, task complexity)	Total number of completed tasks/complexity-weighted task count
Adaptability vs. Static Nature	Static (fixed over time) or adaptive (updated based on environment or feedback)	Weight modification after a task cycle
Aggregation Stability	Whether the function retains consistent behavior when aggregated into group or system-level metrics	Summable KPIs or weighted aggregates
Behavioral Focus	Output-based (goal-focused) or behavior-based	Achieving a quality level/following procedures
Scale	Local, group/cluster, or global	For one agent/a group/the entire system
Inter-Agent Dependency	Independent or dependent	Executed regardless of other agents/depends on other agents' decisions and states

Let us examine the classification of typical agent objective functions in a distributed production system based on the specified set of criteria (Table 3). Each column corresponds to an agent's objective function and reflects its properties in terms of orientation, nature, parameterization, scale, and other characteristics.

Table 3. Description of Agent Objective Functions by Criteria.

Criteria	Customer	Supplier	Enterprise	Structural Unit	Worker	Employee
Goal Orientation	Individual—interests focus on meeting specific quality and deadline requirements	Individual—aims to optimize its own timelines and costs	Mixed—accounts for both internal and external objectives	Mixed—both local task execution and contribution to broader goals matter	Individual—metrics tied to personal effectiveness	Mixed—personal KPIs and team performance indicators
Nature of the Objective Function	A combination of quality control and delivery time	Time-cost focused	Multi-criteria—includes deadlines, quality, load balancing, and cost	A combination of cost and time performance	Based on time and quality of work	Quality-time based—task execution efficiency
Constraints	Hard—strict order deadlines	Soft—delays may incur penalties, but flexibility remains	Both hard (contracts, regulations) and soft (resources)	Mainly soft (resources, shifts)	Soft—defined by norms and task requirements	Soft—typically procedural
Parameter Detail	Aggregated—considers the order as a whole	Aggregated—averages across logistics indicators	Parameterized—incorporates multiple variables	Parameterized—considers load, errors, and quality	Parameterized—includes errors and task weights	Parameterized—task specificity and errors are accounted for
Adaptability vs. Static Nature	Static—conditions are fixed in the contract	Adaptive—adjusts to demand and supply chain dynamics	Adaptive—can evolve through feedback mechanisms	Adaptive—can respond to disruptions and changes	Adaptive—can change based on KPIs or performance evaluation	Adaptive—responds to feedback
Aggregation Stability	Stable—not highly sensitive to internal changes at the enterprise	Stable—aggregated metrics smooth out fluctuations	High—can be aggregated at the system level	Stable—especially in team-based work	Moderately stable—subject to individual variation	Moderately stable—sensitive to workload
Behavioral Focus	Output-based—final product delivery	Output-based—timely delivery	Output-based—fulfilling the production plan	Behavior-based—managing task execution processes	Behavior-based—focuses on how the person works	Behavior-based—evaluates working style and discipline
Scale	Local—specific to a given customer	Local—applies to a specific supplier	Global—covers the entire system	Intermediate—covers one organizational link	Local—specific to an individual worker	Local—individual agent or role
Inter-Agent Dependency	Yes—depends on the enterprise's performance	Yes—depends on the enterprise and logistics environment	Yes—encompasses all levels	Yes—interacts with other units and the enterprise	Partial—functions are individual, but in coordination with colleagues	Partial—in coordination with colleagues

This classification of agent objective functions highlights the importance of a systems approach to the design of agent behavior and interactions in a multi-agent

production environment. Each agent (from external customer to shop-floor worker) has a specific goal, performance evaluation structure, and degree of interdependence with other system participants.

Goal orientation ranges from strictly individual (e.g., worker, supplier) to mixed and collective (e.g., enterprise, departments). The nature of functions becomes increasingly multi-criteria at higher levels, encompassing time, quality, and risk. Additionally, economic dimensions of agent interactions—though beyond the scope of this study—gain significance at the upper levels. Parameterization and adaptability increase closer to dynamic system elements (workers, employees), where continuous reevaluation and behavioral adjustments are required. Inter-agent dependencies grow with scale and coordination complexity, especially for enterprises and their subdivisions. The scope of an agent's objective function is directly tied to its architectural level and influences both the generalization degree and aggregation potential.

