

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2023

Bc. Martin Žiačik



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

MULTIAGENTNÍ SYSTÉMY PRO DISTRIBUOVANOU VÝROBU

MULTIAGENT SYSTEMS FOR DISTRIBUTED PRODUCTION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Martin Žiačik

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Václav Kaczmarczyk, Ph.D.

BRNO 2023



Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Martin Žiačik

ID: 211194

Ročník: 2

Akademický rok: 2022/23

NÁZEV TÉMATU:

Multiagentní systémy pro distribuovanou výrobu

POKYNY PRO VYPRACOVÁNÍ:

- 1) Proveďte průzkum dostupných a využívaných algoritmů pro řízení výroby (centralizovaných i distribuovaných).
- 2) S využitím konceptu multiagentních systémů navrhnete vhodný přístup (či více přístupů) pro distribuovanou organizaci výroby.
- 3) Navrhnete řídicí algoritmy pro jednotlivé agenty v systému a provedte simulaci na vybrané platformě.
- 4) Porovnejte navzájem navržené algoritmy a stanovte jejich vhodnost použití.
- 5) Celé řešení implementujte jako vzorovou simulovanou výrobní úlohu a zadokumentujte.

DOPORUČENÁ LITERATURA:

Aleš Kubík: Inteligentní agenty

Termín zadání: 6.2.2023

Termín odevzdání: 17.5.2023

Vedoucí práce: Ing. Václav Kaczmarczyk, Ph.D.

doc. Ing. Petr Fiedler, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Výrobné spoločnosti v súčasnosti musia čeliť dynamickému prostrediu, v ktorom sa rýchlo menia ekonomické, technologické a zákaznicke trendy, čo si vyžaduje zvýšené nároky na pružnosť a agilitu riadenia a plánovania výroby. Využitie multiagentných systémov, ktoré svojou koncepciou reflektujú na tieto nároky, môže byť vhodnou alternatívou k tradičnému centralizovanému riadeniu výroby. V práci diskutujem teoretickú stránku tejto problematiky a na základe získaných poznatkov navrhujem model takejto decentralizovanej výroby. Na vytvorenom simulátore následne overujem vhodnosť multiagentného prístupu k riadeniu výroby. Výstupy práce môžu byť použité ako podklad pri nasadzovaní multiagentných štruktúr do reálnych výrobných podnikov. Aj pomocou takýchto koncepčných zmien vo výrobe sa podniky približujú k moderným princípom priemyslu 4.0.

Kľúčové slová

Rozvrh, plánovanie, algoritmus, multiagentný prístup, optimalizácia, CPPS

Abstract

Manufacturing companies today face a dynamic environment in which economic, technological and customer trends are changing rapidly, requiring increased demands for flexibility and agility in production management and planning. The use of multi-agent systems that reflect these demands in their design can be a suitable alternative to traditional centralised production management. In this thesis I discuss the theoretical aspect of this issue and based on the obtained knowledge a model of such decentralized production is proposed. I then verify the suitability of the multi-agent approach to production management on the developed simulator. The outputs of the thesis can form the basis for the deployment of multiagent structures in real production enterprises. Also with the help of such conceptual changes in production, enterprises are moving closer to the modern principles of Industry 4.0.

Keywords

Scheduling, planning, algorithm, multi-agent approach, optimization, CPPS

Bibliografická citácia

ŽIAČIK, Martin. Multiagentní systémy pro distribuovanou výrobu. Brno, 2023.
Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/151545>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Václav Kaczmarczyk.

Prehlásenie autora o pôvodnosti diela

Meno a priezvisko študenta: Martin Žiačik

VUT ID študenta: 211194

Typ práce: Diplomová práca

Akademický rok: 2022/23

Téma záverečnej práce:
Multiagentné systémy pre distribuovanú výrobu

Prehlasujem, že svoju záverečnú prácu som vypracoval samostatne pod vedením vedúceho záverečnej práce a s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej záverečnej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto záverečnej práce som neporušil autorské práva tretích osôb, predovšetkým som nezasiahol nedovoleným spôsobom do cudzích osobných autorských práv a som si plne vedomý následkov porušenia ustanovení § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovení druhej časti, hlavy VI. diel 4 Trestného zákonníka č. 40/2009 Sb.

V Púchove dňa: 10. mája 2023

podpis autora

Pod'akovanie

Ďakujem vedúcemu práce Ing. Václavovi Kaczmarczykovi, Ph.D. za účinnú metodickú, pedagogickú a odbornú pomoc a mnohé cenné rady pri spracovávaní mojej diplomovej práce.

V Púchove dňa: 10. mája 2023

podpis autora

Obsah

ZOZNAM SYMBOLOV A SKRATIEK	9
ÚVOD	11
1. PLÁNOVANIE A ROZVRHOVANIE ZDROJOV VO VÝROBE	12
1.1 PROGNÓZOVANIE	13
1.2 DRUHY STRATÉGIE PRE RÔZNE TYPY VÝROBY	13
1.3 PLÁNOVANIE VÝROBNÝCH KAPACÍT - PREHĽAD METÓD	14
2. ALGORITMY ROZVRHOVANIA VO VÝROBE	17
2.1 DETERMINISTICKÉ ROZVRHOVANIE VÝROBY	17
2.1.1 <i>Formalizmus značenia úloh rozvrhovania</i>	18
2.2 METÓDY A ALGORITMY PRE RIEŠENIE ÚLOH ROZVRHOVANIA	18
2.2.1 <i>Lineárne programovanie</i>	19
2.2.2 <i>Hill climbing</i>	22
2.2.3 <i>Kritéria výberu vhodnej metódy</i>	24
2.3 VYBRANÉ MATEMATICKÉ MODELÝ PLÁNOVACÍCH PROBLÉMOV VO VÝROBNOM PODNIKU	25
2.3.1 <i>Klasifikácia úloh rozvrhovania</i>	25
2.3.2 <i>Minimalizácia účelových funkcií</i>	28
2.3.3 <i>Modely rozvrhovania na jednom stroji</i>	29
2.3.4 <i>Výpočtová náročnosť</i>	30
3. TYPY RIADIACICH ARCHITEKTÚR VÝROBNÝCH SYSTÉMOV	32
3.1 KRITÉRIA PRI VÝBERE ROZVRHNUTIA RIADIACEJ ARCHITEKTÚRY	32
3.2 POPIS ŠPECIFICKÝCH TYPOV RIADIACICH ARCHITEKTÚR	32
3.2.1 <i>Centralizovaná riadiaca architektúra</i>	33
3.2.2 <i>Hierarchická riadiaca architektúra</i>	33
3.2.3 <i>Upravená hierarchická riadiaca architektúra</i>	34
3.2.4 <i>Heterarchická architektúra</i>	35
4. INTELIGENTNÉ VÝROBNÉ SYSTÉMY	37
4.1 VLASTNOSTI INTELIGENTNÝCH VÝROBNÝCH SYSTÉMOV	37
4.2 INTELIGENTNÉ VÝROBNÉ SYSTÉMY ZALOŽENÉ NA AGENTOVOM PRÍSTUPE	38
4.2.1 <i>Komunikácia, ontológia, interakčné protokoly</i>	39
4.2.2 <i>Decentralizované multiagentné systémy</i>	42
4.3 POTENCIÁL A VYUŽITELNOSŤ MAS V PRIEMYSELNOM PROSTREDÍ	43
4.4 SÚVISIACE TECHNOLOGIE PRÍSTUPU INDUSTRY 4.0	44
4.4.1 <i>Architektúra orientovaná na služby v zmysle normy IEC 61499</i>	45
4.5 REFERENČNÁ ARCHITEKTÚRA CPPS	46
4.5.1 <i>Topológia systému CPPS</i>	46
5. ALGORITMY PRE DISTRIBUOVANÚ ORGANIZÁCIU VÝROBY	49
5.1 ROZHODOVANIE V CPPS	49
5.2 MODEL ROZVRHOVANIA ÚLOH V CPPS	50
5.2.1 <i>Robustnosť rozvrhovania v CPPS</i>	51
5.3 INTERAKCIE MEDZI AGENTAMI V CPPS	52

5.3.1	<i>Protokoly založené na spolupráci</i>	52
5.3.2	<i>Negociačné protokoly - trhové mechanizmy</i>	58
5.4	OPERATÍVNE PLÁNOVANIE	58
5.4.1	<i>Metódy rozvrhovania výroby</i>	59
5.4.2	<i>Stratégie preplánovania výroby</i>	59
5.4.3	<i>Metódy preplánovania výroby</i>	61
5.5	NAVRHNUTÝ ALGORITMUS ORGANIZÁCIE VÝROBY CPPS	62
5.5.1	<i>Model výrobného zdroja (Resource agenta)</i>	63
5.5.2	<i>Model inteligentného výrobku (Product agenta)</i>	64
5.5.3	<i>Plánovanie operácií</i>	65
5.5.4	<i>Heuristika vkladania úloh do rozvrhov resource agentov</i>	67
5.5.5	<i>Možné rozšírenie heuristiky vkladania</i>	69
5.5.6	<i>Dynamické preplánovanie</i>	69
6.	IMPLEMENTÁCIA MAS RIADENIA VÝROBY VO VIRTUÁLNEJ REALITE	71
6.1	SIMULOVANÝ MODEL VÝROBNÉHO PODNIKU	71
6.2	VALIDÁCIA SIMULÁCIE RIADIACEJ LOGIKY	73
6.2.1	<i>Použité metódy dynamického rozvrhovania</i>	75
6.3	OPIS IMPLEMENTÁCIE SIMULAČNÉHO PROSTREDIA	76
7.	ZÁVER	83
	LITERATÚRA	84
	ZOZNAM PRÍLOH	89

ZOZNAM SYMBOLOV A SKRATIEK

Skratky:

APS	Advanced Planning Scheduling (Systémy pokročilého plánovania)
ERP	Enterprise resources planning (Plánovanie podnikových zdrojov)
MES	Manufacturing Execution Systém (Manažérsky systém riadenia výroby)
MRP	Material Requirements planning (Plánovanie požiadavkou na materiál)
MAS	Multiagent System (Multiagentový systém)

ZOZNAM OBRÁZKOV

Obr. 1.1: Ganttov diagram poradia zadaných operácií do výroby a ich vplyv na využitie kapacít [3]	13
Obr. 1.2: Historický prierez plánovacích systémov. [7]	16
Obr. 2.1: Ganttov diagram: (a) orientovaný na stroje (b) orientovaný na úlohy.	18
Obr. 2.2: Grafická reprezentácia parametrov rozvrhovanej úlohy [9]	18
Obr. 2.3: Branch-and-bound metóda - stromová štruktúra algoritmu. [8]	22
Obr. 2.4: Schematické znázornenie generovania lokálnych okolí a ich optím. [15].....	23
Obr. 2.5: Funkčné závislosti nákladových funkcií. [8]	28
Obr. 2.6: Ganttov diagram pre rozvrh na jednom stroji	29
Obr. 3.1 Typy riadiacich architektúr (štvrce sú komponenty riadenia a kruhy sú riadené entity) [19].....	33
Obr. 4.1: Príklad správ (otázok a odpovedí) v jazyku KQML [24]	41
Obr. 4.2: Architektúra federovaného multiagentného systému. [23]	42
Obr. 4.3: Architektúra decentralizovaného multiagentného systému. [23].....	42
Obr. 4.4: Implementačné vrstvy na mikroúrovne agenta v CPPS. [33]	47
Obr. 5.1: Štruktúra hierarchických úrovní automatizácie vo výrobnom podniku.	49
Obr. 5.2: Štruktúra úrovní prvkov automatizácie pri použití CPPS architektúry.	50
Obr. 5.3: UML diagram základného CNP protokolu. [37]	53
Obr. 5.4: CNP protokol nepodporuje synchronizáciu pre súbežnú komunikáciu.	54
Obr. 5.5: UML diagram rozšíreného CNP protokolu s potvrdzovaním. [37]	55
Obr. 5.6: Použitie metódy posunu doprava na preplánovanie výrobného rozvrhu. [39].....	61
Obr. 5.7: Popisovaná štruktúra hierarchického delenia technologických operácií.....	64
Obr. 5.8: Vývojový diagram základných častí algoritmu.	67
Obr. 5.9: Ukážka heuristiky vkladania výrobných operácií do rozvrhov resource agentov.....	69
Obr. 6.1: Rozvrhnutie výrobných zdrojov v simulovanom podniku.....	72
Obr. 6.2: Výpis z kódu konštruktora resource agenta - definícia procesných segmentov.....	72
Obr. 6.3: Výpis z kódu konštruktora product agenta - definícia produktových segmentov.....	73
Obr. 6.4: Grafické okno simulátora - začiatok simulovanej výroby s architektúrou CPPS.	74
Obr. 6.5: Grafické okno simulátora - koniec simulovanej výroby s architektúrou CPPS.	75
Obr. 6.6: Priama komunikácia medzi agentami (CNP - Contract Net Protocol).	76
Obr. 6.7: Prehľadový diagram priebehu rezervácie na strane resource agenta, prijímanie správ.	77
Obr. 6.8: Prehľadový diagram priebehu rezervácie na strane product agenta - prijímanie správ.	77
Obr. 6.9: Prehľadový diagram priebehu rezervácie na strane product agenta - odosielanie správ.....	78
Obr. 6.10: Class-diagram - simulátor distribuovanej výroby (1).	79
Obr. 6.11 Class-diagram - simulátor distribuovanej výroby (2).	80

Obr. 6.12: Class-diagram - simulátor distribuovanej výroby (3).....	81
Obr. 6.13: Class-diagram - simulátor distribuovanej výroby (4).....	82

ZOZNAM TABULIEK

Tabuľka 2-1: Názorný príklad rozvrhovacej úlohy - parametre [8].....	21
Tabuľka 2-2: Prehľad základných druhov dedikovaných výrobných systémov [17]	27
Tabuľka 2-3: Časy potrebné na spracovanie úlohy a váhy - rozvrh na jednom stroji.....	29
Tabuľka 5-1: Rozdelenie - metódy rozvrhovania výroby. [39]	59
Tabuľka 5-2: Stratégie preplánovania výroby v kontexte operatívneho plánovania. [39]	61
Tabuľka 5-3: Metódy preplánovania výrobného rozvrhu. [39].....	62

ÚVOD

Turbulentné prostredie a dynamické zmeny v celosvetovej ekonomike v oblasti výrobných podnikov čoraz viac dáva do popredia potrebu flexibilne a systémovo reagovať na zmeny. Jedným z rozhodujúcich predpokladov správnej a včasnej reakcie na výkyvy či už v časovej doméne, alebo na rôznorodé požiadavky zákazníkov, je efektívne nastaviť a následne riadiť operácie vo výrobe. Takýmito operáciami sa transformujú vstupy na výrobky alebo služby, ich optimálnym naplánovaním a rozvrhnutím v priestore a čase sa zaručí splnenie očakávaní zákazníkov a zároveň ekonomické prežitie a prosperita výrobných podnikov. Kľúčovú úlohu pritom zohráva operatívne prispôsobovanie výroby a sústavné zvyšovanie účinnosti využívania výrobných a ľudských zdrojov.

Problematikou zavádzania a optimalizácie operatívneho rozvrhovania sa zaoberá hlavná časť tejto práce. V práci predstavujem bázu teoretických znalostí a analýz o inteligentných výrobných systémoch, ktoré svojím modulárnym prístupom umožňujú zabezpečovať plánovacie a riadiace funkcie podniku. Popisované multiagentové systémy tvoria vhodné paradigmy na vytvorenie takýchto modelov distribuovaných inteligentných systémov riadenia výroby. Tieto systémy pozostávajú z viacerých "agentov", tj. softvérových entít, ktoré sú navrhnuté tak, aby sa správali autonómne s dôrazom na heterarchické usporiadanie. Tento koncept multiagentných štruktúr v rozhodovacom procese výrobného podniku by mal priniesť systémy, ktoré sú spoľahlivejšie, ponúkajú paralelizmus, škálovateľnosť či robustnosť. Nie všetky tieto vlastnosti sú principiálne späté so základným fungovaním MAS, preto je nutný vývoj a prispôbenie konceptu pre potreby riadenia podniku. Napríklad dosiahnutie robustnosti si vyžaduje viac než len zavedenie redundancie, pretože redundancia nevyrieši problémy súvisiace s neistotami a možnou nestabilitou otvoreného systému. Naopak, môže zvýšiť komunikačné zaťaženie. Ak majú byť multiagentové systémy odpoveďou na dynamické požiadavky trhu, je kľúčové vyvinúť interakčné protokoly s akcentom na tvorbu organizačných foriem (kooperačných skupín) a spolupráce autonómnych agentov. Práve vhodné štrukturovanie multiagentných systémov zvyšuje ich potenciál využitia a aj uvádzanú robustnosť. Možné interakcie agentov a alternatívne koordinačné mechanizmy sú rozoberané v teoretických kapitolách práce.

Výzvou je vyvinúť inovatívnu architektúru pre distribuované riadenie výroby s využitím nových paradigiem MAS. Implementácia takéhoto modelu riadenia výroby je predmetom predloženej diplomovej práce. Na overenie teoretických výstupov bol naprogramovaný simulátor v jazyku C#, ktorý modeluje základné vlastnosti MAS a definuje návrhové parametre tohto prístupu v riadení výrobného podniku. Simulátor predstavuje vhodný nástroj na testovanie rôznych algoritmov decentralizovanej organizácie výroby. Popis implementácie simulátora a príklad simulovanej výrobnéj úlohy je popísaný a vyhodnotený v poslednej kapitole tejto práce.

1. PLÁNOVANIE A ROZVRHOVANIE ZDROJOV VO VÝROBE

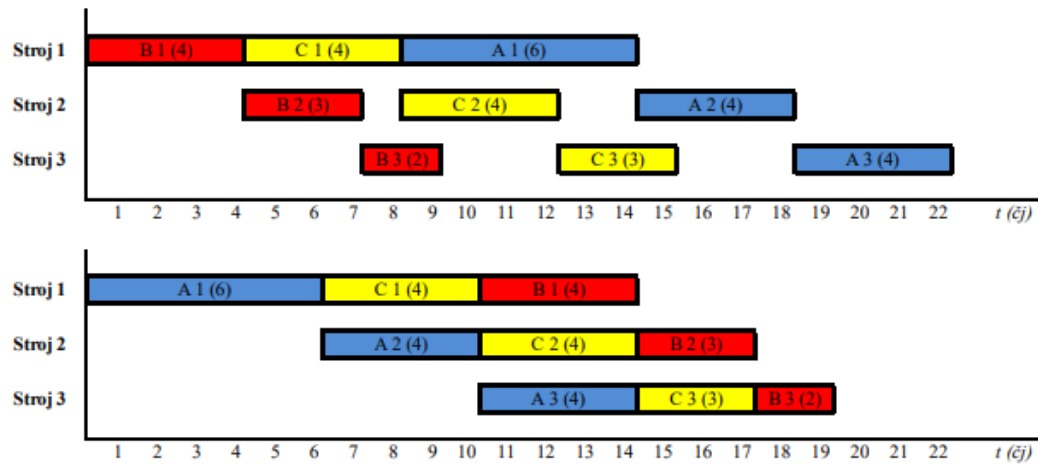
Plánovanie predstavuje jednu z najdôležitejších funkcií pri manažovaní podniku. Správne zvolené plánovanie je strategická funkcia, od ktorej vo veľkej miere závisí ekonomický výsledok podniku, zahŕňa ciele a prostriedky na dosahovanie cieľov a zohľadňuje zdroje podniku s ohľadom na flexibilné prispôsobovanie sa nestálym podmienkam trhu. Konkurencieschopnosť podniku vo veľkej miere závisí od pružnej akceptácie meniacich sa podmienok, od správnych a včasných rozhodnutí v procese plánovania. Táto konkurencieschopnosť v sebe zahŕňa viacero oblastí, ktoré je potrebné zodpovedným spôsobom plánovať. Ide najmä o finančnú efektivitu v pomere vstupných a výstupných veličín, formu stratégie a následnej koordinácie jednotlivých činností vedúcej k dosiahnutiu vytýčených cieľov a tiež k hospodárnemu využitiu časového vkladu vo výrobnom procese. [1]

Plánovanie je charakterizované kauzálnym vzťahom medzi akciami a výberom akcií za účelom dosiahnutia cieľa. Efektívne plánovanie vedie k dosiahnutiu optimálneho fungovania výrobných systémov, eliminuje chyby, ktoré by viedli k plytvaniu finančných prostriedkov alebo času. Dôležitou skutočnosťou determinujúcou efektívne plánovanie je aj správnosť a včasnosť dát z informačných systémov. Plánovanie a riadenie výrobných systémov závisí od konkrétneho charakteru výroby a je špecifické pre daný podnik. Proces rozvrhovania vo všeobecnosti rozplánuje výrobu buď od počiatočného termínu dopredu, alebo od finálneho dozadu. [2]

Výhodou plánovania vo výrobnom podniku je minimalizovanie neproduktívnosti práce zamestnancov, lepšie využitie výrobného zariadenia a zdrojov podniku a tiež slúži ako nástroj pre kontrolu dodržania cieľov. Nevýhodou plánovania sú vysoké vstupné finančné a časové náklady, obmedzenie inovatívnej iniciatívy zamestnancov a v neposlednom rade môže viesť k nadmernej byrokracii. Nevýhodou tradičného plánovania s nízkou flexibilitou je tiež fakt, že často ide iba extrapolované ciele z minulosti.