This classification approach enables not only the standardization of objective functions but also the identification of methods for aligning, optimizing, and integrating them into the overall architecture of a production MAS, as well as to describe a methodology for converting objective function patterns into a mathematical form. This is particularly critical in the development of decision support systems and distributed manufacturing infrastructures.

To implement the objective function patterns in the computational model, it is proposed to perform the following transformation:

Each heterogeneous criterion in the pattern from Table 1 should be reduced to a dimensionless form on a common scale (for example, $[0, 1]$) using a normalization function. Further, to aggregate the indicators into a single scalar goal function, it is assumed to use linear or nonlinear convolution with weighting coefficients reflecting the priorities of agents. The classification criteria from Table 2 are a direct source for determining and adjusting these weights. To model the joint behavior of agents, a pair of agents will be compared to the corresponding game or cooperative protocol, which will be considered in the results of this study.

Thus, the presented classifications serve as instructions for converting conceptual objective function patterns into formal agent models and rules for their interaction.

3. Results

Let us consider a typical distributed manufacturing system that implements an intelligent scheduling system for orders and production shifts. Within this system, agents represent the interests of external customers, the enterprise, structural units, and workers. To coordinate agent schedules, prevent resource conflicts, and minimize deadline deviations, a classification of agent objective functions is applied. This classification enables the clear delineation of priorities: the customer seeks to minimize lead times and costs; the enterprise aims to fulfill all orders with optimal resource utilization; departments strive for balanced resource allocation; and workers prefer stable shifts with a prioritized task plan. During preliminary analysis, potential agent conflicts are identified [31–33]—for instance, between departments competing for resources and between departments and workers whose workload expectations may diverge. To enhance alignment between agent objective functions, the system implements a hierarchical cooperative interaction model [34,35] and introduces standardized rules for task prioritization and production schedule coordination. As a result, the proposed approach makes it possible to adapt the production schedule to the changing input conditions of the system, reallocate resources, and reduce the frequency of conflicts among agents. This affects the reliability of the production schedule, as it

takes into account multi-agent interaction, without which the schedule will not produce a complete and sustainable result.

Based on the above, we examine the interrelations among the objective functions of different agent types (Table 4). Rows represent influencing agents, while columns represent agents receiving the impact. Each cell indicates whether and how one agent's objective function affects another.

Table 4. Classification of Interactions Between Agent Objective Functions.

		Impact Receptor					
		Customer	Supplier	Enterprise	Department	Employee	Worker
Impact Source	Customer	-	Independent	Constrains	No interaction	No interaction	No interaction
	Supplier	Independent	-	Constrains	No interaction	No interaction	No interaction
	Enterprise	Indirect impact	Indirect impact	-	Manages	Indirect impact	Indirect impact
	Department	No impact	No impact	Subordinate	-	Manages	Manages
	Employee	No impact	No impact	Feedback	Subordinate	-	Manages
	Worker	No impact	No impact	Triggers events	Feedback	Subordinate	-

The relationships between agent objective functions determine which priorities dominate at various management levels and how coordination is established among them. For example, the individual goals of customers and suppliers define the external constraints that the enterprise must account for in strategic planning. The enterprise's objective function translates these constraints into limited resources and priorities that influence the actions of departments. Departments then develop tactical schedules based on production constraints and feedback from employees and workers. When agent objectives are well-aligned and incorporate adaptive mechanisms, the schedule becomes resilient to changes and local disruptions. However, misalignment of functions or resource competition among departments may lead to conflicting interests, cascading delays, and systemic instability. The enterprise's objective function indirectly influences those of the customer and supplier: during negotiations and bidding processes, it balances internal capabilities and external demands, shaping execution conditions, prices, and deadlines, which are then reflected in the expectations and strategies of external agents.

By applying the approaches proposed in [36] to the examined organizational agents, we formalize their interactions through models of cooperation and competition, utilizing both optimization and game-theoretic mechanisms.