Rozsah diskutovanej domény plánovania a rozvrhovania výroby nie je vôbec banálny. Problematika je typická protichodnými požiadavkami, ktoré sú kladené na výrobu. Na jednej strane je snaha o maximálne využitie výrobných kapacít pri čo najkratších prestojoch a minimálnych zásobách. Na strane druhej je realita obmedzených zdrojov, materiálových, ľudských či časových. Aj preto je úsilie viacerých firiem sústredené najmä na zoštíhľovanie svojich výrobných procesov, pričom sa značná pozornosť venuje na tvorbu detailného rozvrhu výrobných operácií. Dôležitosť plánovania vo výrobnom podniku demonštruje nasledujúci obrázok Obr. 1.1. Na obrázku sú zobrazené dva rozvrhy s odlišným poradím výroby produktov (A,B,C). Poradie operácie nad produktom je uvedené za menom produktu. Ich časová náročnosť je uvedená v zátvorke (napr. tretia

operácia nad dielcom A s časovou náročnosťou 2 je A3(2)). Z obrázka je zrejmé aj bez ďalších analýz, že rôzna sekvencia vykonávania operácií má značný vplyv na celkovú dobu trvania výroby.



Obr. 1.1: Ganttov diagram poradia zadaných operácií do výroby a ich vplyv na využitie kapacít [3]

1.1 Prognózovanie

Prognózovanie redukuje neurčitost' pri vytvorení plánov a odhad budúcich objednávok a je nástrojom na odhad predaja výrobkov, od čoho sa následne odvíja odhad spotreby materiálov, energie a ďalších zdrojov. Metódy prognózovania delíme na dve hlavné skupiny:

- Kvantitatívne sú založené na analýze a spracovaní sekvencií historických údajov a pozorovaní získaných v určitom pravidelnom časovom intervale. Za predpokladu, že vzťahy medzi hodnotami v minulosti budú pokračovať aj v budúcnosti, môžeme ich extrapoláciou získať prognózu. Dôležitými parametrami prognóz je použitý časový horizont, t.j na aký dlhý čas je potrebné stanoviť prognózu. Na základe tohto parametra delíme prognózy na strategické a taktické.
- Kvalitatívne sú založené na odhade z referencií od zákazníkov a predajcov. Uplatnenie týchto metód je výhodné, ak nemáme dost' relevantných informácií na použitie kvantitatívnych metód. Medzi nevýhody môžeme zaradiť subjektívne posúdenie trhu závislé na individualite predajcov - ľudský faktor a s ním spojená chybovosť. [4]

1.2 Druhy stratégie pre rôzne typy výroby

Rôzne stratégie firiem vyplývajúce z potrieb ich zákazníkov vedú k rozlišovaniu výroby z pohľadu rozpojenia objednávky. Tento pohľad na problematiku umožňuje rozlíšiť kedy sa vyrába na základe prognózovania a kedy je výroba závislá na objednávke zákazníka.

- **výroba na sklad:** pri tomto type výroby sa vychádza z predikcie očakávaných objednávok. Očakávaný dopyt je stanovený vhodne zvoleným prognózovaním vid'. kapitola 1.1. Výsledkom by malo byť vytváranie skladových zásob, ktoré svojim množstvom reflektujú reálny záujem zo strany zákazníkov. Dôležitosť je kladená najmä na riadenie dodávateľského reťazca na základe informácií o stave zásob. Pri plánovaní výroby pri tejto forme stratégie sa použité plánovacie metódy nazývajú tlačné (tj. tovar je zo skladov tlačný k zákazníkovi).
- **výroba na zákazku:** tento prístup k výrobe je väčšinou využívaný na uspokojenie špecifických potrieb zákazníka. Do popredia kladie individuálny prístup ku každej zákazke. Rovnako aj pri tejto stratégii je dôležité čo najpresnejšie odhadnúť reálny záujem zákazníkov a prispôbiť výrobu na najpredávanejšie konfigurácie daného produktu. Stratégia výroby na zákazku vyžaduje iné metódy plánovania, vo všeobecnosti nazývané ako ťažné. (tj. zákazník si ťahá produkt individuálnym výrobným procesom).
- **montáž na zákazku:** je spojením výroby na zákazku a výroby na sklad. Produkt je zostavovaný podľa individuálnych špecifikácií, za použitia komponentov vyrábaných na sklad. Projektant v tomto prípade navrhuje použitý materiál, výrobné fázy a celú konštrukciu výrobku.
- **inžinierske práce na zákazku:** tento typ výroby vyžaduje vysokú mieru pridanej hodnoty na strane výrobcu. Výrobca musí podľa predstavy zákazníka vytvoriť špecifické technické riešenie ako výsledný produkt realizovať [5]

1.3 Plánovanie výrobných kapacít - prehľad metód

Napriek výraznému rozvoju softvérových riešení, metodík a algoritmov rozvrhovania vo výrobe, je v praxi stále využívané aj manuálne rozvrhovanie. Svoje uplatnenie nachádza najmä vo výrobných podnikoch, kde z pohľadu usporiadania výroby by pokročilejšie plánovanie z princípu neprineslo výraznú pridanú hodnotu, napríklad vo výrobe využívajú model flow job v kombinácii s jedným hlavným produktom bez obmien. Rozvrhovanie potom pozostáva z jednoduchého monitorovania priebehu výroby, neuvažujúc jej obmedzenia (úzke miesta). Riadenie výroby prebieha pomocou prioritných pravidiel, štartovacie časy sú determinované odhadom na základe skúseností operátora a dostupnosti materiálu. Najčastejšie je využívané pravidlo FIFO - prvý vstupujúci je aj prvý vystupujúci zo systému. Manuálne rozvrhovanie rozvrhuje priority iba vzhľadom na požadované termíny ukončenia zákazky, tj. najvyššiu prioritu má zákazka, ktorá má najväčšie oneskorenie. Uvedená prioritizácia principiálne vedie k lavínovému efektu, kde okrem už oneskorených zákaziek sa oneskorí aj tie, ktoré mali byť ukončené včas. Do diskutovanej problematiky manuálneho plánovania možno zaradiť aj rozvrhovanie výroby s použitím tabuľkových kalkulačiek alebo plánovacích tabuľ. Avšak opäť je rozsah využitia obmedzený len na dostatočne malé a jednoduché

výrobné systémy, nakoľko pre rozvrhovacie problémy väčšej zložitosti už nie sú schopné prinášať reálne riešenia. Použitie týchto aplikácií v rozvrhovaní zložitejších zákaziek s rozdielnou trajektóriou výrobným systémom vyžaduje vysoký čas potrebný pre generovanie optimálneho rozvrhu. [6]

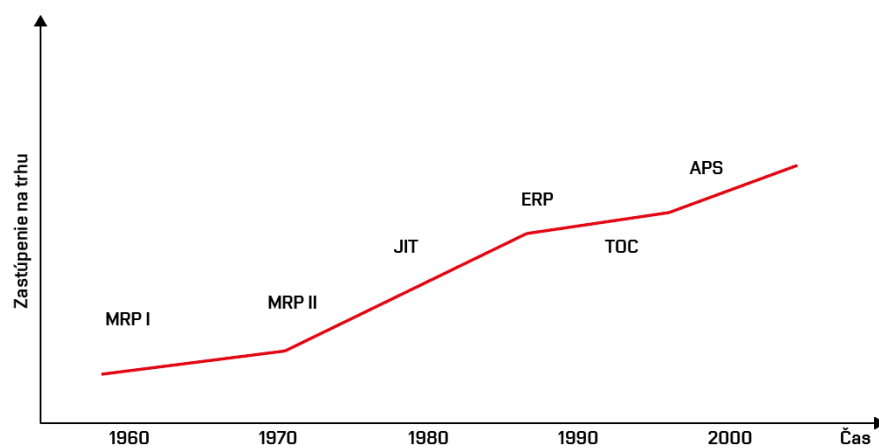
Až rozvoj podnikovej informatiky sprístupnil plnú integráciu pokročilých metód plánovania a riadenia do výrobných podnikov. Jadrom takéhoto riadenia podniku sú ERP systémy, ktoré umožňujú koordináciu všetkých disponibilných podnikových zdrojov a obsahujú širokú škálu modulov napr. ekonomické či logistické. Medzi nimi aj moduly na rozvrhovanie a plánovanie. Tie sa navzájom diferencujú podľa aplikovaného princípu riadenia výroby. Poznáme dva základné princípy logického toku produktu cez výrobný podnik, sú to princíp tlaku a princíp ťahu. Tlačný princíp stanovuje, že výrobok bude vyrábaný na základe prognózovania a je tlačný cez jednotlivé výrobné procesy až k zákazníkovi. Pre ťahový princíp je charakteristické, že výroba je iniciovaná zákazníkom v podobe objednávky. [6]

Metódy na rozvrhovanie materiálových zásob v minulosti vychádzali z konceptu riadenia výroby podľa minimálnych zásob. Tento statický spôsob bol rigidný na zmeny a zbytočne viazal náklady v nutných zásobách. Rozšírením tohto konceptu je metóda MRP, ktorá už má väzbu na reťazec zásobovania, skladovania a dopravy. Vytvára tak rovnováhu medzi požiadavkami trhu a samotnou výrobou. Výstupom bol zoznam zákaziek a podklady na potrebný materiál spolu s reportom dostupných zásob či iniciovaním prípadného naskladnenia. Nedisponoval údajmi o poradí výroby, alebo o optimalizácií rozvrhovania, čo bolo nutné doriešiť dodatočným rozvrhovaním. Nedostatkom tiež bolo, že priebežná doba výroby bola dopredu daná a nemenná. Koncept viedol k nereálnym výrobným plánom, t.j. nebolo možné rozvrhovať výrobu do obmedzených kapacít. [7; 5]

Doplnením MRP o kapacitné plánovanie a nákup v podobe spätnoväzobnej slučky na reálnu výrobu a predaj vznikla metóda MRP II. Toto doplnenie však predstavovalo iba technologickú zmenu, nie koncepcnú. Zohľadňovaním kapacitných obmedzení výroby spolu so zvýšením kvality a agregácie toku dát sa predchádza nereálnemu plánovaniu typickému pre jeho predchodcu. Metódy MRP a MRP II patria medzi tlačný princíp riadenia výrobného podniku. Výhodou je možnosť integrácie týchto metód do celopodnikového systému plánovania zdrojov ERP.

Medzi ťahový princíp riadenia toku produktov cez výrobný podnik môžeme zaradiť koncept (filozofiu) JIT (Just in Time). Filozofia dáva do popredia samotné riadenie výrobného procesu s prihliadnutím na všetky články dodávateľského reťazca. Medzi hlavné charakteristiky konceptu JIT patrí dôraz na nulové zásoby a nulové straty a maximálna efektívnosť. Filozofia JIT (a jej príbuzná Kanban) je s výhodou používaná všade tam, kde sa vyžaduje štíhla výroba. Ide predovšetkým o opakovanú výrobu rovnakých, alebo príbuzných produktov s homogénnym odbytom.

Systém pokročilého plánovania APS eliminuje nevýhody uvádzaných metód MRP. Predstavuje iba nadstavbu ERP závislú od jeho dát, nie je to systém fungujúci samostatne. Systémy APS v sebe spájajú možnosť plánovať s obmedzenými kapacitami a riadiť výrobu s prihliadnutím na obmedzenia úzkych miest vo výrobe. Umožňujú ucelené plánovanie celého dodávateľského reťazca, s podporou rozhodovania v rámci SCM (Supply Chain Management), t.j. celý dodávateľský reťazec až k zákazníkovi. Výstupom APS systému je realistický a optimalizovaný výrobný plán, ktorý reflektuje reálne obmedzenia a okolnosti vo výrobe. Systémy APS spolu s ERP systémami a ďalšími optimalizačnými a simulačnými nástrojmi sú základnými informačnými technológiami riadenia výroby v podniku. [7; 5]



Obr. 1.2: Historický prierez plánovacích systémov. [7]

2. ALGORITMY ROZVRHOVANIA VO VÝROBE

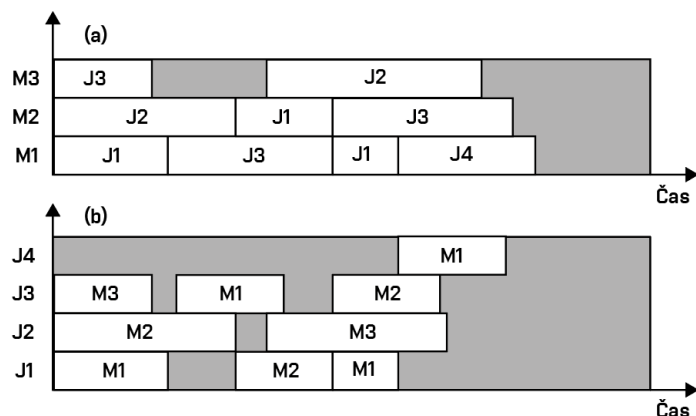
Optimalizácia výrobného procesu zahŕňa celý rad nástrojov z oblasti operačnej analýzy, štatistického zisťovania a manažmentu. Cieľom takejto optimalizácie je nájsť riešenia, ktoré budú vo vysokej miere účinné z pohľadu časovej náročnosti, kvality výroby, náklady na servis. Prostriedkom na nájdenie najlepšieho riešenia v úlohách rozvrhovania, tj. optimálneho rozvrhu je účelová funkcia vyjadrená vhodným matematickým vzorcom.

2.1 Deterministické rozvrhovanie výroby

Pri rozvrhovaní výroby sa každá zákazka rozkladá na množinu usporiadaných menších operácií (úloh) s definovaným termínom ukončenia. Každá z nich vyžaduje spracovanie na určitom stroji po určitú dobu. Vo všetkých vybraných modeloch sa vychádza z predpokladu, že všetky údaje o probléme (t. j. počet úloh, časy spracovania, dátumy zadania, termíny ukončenia, váhy atď.) sú známe v čase „nula“. Plánovač, má teda k dispozícii všetky informácie a môže v čase „nula“ určiť celý plán. Toto paradigma sa označuje ako offline plánovanie. V praxi poznáme aj pojem online plánovanie, kedy plánovač rozhoduje o poradí vykonávaných úloh len na základe znalosti počtu pracujúcich strojov (pri paralelnom modeli rozvrhovania), pričom nepozná počet ešte nespracovaných úloh a nedisponuje ani informáciou o čase ich trvania. Pre tento prípad sa používajú algoritmy LIST prípadne Round Robin. [8]

Cieľom deterministického rozvrhovania je nájsť rozvrh, ktorý určuje, kedy a ktorá zákazka a na ktorom stroji bude spracovávaná, pričom sa kladie dôraz na optimalizáciu určitej účelovej funkcie, ktorá vychádza z modelu rozvrhovania.

Vo všetkých v práci uvažovaných problémoch plánovania sa explicitne predpokladá konečný počet úloh (označujeme n s indexom j) a strojov (označujeme m s indexom i). Ak úloha vyžaduje určitý počet krokov spracovania alebo operácií, potom dvojica (i, j) označuje rozvrhovaný krok spracovania, respektíve operáciu úlohy j na stroji i . Rozvrhy vo všeobecnosti môžu byť reprezentované Ganttovými diagramami, ako je znázornené na obrázku Obr. 2.1. Ganttové diagramy môžu byť orientované na stroje (Obr. 2.1 (a)) alebo na úlohy (Obr. 2.1(b)).



Obr. 2.1: Ganttov diagram: (a) orientovaný na stroje (b) orientovaný na úlohy.

2.1.1 Formalizmus značenia úloh rozvrhovania

Čas spracovania (p_{ij}): reprezentuje čas potrebný na spracovanie úlohy j na stroji i . Ak sa úloha j má spracovať na konkrétnom stroji alebo naopak nezáleží od typu stroja index i vynechávame.

Čas začiatku spracovania (r_j): možno tiež chápať ako čas pripravenosti, najskorší čas, kedy je úloha j uvoľnená do výroby.

Koncový čas spracovania (c_j): koncový čas operácie j .

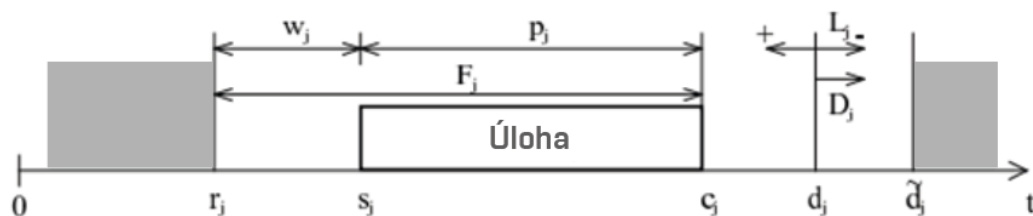
Čas splatnosti (d_j): čas do ktorého by daná úloha mala byť dokončená. Ukončenie po termíne splatnosti je nežiadúce a penalizované. Čas kedy musí byť úloha ukončená sa považuje za deadline a označujeme ho \bar{d}_j .

Váha (w_j): koeficient dôležitosti danej úlohy, označuje jej prioritu vzhľadom na iné úlohy.

Oneskorenie (L_j): penalizovaná odchýlka od definovaného času kedy mala byť úloha dokončená.

Množina (P_j): množina všetkých predchodcov operácie j . Tieto operácie musia byť dokončené pred r_j .

Množina (K): množina reprezentujúca kapacitu všetkých výrobných zdrojov.



Obr. 2.2: Grafická reprezentácia parametrov rozvrhovanej úlohy [9]

2.2 Metódy a algoritmy pre riešenie úloh rozvrhovania

Vo všeobecnosti konvenčné metódy na rozvrhovanie a plánovanie výroby môžeme rozdeliť do niekoľkých skupín podľa viacerých vlastností. [4]

- Metódy operačného výskumu - pracujú s exaktnými matematickými modelmi a algoritmami, ktoré sú účinným nástrojom na reprezentáciu mnohých plánovacích/rozvrhovacích úloh. Rozlišujeme dve skupiny matematických modelov v rámci operačného výskumu, rozhodovacie - obsahujú kritériálnu funkciu a optimalizujú. Druhou skupinou sú modely technologické - neobsahujú kritériálnu funkciu - neoptimalizujú.
- Heuristicky orientované prístupy - využívame najmä v prípadoch, kde nie je známy všeobecný algoritmus pre získanie riešenia problému, prípadne z dôvodu zložitosti rozvrhovacieho problému nie je možné zostrojiť matematický model. Takéto riešenie pozostáva z analýzy jednotlivých čiastkových krokov, pričom sa hodnotí najmä ich prínos ku konečnému výsledku. Z výsledkov jednotlivých krokov sa iteratívne určuje spôsob, akým je nutné spresniť nasledujúce iterácie. V rozvrhovaní výroby sa použitie heuristických metód využíva najčastejšie pri výbere z množiny úloh na základe zvolených pravidiel, prípadne podľa skúseností pracovníkov.
- Úplné metódy - na ich konci je zaručené nájdenie optimálneho riešenia úloh rozvrhovania.
- Neúplné metódy - (stochastické) neprehľadávajú celý stavový priestor, ale iba jeho časť. Na ich konci je zaručené iba suboptimálne riešenie. S výhodou sa používajú najmä pre rozsiahle úlohy, kedy by úplné metódy zlyhávali v rozsiahlych kombinatorikách.
- Neštandardné metódy - meta-heuristiky, patria sem napríklad neurónové siete, teória učenia strojov, fuzzy logika či genetické algoritmy. Tieto výsledky a metódy umelej inteligencie sú zdrojom nových možností riešenia problémov v oblasti výroby.