Firstly, a multi-agent organizational system can be represented as a hierarchical model with decentralized or centralized elements, where each agent pursues its own objective. Through cooperative optimization, the actions of agents can be viewed as joint minimization of the enterprise's total costs (e.g., model predictive production control). However, in the presence of limited resources and divergent interests (e.g., among structural units), it becomes necessary to apply cooperative or non-cooperative game-theoretic approaches.

Secondly, agents at the enterprise level (such as production and maintenance departments) participate in distributed cooperative optimization, minimizing the overall costs while preserving local autonomy:

- each agent minimizes its own local cost function (e.g., downtime, defects, excessive inventory);
- the global enterprise performance function is formed as an aggregate of the local functions.

Coordination among agents is achieved via agreed-upon messages or limited iterations of intention exchange (in BDI terms). For example, the production unit and logistics service jointly minimize delivery delays and storage losses by synchronizing their plans through a

limited number of iterations—e.g., using distributed optimization methods over a directed graph [37].

Thirdly, each agent type can aggregate, process, and transmit information according to its role and level of responsibility. At the lowest level, workers record primary data—telemetry, operation statuses, and production disruptions. Employees aggregate these into informative insights, such as deviations from the plan or task execution statistics. Departments, with knowledge of the local schedule and resources, transform information into knowledge by identifying patterns, bottlenecks, or potential conflicts. The enterprise, receiving only aggregated knowledge, is capable of generating strategic wisdom—making informed decisions on resource reallocation or adjustments to the global schedule. Suppliers and customers, located at the external level, operate using high-level generalizations and forecasts, formulating requirements and constraints without overloading the system with granular data. This approach reduces communication overhead, increases robustness to information noise, and enables decision-making based on relevant, abstracted knowledge without sacrificing local adaptability [38,39].

Fourthly, the enterprise, production units, and development departments can be modeled as participants in a static cooperative game, solving tasks such as coordination, team formation, and resource allocation [13]:

- agents propose resource allocation strategies;
- Pareto optimization or other multi-objective methods are applied;
- the outcome is a compromise strategy (e.g., a balanced production schedule).

When structural units compete for limited resources (e.g., working time, labor funds, equipment), their behavior is described by a non-cooperative game:

- each agent maximizes its own utility (e.g., plan fulfillment);
- a Nash equilibrium defines a stable resource allocation;
- dynamic learning methods, such as Q-learning [40], may be applied to identify the most beneficial resource distribution.

At the same time, such non-cooperative games occur in parallel with cooperative ones at other levels. For example, at the upper level, the enterprise and its departments may engage in a cooperative game in which strategic goals, overall capacity utilization, and long-term priorities are aligned. This interaction sets global boundaries and constraints, such as budget allocation, quality targets, or the overall production plan. At the local level, within these boundaries, departments participate in a non-cooperative game, competing for specific resources (e.g., machine time, wage funds) and striving to maximize their own local efficiency. While their strategies may align with global objectives, they do not necessarily align with each other. Thus, cooperative games form a “corridor of agreed solutions”, within which local non-cooperative competition for self-optimization occurs. If the cooperative boundaries are too narrow or poorly synchronized, non-cooperative behavior can lead to systemic conflicts. In such cases, predefined cooperative agreements and profit redistribution mechanisms (e.g., via smart contract mechanisms [41]) can help limit destructive competition and increase system resilience. This two-level control loop can be further reinforced through iterative procedures: the outcomes of the non-cooperative phase are analyzed and lead to the adjustment of cooperative conditions in the next planning iteration [42].

External environment agents, such as customers and suppliers, define constraints and act as external players. Their behavior can be formalized as game parameters or rules (e.g., contract terms, regulations, penalties) (Table 5).

Table 5. Examples of Agent Interactions and Corresponding Game Models.

Agent Pair	Interaction Type	Game Model	Cooperation/Competition
Enterprise—Departments	Multi-level	Hierarchical cooperative game	Goal-driven cooperation
Departments—Departments	Horizontal	Dynamic cooperative game	Schedule and resource alignment
Workers—Workers	Local horizontal	Cooperative game	Cooperation/partial competition
Customer—Enterprise	Contract game	Non-cooperative, dynamic game	Negotiation of conditions
Enterprise—Supplier	Network-based game	Multi-agent trading game	Negotiation

The interaction between the enterprise and its departments represents a multi-level hierarchy, where the enterprise sets goals and departments implement them—this is modeled as a cooperative game. Horizontal interactions between departments involve both task coordination and resource competition, modeled through a dynamic cooperative game. Local interactions between employees or workers are cooperative, with the possibility of partial competition [43–45]. The customer-enterprise interaction is a non-cooperative contractual game, where each party pursues its own benefit [41–45]. The enterprise-supplier interaction can be implemented through a networked trading game, where delivery terms are negotiated in a dynamic market environment.

In summary, the behavior of agents in an organizational system can be consistently described as a transition from cooperative distributed optimization (at the level of joint production task execution) to cooperative and non-cooperative games (at the level of resource and strategy allocation). This layered modeling approach enables the development of adaptive management models, resilient to conflicts of interest and scalable to real-world industrial conditions.

Let us examine the impact of interrelations between typical agent objective functions, grouped by agent class, on the stability of scheduling solutions in distributed manufacturing systems.

The customer does not generate schedules directly but defines high-level constraints: deadlines, quality requirements, and penalties. These parameters are aggregated by the enterprise into order priorities and influence strategic and tactical decisions. For instance, a customer's objective function with a high priority coefficient (α) causes the enterprise to postpone less urgent orders and allocate more resources to the department handling the critical order. The priority coefficient α increases in product quality requirements to Q_{product} and is interpreted at lower levels of the hierarchy as w_i . Thus, the higher the α value, the greater the scheduling pressure that is transmitted down the agent hierarchy. Under the multitasking conditions of enterprise operations, this necessitates dynamic reassignment of orders and resources within the system.

The supplier also does not generate schedules directly but imposes external constraints, primarily concerning delivery deadlines and supply volumes. These constraints are translated by the enterprise into resource availability plans and affect the feasible time windows for initiating production operations. A delay in the delivery of a critical resource (reflected by a high weight T_{delivery} in the supplier's objective function) can result in schedule shifts, priority changes, or reallocation of orders to other workstations. Thus, the more dependent the production plan is on timely deliveries, the greater the supplier's impact on the schedule stability of the enterprise. In the context of limited logistics flexibility, this requires adaptive planning mechanisms and early-stage consideration of failure probabilities.

The enterprise operates at the strategic planning level, aggregating customer demands, supplier constraints, and department capabilities. Its objective function reflects a balance between quality, timeliness, cost, and risk. The enterprise's influence on schedule formation lies in the strategic allocation of orders and resources, priority setting, and schedule con-

figuration. For example, if the weight of the D_{deadline} indicator (on-time delivery) is high, the enterprise may reduce operational buffers, outsource part of the work, or restructure the department workloads. These decisions have a top-down impact on departmental schedules and require continuous synchronization with lower system levels.

The department generates the detailed operational schedule within the boundaries of resources allocated by the enterprise. Its objective function focuses on plan fulfillment and defect reduction. It must consider current workloads, personnel availability, and equipment readiness. The department's influence on scheduling is reflected in operation reallocation, order sequencing, and task reassignment among employees. For example, if production output lags behind the plan, the department agent might activate a backup shift or redistribute orders to less loaded workstations. Such localized intervention helps stabilize order fulfillment without revising the strategic plan, but it requires precise coordination with employee and worker agents.

The employee directly executes tasks from the department schedule, but their agent can adapt the task flow depending on task complexity, error rates, and execution pace. The employee's objective function accounts for performance, time expenditure, and accuracy. Its influence on the schedule is realized through feedback mechanisms that trigger task reordering or reassignment. For example, if an employee consistently exceeds the allowable number of errors (high N_{errors} value), the system may assign simpler tasks or reduce workload. This change propagates to the department level and requires task redistribution among other employees.