2.2.1 Lineárne programovanie

Patrí medzi úplné (presné) metódy operačného výskumu, ktoré maximalizujú alebo minimalizujú kritériálnu funkciu vhodnou voľbou hodnôt koeficientov obmedzených premenných. Predpokladom použitia tejto optimalizačnej metódy je linearita optimalizovanej kritériálnej funkcie a aj všetkých jej obmedzení. [10]

Symbolicky je možné takúto úlohu definovať:

$$\min C_{n+1} \quad 2-1$$

$$C_h \leq C_j - p_j \quad j = 1, \dots, n + 1; h \in P_j$$

$$\forall J^* \neq J: C_{ij} + d_{ij} \leq C_{ij}^* \text{ alebo } C_{ij}^* + d_{ij}^* \leq C_{ij} \quad 2-2$$

$$C_j \geq 0 \quad j = 1, \dots, n + 1 \quad 2-3$$

Cieľ je minimalizovať celkovú dobu trvania zákazky (rovnica 2-1) pri platných obmedzeniach, že operácia nemôže začať skôr, než sú ukončené predchádzajúce - precedenčné obmedzenie. Nerovnice (2-3) zastupujú dvojicu disjunktných obmedzení, z ktorých bude zakaždým platiť práve jedno. Tým sa zabezpečí podmienka, že v danom okamihu môže na pracovisku prebiehať iba jedna operácia. Ďalšie možné príklady obmedzenia sú definované viď. kapitola 2.3.1. Vzťah (2-4) deklaruje nezápornosť koncových časov vykonaných operácií. Takýmto postupom dostávame príslušný počet konjunktívnych množín nerovnic definujúcich rozvrhovací problém. [11]

Riešenie problému lineárneho programovania nachádzame použitím grafickej, simplexovej alebo duálnej metódy. Grafická metóda je z menovaných najintuitívnejšia. Funkčné závislosti obmedzení zastupujú priamky v 2D priestore, ktoré vymedzujú množinu prípustných hodnôt. Nevýhodou je použiteľnosť iba v najjednoduchších zadaniach lineárneho problému. Ďalšia z metód je simplexová metóda, predstavuje postupne prípustné riešenia, pričom ich optimalitu určuje z hodnôt minimalizovanej kritériálnej funkcie. Zadanie úlohy musí byť v kanonickom tvare, ktorý sa prepíše do tzv. simplexovej tabuľky. V tabuľke pomocou matematických operácií vytvárame jednotkovú maticu a z nej už vychádza výsledok. Duálna metóda rieši opačnú úlohu nad kritériálnou funkciou, tj. ak je úlohou tej primárnej maximalizácia kritériálnej funkcie, duálna rieši minimalizáciu a naopak. Duálna úloha lineárneho programovania je rovnocenná s primárnou (pôvodnou) úlohou a vo výsledku nezáleží na tom, ktorá je primárna a ktorá duálna. Obidve sa blížia k optimálnemu riešeniu z inej strany. [12]

Podmnožinou tejto metódy je celočíselné programovanie (integer programming IP), ktoré je s výhodou využívané najmä pri rozvrhovacích problémoch. Metóda vetiev a medzí (Branch and Bound) je jedným z najrozšírenejších algoritmov optimalizácie pri riešení práve úloh celočíselného programovania. Vo všeobecnosti sa nejedná o jediný algoritmus, ale o celú skupinu metód individuálne prispôbených na riešenie konkrétneho optimalizačného problému. Patria sem najdôležitejšie metódy vetvenia (branching), metódy hľadania medzí (bounding) a mechanizmus prerezávania (pruning). Branch-and-bound používa pre obmedzenie prehľadávaného priestoru tzv. hornú a dolnú hranicu nákladov. Voľba týchto hraníc (medzí) určuje inicializačný interval, ktorý je dôležitý, pretože výrazne ovplyvňuje celkový čas potrebný na nájdenie riešenia. V rozvrhovaní sú to napríklad údaje o najskoršom a najneskoršom možnom štarte každej z rozvrhovaných úloh. Rýchlosť konverencie k optimálnemu riešeniu tiež determinujú faktory, akými sú kvalita vyhodnocovacej funkcie, zvolené poradie prehľadávania pri orezávaní a v neposlednom rade tiež spôsob reprezentácie úlohy najmä v súvislosti s disjunktnými obmedzeniami, ktoré sú najväčším zdrojom komplexnosti rozvrhovacích úloh. [11]

Algoritmus postupuje smerom do hĺbky z vyššej úrovne na úroveň nižšiu, až kým nepreskúma všetky vetvy na poslednej úrovni. Dôležité je, že si pamätá doposiaľ najlepšie riešenie, ktoré pri prechode na ďalší uzol porovná s odhadom riešenia (formou

vyhodnocovacej funkcie) pod ním. Ak je horšie než už dosiahnuté optimum, je zrejmé, že neexistuje možnosť, aby v ktoromkoľvek uzle z daného pod-priestoru bolo nájdené lepšie riešenie. Uzol sa teda nepreskúmava a oreže sa. Existuje niekoľko spôsobov, ako možno získať hraničnú hodnotu vyhodnocovacej funkcie. Pre ilustráciu bol zvolený nasledovný spôsob - naplánovať v každom z uzlov zvyšný počet úloh podľa preemptívneho pravidla EDD (úlohy s najskorším termínom dokončenia úlohy ako prvé) a s dôrazom na minimalizáciu oneskorenia úlohy (viď. kapitola 2.3.1). [13] Behom riešenia vzniká stromová štruktúra, zastupujúca prehľadávaný stavový priestor.

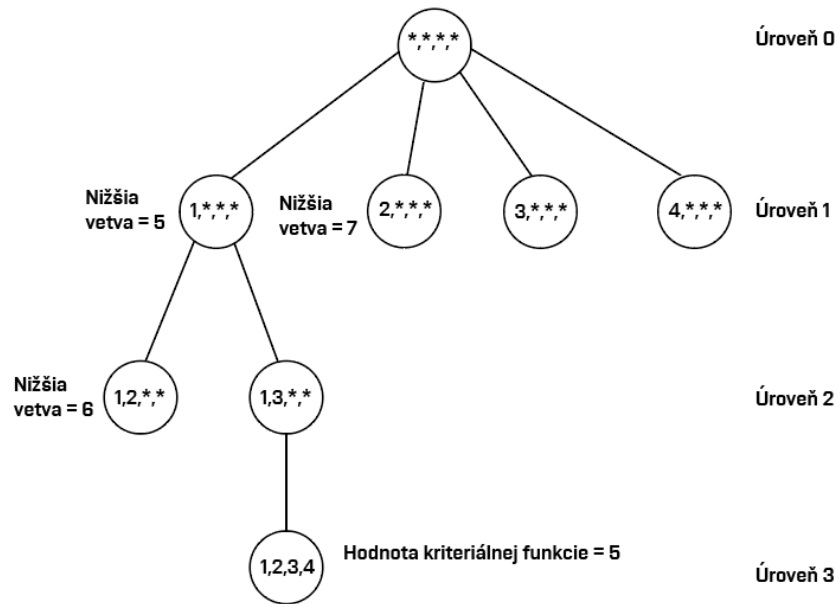
Tabuľka 2-1: Názorný príklad rozvrhovacej úlohy - parametre [8]

Úloha	1	2	3	4
(pj)	4	2	6	5
(rj)	0	1	3	5
(dj)	8	12	11	10

Ilustračný príklad algoritmu začína koreňovým uzlom, uzol v bode 0. Do tohto uzla nie je vložená žiadna rozvrhovaná úloha. Nižšie na úrovni 1. má každý uzol konkrétnu úlohu umiestnenú na prvú pozíciu v rozvrhu. V každom z týchto uzlov je teda ešte $n-1$ úloh, ktorých pozícia v rozvrhu nebola určená, tj. (1, *, *, *), (2, *, *, *), (3, *, *, *) a (4, *, *, *). Je zrejmé, že uzly (3, *, *, *) a (4, *, *, *) môžu byť okamžite vylúčené, nakoľko nesplňujú kritérium optimality rozvrhu - úloha 3 je uvoľnená až v čase 3. Ak by napr. úloha 2 začala svoje spracovanie v čase 1, úloha 3 môže rovnako začať v čase 3. Uvedené platí aj pre úlohu 4. [8]

Výpočet medze pre uzol (1, *, *, *) podľa zvoleného preemptívneho pravidla EDD vedie k rozvrhu, v ktorom sa úloha 3 spracúva počas časového intervalu [4,5], následne je prerušená úlohou 4 počas časového intervalu [5,10]. Úloha 3 sa potom dokončuje počas intervalu [10,15]. A posledná úloha 2 počas intervalu [15,17]. Algoritmus postupuje do druhej (nižšej) úrovne v stromovej štruktúre na uzol (1, 3, *, *). Hodnota vyhodnocovacej funkcie $L_{max} = 5$ a je určená už nepreemptívnym rozvrhom úloh: 1,3,4,2., tento rozvrh je z vyšetovaného hľadiska globálne optimálny. Podobným spôsobom možno určiť hraničnú medzu aj pre uzol (2, *, *, *), tá je 7, čo je viac ako už

dosiahnuté optimum, preto je uzol a k nemu prislúchajúci pod-priestor automaticky vylúčený. [8]



Obr. 2.3: Branch-and-bound metóda - stromová štruktúra algoritmu. [8]

Algoritmus - Všeobecná branch and bound (BAB) procedúra [14]

```

1:   Solution = ∅
2:   procedure BAB(x)
3:       bound1
4:       . . .                                ▷ kontrahovacia sekcia
5:       bound1
6:       prune1
7:       . . .                                ▷ orezávacia sekcia
8:       prunek
9:       if x ≠ ∅ then
10:          if size(x) > ε* then
11:             [x1, x2] ← split(x)
12:             BAB(x1)
13:             BAB(x2)
14:          else
15:             Solution ← x                    ▷ pridanie do riešenia
16:          end if
17:       end if
18:   end procedure

```

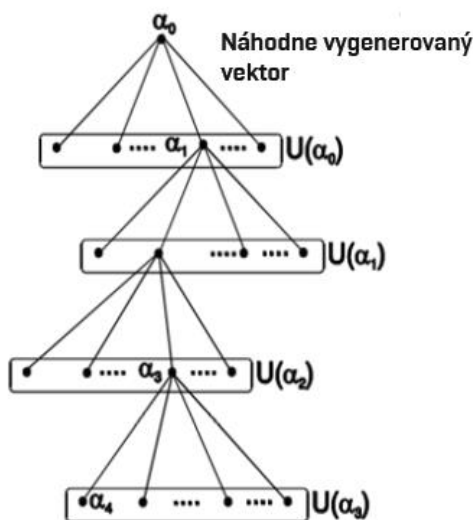
*ε = koeficient veľkosti, definuje presnosť implementovanej mriežky prerezávania.

2.2.2 Hill climbing

Patrí medzi neúplné (stochastické) metódy založené na heuristickom prístupe. Horolezecký algoritmus (hill climbing) prehľadáva stavový priestor optimalizačných úloh iba lokálne a teda prináša len suboptimálne riešenia. Tieto suboptimálne riešenia

sú ale v praxi častokrát akceptovateľné. Uplatnenie nachádza najmä pri komplexných úlohách, kedy úplné metódy zlyhávajú kvôli enormnému nárastu potreby na výpočtový výkon. Kritický parameter úspešnosti pri tejto metóde je silne závislý od kvality zvolenej funkcie susednosti, tak aby efektívne prehľadávala stavový priestor. Táto funkcia priradzuje každému stavu skupinu ďalších - “susedných” stavov v priestore prehľadávania. Algoritmus začína na náhodne, alebo pomocou niektorej z heuristík, vygenerovanom mieste, odkiaľ sa presunie do susedného stavu, ktorý je vzhľadom na optimalizačnú funkciu výhodnejší (s minimálnou hodnotou vyhodnocovacej funkcie). Nový stav sa použije v nasledujúcom iteračnom kroku ako stred novej oblasti. Proces sa jednoducho opakuje, predpísaný početkrát, pričom sa stále ukladá najlepšie riešenie v priebehu histórie algoritmu. Limitne je schopný nájsť aj globálne optimálne riešenie, ale počet susedných skúmaných elementov by sa blížil počtu všetkých možných rozvrhov. V tom prípade by už algoritmus nebol iteračný, pretože by sme našli optimálne riešenie v rámci jedného prechodu algoritmu.

Na podobnom princípe fungujú aj iné stochastické metódy optimalizácie rozvrhovacích úloh ktorými sú: tabu search (metóda zakázaného prehľadávania), alebo metóda simulovaného žihania (Simulated annealing). Ilustračné znázornenie prechodu hill climbing algoritmu vid'. Obr. 2.4. α zastupuje lokálne optimum a $U(\alpha)$ dané skúmané okolie. [15]



Obr. 2.4: Schematické znázornenie generovania lokálnych okolí a ich optím. [15]

Zásadným nedostatkom popisovanej metódy je riziko, že algoritmus uviazne v lokálnom optime, z pohľadu kvality susedných riešení. Tento problém je možné odstrániť zväčšením množiny susedov (počtu elementov v $U(\alpha)$), prípadne opakovať metódu pre rôzne vygenerované počiatkové riešenia. V literatúre je možné tiež nájsť pokročilejšie formy tohto algoritmu, napríklad horolezecký algoritmus s učením, ktorý obsahuje navyše parameter určujúci rýchlosť konvergencie pri zlepšovaní optimality.

Algoritmus - Hill Climbing procedúra [15]

```
1: procedure Hill_Climbing(x)
2:    $f_{\text{optimal}} := \infty$ ;  $t := 0$ ;
3:   while  $t < t_{\text{max}}$  do
4:      $\alpha^* = \arg \min ( \mathcal{T}(\alpha) )$  // najlepšie riešenie v danom okolí
5:     if  $f( \mathcal{T}(\alpha^*) ) < f_{\text{optimal}}$  then
6:       {
7:          $f_{\text{optimal}} := f( \mathcal{T}(\alpha^*) )$ 
8:          $\alpha_{\text{optimal}} := \alpha^*$ 
9:       }
10:    end if
11:     $\alpha := \alpha *$ 
12:  end while
13: end procedure
```

2.2.3 Kritéria výberu vhodnej metódy

Kritérium splniteľnosti alebo optimalizácie - výber kritéria by mal reflektovať na našu prioritu. Buď do popredia kladieme splnenie všetkých ohraničení (úlohy splniteľnosti) alebo riešenie, ktoré je optimálne podľa zvoleného kritéria. Pri pojme ohraničenia je dôležité spomenúť delenie na tvrdé a preferenčné ohraničenia. Tvrdé ohraničenie v rozvrhovaní je definované napríklad limitnými vlastnosťami technológie výroby. Preferenčnými možno chápať ohraničenia, ktoré sú vítané, ale nie nevyhnutné. Takéto preferenčné obmedzenia možno premietnuť do nákladov úlohy rozvrhovania a problém riešiť ako optimalizačný.

Kritérium časovej náročnosti alebo optimálnosti - voľba vhodného algoritmu bude závislá od rozhodnutia, ktorá požiadavka bude v konkrétnom prípade dôležitejšia. Ak je priorita kladená na nájdenie zaručene optimálneho riešenia, je vhodné použiť niektorú z úplných metód. Problémom zostáva ich časová náročnosť, ktorá exponenciálne narastá s veľkosťou rozvrhovacej úlohy. Pri dlhodobom plánovaní nie je časová náročnosť výpočtu až tak kritický parameter, ale dôraz je kladený najmä na optimálnosť riešenia. Naopak, v časovo kritických situáciách rozvrhovania výroby (mimoriadny stav) je použitie niektorej zo stochastických (neúplných metód) plne dostačujúce aj za cenu iba suboptimálneho riešenia. [11]

Kritérium špecifického popisu problému - špecificita zadania rozvrhovacej úlohy a jeho formálna reprezentácia determinuje použitie vhodnej metódy. Napríklad metóda lineárneho programovania vyžaduje množinu lineárnych rovníc a nerovnic a matematický model kriteriálnej funkcie, ktorej hodnotu sa snaží maximalizovať, resp. minimalizovať. Stochastické metódy je výhodné použiť na široké spektrum úloh. Ich opodstatnené použitie ale predpokladá podobnosť medzi nákladmi a riešeniami rozvrhovacích úloh, inak sú porovnateľné s algoritmom backtracking - návrat k exponenciálnej časovej náročnosti. [11]

Implementačné kritérium - zvolená metóda na riešenie rozvrhovacej úlohy by mala byť algoritmizovateľná v dostatočnej miere pre našu aplikáciu. Zároveň by algoritmus svojou náročnosťou a rozsahom nemal prevýšiť pridanú hodnotu, ktorú prináša. Dôraz pri výbere metódy z pohľadu implementácie je kladený na flexibilitu, efektívnosť prípadnú rozšíriteľnosť rozvrhovacích aplikácií. Za dobre definované algoritmy s priamočiarou implementáciou možno považovať algoritmy lineárneho programovania. Väčší odborný vklad od riešiteľa vyžaduje algoritmus stochastickej metódy hill climbing. [11]

2.3 Vybrané matematické modely plánovacích problémov vo výrobnom podniku

V posledných päťdesiatich rokoch sa venovalo značné množstvo úsilia zameraného na výskum deterministického plánovania. V odbornej literatúre je možné nájsť široké spektrum deterministických modelov, s väčšou či menšou mierou abstrakcie, spolu s prislúchajúcim zápisom, ktorý zodpovedá ich štruktúre. Väčšina z nich je implementovaná aj v rôznych softvérových produktoch pre plánovanie výroby, najčastejšie v APS systémoch. V literatúre sa problém rozvrhovania často obecnnejšie označuje RCPSP "Resource-Constrained Project Scheduling Problem".

2.3.1 Klasifikácia úloh rozvrhovania

Široké spektrum rôznych typov úloh na rozvrhovanie vyžaduje zadenovanie spôsobu klasifikácie. Na opis problému rozvrhovania sa používa Grahamova notácia, ktorá pozostáva z troch častí označených δ , β , γ . [8]

Prvý zo symbolov δ - označuje cestu operácií (úloh) cez stroje, tj. určuje druh modelu rozvrhovania strojného vybavenia. Symbol δ môže nadobúdať hodnoty:

Jednostupňová výroba - jeden stroj (1) - najjednoduchšie strojové prostredie, rozvrhovanie výroby prebieha na jednom stroji. Prirodzené číslo nasledujúce za symbolom δ vyjadruje počet strojov, s ktorými v úlohe počítame.

Viacstupňová výroba - paralelné výrobné zdroje

- **Paralelné identické stroje (P_m):** reprezentuje m identických strojov situovaných paralelne. Ak úloha j vyžaduje spracovanie jednej operácie, je možné ju spracovať na ktoromkoľvek z m strojov. V tejto súvislosti možno vytvoriť aj podskupiny strojov M_j , kedy by úloha nemohla byť spracovaná na hociktorom stroji, ale len stroji patriacemu do danej podskupiny. Reprezentovalo by sa tým obmedzenie vstupujúce do klasifikácie β .
- **Paralelné stroje s rôznymi rýchlosťami (Q_m)** označuje prípad, keď každý z identických strojov má inú rýchlosť vykonávania úloh definovanú ako:

$p_{ij} = p_j / v_i$ Rýchlosť stroja (v_i) je ale rovnaká pre všetky úlohy tzn. je uniformná.

Dedikované druhy (špecializované z pohľadu strojného vybavenia) - vyznačujú sa tým, že každá úloha môže bežať len na špeciálne určenom stroji. Úlohy sa rozdeľujú do skupín, tzv. zákaziek: $J_k = [J_{1,k}, \dots, J_{n,k}]$. Rozlišujeme štyri základné modely týchto úloh

- **Flow shop (F_m):** v preklade „prúdový problém“ označuje model m strojov v sériovom usporiadaní. Každá úloha musí byť spracovaná na každom stroji a úlohy musia sledovať rovnakú cestu, tj. musia byť spracované na stroji 1, potom na stroji 2, atď. Pri vykonávaní môžu vzniknúť medzi rozvrhovanými strojmi fronty, ktoré zvyčajne pracujú na princípe FIFO (first in first out). Vtedy hovoríme o permutačnom flow shop modeli (zn. „prmu“).
- **Job shop (J_m):** v preklade „sekvenčný problém“. Všetky úlohy majú svoju individuálne predurčenú cestu výrobnými strojmi. Každá zákazka má svoj technologický postup, pričom neplatí pravidlo, že na všetky zákazky musia využiť všetky stroje. V tomto modeli rozdeľujeme úlohy na tie, z ktorých každá prejde cez každý stroj len raz, a úlohy, kde nastáva recirkulácia (zn. „rcrc“).
- **Open shop (O_m):** v preklade „otvorený problém“ je špecifickým modelom rozvrhovania vo výrobe, v ktorom nie je poradie operácií na zákazke dopredu zadefinované. Každá zákazka musí byť spracovaná na každom z m strojov, avšak niektoré z týchto časov môžu byť nulové. Model nedisponuje explicitnými obmedzeniami smerovania každej úlohy naprieč strojným vybavením. Rozličné úlohy majú rozličné trasy aj časy spracovania.
- **Re-Entrant (RMSs) HMLV (High-Mix Low-Volume Production System):** ide spravidla o automatizované výrobné systémy, ktoré umožňujú produkovať na rovnakej linke rozdielne, ale technologicky príbuzné výrobky s rešpektovaním individuálnych požiadaviek zákazníka. V mnohovýrobných výrobných systémoch sa kladie dôraz najmä na variabilitu výrobných kapacít. Odlišnosťou od tradičných výrobných systémov je prekrývanie dávkového a diskrétného fungovania výroby. Typickým príkladom výrobného systému HMLV je výroba polovodičov. Pri rozvrhovaní takto kombinatoricky náročných úloh sa využívajú najmä fuzzy a genetické algoritmy. [16]

V literatúre je možné nájsť aj kombinácie pomenovaných modelov, ktoré zovšeobecňujú vlastnosti viacerých v jednom modeli, napríklad flexibilný job shop (FJ_c) alebo flexibilný flow shop (FF_c). Kde namiesto m strojov pracujeme s indexom c označujúcim úroveň. Tieto úrovne obsahujú niekoľko sériovo/paralelne usporiadaných identických strojov. [8]

Tabuľka 2-2: Prehľad základných druhov dedikovaných výrobných systémov [17]

Typ modelu rozvrhovania	Počet úloh v rámci zákazky	Poradie úloh v rámci zákazky
Flow shop	Rovnaký pre všetky výrobné zákazky Jk	Pevne dané, rovnaké pre všetky výrobné zákazky Jk
Job shop	Rôzny pre jednotlivé výrobné zákazky Jk	Pevne dané, rôzne pre jednotlivé zákazky Jk
Open shop	Rovnaký pre všetky výrobné zákazky Jk	Nedefinované (ľubovoľné)

β určuje charakteristiky spracovania, najčastejšie obmedzenia. Môže obsahovať aj niekoľko vstupov. Symbol β môže nadobúdať hodnoty:

Preemptions (prmp): (právo na prerušenie) reprezentuje stav, kedy nie je nevyhnutné udržať vykonávanie úlohy od začiatku až po jej ukončenie. Teda plánovač má právo zastaviť spracovávanie úlohy s nižšou prioritou, tú uložiť tak, aby jej postup nebol stratený, a využiť zdroje a prostriedky na ukončenie úlohy s vyššou prioritou.

Prednostné obmedzenie (prec): toto obmedzenie je jedno z najčastejších v rozvrhovacích problémoch. Reprezentuje podmienku, aby jedna alebo viacero úloh boli riadne ukončené pred začatím spracovania ďalšej úlohy. Každá úloha má teda určeného svojho predchodcu a nástupcu. Iným typom obmedzenia sú precedencie vo forme disjunktneho ohraničenia. Vzniká pri zdieľaní výrobného stroja viacerými úlohami, pričom naraz je možné obslúžiť len jednu z úloh. Sú zdrojom prestojov vo výrobe.

Obmedzenia spôsobilosti strojov (M_j): pri paralelnom usporiadaní strojov označuje takú skupinu strojov, ktoré sú schopné danú úlohu j spracovať.

Blokovanie (block): Výkonnosť rozvrhovania vo výrobe najmä v modeli flow shop ovplyvňuje veľkosť zásobníka medzi dvoma za sebou idúcimi strojmi. Ak sa zásobník naplní (prípadne je nulový), predchádzajúci stroj nemôže uvoľniť úlohu nasledujúcemu a tak poskytnúť svoje prostriedky ďalšej úlohe.

Bez čakania (No-wait, nwt): Vyskytuje sa všade tam, kde počas vykonávania úlohy nemôže nastať prestoj medzi dvoma strojmi, najčastejšie v permutačnom (FIFO) flow shop modeli. Znamená to, že čas začatia úlohy musí byť rozvrhnutý tak, aby zabezpečoval, že úloha prejde procesom výroby bez toho, aby musela čakať na nejaký stroj. [8]

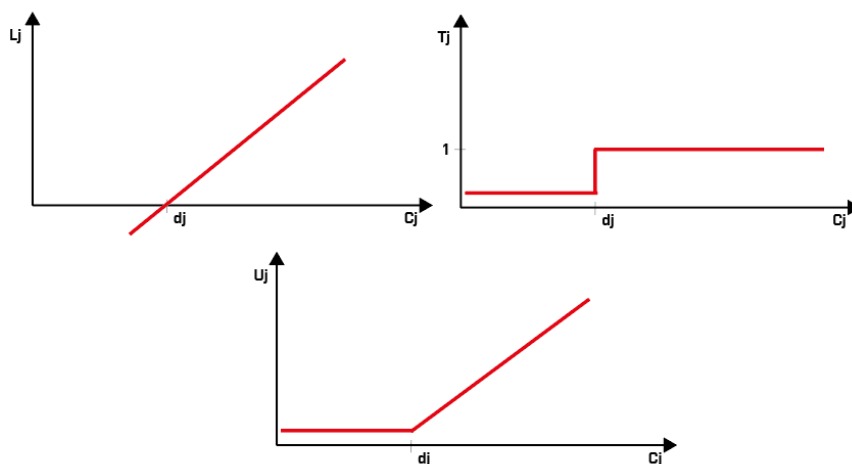
γ obsahuje cieľové hodnoty a účelové funkcie, ktoré majú byť minimalizované, tj. špecifikuje optimalizačné kritérium. Pri zostavovaní optimálneho rozvrhu využívame tieto pravidlá (funkcie), ktorými definujeme logiku prechodu úloh cez výrobný systém, s cieľom optimalizovať hodnotu vybraného kritéria. Popisované funkcie možno nazvať aj pravidelné výkonové merania. Sú neklesajúce v rozmedzí (C_1, \dots, C_n). V praxi najčastejšie používané kritéria optimálnosti možno rozdeliť do skupín podľa zamerania. Sú to kritéria založené na časoch ukončenia výroby, časoch dodávok (oneskorenia), alebo skladových spotrebných nákladov. Vybrané vstupné hodnoty do γ poľa môžu byť napríklad:

Výrobný rozsah (C_{\max}): definovaný ako $\max(C_1, \dots, C_n)$ je ekvivalentný času ukončenia poslednej úlohy rozvrhovanej zákazky. Najčastejšie pri tvorbe optimálneho rozvrhu je predmetom minimalizácie práve čas ukončenia poslednej z úloh.

Diskontovaný celkový vážený čas ukončenia ($\sum w_j(1 - e^{-rC_j})$): zovšeobecnená funkcia nákladov, kde sú náklady diskontované v rozsahu r , tj. $0 < r < 1$ na jednotku času. Z definície vyplýva, že ak úloha nie je riadne ukončená v čase t , tak vznikajú dodatočné náklady $w_j r e^{-rt} dt$ počas časovej periódy $[t, t+dt]$. Naopak pri dokončení úlohy v perióde $[0, t]$ sú vzniknuté náklady $(1 - e^{-rt})$

Celkový vážený čas ukončenia ($\sum w_j C_j$): v literatúre označovaný aj ako vážený čas plynutia. Podáva nám informáciu o čase ukončenia n úloh s ohľadom na váhy, ktoré definujú náklady na skladovanie a tvorbu zásob spôsobených rozvrhom. Jeho zovšeobením môžeme dostať nákladovú funkciu celkového váženého meškania: $(\sum w_j T_j)$ a následným rozšírením aj funkciu váženého množstva omeškaných úloh $(\sum w_j U_j)$. Na obrázku Obr. 2.5 sú vyobrazené popisované kritériálne funkcie. Funkcie sú súčasne aj regulárne, nakoľko nie je možný ich nárast bez toho, aby sa nepredĺžil termín ukončenia (tj. čas splatnosti (d_j)) aspoň jednej z úloh.

- Oneskorenie úlohy j je zadefinované: $L_j = C_j - d_j$,
- Meškanie úlohy j je zadefinované: $T_j = \max(C_j - d_j, 0) = \max(L_j, 0)$,
- Množstvo omeškaných úloh: $U_j = 1$ ak $C_j > d_j$; 0 za iných podmienok.



Obr. 2.5: Funkčné závislosti nákladových funkcií. [8]

2.3.2 Minimalizácia účelových funkcií

Pri vytváraní výrobného rozvrhu je dôležitá správna voľba kritéria, prípadne viacerých kritérií, na základe ktorých sa hodnotí a následne optimalizuje výrobný rozvrh. Algoritmy rozvrhovania výroby využívajú pri priradovaní úloh k výrobným zdrojom množstvo heuristických pravidiel. Rozhodovanie môže byť realizované napríklad podľa dĺžky

operačného času, podľa termínu dokončenia, podľa počtu zostávajúcich operácií, či podľa ceny. Medzi najpoužívanejšie algoritmy rozvrhovania výrobných zdrojov môžeme zaradiť Johnsonov, Jacsonov, McNaughtonov, Min-max algoritmus, či Palmerovu heuristiku. [16]

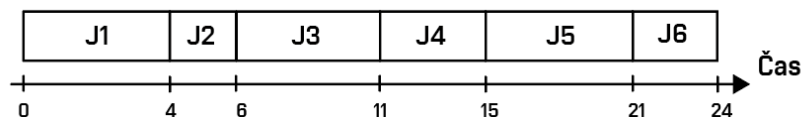
2.3.3 Modely rozvrhovania na jednom stroji

Koncepcia modelu rozvrhovania nad jedným strojom nie je v praxi rozšírená. Ide o veľmi jednoduchý prípad, na rozdiel od ostatných modelovaných prostredí. Avšak ponúka základ pre heuristiky použité v komplikovanejších strojových prostrediach. Pri analýze problémov rozvrhovania je možné komplexnejšie strojové prostredia rozložiť na pod-problémy, pri ktorých využívame znalosť tohto modelu. Existuje niekoľko špeciálnych prípadov, ktoré sa dajú riešiť triviálne (polynomiálny algoritmus). Väčšina týchto úloh však patrí do skupiny nepolynomiálnych úloh.

Je k dispozícii jeden stroj, ktorý spracováva najviac jednu operáciu z množiny n operácií $J = \{J_1, \dots, J_n\}$ Parametre vid'. Tabuľka 2-3.

Tabuľka 2-3: Časy potrebné na spracovanie úlohy a váhy - rozvrh na jednom stroji.

	J ₁	J ₂	J ₃	J ₄	J ₅	J ₆
p _j	4	2	5	4	6	3
w _j	6	8	7	4	2	1



Obr. 2.6: Ganttov diagram pre rozvrh na jednom stroji

Predpokladajme, že na optimalizáciu rozvrhu modelovej situácie použijeme účelovú funkciu celkového váženého času ukončenia úloh. Často býva interpretovaná ako miera rozpracovanosti výroby. Rozvrhovací problém by mal podobu: $1 || \sum w_j C_j \rightarrow \min$.

Algoritmus aplikuje pravidlo známe z teórie rozvrhovania WSPT (najkratší vážený čas spracovania ako prvý). Podľa tohto pravidla sú úlohy usporiadané jednoducho zostupne podľa hodnoty w_j / p_j

Algoritmus: $1 || \sum w_j C_j \rightarrow \min$ [13]

```

ENUMERATE WHILE NOT
w1/p1 ≥ w2/p2 ≥ ... ≥ wn/pn;
C0 := 0;
FOR i := 1 TO n DO
Ci := Ci-1 + pi

```

Výpočtová náročnosť rozvrhu sa začne zvyšovať pri uvažovaní precedenčného obmedzenia medzi jednotlivými zákazkami. Je zrejmé, že aplikovaním predchádzajúceho algoritmu by sme dostali neprípustný rozvrh, ktorý by nereflektoval na obmedzenie vo výrobe. Prirodzený postup by mohol byť postup úplnej enumerácie všetkých prípustných riešení a výber toho s najmenšou hodnotou účelovej funkcie. Tento prístup by však viedol k neupočítateľnému exponenciálne rastúcemu prehľadávaniu $n!$ prípustných permutácií.

Ďalšie možné kritérium pri hľadaní optimálneho rozvrhu je minimalizovať vážený súčet omeškaných úloh. Optimálny rozvrh podľa heuristik vzniká rozdelením úloh do dvoch množín. Pre prvú množinu J platí podmienka pre všetky úlohy j $d_j \leq \sum_{k=1}^n p_k$ tj. obsahuje len úlohy, ktoré môžu byť v optimálnom rozvrhu včas dokončené. Tieto úlohy sú radené v poradí neklesajúcich d_j tzn. $d_1 < d_2 < \dots < d_n$, podľa pravidla EDD (najskorším termínom dokončenia ako prvé). Druhá množina úloh J odkazuje na úlohy, ktoré už boli vylúčené a nesplnia ich čas splatnosti. Tieto oneskorené úlohy už môžu byť zostavené v ľubovoľnom poradí. V nasledujúcej ukážke algoritmu vychádzam z predpokladu, že váhy všetkých zákaziek sú rovnaké, teda vo výpočte ich neuvažujem. Zoradené úlohy podľa času splatnosti (d_j) postupne pridávame do množiny J až do momentu, kedy $C_j > d_j$ potom z množiny včas dokončených zákaziek odstránime tú s najväčšou prácnosťou. Algoritmus s obecnými hodnotami váh je zložitejší a vedie k úplnej enumerácii. [13]

Algoritmus: $1 \parallel (\sum w_j U_j \rightarrow \min)$ [13]

```

ENUMERATE WHILE NOT
 $d_1 \leq d_2 \leq \dots \leq d_n$ ;
2.  $J := \emptyset$ ;  $t := 0$ ;
3. FOR  $i := 1$  to  $n$  DO
BEGIN
4.  $J := J \cup \{i\}$ ;
5.  $t := t + p_i$ ;
6. IF  $t > d_i$  THEN
7. Find job  $j$  in  $J$  with a largest  $p_j$  value;
8.  $J := J \setminus \{j\}$ ;
9.  $J^d := J^d \cup \{j\}$ ;
10.  $t := t - p_j$ 
END

```

V praxi predstavuje toto kritérium mieru výkonu rozvrhu, ktorá sa vyhodnocuje, avšak nemôže byť jediným prostriedkom merania splnenia časov splatnosti. Jeho nedostatkom je, že niektoré úlohy budú musieť čakať na vykonanie neprimerane dlhý čas. Z tohto pohľadu je vhodnejšie optimalizačné kritérium celkového oneskorenia: $1 \parallel \sum w_j T_j$. [13]

2.3.4 Výpočtová náročnosť

Miera úsilia, ktoré treba vynaložiť pri hľadaní optimálneho riešenia prirodzene vzrastá s komplexnosťou úlohy. Výpočtový čas daného algoritmu je definovaný ako horná medza

počtu iteračných prechodov algoritmu vzhľadom na veľkosť vstupu. Z hľadiska výpočtovej náročnosti rozlišujeme dve triedy algoritmov P a NP. Trieda P obsahuje všetky rozhodovacie problémy, ktoré sú riešiteľné v polynomiálnom čase. To znamená, že na riešený problém existuje algoritmus, ktorého výpočtový čas je zhora ohraničený polynomiálnou funkciou. Napríklad algoritmus pre minimalizáciu kritériálnej funkcie $1 || \sum w_j C_j$ je polynomiálny. Trieda NP problémov obsahuje také problémy, pre ktoré nebol nájdený algoritmus riešiteľný v polynomiálnom čase. Problém $1 || (\sum w_j U_j \rightarrow \min)$ je problémom triedy NP. Tieto algoritmy sú výpočtovo veľmi náročné, preto je žiadúce sa vyhnúť úplnej enumerácii, a tým znížiť časovú a pamäťovú náročnosť algoritmu. Za týmto účelom sa používajú rôzne stochastické metódy založené na heuristickom prístupe, napríklad popisovaná metóda vetiev a medzí (Branch and Bound). [8]

3. TYPY RIADIACICH ARCHITEKTÚR VÝROBNÝCH SYSTÉMOV

Cieľom tejto kapitoly je teoreticky popísať vývoj riadiacich štruktúr najmä pre pokročilé výrobné systémy a identifikovať kľúčové rozdiely medzi nimi. Ďalej sa uvádzajú základné charakteristiky, výhody, nevýhody a typické určenie jednotlivých výrobných architektúr. Výrobné riadiace architektúry pre automatizované systémy môžeme definovať ako súbor pravidiel vo forme softvérových programov. Tento pojem stelesňuje mnoho rozhodovacích a koordinačných povinností, vrátane plánovania, pridelovania výrobných zdrojov až cez distribúciu rozhodovacích povinností konkrétnym riadiacim komponentom. Následne typ riadiacej architektúry špecifikuje vzájomné vzťahy a väzby medzi jednotlivými časťami architektúry.

3.1 Kritéria pri výbere rozvrhnutia riadiacej architektúry

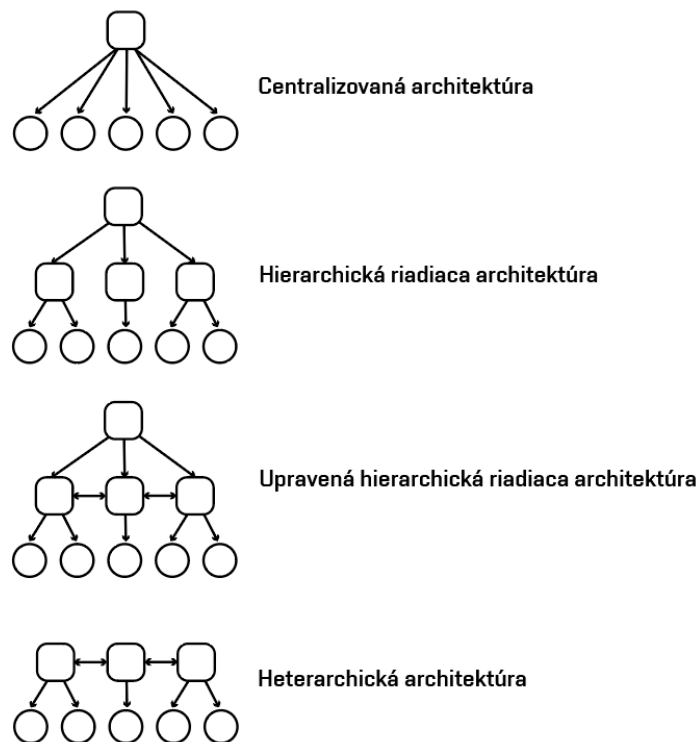
Kritéria na ich výkonnosť sa určujú zo sledovaných parametrov ako napríklad: dlhodobá spoľahlivosť, odolnosti proti poruchám, modifikovateľnosť či dodatočná rozširiteľnosť. Veľký dôraz je v súčasnosti prikladaný najmä na požiadavku flexibility, a tým na vznik pružného výrobného systému (flexible manufacturing system - FMS). Flexibilita v zmysle dynamickej rekonfigurácie výrobného procesu - pridávanie, prispôsobovanie rôznych komponentov výrobných systémov počas prevádzky. Takáto rekonfigurácia môže byť potrebná pri vykonávaní preventívnej údržby, tj. odstávky nejakej časti výroby. Okrem toho by zmeny v konfigurácii výrobnej architektúry mali reflektovať na dynamickú (meniacu sa) kapacitu výroby podľa aktuálneho dopytu. V konečnom dôsledku by takáto schopnosť v prípade výpadku jedného stroja viedla ku korekčnej reakcii celého systému a zachovaní výkonnosti výroby správnym presmerovaním. Je zrejmé, že voľba architektúry riadenia determinuje vo veľkej miere celkovú účinnosť riadiaceho systému. [18]

3.2 Popis špecifických typov riadiacich architektúr

Každá architektúra predstavuje jedinečný prístup k návrhu riadiaceho systému výroby.

Z hľadiska vzájomných väzieb a usporiadania rozlišujeme štyri typy riadiacich architektúr výrobných systémov, vid'. Obr. 3.1.

- Centralizovaná
- Hierarchická
- Hybridná - upravená hierarchická
- Heterarchická



Obr. 3.1 Typy riadiacich architektúr (štvorce sú komponenty riadenia a kruhy sú riadené entity) [19]

3.2.1 Centralizovaná riadiaca architektúra

Historicky najstaršou koncepciou riadiacej architektúry je centralizovaný prístup k riadeniu. Z princípu svojho rozvrhnutia disponuje výhodou, že ide o jednoduchú architektúru s globálnym uskladnením všetkých relevantných údajov z riadenia do centrálnej jednotky. Pokrok v oblasti automatizovanej výroby so stále väčším dôrazom na výkonnosť a flexibilitu ale naráža na množstvo nevýhod centralizovaného riadenia. Rýchlosť reakcie je zvyčajne pomalšia ako v distribuovaných systémoch, predovšetkým ak začne systém naberať na komplexnosti a slúži v reálnom čase viacerým užívateľom. Problematické tiež zostáva robiť softvérové alebo hardvérové zmeny v riadiacej jednotke z dôvodu takmer žiadnej škálovateľnosti systému. Kritickým bodom je aj odolnosť voči poruchám, nakoľko ak centrálna jednotka zlyhá, je znefunkčnený celý výrobný systém. Informačný tok povelov a príkazov prebieha iba jedným smerom z jednej centrálnej riadiacej jednotky do všetkých k nej pridružených výrobných zariadení. Plánovanie a rozvrhovanie výroby sa vykonáva dopredu vždy na určitý časový horizont, preto nedokáže reflektovať na výkyvy vo výrobe. [19]

3.2.2 Hierarchická riadiaca architektúra

Hierarchická architektúra zahŕňa jednu zloženú riadiacu organizačnú štruktúru do niekoľkých menších riadiacich jednotiek. Rozhodovací proces je distribuovaný do viacerých riadiacich úrovní za účelom zníženia výpočtového výkonu centrálnej

riadiacej jednotky. Riadiace rozhodnutia sa vykonávajú zhora nadol, t.j. od vyšších riadiacich jednotiek do nižších so spätnou väzbou, pričom táto funguje zdola-nahor. Na vrchole hierarchie je jeden riadiaci člen, ktorý je zodpovedný za stanovovanie globálnych cieľov a dlhodobých stratégií, koordinuje celú hierarchickú štruktúru k jednotnému postupu pri dosahovaní cieľa.