The worker executes scheduled operations at the physical level and generates event flows: task starts and completions, deviations, and failures. While the worker does not directly influence scheduling decisions, their behavior triggers local schedule recalculations or initiates compensatory actions. High productivity or early task completion may lead to the release of the next task, minimizing idle time. Thus, even minor deviations at the worker level can cascade upward, affecting the schedules of employees and departments, and in rare cases, even reaching the enterprise level, particularly in critical production chains (Table 6).

Table 6. Influence of Agent Objective Functions on Schedule Formation.

Agent Level	Influence via Objective Function	Transmission Mechanism
Customer	Sets execution priorities	Contract → Enterprise
Supplier	Defines resource availability limits	Coordination → Enterprise
Enterprise	Balances global KPIs and manages strategy	Task and resource allocation → Department
Department	Optimizes plan within allocated resources	Operation detailing → Worker/Employee
Employee	Modifies execution based on performance	Process feedback and local adaptation → Department
Worker	Executes tasks, generates events	Execution and function-triggering events → Department

The objective functions of agents at different levels reflect their intentions and constraints:

- External agents (customers and suppliers) define conditions and constraints that the system must respond to adaptively.
- Strategic agents (enterprise) define global goals and strategic resource management rules.
- Tactical and operational agents (departments, employees, workers) execute operational tasks and provide feedback.

Feedback in such a hierarchical system plays a crucial role in ensuring the adaptability and stability of the production process. At the lower level, a worker—through task execution events (e.g., delays, downtime, deviations from plan)—triggers local rescheduling within the department. This impacts the objective functions of employees and the

department itself, allowing for adjustments in task details or resource allocation. These changes aggregate upward through the hierarchy: the enterprise receives information about decreased performance or changes in resource availability and, based on its objective functions, can recalculate strategic priorities or initiate recoordination with the supplier and customer. Thus, the feedback cycle enables agent objective functions to adapt not only in response to initial customer requirements but also to the actual execution state at lower levels, maintaining a balance between schedule stability and flexibility.

This structure enables a hierarchical decomposition of the scheduling problem in a manufacturing system, allowing local schedule adaptation without disrupting the global coherence of enterprise operations. When agent objective functions across different levels are misaligned (e.g., a department tries to minimize defects at the expense of deadlines, while the enterprise prioritizes on-time delivery), structural instability can arise:

- conflicts of interest;
- mutual resource blocking;
- cascading delays.

In such cases, a centralized goal alignment policy is required, either through a shared enterprise strategy or cooperation protocols between departments. The classification and taxonomy of relationships determine the requirements for future planning algorithms, indicating the need for mechanisms such as:

- local adaptation (e.g., task redistribution among employees),
- macro-level rescheduling (at the enterprise level),
- dynamic priority confirmation.

Thus, the main result of this research is a structure and a set of design principles for developing multi-agent scheduling algorithms that can provide adaptive resilience—the system's ability to recover from local disruptions without requiring a complete global schedule revision.

4. Discussion and Conclusions

The developed classification and taxonomy of agent target functions provide the foundation for modeling distributed production systems as hierarchical multi-agent structures. The analysis shows that the stability of planning decisions in such systems will depend crucially on taking into account the consistency, adaptability, and hierarchical structure of the goals of the agents. While individual agent objectives enhance local efficiency, they may reduce system-wide alignment and stability. Conversely, centralized approaches offer high robustness but must be implemented carefully in dynamic environments. Therefore, scheduling algorithms should give preference to mixed, multi-criteria, and adaptive functions that strike a balance between coordination and autonomy. Therefore, based on the taxonomy of interactions, we conclude that a planning solution in a multi-agent production system will be more stable when:

- Agent objective functions are coordinated across hierarchical levels;
- Adaptive, multi-criteria models with soft constraints are used;
- Mechanisms for both local adaptation and global revision are in place to handle disruptions;
- The entire system operates through information exchange and feedback between agents.

This approach supports achieving a more sustainable equilibrium between production efficiency and operational flexibility, which is especially critical in volatile external environments. The stability of the approach is achieved by considering the multi-agent nature of a complex system. And also due to not only the global, but also the local goals of each agent and their interaction within the system. This approach allows for taking into account individual factors that undoubtedly affect scheduling in a distributed production system.