Osobitným prípadom hierarchickej architektúry s jednou úrovňou riadenia je centralizovaná architektúra. Vyššie úrovne hierarchickej architektúry budú klásť rôzne nároky na výpočtovú techniku, od výkonných počítačov na najvyššej úrovni až po mikropočítače na najnižšej úrovni. Podobne ako v centralizovanej architektúre sa podrobnosť informácií zvyšuje s každou vyššou úrovňou riadenia, zatiaľ čo časový interval na ich zohľadnenie sa znižuje. Čím vyššie sa v hierarchii nachádzame, tým abstraktnejšie príkazy systém spracováva.

Prínosom hierarchickej architektúry je obmedzenie veľkosti a zložitosti jednotlivých riadiacich modulov, čo vyplýva z hierarchického princípu distribuovania rozhodovacej inteligencie. Rozdelenie úloh do jednotlivých úrovní prináša lepšiu časovú odozvu a priblíženie sa spracovávaniu v reálnom čase. Takisto sa na každej úrovni poskytujú aj informácie z podriadených modulov smerom k nadriadenému modulu. To umožňuje uzavretie riadiacej slučky, čo je nutným predpokladom pre adaptívne správanie systému. Hierarchicky štruktúrované systémy majú sklon byť rigidné a obmedzené svojim formalizmom z dôvodu vopred určených postupov riadenia a vzájomnej závislosti modulov na sebe. Štruktúra týchto systémov sa stáva pevnou už v počiatočných implementáciách takéhoto systému riadenia. Preto sa nedosahuje takej flexibility pri už navrhnutých systémoch, a to aj napriek vrodenej modularite. Konceptia nadradených a podradených úrovní predpokladá značné znalosti o úrovni pod a nad, čo pri rozširovaní systému býva problém. Riadené moduly nedisponujú takmer žiadnou autonómiou, preto nie sú schopné reagovať na neurčitosti prichádzajúce z vonkajšieho prostredia. Tolerancia chýb je vyššia ako pri centralizovanej architektúre (výpadok podriadenej jednotky nutne neznamená výpadok celého systému). Stále ale platí, že výpadok centrálny jednotky zasiahne vždy celý systém riadenia. [19]

3.2.3 Upravená hierarchická riadiaca architektúra

Dostupnosť výpočtového výkonu a technológie lokálnych sietí (LAN) umožnila vznik modifikovanej riadiacej architektúry. Už z názvu je zrejmé, že upravená hierarchická forma architektúry má veľa znakov spoločných s predchádzajúcou architektúrou, pričom eliminuje jej viaceré nedostatky a obmedzenia. Disponuje horizontálnym tokom informácií medzi modulmi, čo umožňuje rozšírenie autonómneho rozhodovania aj do nižších úrovní architektúry. Zvýšená miera samostatnosti podriadených jednotiek vyžaduje viac interakcií s okolím a viac informácií, viac informácií prináša viac inteligencie do podradených modulov.

Relatívne decentralizovaný prístup takto modifikovanej hierarchickej architektúry

priniesol viac pohotovú reakciu na miestne podmienky, viac robustnosti pri vzniku náhodných porúch a neurčitostí. Výsledkom je zvýšenie efektivity, agilnosti, modularity či vyššej miery odolnosti proti poruchám. Avšak zlepšenie je za cenu implementačne zložitejšieho systému. [19]

3.2.4 Heterarchická architektúra

Táto forma riadenia automatizovaných systémov prekonáva nevýhody, ktoré obmedzujú predchádzajúce tri formy riadenia. Úplne eliminuje koncepciu delenia na nadriadené a podriadené entity, do popredia sa dostáva kooperatívny prístup a snaha o úplnú lokálnu autonómiu entít. Táto forma architektúry je úplne decentralizovaná, tvorená je nezávislými entitami, ktoré majú svoj vlastný riadiaci algoritmus a dodržia určité pravidlá. Dekompozícia systému na súbor na sebe nezávislých entít je vykonaná prirodzene podľa obsahovej a funkčnej príslušnosti.

Rozhodnutia sa vykonávajú lokálne v mieste zberu informácií. V systéme sa preto minimalizujú alebo úplne vylúčia globálne databázy a pracuje sa len s lokálnymi databázami, ktorými disponuje každý modul architektúry.

Spolupráca medzi subjektmi sa zabezpečuje prostredníctvom vyjednávacieho postupu. Výrobné bunky navzájom rokujú o usporiadaní adaptívneho plánovania vzhľadom na vonkajšie podnety, zmeny vo výrobe či poruchy. Medzi hlavné výhody takéhoto usporiadania patrí vysoká odolnosť voči poruchám, flexibilita, vyššia miera prípadnej redundancie. Implicitná odolnosť voči chybám sa dosahuje vďaka lokálnej autonómii komponentov. Zlyhanie niektorého z nich nevedie k zlyhaniu celého systému, ale k rekonfigurácii zvyšných výrobných kapacít. Medzi nedostatky heterarchickej architektúry patrí riziko nedeterministického a nepredvídateľného správania systému ako celku, nakoľko absencia nadradenej komponenty neumožňuje globálnu optimalizáciu. Niektoré problémy tohto prístupu k riadeniu súvisia aj s heterogenitou použitých komponentov a ich vzájomnou nekompatibilitou. Problémom tiež ostáva nedostatočné vymedzenie postupov a noriem na koordináciu vývoja distribuovaných konceptov riadenia. [20]

Pri syntéze heterarchických výrobných systémov je veľmi dôležitá kombinácia dôkladnej funkčnej analýzy v smere zhora nadol a usporiadanie toku implementačných rozhodnutí v smere zdola nahor. Dôraz je kladený na vytvorenie kooperatívnej heterarchie (založenej na vyjednávaní), ktorá spĺňa pravidlá:

- subjekty majú rovnaké práva na prístup k zdrojom,
- subjekty majú vzájomne nezávislé spôsoby fungovania,
- entity majú rovnaký vzájomný prístup a dostupnosť k sebe navzájom,
- subjekty prísne dodržiavajú pravidlá celkového systému.

Kľúčové pre každú entitu v systéme je konať v súlade s maximalizovaním výkonnosti celého systému a zabezpečovať jeho optimálnu prevádzku a prežitie aj v havarijných stavoch. [21]

V tejto práci sa budem ďalej zaoberať návrhom takéhoto heterarchického distribuovaného systému riadenia, prioritne založenom na agentovom prístupe.

4. INTELIGENTNÉ VÝROBNÉ SYSTÉMY

Definícia inteligentného výrobného systému (Intelligent Manufacturing System - IMS) nie je naprieč literárnymi zdrojmi jednotná. Za inteligentný systém sa dá považovať systém s vlastnosťou, pomocou ktorej je schopný pochopiť príčiny zmien na svojom vstupe a tieto informácie využiť na učenie, ktoré sa opakovaním prehĺbuje. Možné zmeny na vstupoch systému sa nedajú exaktne vyšpecifikovať ani predpokladať, preto si inteligentný systém kladie za cieľ napodobňovať schémy ľudského správania a pri vyhodnocovaní podnetov uplatňovať najmä tvorivosť. Na inteligentný výrobný systém možno nazerať aj z pohľadu umelej inteligencie ako vednej disciplíny. Tá v sebe zahŕňa množstvo vedných odborov, preto sa definícia toho, čo je považované za inteligentné správanie, dá nájsť v rôznej podobe. Vo všeobecnosti sa považuje systém za inteligentný jednoducho vtedy, keď by sme jeho konanie označili inteligentným, ako keby sa jednalo o človeka. [18]

Za inteligentný výrobný systém budeme považovať systém, ktorý pozostáva z distribuovaných entít. Distribuovanie komplexného problému medzi viaceré časti inteligentného systému prináša viaceré inováčne využitia takto navrhnutého systému. Náročné problémy materiálového, informačného alebo riadiaceho charakteru rozdeľuje na menšie podľa príslušnosti k niektorej z logicky vyplývajúcich tried (manipulácia, doprava, skladovanie, objednávanie a zabezpečovanie vstupných komodít do výroby).

4.1 Vlastnosti inteligentných výrobných systémov

Vlastnosti, ktoré determinujú správanie IMS reflektujú na hlavnú myšlienku - ako zvýšiť výkonnosť výrobných systémov a ich schopnosť adaptácie na rýchlo sa meniace požiadavky vstupujúce do výrobného procesu od zákazníkov. Nemenej dôležité sú požiadavky na zvyšovanie produktivity a znižovanie nákladov. Aspekt "Time-to-Market" väčšiny výrobných spoločností vyžaduje veľmi rýchle uvedenie výrobku na trh a v prípade úspechu veľmi rýchle zvýšenie výroby. Navyše výrobky sa stávajú zložitejšími, s vyššou mierou individuálnej konfigurácie, čo predstavuje zjavnú výzvu pre tradičné riadenie výroby. Rýchlosť reakcie je kritický parameter aj pri odolnosti voči poruchám, kedy by rôzne abnormálne stavy v systéme mali byť riešené v reálnom čase. Dizajn IMS by mal zvládať aj neurčitosti a nekompletné informácie, plynúce z reálnej výroby. Ďalšie vlastnosti vychádzajú najmä z popisu heterarchickej architektúry riadiaceho systému, na ktorom je koncept IMS založený.

Za zmienku stojí jedna z paradigiem IMS pre výrobný podnik, ktorá podporuje nutnosť implementácie hierarchických štruktúr pre zaručenie riešenia konfliktov medzi jednotlivými autonómnymi entitami. Cieľom IMS je spojiť stabilitu a dynamickú flexibilitu heterarchických systémov. Výsledkom je hybridná štruktúra, inšpirovaná napríklad fungovaním živých organizmov, alebo mechanizmami fungovania ľudskej

spoločnosti. Inšpiráciu v prírode možno nájsť pri vyhľadávaní potravy pomocou prístupu založeného na feromónoch. Hmyz, najmä kolónie mravcov sú prototypom takýchto organizácií. V tomto prípade každý jeden mravec sa riadi len veľmi jednoduchým algoritmom, ale kolónia ako celok je schopná riešiť a prispôbiť sa širokej škále okolností. Napríklad v kolónii je dôležité, aby mravce našli blízke zdroje potravy a vytýčili krátke cesty medzi hniezdom a potravou. Kolónia je schopná vybudovať minimálnu štruktúru ciest, ktoré spájajú hniezdo s potravou. Kľúčovou myšlienkou, ktorá je tu zachytená, je koncept kolektívnej inteligencie. Ukazuje sa, že systém začne vykazovať kolektívnu inteligenciu pri riešení problému väčšieho rozsahu, ktorý zjavne presahuje rámec agregácie individuálnej inteligencie. Žiaden mravec nedrží "rozvrh" na dodávanie potravy, neexistuje centrálny plánovač. To je hlavný rozdiel oproti tradičnému pohľadu, riešenie sa už nevníma ako výsledok, ale ako proces. Vo výrobe by sa tento pokrokový prístup prejavil väčšou stabilitou voči poruchám, lepšou adaptabilitou a flexibilitou voči zmenám či efektívnym využitiam dostupných zdrojov. [22]

4.2 Inteligentné výrobné systémy založené na agentovom prístupe

Na vyššie popisované požiadavky inteligentných výrobných systémov (IMS) reflektujú multiagentné systémy a heterarchické systémy založené na holonickom princípe. Práve heterarchická architektúra umožňuje plne decentralizovať rozhodovacie a plánovacie kompetencie. Teória multiagentných systémov svojou koncepciou spadá do viacerých vedných disciplín, medzi nimi aj do oblasti výskumu umelej inteligencie. Zameriava sa prioritne na systémy zložené z viacerých na sebe nezávislých prvkov v prostredí, ktoré disponujú distribuovanou inteligenciou a sú vo vzťahu kooperatívnej heterarchie. Tento koncept predstavuje moderný spôsob spolupráce samostatných entít v priemysle 4.0. Agentový prístup v oblasti riadenia a plánovania výroby, logistiky vnútorných a vonkajších potrieb podniku prestáva byť doménou len výskumných úloh univerzít či výskumných pracovísk, ale stáva sa nástrojom zvýšenia konkurencieschopnosti podnikov. Výrobné systémy založené na multiagentnom princípe pozostávajú z virtuálnych entít - agentov. Agent ako autonómny výpočtový systém napĺňa predefinované ciele a svojim správaním zvyšuje výkonnosť systému v závislosti od svojich zdrojov, schopností, vnímania a komunikácie. Pre úspešné napĺňanie cieľov agenti využívajú princípy kooperácie, vzájomnej koordinácie a vyjednávacích mechanizmov. Takáto entita v multiagentnom systéme má nasledujúce vlastnosti.

- Autonómny - agent vykonáva svoju činnosť bez zásahu človeka tj. samostatne,
- Reaktívny - agenti vnímajú prostredie a včasne reagujú na prípadné zmeny,
- Proaktívny - agent je schopný prevziať iniciatívu a cielene upraviť svoje reakcie,

- Sociálny - medzi agentami dochádza k interakciám prostredníctvom vhodného komunikačného jazyka, ontológií a interakčných protokolov.

Individuálne správanie agentov je determinované množstvom aspektov, od samotnej vnútornej štruktúry agenta až po architektúru prostredia, v ktorom je agent nasadený. [23] Klasifikácia agentov z pohľadu zložitosti a miery inteligencie (racionality), ktorou disponujú:

- **Reaktívny agent** - (alternatívny prístup - nemysli, konaj). Principiálne ide o agenta s najjednoduchšou vnútornou architektúrou. Vyznačuje sa tým, že nemá kapacitu modelovať obraz o svojom prostredí, tj. nemá modul vnútornej reprezentácie. Následkom toho nie je schopný ani predvídať, akú akciu vykoná. Agent iba bezprostredne reaguje na podnety vychádzajúce z prostredia, v ktorom operuje. Súčasťou reaktívneho agenta sú kompetenčné moduly, ktorých podoba je väčšinou vo forme funkcie alebo procedúry obsahujúcej invokačnú podmienku a dôsledkovú časť (špecifikácia správania agenta po prijatí príslušného stimulu z prostredia). Delíme ich na agentov s pamäťou a bez pamäte. Agent s pamäťou si pamätá stav, do ktorého ho prostredie v minulosti dostalo a ak sa situácia opakuje zachová sa rovnako. Naopak agent bez pamäte nemá žiadne vnútorné stavy, ktoré by zastupovali pamäť, preto vždy vykoná prvú možnú akciu, ktorú môže v definovanej situácii vykonať.
- **Deliberatívny agent** - (klasický prístup - najprv myslí, potom konaj). Udržiava obraz o okolitom prostredí, dokáže cielene vytvárať plán svojich reakcií, pričom udržiava reprezentáciu svojich vnútorných vzťahov. Oproti reaktívnemu agentovi disponuje modulom vnútornej reprezentácie, je schopný deduktívneho myslenia, pomocou ktorého vyberá postupnosť akcií vedúcich k cieľu. Vlastnosti tohto typu agenta ho predurčujú do prostredí, kde je deterministicky špecifikované riešenie daného problému. Podmnožinou deliberatívneho agenta sú aj sociálni agenti. Tí rozširujú popisované vlastnosti o možnosť s okolitými agentami kooperovať v zmysle optimalizovania svojich činností v prospech okolitých agentov a tým aj k dosiahnutiu spoločného cieľa systému.
- **Hybridný agent** - (alternatívny prístup - myslí a konaj nezávisle). Vznikol ako kombinácia vlastností reaktívneho a deliberatívneho agenta. Takýto agent operatívne prispôsobuje svoje vlastnosti a dokáže sa správať buď iba ako jednoduchý reaktívny agent, pričom ak to situácia vyžaduje, dokáže vybudovať vlastný plán. [23]

4.2.1 Komunikácia, ontológia, interakčné protokoly

Jednotlivé entity multiagentných štruktúr majú obmedzené informácie a kompetencie vo výrobnom procese, preto je nevyhnutné zdieľanie výsledkov individuálneho úsilia naprieč systémom k dosahovaniu spoločných cieľov. Preto je nutná komunikácia medzi

agentami, ktorej cieľom je prepojenie každého agenta s globálnym fungovaním systému. Kooperatívne usporiadanie v multiagentných systémoch a s tým spojené dosiahnutie vyššieho stupňa racionality so sebou prináša veľa implementačných nárokov. Interakčné mechanizmy musia implementovať špecifické vlastnosti rolí, väzieb a vzájomných závislostí medzi agentami, čo si vyžaduje štandardizáciu. [23]

Kooperatívne mechanizmy medzi agentami vyžadujú komunikačné protokoly. V prostrediach s rôznymi platformami a heterogénnymi agentami je kladený veľký dôraz na štandardizovaný rámec komunikácie s cieľom zaručiť syntaktickú interoperabilitu medzi rôznymi MAS. Štandard FIPA ACL alebo KQML je manipulačný jazyk založený na komunikačných aktoch. To znamená, že KQML nedefinuje formu samotného významu správy. Definuje len od koho, komu a kedy bola správa doručená a v akej forme odosielateľ očakáva odpoveď. Implementácia tohto jazyka má zadané dve úrovne. Prvou je úroveň rozlíšenia rečového aktu a druhou je úroveň samotného komunikovaného obsahu. Jazyk obsahu správy môže byť akýkoľvek, ktorý je adekvátny pre vyjadrenie obsahu a rozumejú mu obaja agenti. [24]

Komunikáciu zastrešuje prenosová vrstva založená väčšinou na protokole TCP. Pri použití protokolu UDP je potrebné kontrolovať chyby, poradie paketov atď., čo zvyšuje nároky na zdroje aplikácie najmä procesor a pamäť. Na ilustráciu je uvedený príklad komunikačného aktu žiadosti agenta *B* o informácie od agenta *A* v jazyku KQML vid'. Obr. 4.1. Agent *B* môže byť v kontexte výrobného podniku agent, ktorý prostredníctvom vytvoreného kanála periodicky vyjednáva svoju výrobu s ostatnými agentami. Agent *A* v pozícii agenta výrobného zdroja odpovedá na požiadavku agenta *B* poskytnutím referencie na svoju rezervačnú frontu, na základe ktorej plánuje svoju činnosť.

```
(subscribe
  :sender B
  :receiver A
  :timestamp 1113340454
  :reply-with query_1
  :language KQML
  :ontology Manufacturing system
  :content (ask if
    :sender B
    :receiver A
    :in-reply-to query_1
    :language Prolog
    :ontology Manufacturing_system
    :content reservation request(time)
            state(in progress)
  )
)

(tell
  :sender A
  :receiver B
  :timestamp 1113341454
```

```

:reply-with query_1
:language Prolog
:ontology Manufacturing system
:content reservation queue(array[X]);
        state(in operation) ;
)

```

Obr. 4.1: Príklad správ (otázok a odpovedí) v jazyku KQML [24]

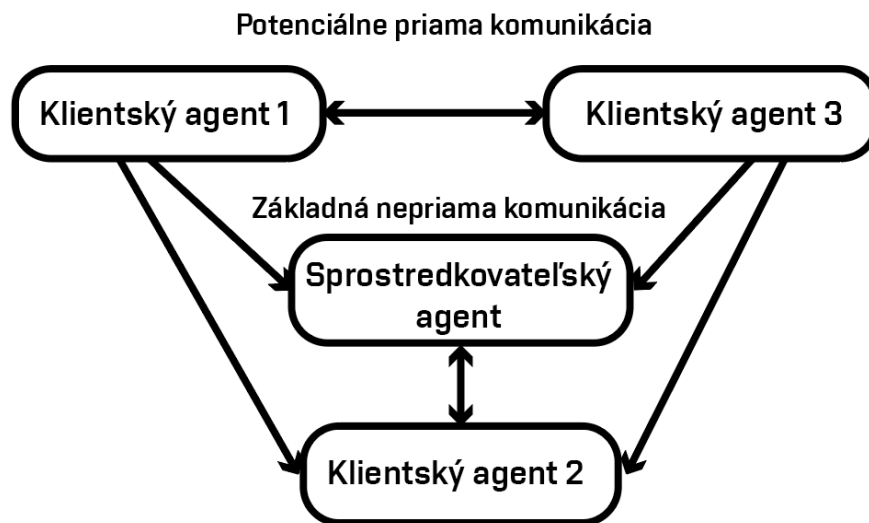
Každú správu tvoria dvojice parametrov a hodnôt. Pre každý typ rečového aktu existuje *performatív* (tučne zvýraznené príkazy na Obr. 4.1). Performatív definuje, či je obsah tvrdenie, dotaz, príkaz alebo akýkoľvek iný vzájomne dohodnutý rečový akt. Opisuje tiež formu, akou by mala byť odpoveď doručená. KQML implementačne podporuje služby umožňujúce agentom sa navzájom lokalizovať v distribuovanom prostredí prostredníctvom nepovinných vlastností v správach, akými sú jedinečné identifikátory, mená a adresy všetkých zúčastnených agentov. [25]

Výhody komunikačných jazykov KQML pre multiagentné štruktúry:

- Nezávislosť komunikačného jazyka od jazyka obsahu správy,
- Oddelenie obsahovej vrstvy od vrstvy komunikačného aktu,
- Deklaratívna forma.

Absencia priameho počítačového spracovania prekladača pre jazyk KQML vedie k snahe spojiť výhody jazyka KQML s jazykom XML, ktorý je zaužívaným štandardom pre komunikáciu medzi počítačmi. Celú správu by tak interpretoval len XML parser. Použitie sémantiky a ontológií v komunikovaných správach je špecifické pre rôzne domény výrobných systémov. Existujúce normy v priemyselnom odvetví ISA-95 a ISA-88 by mali reflektovať na túto potrebu vytvorením príslušného modelu. [23] Keďže prieskum definícií a opis agentových protokolov by viedol k hĺbkovej diskusii, ktorá nie je potrebná na všeobecný opis, v práci sa im nebudem viac venovať.

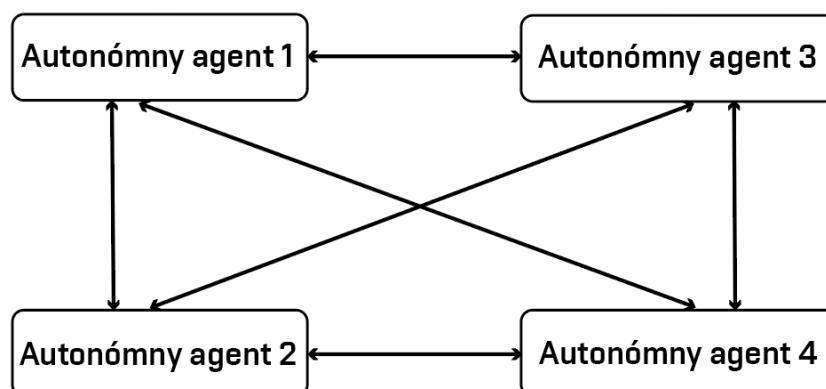
Samotnú komunikáciu môžeme rozdeliť z pohľadu centralizácie hľadania požadovaných agentov na priamu a nepriamu. Ak agenti spolu nekomunikujú priamo, ale je v systéme zastúpený sprostredkovateľský agent, ktorý vybavuje ich požiadavky, hovoríme o nepriamej komunikácii. Môžeme to označiť ako federované MAS, teda predstupeň decentralizovaných MAS. Klientski agenti tak nemusia poznať adresy a poskytované služby ostatných agentov, stačí im adresa mediátorského agenta. Ten má k dispozícii databázu údajov, adries a schopností, pričom sa stará aj o pravidelnú aktualizáciu týchto dát. V systéme vystupuje ako komunikačný most, čo umožňuje implementačnú voľnosť a zjednodušenie vzťahov medzi ostatnými agentami. Zlyhanie sprostredkovateľských agentov ohrozuje celú aplikáciu, preto je nutné dodržať funkčnú redundanciu týchto agentov. Nepriama komunikácia redukuje záťaž na komunikačné nároky v systéme.



Obr. 4.2: Architektúra federovaného multiagentného systému. [23]

4.2.2 Decentralizované multiagentné systémy

Decentralizované (autonómne) MAS sú bez centrálného agenta a bez agenta sprostredkovateľa, agenti komunikujú len na základe svojich sociálnych poznatkov. Sú tvorené autonómnyimi agentami, ktorí využívajú priamu komunikáciu naprieč systémom. Vlastnosti decentralizovaného MAS determinuje povaha interakcií medzi agentami - statické a dynamické. Statické charakterizuje vyššia miera centralizácie, dynamické decentralizované MAS sú najrobustnejšie využívajúce koordináciu aktivít v podobe negociačných protokolov. Z princípu vysokej miery autonómnosti agentov vyplývajú vyššie nároky na komunikačné prostriedky. Pre systém s N agentami musí existovať až $N(N-1)/2$ point-to-point komunikačných kanálov. Réžia takejto komunikácie je zložitá aj pre nutnosť vyhľadávania agentov, nakoľko agenti nemajú informáciu o ostatných entitách v systéme. Potenciálnym rizikom tejto koncepcie usporiadania agentov je aj možné nepredvídateľné správanie systému vid'. kapitola 4.3.



Obr. 4.3: Architektúra decentralizovaného multiagentného systému. [23]

4.3 Potenciál a využiteľnosť MAS v priemyselnom prostredí.

Koncepcia technológie MAS je implementačne relatívne mladá, preto sa štruktúry založené na agentovom princípe zvyčajne spájajú s nedostatočnou, či úplne chýbajúcou metodikou vývoja, a teda aj slabým prijatím v priemysle. V súčasnosti najviac rozšírené centralizované riadenie výroby spolieha na MES a MOM systémy, do ktorých sa musia prenášať všetky relevantné údaje. Takýto prenos je štruktúrovaný vertikálne, podliehajúc hierarchickým princípom. Paradigma MAS nahrádza centralizované riadenie distribuovaným, pri ktorom interakcie medzi jednotlivými entitami (agentami) vedú k vzniku "inteligentného" globálneho správania. Interakcie sú integrované horizontálne aj vertikálne. To umožňuje dosiahnuť vysoký stupeň autonómie a spolupráce bez pevnej rigidnej štruktúry a následne poskytnúť modularitu, flexibilitu, robustnosť a agilitu. Multiagentný prístup má vysoký potenciál pridanej hodnoty najmä v komplexne prepojených systémoch s vysokou mierou zložitosti. Tradičný prístup riadenia takýchto systémov ako centralizovaných (monolitických), alebo hierarchických, je nielen neefektívny, ale častokrát je ich centralizácia prakticky nemožná z dôvodu obmedzenia množstva zdrojov jediného centrálného agenta. Z tohto pohľadu je prirodzená snaha o dekompozíciu do viacerých funkčných jednotiek - agentov. Vo výrobnom systéme to umožňuje efektívnejšie využiť výpočtový výkon a tým skrátiť čas splnenia úlohy, čo súčasne umožňuje zvýšiť kvalitu výsledného riešenia. [18]

Na teoretickej úrovni existujú minimálne tri kritériá, potrebné pre naplnenie predpokladov použiť agentovo orientovanú architektúru v priemyselnom prostredí. Aplikácie (nielen priemyselného charakteru), ktoré nespĺňajú všetky tri kritériá, by sa nemali riešiť pomocou agentovej technológie.

- vykazovať prirodzenú distributivitu, napr. autonómne subjekty, geografické rozdelenie, distribuovaná inteligencia,
- mať potrebu flexibilnej interakcie, napr. neexistuje apriórne priradenie úloh aktérom, tj. žiadne pevne stanovené procesy,
- technológia zasadená do dynamického prostredia s premennými vlastnosťami vyžadujúcimi bezprostrednú intervenciu systému, často vyžaduje schopnosť on-line rozhodovania, ktoré vychádza z protokolov vyjednávania/spolupráce.

Výrobné systémy, ktoré sú veľmi dynamické pri vytváraní vzťahov so zákazníkmi a dodávateľmi môžu nájsť účelné riešenia riadenia a rozvrhovania práve využitím MAS. Takéto vhodné nasadenie MAS je v podniku s viacerými závodmi, ktorý vyrába výrobky na objednávku stratégiou make-to-order pričom zložitosť výrobku vyžaduje spoluprácu viacerých odvetví. Naopak neefektívne nasadenie MAS by bolo vo výrobnom podniku tvorenom jediným závodom s malým konsolidovaným počtom dodávateľov, ktorý sa vyznačuje pomerne stabilnou výrobou niekoľkých typov jednoduchých výrobkov využívajúci stratégiu make-to-stock. Ak kontext, v ktorom sa vykonávajú rozhodovacie činnosti, umožňuje použiť uspokojivý off-line, centralizovaný a optimálny prístup, prechod na MAS nie je účelná.

Hlavným záujmom ostáva otázka vývoja usmernení a formálnych prístupov zameraných na metodológiu agentového inžinierstva, ktoré poskytujú technologickú základňu potrebnú na transformáciu tradičného priemyselného vybavenia na entity inteligentných výrobných systémov s čo najjednoduchšou integráciou. [26]

Dôležité je ale podotknúť, že decentralizovaný prístup k riadeniu výroby prináša aj nevýhody. Jedným z hlavných nedostatkov tohto konceptu, ktorý bráni plnému využitiu v priemyselnom prostredí, je nedeterministický charakter a nepredvídateľnosť výsledku rozhodovacích procesov. Predikcia je prakticky nemožná z dôvodu, že budúce stavy sú vždy výsledkom vzájomnej interakcie agentov, čo môže vyústiť do nežiadúceho až nebezpečného správania systému, sa odborné nazýva emergencia. Je to jav, kedy neplatí princíp superpozície, tj. výsledné vlastnosti celku nie sú súčtom vlastností jeho hierarchicky nižších častí. Pri kooperácii autonómnych agentov v heterarchickej štruktúre môžu vzniknúť vlastnosti, ktoré nie sú definované ani v jednom z nich. Odpoveďou na uvedené je rozšíriť agentový prístup o využitie holonického konceptu, v ktorom sú niektoré entity časťami väčších celkov, pri dodržaní hierarchických princípov. Každý holón môže byť súčasťou iných holónov a tým vytvárať holonický výrobný systém. Architektúra sa tak svojou organizáciou približuje už popisovanej upravenej hierarchickej štruktúre (semi-heterarchickej štruktúre). [27] Pre zvýšenie potenciálu využitia MAS v organizácii výroby je nutné výskumnú činnosť zamerať na najslubnejšie oblasti: [28]

- Mechanizmy a protokoly vyjednávania - cieľom výskumu sú sofistikovanejšie vyjednávacie mechanizmy a protokoly [29] bližšie v kapitole 0.
- Integrácia agentových a tradičných prístupov - v tejto súvislosti sa naznačuje niekoľko možností výskumu, ktoré vychádzajú z kombinácií vyžadujúcich predbežné (off-line) a reaktívne on-line plánovanie.
- Teoretické skúmanie metodiky - najmä konsolidovať súčasné výsledky výskumu a tým uľahčiť implementáciu reálnych priemyselných aplikácií. Toto skúmanie zahŕňa mnoho aspektov, zamerať by sa malo na referenčné rámce semi-heterarchickej architektúry.
- Hodnotenie a porovnávanie výkonnosti konkrétnych implementácií MAS - neexistuje žiadna štandardná metodika založená na spoločnom porovnaní výkonnosti s dostatočnou mierou komplexnosti.

V nadväznosti na predchádzajúce skutočnosti boli zadané scenáre, v ktorých môžu MAS najlepšie preukázať svoju hodnotu. MAS poskytujú infraštruktúru umožňujúcu distribuovaným subjektom vymieňať si informácie a vzájomne spolupracovať.

4.4 Súvisiace technológie prístupu Industry 4.0

Predstavené architektúry inteligentných výrobných systémov so sebou prinášajú násobne vyššie nároky na konektivitu a autonómnosť distribuovaných entít - agentov.

Technológie a prístupy priemyslu 4.0, ktoré sú v tomto kontexte relevantné sú cloud-computing, internet vecí (IoT), Big-Data a prepojenia RFID. [30]

Na IMS sa dá pozerat' aj optikou cloudovej platformy, ktorá poskytuje cloud-computing nad agentami reálnych zariadení v továrni (PLC, aktuátory, snímače), čo umožňuje virtualizáciu a dynamické škálovanie takýchto zdrojov.

Internet vecí (IoT) zahŕňa rôzne technológie, ktoré umožňujú integráciu relevantných objektov do on-line sveta. Takáto prepojenosť poskytuje sledovanie a zhromažďovanie dát v reálnom čase. Na veľké objemy takto získaných údajov sa budú vzťahovať techniky spracovania rozsiahlych a nehomogénnych databáz - Big-Data.

Rádiofrekvenčná identifikácia (RFID) je technológia, ktorá poskytuje spôsob výmeny relevantných informácií o základnej identite medzi entitami v agentových štruktúrach aj na veľké vzdialenosti. Môže prebiehať pomocou kariet alebo transpondérov RFID.

Nutné tiež spomenúť digitálne dvojča ako ďalší koncept priemyslu 4.0, ktorý je spojený s integráciou CPPS. Digitálne dvojča môže vytvárať detailné virtuálne modely fyzických objektov vo virtuálnom priestore s cieľom simulovať ich správanie v reálnom svete a poskytovať spätnú väzbu už vo fáze vývoja systému, prípadne môže slúžiť ako referencia pre odhaľovanie chýb počas prevádzky. Cieľom je takto vytvoriť inteligentné továrne, ktoré efektívne reflektujú na požiadavky mnohostranného riadenia výroby. [31]

4.4.1 Architektúra orientovaná na služby v zmysle normy IEC 61499

Jednou z integrálnych súčastí modelovania multi-agentových štruktúr v priemysle je SOA (architektúra orientovaná na služby). Koncepcia SOA vznikla s cieľom uľahčiť vytváranie distribuovaných systémov. Definuje prístup výrobných riadiacich algoritmov k dynamickej rekonfigurácii založenej na norme IEC 61499 s akcentom na rast a interoperabilitu. Architektúra SOA má pri správnej implementácii predpoklad zjednocovať technológiu na všetkých podnikových informačných úrovniach od senzorov a aktuátorov až po úroveň ERP. Takáto architektúra je jedným z hlavných pilierov priemyslu 4.0. Rozhranie medzi fyzickým svetom a modelom riadiacich algoritmov zabezpečuje modul Asset Administration Shell (AAS). Každý autonómny agent decentralizovanej výroby napríklad produkt, stroj, robot, dopravník, linka alebo regál v sklade je vybavený týmto modulom, ktorý zastrešuje množstvo funkcií. Zabezpečuje réžiu identifikácie a komunikácie, monitorovanie stavu či životného cyklu, alebo konfiguráciu vlastností agenta. V architektúre SOA je modul zodpovedný aj za inzerovanie svojich služieb a rovnako tak využívanie služieb ostatných agentov v sieti. Na administráciu špecifikácií a riadenie vyjednávania takýchto služieb sa používajú webové a cloudové riešenia. Štandardizácia rozhrania modulu AAS nad každým agentom by priniesla vysokú škálovateľnosť systému najmä pri začleňovaní nového agenta - plug & produce (P&P). [31]

4.5 Referenčná architektúra CPPS

Architektúry inteligentných výrobných systémov založených na multiagentnom alebo holonickom prístupe vyžadujú formalizovať základné typy agentov, ktoré reprezentujú entity vo výrobnej oblasti. Jednou z nich môže byť architektúra PROSA. Definuje štyri hlavné holóny, od ktorých je odvodený aj jej názov.

Sú to: Product (výrobok), Resource (zdroj) Order (objednávka) a Staff (personál). Ďalšou veľmi rozšírenou architektúrou je ADACOR. Vychádza z konceptu PROSA, pričom definuje dohliadací holón (supervisor), ktorý je v roli koordinátora rozvrhovania a plánovania operácií. [18]

Koncepty v oblasti automatizácie zastrešuje názov kyberneticko-fyzikálne výrobné systémy (CPPS). CPPS je inovatívnou a do budúcnosti sľubnou paradigmou riadenia výroby. Architektúra inteligentnej výroby CPPS je definovaná kombináciou vstavaného systému, t.j. fyzickej časti vybavenej prvkami procesnej automatizácie, a kybernetickej časti zastupujúcou softvérovú výbavu agenta. Tento koncept zastupuje možnosť, ako by sa organizácia riadenia výroby mohla zmeniť z tradičnej, statickej, hierarchickej, uzavretej organizácie na dynamickú, otvorenú, samo-organizujúcu sa organizáciu. Hlavnou výhodou takéhoto prístupu je vyššie uvádzaná vlastnosť automatizovanej rekonfigurácie, ak je plánovaná výroba nejakým spôsobom zmenená. V prípade takejto abnormality vo výrobe, či už pozitívnej (pridanie zdrojov, navýšenie výkonu stávajúcich zdrojov), alebo negatívnej v podobe výpadku niektorého z agentov, negociačné mechanizmy iniciujú procedúru vyjednávania s inými strojmi alebo výrobkami, preto je systém principiálne odolný voči zmenám a poruchám. [32]

4.5.1 Topológia systému CPPS

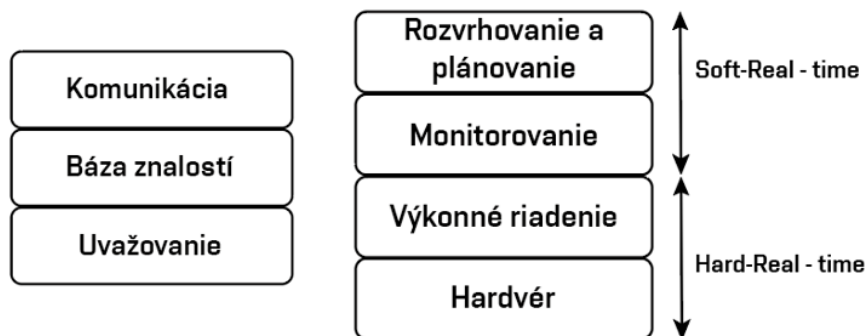
Návrh samotných rolí, kompetencií a vlastností agentov vystupujúcich v CPPS je jedným z prvých a najdôležitejších krokov návrhu výrobného systému.

Topológia agentov na makroúrovni v CPPS pozostáva z dvoch najdôležitejších typov agentov - Resource agent a Product agent. V závislosti na implementácii môžu byť v konkrétnom návrhu CPPS vyšpecifikované aj ďalšie vedľajšie typy agentov.

Na mikroúrovni konkrétneho agenta sa topológia pri vysokej miere abstrakcie a zjednodušenia skladá z komunikačného modulu, znalostnej bázy a uvažovacieho (argumentačného) modulu. Špecifikácia komunikačného modulu by mala definovať komunikačný kanál a príslušnú ontológiu, viď kapitola 4.2.1. Keďže agent je softvérová aplikácia, mala by jasne odkazovať na architektúru počítačovej siete ISO/OSI, ktorá špecifikuje fyzické a softvérové vlastnosti spojenia. Špecifikácia bázy znalostí zapuzdruje know-how agenta v určitej oblasti. V kontexte výrobného podniku zahŕňa informácie v poli pôsobnosti konkrétneho agenta (technologický postup, zoznam možných výrobných operácií, pravidlá a obmedzenia). Modul uvažovania súvisí s definíciou bázy znalostí. Dokáže obsah týchto znalostí modifikovať, tj. pridávať pravidlá, alebo upravovať jej parametre. Modul by mal tiež disponovať inteligenciou,

čo znamená, že sa dokáže učiť zo skúseností. Takáto vlastnosť modulu uvažovania jednoznačne predstavuje dôležitý aspekt výskumu MAS. [33]

Ďalší prístup k procesu navrhovania súvisí s definíciou časových obmedzení pri interakcii s vonkajším svetom. Široké portfólio scenárov fungovania agenta vyžaduje rôzny časový determinizmus na interakcie. To vedie ku klasifikácii na tvrdé a mäkké aplikácie reálneho času na úrovni činnosti agenta, ktoré musia byť definované už v štádiu návrhu. Dve horné vrstvy majú mäkké požiadavky na reálny čas, zatiaľ čo nízkoúrovňové úlohy riadenia fyzických zariadení vyžadujú tvrdý reálny čas, vid' Obr. 4.4.