The results obtained contribute to the development of methodological and instrumental means of decision support for sustainable production, expanding existing research in the field of modeling intelligent agents and their interaction within complex production systems [29–34]. The research provides a conceptual basis for further implementation of the multi-agent accounting algorithm for scheduling in distributed production systems, taking into account the multidirectional objective functions of management entities.

Author Contributions: Conceptualization Z.V.B.; methodology I.V.V.; validation K.N.P.; formal analysis D.A.A.; supervision A.M.G. All authors have read and agreed to the published version of the manuscript.

Funding: The research is funded by the Russian Science Foundation, project No. 25-21-00322, <https://rscf.ru/en/project/25-21-00322/>.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Novikov, D.A. *Cybernetics: From Past to Future*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 47.
2. White, B.E. *Managing Complexity by George Rzevski and Petr Skobelev Southhampton, Boston: WIT Press, 2014 (ISBN: 978-1-84564-936-4); INCOSE: Pasadena, CA, USA, 2015.*
3. Lakemond, N.; Holmberg, G.; Pettersson, A. Digital transformation in complex systems. *IEEE Trans. Eng. Manag.* **2021**, *71*, 192–204. [[CrossRef](#)]
4. Caridi, M.; Cavalieri, S. Multi-agent systems in production planning and control: An overview. *Prod. Plan. Control* **2004**, *15*, 106–118. [[CrossRef](#)]
5. Rodionov, D.G.; Alfer'yev, A.D. Sustainability of the optimal plan of innovative production of an industrial enterprise. *St. Petersburg State Polytech. Univ. J. Econ.* **2020**, *13*, 106–119.
6. Moghrabi, I.A.R.; Bhat, S.A.; Szczuko, P.; AlKhaled, R.A.; Dar, M.A. Digital Transformation and Its Influence on Sustainable Manufacturing and Business Practices. *Sustainability* **2023**, *15*, 3010. [[CrossRef](#)]
7. Rahnama, H.; Johansen, K.; Larsson, L.; Rönnbäck, A.Ö. Collaboration in value constellations for sustainable production: The perspective of small technology solution providers. *Sustainability* **2022**, *14*, 4794. [[CrossRef](#)]
8. Pospelov, K.N.; Burlutskaya, Z.V.; Gintciak, A.M.; Troshchenko, K.D. Multiparametric Optimization of Complex System Management Scenarios Based on Simulation Models. *Int. J. Technol.* **2023**, *14*, 1748. [[CrossRef](#)]
9. Dvorkin, Y.; Wang, Y.; Pandzic, H.; Kirschen, D. Comparison of scenario reduction techniques for the stochastic unit commitment. In Proceedings of the 2014 IEEE Pes General Meeting | Conference & Exposition, National Harbor, MD, USA, 27–31 July 2014; IEEE: Washington, DC, USA, 2014; pp. 1–5.
10. Heitsch, H.; Römis, W. A note on scenario reduction for two-stage stochastic programs. *Oper. Res. Lett.* **2007**, *35*, 731–738. [[CrossRef](#)]
11. Hu, D.; Li, Z.; Ye, Z.; Peng, Y.; Xi, W.; Cai, T. Multi-agent graph reinforcement learning for decentralized Volt-VAR control in power distribution systems. *Int. J. Electr. Power Energy Syst.* **2024**, *155*, 109531. [[CrossRef](#)]
12. Lang, J.; Zhao, J. Modeling and optimization for oil well production scheduling. *Chin. J. Chem. Eng.* **2016**, *24*, 1423–1430. [[CrossRef](#)]
13. Sharko, P.A.; Burlutskaya, Z.V.; Zubkova, D.A.; Gintciak, A.M.; Pospelov, K.N. AI-Supported Decision Making in Multi-Agent Production Systems Using the Example of the Oil and Gas Industry. *Appl. Sci.* **2025**, *15*, 5366. [[CrossRef](#)]
14. Caridi, M.; Sianesi, A. Multi-agent systems in production planning and control: An application to the scheduling of mixed-model assembly lines. *Int. J. Prod. Econ.* **2000**, *68*, 29–42. [[CrossRef](#)]
15. Shukla, O.J.; Soni, G.; Kumar, R.; Ahluwalia, R.S. A review of multi agent-based production scheduling in manufacturing system. *Recent Pat. Eng.* **2021**, *15*, 15–32. [[CrossRef](#)]

16. Siatras, V.; Bakopoulos, E.; Mavrothalassitis, P.; Nikolakis, N.; Alexopoulos, K. Production scheduling based on a multi-agent system and digital twin: A bicycle industry case. *Information* **2024**, *15*, 337. [\[CrossRef\]](#)
17. Biasoto, L.P.; de Carvalho, V.R.; Sichman, J.S. Applying Multi-Agent Negotiation to Solve the Production Routing Problem With Privacy Preserving. *arXiv* **2024**, arXiv:2406.09214. [\[CrossRef\]](#)
18. Motsch, W.; Wagner, A.; Ruskowski, M. Autonomous Agent-Based Adaptation of Energy-Optimized Production Schedules Using Extensive-Form Games. *Sustainability* **2024**, *16*, 3612. [\[CrossRef\]](#)
19. Varriale, V.; Cammarano, A.; Michelino, F.; Caputo, M. Critical analysis of the impact of artificial intelligence integration with cutting-edge technologies for production systems. *J. Intell. Manuf.* **2025**, *36*, 61–93. [\[CrossRef\]](#)
20. Guan, Y.; Wang, D.; Chu, Z.; Wang, S.; Ni, F.; Song, R.; Zhuang, C. Intelligent agents with llm-based process automation. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 5018–5027.
21. Bolsunovskaya, M.V.; Gintciak, A.M.; Fedyaevskaya, D.E.; Petryaeva, A.A.; Burlutskaya, Z.V. Complex modeling of the oil production processes: Analytical review. *Autom. Informatiz. Fuel Energy Complex* **2023**, *2*, 51–62. [\[CrossRef\]](#)
22. Shang, J.; Yan, J.; Ren, F. BDI Agents Based Dynamic Resource Allocation in Emergency Scenarios. In Proceedings of the 2024 IEEE International Conference on Agents (ICA), Wollongong, Australia, 4–6 December 2024; IEEE: Washington, DC, USA, 2024; pp. 45–49.
23. Mentzas, G.; Hribernik, K.; Stahre, J.; Romero, D.; Soldatos, J. Human-Centered Artificial Intelligence in Industry 5.0. *Front. Artif. Intell.* **2024**, *7*, 1429186. [\[CrossRef\]](#)
24. Corbo, L.; Kraus, S.; Vlačić, B.; Dabić, M.; Caputo, A.; Pellegrini, M.M. Coopetition and innovation: A review and research agenda. *Technovation* **2023**, *122*, 102624. [\[CrossRef\]](#)
25. Zakharchenkov, K.V.; Mrochek, Z.A.; Mrochek, T.V. Algorithm for solution of multicriterion problem of production planning of pipes and shaped products. *Syst. Anal. Appl. Inf. Sci.* **2018**, *4*, 4–10. [\[CrossRef\]](#)
26. Xiong, H.; Shi, S.; Ren, D.; Hu, J. A survey of job shop scheduling problem: The types and models. *Comput. Oper. Res.* **2022**, *142*, 105731. [\[CrossRef\]](#)
27. Hartmann, S.; Briskorn, D. An updated survey of variants and extensions of the resource-constrained project scheduling problem. *Eur. J. Oper. Res.* **2022**, *297*, 1–14. [\[CrossRef\]](#)
28. Wang, Y.; Garcia, E.; Casbeer, D.; Zhang, F. (Eds.) *Cooperative Control of Multi-Agent Systems: Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 2017.
29. Dorri, A.; Kanhere, S.S.; Jurdak, R. Multi-agent systems: A survey. *IEEE Access* **2018**, *6*, 28573–28593. [\[CrossRef\]](#)
30. Cruz Salazar, L.A.; Ryashentseva, D.; Lüder, A.; Vogel-Heuser, B. Cyber-physical production systems architecture based on multi-agent's design pattern—Comparison of selected approaches mapping four agent patterns. *Int. J. Adv. Manuf. Technol.* **2019**, *105*, 4005–4034. [\[CrossRef\]](#)
31. Lima, R.M.; Sousa, R.M.; Martins, P.J. Distributed production planning and control agent-based system. *Int. J. Prod. Res.* **2006**, *44*, 3693–3709. [\[CrossRef\]](#)