Obr. 4.4: Implementačné vrstvy na mikroúrovne agenta v CPPS. [33]

- **Resource agent** - predstavuje inteligentnú softvérovú nadstavbu nad fyzickým výrobným zariadením. Agent výrobného zariadenia môže reprezentovať všetky zdroje naprieč výrobou od dopravníkov, žeriavov, AGV systémov, obrábacích či CNC strojov, až po ľudské zdroje. Transparentne reprezentuje a udržiava aktuálny zoznam služieb, ktorý vie poskytnúť on alebo jeho fyzická časť. Skladá sa z viacerých modulov, ktoré vďaka vzájomnému prepojeniu plnia diagnostickú, riadiacu a plánovaciu funkciu. Každý agent výrobného zdroja dokáže kooperovať s ostatnými agentami a v prípade zmien vo výrobe prípadne prebrať naplánovaný harmonogram na seba. [18]
- **Product agent** - predstavuje inteligentnú softvérovú nadstavbu nad vyrábaným produktom. Takýto inteligentný výrobok by mal disponovať najmä prostriedkami pre proaktívne alokovanie výrobných požiadaviek pomocou komunikačného rozhrania s orientáciou na resource agentov. Mal by byť schopný monitorovať svoj stav, rovnako stav výrobných zariadení, prípadne iných product agentov. Na základe týchto informácií plne riadiť a optimálne plánovať svoju výrobu s prihliadnutím aj na vonkajšie faktory vstupujúce do procesu výroby. Obsahuje procedúry na odstránenie svojej fyzickej časti a uvoľnenie výrobného procesu v prípade neúspešného vyrobenia. Optikou manažmentu štíhlej výroby prípadne PLM (manažmentu životného cyklu výrobku) môže inteligencia výrobku zohrávať rolu pri zbere dát pre spätnú väzbu vývoja ďalšej generácie výrobku či proaktívnej údržbe. Agent inteligentného výroku teda nemusí ukončiť

svoj výrobný cyklus ihneď ako je kompletne vyrobený. Umiestnenie inteligencie výrobku môže byť dedikované na vzdialenom serveri, pričom prebieha komunikácia medzi aktérmi, alebo je zabudované priamo vo výrobku. Z dôvodu, že veľká väčšina inteligentných výrobkov sa bude skladať z ďalších inteligentných výrobkov s rôznou úrovňou inteligencie, je jednou z vlastností product agenta aj miera agregácie inteligencie. Hierarchicky najvyššia (posledná) entita v procese výroby bude vystupovať buď ako inteligentná položka, kedy nie sú rozlíšiteľné jej dielčie komponenty, alebo ako inteligentný kontajner, pri ktorom k agregácií inteligencie nedochádza. [18]

- **Iné typy agentov**

V [34] sú okrem dvoch hlavných agentov, s ktorými v topológii CPPS už uvažujeme, napríklad zásobovací (obstarávací) agent, ktorý uspokojuje materiálne požiadavky zdrojov. Ďalej LMA (Logistic Management Agent) je agentom obsahujúci logistické parametre, akými sú dopravná kapacita a dopravné časy. Rozhranie medzi agentami a človekom môže byť implementované tiež ako samostatný agent inteligentnej objednávky. Spravoval by réžiu všetkých ponúkaných služieb systému a aktuálnej rozpracovanosti zadaných objednávok. Taktiež by vytváral a inicioval product agentov. [18]

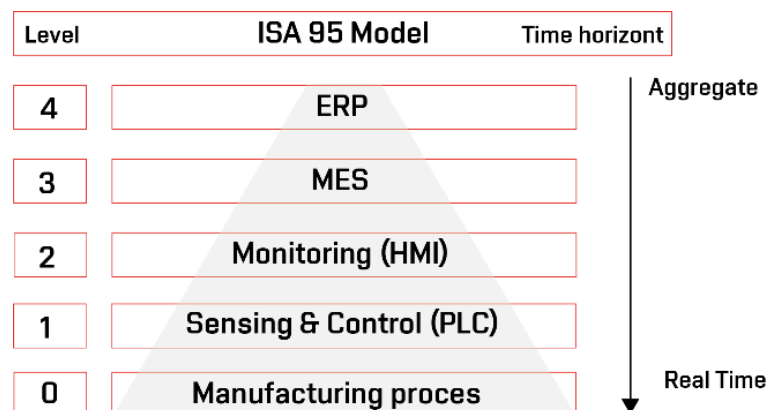
Pomocou vhodných distribuovaných riadiacich algoritmov (viď kapitola 5) môžu jednotlivé stroje (resource agent) a výrobky (product agent) prijímať vlastné rozhodnutia o riadení výroby, týkajúce sa pridelovania zdrojov a koordinácie pomocou automatizovanej formy vyjednávania. Inteligentný výrobok tak vyhľadáva služby inteligentných zdrojov, čím si riadi svoje vlastné vyrobenie.

5. ALGORITMY PRE DISTRIBUOVANÚ ORGANIZÁCIU VÝROBY

Popisované inteligentné výrobné systémy založené na agentovom prístupe sú schopné dosiahnuť efektívnosť vo všetkých úrovniach priemyselných činností. To jasne ukazuje, že paradigma CPPS je perspektívna architektúra a tvorí základný kameň novej výrobnéj paradigmy pre rozhodovacie procesy priemyslu 4.0. Tento inovatívny pohľad na plánovanie výroby je možný aj vďaka masívnemu rozvoju výpočtových zdrojov používaných v inteligentnej výrobe, ktorý umožňuje postupný nástup dynamických (online) plánovacích mechanizmov. Vysoký stupeň prepojenia medzi fyzickými prostriedkami výroby a softvérovými nástrojmi v systémoch CPPS poskytuje možnosť priameho prepojenia výrobnéj haly so systémom na podporu rozhodovania ERP.

5.1 Rozhodovanie v CPPS

Pri analýze distribuovania rozhodovacích procesov s akcentom na princípy priemyslu 4.0 budeme vychádzať z normy ANSI/ISA 95. Táto norma poskytuje všeobecne akceptovanú reprezentáciu architektúry naprieč výrobným podnikom. Hierarchicky organizuje jednotlivé úrovne rozhodovania, rozlišuje sa päť úrovní, vid'. Obr. 5.1. Od 0. úrovne spojené s fyzickým procesom výroby cez 1. úroveň zariadení, ktoré merajú a manipulujú s fyzickým procesom - snímače, aktuátory a súvisiace prístroje. Úroveň 2 predstavuje dispečerské riadenie a dohľad (SCADA). Úroveň 3 MES systémy okrem mnohého iného spolu s ERP zahŕňajú riadenie operácií a proces plánovania. Nakoniec úroveň 4 ERP zastupujúca obchodné činnosti celého podniku a koordináciu všetkých disponibilných podnikových zdrojov. Tento pohľad má určité obmedzenia, najmä pokiaľ ide o agilitu, pretože riadenie ovplyvňuje dynamiku výroby iba prostredníctvom systému ERP. Získavanie informácií z výrobného procesu a naopak modelovanie rozhodovania je striktne hierarchické. [30]



Obr. 5.1: Štruktúra hierarchických úrovní automatizácie vo výrobnom podniku.

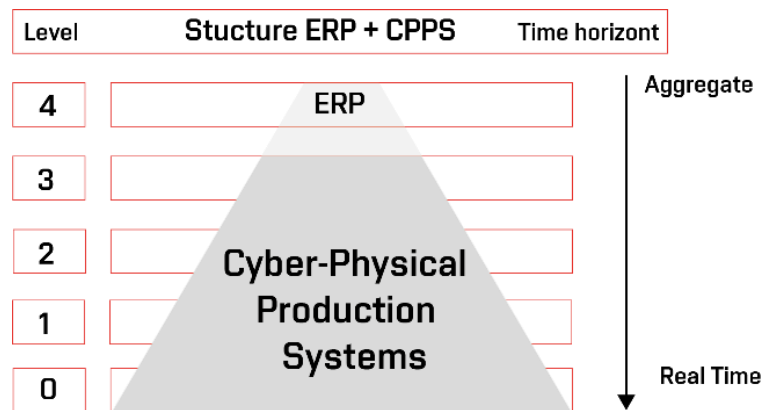
Rozhodovanie v CPPS mení tento tradičný spôsob prijímania rozhodnutí v oblasti priemyselného riadenia. Kapacity, ktorými disponuje architektúra CPPS, umožňujú integráciu prvých 4 úrovní hierarchického modelu ISA-95 do jednej monolitckej distribuovanej architektúry, vid'.

Obr. 5.2. Model CPPS poskytuje prirodzený spôsob ako integrovať dispečerské plánovanie výrobných procesov a aj dohľadové funkcie spolu so samotným riadením fyzickej realizácie. Práve z tejto integrácie plynú priame výhody, ako hlavne už niekoľkokrát uvádzané zvýšenie flexibility výroby, a tiež implicitná odolnosť voči neočakávaným udalostiam vo výrobe. [35]

Systémy ERP v novom modeli rozhodovania ostávajú plniť funkciu prepojenia s ľudskými rozhodovacími zámermi na úrovni súhrnného alebo strategického plánovania. Systém ERP získa a pretransformuje tieto usmernenia pre nižšiu vrstvu CPPS, ktorá plne autonómne zastrešuje zvyšok výrobného procesu a poskytuje možnosť simultánnej optimalizácie rozvrhovania výroby pri vzniku abnormalít v procese výroby. [30]

Za zmienku tiež stojí potreba vertikálnej koordinácie a agility architektúry CPPS s komponentami ERP systému napr. logistické, obchodné, servisné, dopravné, dodávateľské reťazce atď. Ich vzájomná koordinácia (napríklad za použitia protokolu CNP) je nutným predpokladom pre dosiahnutie plnej prepojenosti inteligentnej továrne.

Jedným z hlavných cieľov budúcich výskumných aktivít zostáva riešenie problémov s integráciou vzájomne nie vždy kompatibilných entít a predefinovanie tradičných metód a algoritmov, aby naplňali distribuovaný rámec. [33]



Obr. 5.2: Štruktúra úrovní prvkov automatizácie pri použití CPPS architektúry.

5.2 Model rozvrhovania úloh v CPPS

Cieľom modelu organizácie rozvrhovania v multiagentných štruktúrach CPPS je zabezpečiť splnenie plánu všetkých produktových agentov v systéme a zároveň zabezpečiť čo najlepšiu využiteľnosť zariadení. Historicky je výrobný podnik organizovaný podľa hierarchických úrovní pyramídy automatizácie. Takýto

centralizovaný pohľad na plánovanie a riadenie výroby pomocou celopodnikových informačných systémov ERP vytvorí plán výroby a tá sa následne rozdelí na niekoľko strojov. V tejto práci sa zameriam najmä na tento posledný krok. Bude predstavený prístup, ako možno decentralizovane, dynamicky a autonómne plánovať a hodnotiť vhodnosť výberu výrobných zdrojov pri spracovaní úloh. Tieto úlohy je možné modelovať ako alokačné problémy, pri ktorých sa zásoby a zdroje priradujú určitému počtu dopytov tak, aby sa dodržali existujúce obmedzenia. Avšak na rozdiel od klasického statického plánovania je množina úloh modifikovaná príchodom nových udalostí, akými sú pridávanie, zmeny alebo odstránenie objednávok. Problém sa tak stáva dynamickým a zahŕňa rozhodnutia v reálnom čase. Na riešenie takejto dynamickej povahy sa efektívne využíva uvedený koncept CPPS agentov inteligentných produktov a inteligentných zdrojov, ktoré sú schopné vyjednávať cestu produktu výrobou. Takto koncipovaná decentralizovaná výroba by mala mať možnosť nastaviť aj stupeň priority pre jednotlivé objednávky. Product agent s vyššou prioritou by v alokačných rezervačných frontách resource agentov bol uprednostnený s cieľom najrýchlejšieho smerovania výrobou. Ako vhodné riešenie sa ponúka použitie interakcií kooperácie a negociácie. Prostredníctvom týchto stratégií produktový agent získava informácie o dostupnosti, kapacite a stave výrobných zariadení.

5.2.1 Robustnosť rozvrhovania v CPPS

Pre oblasť riadenia robustnosti v multiagentových systémoch je dôležité rozpoznávanie anomálie a iniciovanie výnimiek pre vysporiadanie sa so zlyhaním. Dôležité sú aj mechanizmy adaptívnej obnovy systému do požadovaného stavu.

- Škálovanie v architektúre CPPS - definuje schopnosť zvládnuť pridávanie/vypadávanie agentov bez neprimeranej straty efektivity alebo nárastu zložitosti. Dosiachnutie škálovateľnosti vyžaduje využívanie otvorených foriem interakcie, mediátorských agentov tak, aby systém vedel reagovať na zmenenú veľkosť komunity agentov. Z pohľadu rozvrhovania úloh je meradlom výkonnosti v tomto bode počet vykonaných úloh a počet správ na hodnotenie flexibility interakčnej stratégie. [36]
- Reakcia na zmeny prostredia - definuje schopnosť zotavenia systému, ak sa zmenia požiadavky na multiagentový systém zo strany zákazníkov, napríklad náhla zmena počtu objednávok. Meranie výkonnosti z pohľadu rozvrhovania môže byť počet vykonaných úloh a ceny (hodnotí náklady vzniknuté prispôbením).
- Reakcia na vypadnutie agenta - prístupy k zvládaniu zlyhaní agenta je detekcia zlyhávajúceho agenta a následne schopnosť delegovať naplánované úlohy na iných agentov. V systémoch s interakčným protokolom CNP je kompenzovanie zlyhaní implicitne podporované, vid' kapitola 5.3.1. Riešenie môže spočívať aj v tieňových agentoch, ktorí poskytujú formu redundancie

v systéme. Mieru výkonnosti rozvrhovania môžeme vyjadriť ako počet zlyhaných úloh. [37]

5.3 Interakcie medzi agentami v CPPS

Interakcie medzi agentami sú do veľkej miery ovplyvnené typom organizácie, v ktorej operujú. Spektrum foriem organizácie možno definovať od autonómnej formy koordinácie vzájomne samostatných agentov až po organizáciu, v ktorej sa čiastočne/úplne vzdávajú svojej samostatnosti. Väčšie holonické štruktúry - korporácie, aliancie, virtuálne továrne so striktné zadanými dlhodobými vzťahmi členov agregujú agentov a tým menia aj ich interakcie.

V práci sa zameriam na formu autonómnych agentov, ktorí si vzájomne poskytujú len služby bez ďalšej koordinácie, vychádzajúc z predstavenej topológie CPPS. Usporiadanie agentom poskytuje delegovanie úloh na základe výmeny, vymieňajú si úlohy za určitý druh úžitku. Agenti majú vzťah v rovine poskytovateľ - zákazník, nie sú súčasťou spoločnej organizácie. Interakcia je tak krátkodobá, založená výlučne na čas potrebný pre vykonanie úlohy.

5.3.1 Protokoly založené na spolupráci

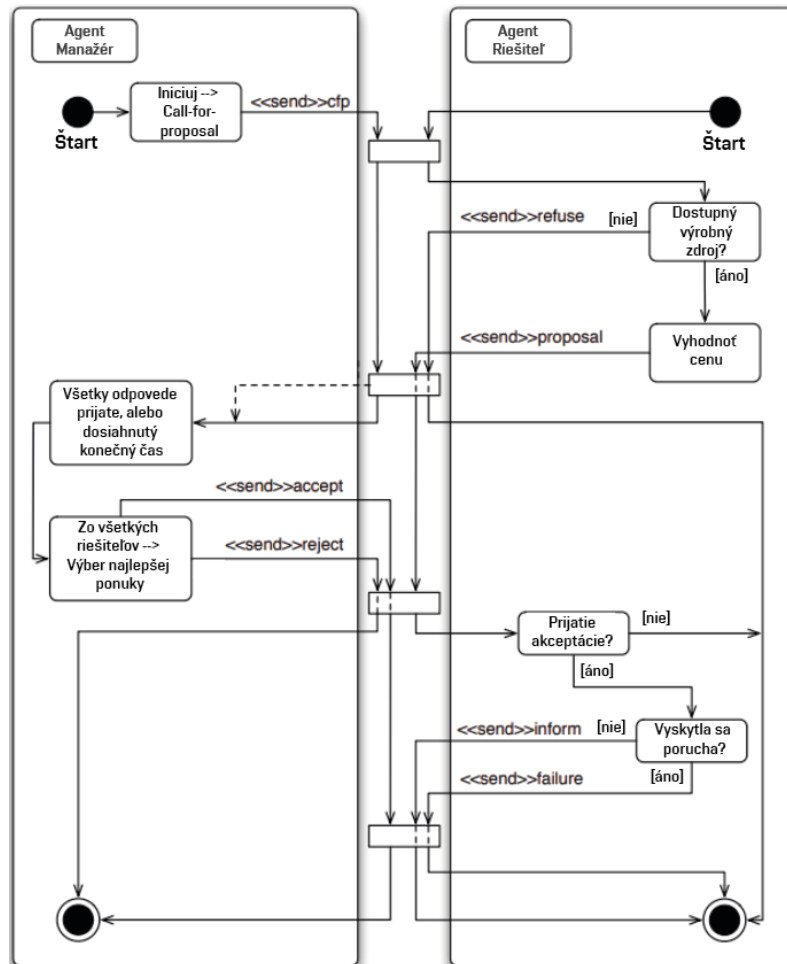
Ako už bolo predstavené v predchádzajúcich kapitolách, agent je samostatná entita, avšak v štruktúrach CPPS existuje mnoho spôsobov, akými agenti môžu spolupracovať. Jedným zo spôsobov je aj tvorba koalícií. Teoretická báza pre výber vhodnej koalície je popísaná v publikácii [38]. V tomto článku je rozobraný problém vytvárania koalícií, ich štruktúrny návrh a vhodnosť spájania agentov do skupín. Do takýchto koalícií môžeme koordinovať len agentov, ktorí majú podobné plány, ciele, disponujú spoločnými parametrami a operujú v rovnakom prostredí.

Vo všeobecnosti existujú dva mechanizmy spolupráce: agenti môžu spolupracovať, aby spojili svoje rôzne schopnosti (kvalitatívna motivácia), alebo agenti s rovnakými schopnosťami môžu spolupracovať na zvýšení kapacity (kvantitatívna motivácia).

Pridelovanie úloh spolupracujúcim agentom a prípadná dynamická (re)alokácia zdrojov v multiagentovom systéme je jednou z kľúčových vlastností takýchto systémov. Spolupráca vnútri modelovaných organizácií si vyžaduje prostriedky koordinácie. Štandardným mechanizmom je protokol CNP - prináša spojenie pre kooperatívne riešenia problémov autonómnyimi agentami. V upravenej forme bol použitý aj v predloženej implementácii CPPS. V nasledujúcich častiach demonštrujem jeho funkčnosť a niektoré problémy spojené s jeho použitím.

Kontrakčná sieť pre distribuované riešenie problému definuje vo všeobecnosti dva typy agentov, riešiteľské a manažérske. Manažérsky agent navrhuje a zasiela predom definovanú formu správy so špecifikáciou požadovanej úlohy. Podľa možnosti komunikačnej infraštruktúry zasielanie funguje pre všetkých broadcastom, prípadne sa požiadavka zasiela iba vybranej skupine kompetentných agentov. Riešiteľský agent

vyhodnotí prijaté požiadavky a s ohľadom na svoje schopnosti posielajú ponuku. Riešiteľský agent môže zastávať aj aktívnu funkciu a iniciatívne inzerovať svoje služby. Manažérsky agent si takto vytvára otvorený zoznam ponúk, z ktorých vyberá tie najvhodnejšie. Takáto výmena správ medzi agentami je rokovanie o spôsobe dosiahnutia spoločného cieľa, pričom predpokladá obojsmernú komunikáciu. [23]



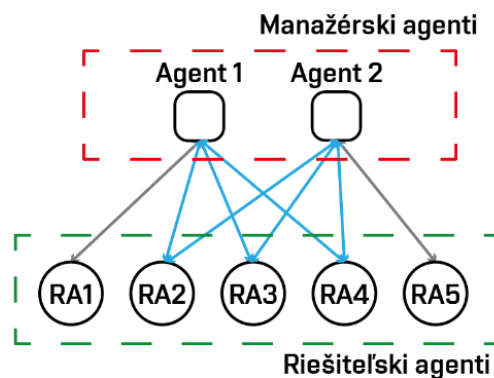
Obr. 5.3: UML diagram základného CNP protokolu. [37]

Efektívnosť protokolu CNP je veľmi závislá na množstve prenášanej komunikácie. Ak sa interakcia medzi agentami zvýši, CNP sa stáva neefektívnym. Pri rozsiahlych aplikáciách protokolu efektívnosť klesá najmä na implicitnú vlastnosť všeobecného vysielania - broadcast. Jeden z prístupov k obmedzeniu všeobecného vysielania je použitie moderátorského agenta na organizáciu komunikácie, ktorý by sprostredkoval spojenie len s kompetentnými agentami pre danú úlohu.

Protokol CNP naráža na svoje limity najmä pri paralelnom vykonávaní viacerých koordinačných spojení. Základná forma CNP postačuje, ak v danom čase prebieha

alokácia len pre jeden zdroj úloh, tj. existuje len jeden manažér (agent inteligentného výrobku), ktorý iniciuje spoluprácu.