32. Canbaz, B.; Yannou, B.; Yvars, P.A. Preventing design conflicts in distributed design systems composed of heterogeneous agents. *Eng. Appl. Artif. Intell.* **2014**, *28*, 142–154. [\[CrossRef\]](#)
33. Hillali, Y.; Zegrari, M.; Chafik, S.; Alfathi, N. An intelligent framework for Multi-Agent System based on dynamic balancing of production 4.0. *IEEE Access* **2025**, *13*, 165695–165717. [\[CrossRef\]](#)
34. Kovalenko, I.; Balta, E.C.; Tilbury, D.M.; Barton, K. Cooperative product agents to improve manufacturing system flexibility: A model-based decision framework. *IEEE Trans. Autom. Sci. Eng.* **2022**, *20*, 440–457. [\[CrossRef\]](#)
35. Zhang, Y.; Zhu, H.; Tang, D.; Zhou, T.; Gui, Y. Dynamic job shop scheduling based on deep reinforcement learning for multi-agent manufacturing systems. *Robot. Comput.-Integr. Manuf.* **2022**, *78*, 102412. [\[CrossRef\]](#)
36. Wang, J.; Hong, Y.; Wang, J.; Xu, J.; Tang, Y.; Han, Q.L.; Kurths, J. Cooperative and competitive multi-agent systems: From optimization to games. *IEEE/CAA J. Autom. Sin.* **2022**, *9*, 763–783. [\[CrossRef\]](#)
37. Wang, D.; Wang, Z.; Chen, M.; Wang, W. Distributed optimization for multi-agent systems with constraints set and communication time-delay over a directed graph. *Inf. Sci.* **2018**, *438*, 1–14. [\[CrossRef\]](#)
38. Wan, J.; Li, X.; Dai, H.N.; Kusiak, A.; Martinez-Garcia, M.; Li, D. Artificial-intelligence-driven customized manufacturing factory: Key technologies, applications, and challenges. *Proc. IEEE* **2020**, *109*, 377–398. [\[CrossRef\]](#)
39. Rozs, R.; Ando, M. Collaborative systems, operation and task of the manufacturing execution systems in the 21st century industry. *Period. Polytech. Mech. Eng.* **2020**, *64*, 51–66. [\[CrossRef\]](#)
40. Gankin, D.; Mayer, S.; Zinn, J.; Vogel-Heuser, B.; Endisch, C. Modular production control with multi-agent deep q-learning. In Proceedings of the 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Västerås, Sweden, 7–10 September 2021; IEEE: Washington, DC, USA, 2021; pp. 1–8.
41. Leng, J.; Sha, W.; Lin, Z.; Jing, J.; Liu, Q.; Chen, X. Blockchain smart contract pyramid-driven multi-agent autonomous process control for resilient individualised manufacturing towards Industry 5.0. *Int. J. Prod. Res.* **2023**, *61*, 4302–4321. [\[CrossRef\]](#)

42. Nongaillard, A.; Mathieu, P.; Jaumard, B. A multi-agent resource negotiation for the utilitarian welfare. In *International Workshop on Engineering Societies in the Agents World*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 208–226.
43. Tennant, E.; Hailes, S.; Musolesi, M. Dynamics of moral behavior in heterogeneous populations of learning agents. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, Madrid, Spain, 20–22 October 2024; Volume 7, pp. 1444–1454.
44. Parker, T. Ethical Evaluation in Multi-Agent Systems. Doctoral Dissertation, Université de Toulouse, Toulouse, France, 2024.
45. Cointe, N.; Bonnet, G.; Boissier, O. Ethics-based cooperation in multi-agent systems. In *Advances in Social Simulation: Looking in the Mirror*; Springer International Publishing: Cham, Switzerland, 2020; pp. 101–116.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.