Situácia sa dramaticky zmení, ak predpokladáme, že existuje niekoľko produktových agentov alokujúcich výrobné operácie súčasne a uvažujeme o skutočnosti, že resource agenti majú len obmedzené zdroje. CNP implicitne nepodporuje súbežne vykonávať viacero inštancií protokolu, vid'. Obr. 5.4. modré spojenia sú problémom. Protokol žiadnym spôsobom explicitne nedefinuje, koľko ponúk by mal riešiteľský agent urobiť. V skutočnosti je racionálna voľba riešiteľa, aby ponúkal svoje zdroje v čo najväčšom počte aukcií. Takéto súbežné vedenie aukcií s viacerými iniciátormi implikuje stav nazývaný Problém nedočkavého uchádzača „eager bidder problem“. Ak resource agent rezervuje poopytované zdroje príliš skoro, napríklad už pri predkladaní svojej ponuky, môže sa stať, že produktový agent nakoniec ponuku neprijme a resource agent zostane nevyužitý. Ak sa chce tomuto problému vyhnúť a zdroje prideluje neskôr, napr. po získaní akceptácie ponuky, môže nakoniec získať viac akceptovaných ponúk, ako mu umožňujú jeho fyzické obmedzenia. [37]

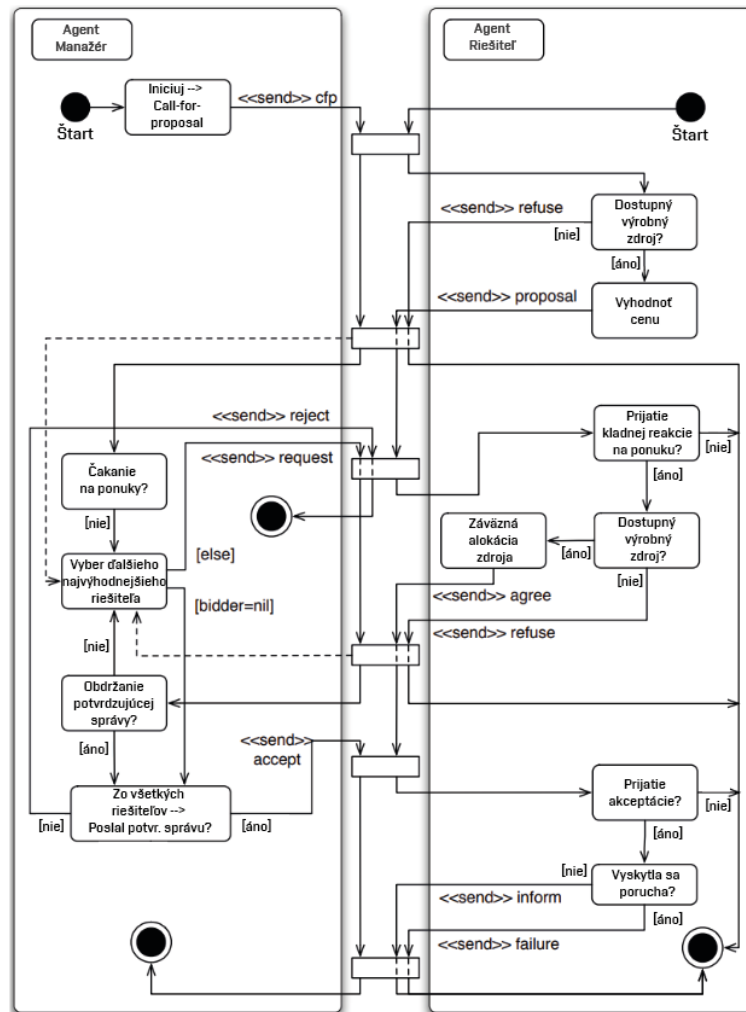


Obr. 5.4: CNP protokol nepodporuje synchronizáciu pre súbežnú komunikáciu.

Ako najjednoduchšie konzervatívne riešenie môže byť uvažovaný prvý prípad, keď resource agent prideluje svoje zdroje hneď v čase odoslania ponuky. Používame tak aukciu s jedným behom, lebo resource agenti môžu v každej inštancii protokolu urobiť len jednu ponuku. Uvedené nutne vedie k neúplnému prideleniu, pretože množstvo iniciátorov (product agentov) nedostane na svoju žiadosť žiadnu ponuku, zatiaľ čo iní ich dostanú niekoľko. A keďže majú na pridelenie len jednu úlohu, ponuky všetkých resource agentov, okrem jedného, ktoré im boli zaslané, sú zbytočné. Je zrejmé, že iteratívnym opakovaním procesu CNP nakoniec dosiahneme správne priradenie pre všetky úlohy, ak necháme iniciátorov opakovane oznamovať svoje úlohy. Avšak takýto pravdepodobnostný algoritmus je veľmi ťažko analyticky vyhodnotiť a zo svojej podstaty je neefektívny.

Rozšírený protokol CNP s potvrdzovaním

Efektívnejší prístup k popisovanému problému je založený na prepracovaní základnej podoby protokolu tak, aby čo najviac odkladal čas záväzku. V pôvodnom CNP protokole účastník prijíma svoj záväzok v ponukovej fáze. V tomto rozšírenom sa záväzok prijíma až vtedy, keď iniciátor (product agent) explicitne požiada riešiteľského agenta prevziať úlohu. UML diagram funkčnosti takéhoto rozšíreného protokolu CNP vid'. Obr. 5.5.



Obr. 5.5: UML diagram rozšíreného CNP protokolu s potvrdzovaním. [37]

Rovnako ako jednoduchý CNP protokol, aj pri tomto type hovoríme o aukcii s jedným behom, keďže každý riešiteľ je obmedzený vystaviť len jednu ponuku. Začiatok tohto procesu kooperácie vychádza z iniciatívy product agenta, ktorý v prvom kroku vyzbiera všetky ponuky od riešiteľských agentov. Do ďalšieho kroku sa presunie po zozbieraní všetkých ponúk, alebo po uplynutí definovaného času, ktorý ošetruje prípadné individuálne zlyhanie doručenia ponúk. Následne product agent (iniciátor, manažér) ponuky zoradí podľa svojich kritérií výhodnosti vzostupne a osloví riešiteľa s najlepšou ponukou. Oslovený riešiteľský agent v tomto bode vie, že úlohu určite dostane pridelenú, stačí iba súhlasiť s jej vykonaním. Táto skutočnosť mu umožňuje bezpečnú alokáciu

zdrojov a zaslanie potvrdenia pre iniciátora. Ak vykonanie úlohy oslovený resource agent odmietne, alebo ak uplynul termín, iniciátor osloví s požiadavkou o alokáciu zdrojov ďalšieho riešiteľa v zoradenom zozname. Iterácia sa zastaví, akonáhle jeden z účastníkov odpovie súhlasnou správou. Všetkým ostatným agentom sa následne pošle správa s odmietnutím návrhu. Komunikačný akt sa ukončuje zo strany iniciátora, ktorý posielala finálnu správu s akceptáciou (nemá dôvod ponuku odmietnuť, pretože je to najvýhodnejší z dostupných agentov, ktorý je schopný vykonať úlohu).

Popisovaný rozšírený CNP protokol v najlepšom prípade vyžaduje, aby každý iniciátor poslal taký počet správ, koľko je riešiteľských agentov, navýšený o dve správy - žiadosť o potvrdenie a odpoveď na ňu. V tom najhoršom prípade musí iniciátor kontaktovať všetkých riešiteľov, aby zistil, že nikto z nich nemôže/nevie úlohu vykonať. To vedie ku navýšeniu novej komunikačnej réžie a následne aj času, hlavne v prípade čakania na časový limit pri zlyhaní niektorého zo zúčastnených agentov.

Menšou nevýhodou tohto prístupu je tiež fakt, že iniciátor potrebuje určitú réžiu na triedenie zoznamu ponúk riešiteľov. Ale vo výsledku rozšírený protokol CNP poskytuje robustnejšie riešenie ako CNP.

Štatistický prístup

Základnou myšlienkou štatistického prístupu je riskovať porušenie rezervačných záväzkov, pokiaľ je pravdepodobnosť tejto udalosti z dlhodobého hľadiska nižšia ako určitá prahová hodnota. V kontexte multiagentného systému organizácie výroby to znamená určiť počet ponúk pre daný prah rizika, ktoré môže riešiteľský agent X poslať (koľkých aukcií sa môže zúčastniť) nad rámec množstva zdrojov, ktoré má k dispozícii.

Uvažujeme:

N_{PA} - množstvo iniciátorov, product agentov,

N_{RA} - množstvo riešiteľov, resource agentov,

B_X - počet ponúk, ktoré resource agent X môže poslať, pre zjednodušenie rovnaký počet ponúk posielajú aj ostatní zúčastnení agenti $B_X = B_A$,

τ - určený prah rizika.

Výpočet pravdepodobnosti p_c riešiteľského agenta, že si vyberie konkrétneho iniciátora na vystavenie ponuky:

$$p_c = \frac{1}{N_{PA}} * B_A \quad 5-1$$

Pravdepodobnosti, že konkrétny riešiteľský agent X dostane jednu prijatú ponuku, keď ju pošle náhodne vybranému iniciátorovi. Výpočet uvažuje pravdepodobnosť, že nikto okrem agenta X si nevybral daného iniciátora:

$$(1 - p_c)^{N_{RA}-1} \quad 5-2$$

Následne pravdepodobnosť p_a pre agenta X , že bude prijatá jediná jeho ponuka vid'. rovnica 5-3 je súčinom počtu všetkých možných permutácií a pravdepodobnosti výberu konkrétneho iniciátora umocnená počtom už vykonaných kôl. Následne súčin pokračuje pravdepodobnosťou akceptácie ponuky a posledný člen súčinu je pravdepodobnosť vybratia konkrétneho riešiteľa iniciátorom.

$$p_A = \sum_{i=0}^{N_{RA}-1} \binom{N_{RA}-1}{i} * p_c^i * (1 - p_c)^{N_{RA}-1-i} * \frac{1}{i+1} \quad 5-3$$

Po získaní pravdepodobnosti prijatia ponuky jedným iniciátorom p_a môžeme dopočítať pravdepodobnosť resp. riziko, že bude prijatá viac ako jedna ponuka pri zúčastnení sa viacerých aukcií. Vystavenie viacerých ponúk je z matematického pohľadu reťazec pokusov, pričom každý má rovnakú pravdepodobnosť, že bude ponuka prijatá. Preto môžeme vypočítať pravdepodobnosť prijatia viac ako jednej ponuky ako Bernoulliho reťaz:

$$P_X = \sum_{i=2}^{B_X} \binom{N_X}{i} * p_A^i * (1 - p_A)^{N_X-i} \quad 5-4$$

V rovnici 5-4 uvažujeme počet zúčastnených minimálne 2 (inak celá pravdepodobnosť nedáva zmysel). V tomto kroku je možné stanoviť maximálny počet vystavených ponúk agenta X tak aby neprekročil prah rizika. Rozdelenie rizika je funkciou premenných B_X a B_A pričom musí platiť: $P_X < \tau$. Analyticky pohľad na rozdelenie rizika odráža skutočnosť, že riešiteľský agent X zvyšuje riziko porušenia rezervačných záväzkov voči iniciátorovi, keď predkladá čoraz viac ponúk. Riziko pre agenta X naopak klesá, ak ostatní agenti predkladajú viac ponúk, a teda znižujú pravdepodobnosť, že bude vybraný X . Inými slovami: ak B_A je pevná, riziko monotónne rastie s rastúcim B_X , ak je B_X pevné, riziko klesá monotónne s rastúcim B_A . Optikou teórie hier je voľba počtu ponúk B_X pre riešiteľského agenta strategickým Nashovým ekviliбриom, ktoré posudzuje racionalitu zvyšovania (nárast rizika) alebo znižovania počtu ponúk (zmenšovania očakávaného výsledku).

Tento prístup ku koordinácii a delegovaniu úloh v multiagentných systémoch je prijateľný len v oblastiach, ktoré umožňujú neúplnosť a nekompletnosť pridelovania úloh s uvedenou pravdepodobnosťou prahu rizika. [37]

Kombinácia štatistického prístupu a CNP s potvrdzovaním

Dôležitou vlastnosťou štatistického prístupu je, že sa dá kombinovať s CNP, alebo s rozšíreným CNP protokolom. Kombinácia štatistického prístupu vstupujúceho do fázy podávania ponúk riešiteľských agentov a deterministicky určitého pridelovania zdrojov vďaka potvrdzovaniu prináša zlepšenie efektivity a zníženie počtu správ medzi účastníkmi. Dôležité je ale podotknúť, že takáto kombinácia nezaručuje úplné priradenie, pretože štatistickým obmedzením počtu vystavených ponúk môže nastať stav, že niektorí

iniciátori nedostanú vôbec žiadnu ponuku a nemajú možnosť pridelit' svoje úlohy. Tento nedostatok sa podobne ako pri základnom protokole CNP dá vyriešiť zavedením iteratívneho opakovania.

Uvedené nedostatky CNP protokolov a postupný rozvoj alternatívnych prístupov pre súbežné pridelovanie úloh pripravil základ pre protokoly založené na teórii aukcií. Trhové mechanizmy (aukcie) zovšeobecňujú CNP a tvoria dôležitú súčasť výskumu multiagentových systémov.

5.3.2 Negociačné protokoly - trhové mechanizmy

Okrem popisovaného dosahovania kooperatívneho správania sa negociácia v multiagentných systémoch využíva aj na riešenie konfliktných situácií - najčastejšie spor (súťaž) o zdroje. Na rozdiel od predchádzajúcich kooperačných protokolov v procese aukcie sú jej účastníci v konkurenčnom vzťahu. Trhové mechanizmy založené na princípoch ekonómie poskytujú pokročilejšie metódy, ktoré v analytických metódach pridelovania úloh nie sú zastúpené. Pomyselný trh zvyčajne určuje cenu výrobného zdroja v čase a priestore, na ktorú reagujú produktívni agenti ponukou koľko sú ochotní za tento výrobný zdroj (podiel na ňom) zaplatiť. Konvergenciou ku konečnej dohode (stav v ktorom agenti už nemenia svoje požiadavky na zdroje vzhľadom na aktuálne ceny) sa funkcia užitočnosti maximalizuje.

Priebeh negociácie determinujú pravidlá: [29]

- Miera kooperácie (od minimálnej až po rozlične veľké koalície spolupracujúcich agentov),
- Regulácia (regulačné opatrenia ktorým podliehajú agenti pri negociácii),
- Množstvo a typ agentov (proaktívne vs. reaktívne vystupovanie v procese vyjednávania),
- Výber modelu vyjednávania - z pohľadu času potrebného na dohodu, čo je dôležitý faktor z hľadiska nákladov na samotnú komunikáciu a blokovanie výpočtového výkonu.

V práci sa oblasti použitia trhových mechanizmov prioritne ďalej nevenujem.

5.4 Operatívne plánovanie

Dynamické plánovanie je problematika diskutovaná najmä v kontexte rastúcej možnosti získavania informácií v reálnom čase. Rozdiel s vyššie opísanou doménou off-line rozvrhovania a plánovania výroby spočíva v tom, že dynamické on-line plánovanie aktualizuje výrobný plán so spätnou väzbou na reálne dynamicky sa meniace podmienky a nepredvídané udalosti narúšajúce výrobu. Ak výrobné zariadenie vypadne z prevádzky z dôvodu nejakej poruchy, operácie, ktoré sú mu pridelené, sa musia presmerovať na iné stroje, čím sa zmení plánovanie výrobného procesu. Takáto zmena harmonogramu je nutná, aby sa minimalizoval vplyv abnormalít na výkonnosť systému výroby. Oproti statickému - deterministickému plánovaniu to dynamické so sebou prináša parametre

rozvrhov, akými sú stabilita, robustnosť a ne-konzistentnosť „nervozita rozvrhu“. Tieto parametre slúžia na určovanie výkonnosti algoritmov dynamického plánovania. Uvedený problém nie je vôbec triviálny a aj preto existuje mnoho stratégií, politik a metód založených na širokej škále experimentálnych a praktických prístupov k zmenám harmonogramu. V tejto kapitole uvediem len základný rámec informácií dynamického plánovania súvisiacich s témou diplomovej práce. [39]

5.4.1 Metódy rozvrhovania výroby

Deterministické, statické metódy rozvrhovania možno považovať za špeciálny prípad rozvrhovania, kde existuje konečná množina úloh bez uvažovania neistôt a abnormalít plynúcich z reálnej výroby. Bližšie boli tieto metódy rozoberané v úvode diplomovej práce. Podobnými sú aj stochastické, statické metódy rozvrhovania, kde rovnako existuje konečná dopredu definovaná množina úloh, ale niektoré parametre úloh nie sú špecifikované. Napríklad časy spracovania úloh môžu mať štatisticky prípustný rozptyl, skutočné časy začatia a ukončenia úloh nebudú zodpovedať očakávaným. Vykonávanie plánu si tak následne vyžaduje minimálne nejaké pravidlo alebo politiku na kompenzovanie chyby v pláne. Napríklad možnosť upraviť rozvrh v určitom okamihu počas vykonávania tak, aby sa reagovalo na skutočnosť. Takýto manévrovací priestor pre kompenzovanie abnormalít môžu priniesť umelo pridané časy nečinnosti do pôvodného rozvrhu. [40] Oproti tradičným statickým metódam stoja dynamické. Každá úloha si bezprostredne pred spracovaním vyžaduje naplánovanie. Z pohľadu variability času uvoľnenia do výroby a samotného toku zákazky výrobou rozlišujeme tri dôležité prípady. Od cyklicky sa opakujúceho, až po reentrantný tok bez časovej stability, či prognózovateľnej opakovateľnosti, vid'. Tabuľka 5-1.

Tabuľka 5-1: Rozdelenie - metódy rozvrhovania výroby. [39]

Metódy rozvrhovania výroby				
Statické (konečný počet zákaziek)		Dynamické (neznámy počet zákaziek)		
Deterministické (všetky parametre výroby apriórne známe)	Stochastické (malá miera neistoty v niektorých parametroch)	Žiadna miera variability (cyklická výroba)	Variabilný čas uvoľnenia do výroby (flow shop)	Variabilita v celom procese výrobou (job shop)

5.4.2 Stratégie preplánovania výroby

V operatívnom plánovacom prostredí s neistým príchodom dopredu nedefinovaných úloh rozlišujeme dve stratégie riadenia výroby - dynamické alebo prediktívno-reaktívne. Typické pre dynamické stratégie plánovania je, že nevytvárajú výrobné plány. Namiesto toho decentralizované formy riadenia výroby pridávajú úlohy výrobným zdrojom, pričom

vychádzajú iba z informácií dostupných v okamihu odoslania. Takéto schémy využívajú dispečerské priradovacie pravidlá alebo iné heuristiky na určenie priorit úloh čakajúcich na spracovanie. Bežné dispečerské pravidlá využívajú časy spracovania a dátumy splatnosti v rôznych kombináciách. Väčšinou sú rozšíreniami metodík, ktoré sa osvedčili a fungujú pri jednoduchých problémoch deterministického plánovania (napríklad najkratší čas spracovania (SPT) a najskorší termín odovzdania (EDD)). Výhodou je, že výpočtová náročnosť dispečerských pravidiel je nízka a je možné ju rozšíriť aj o implementáciu pravidiel JIT, prípadne Kanbanu či iné mechanizmy založené na ťahaní. [34]

V prípade výrobných zdrojov bez nastavovacích časov navrhované pravidlá dispečingu uprednostňujú úlohu podľa rozdielu medzi jej dátumom splatnosti a očakávaným časom do dokončenia úlohy. V prípade zdrojov s potrebným časom nastavenia sa navrhované pravidlá dispečingu zameriavajú na dokončenie všetkých čakajúcich úloh jedného typu pred vykonaním nastavenia a spracovaním úloh iného typu. Všetky skúmané pravidlá udržiavajú stroj v prevádzke, ak na spracovanie čakajú nejaké úlohy. (To znamená, že stroj nemôže ignorovať čakajúce úlohy.)

Prediktívno-reaktívne plánovanie sa vo všeobecnosti skladá z dvoch základných krokov. V prvom kroku sa vytvára počiatočný plán výroby. Ten sa v kontrolnej fáze porovnáva so skutočným priebehom. Ak ich rozdiel prekročí stanovenú hranicu, nasleduje druhý krok, v ktorom sa tento plán aktualizuje v reakcii na poruchu, alebo inú udalosť s cieľom minimalizovať jej vplyv na výkonnosť systému. Prediktívno-reaktívna stratégia plánovania zahŕňa tri typy stratégií zmeny plánovania: periodické, riadené udalosťami a hybridné. Pri riadení preplánovania iba na základe abnormálnych udalostí sa v rozsiahlych výrobných systémoch s mnohými udalosťami v rýchлом slede za sebou môže vyskytovať stav permanentného upravovania rozvrhu spojeného s nadmernými výpočtovými požiadavkami.

Príklady typov narušení, ktoré môžu iniciovať zmenu výrobného plánu:

- Porucha výrobných strojov,
- Príchod urgentnej (prioritnej) objednávky,
- Zrušenie objednávky,
- Zmena času splatnosti (odovzdania),
- Nereálny predpoklad času trvania výrobku,
- Problémy s kvalitou a nedostupnosťou materiálu

Aby nedochádzalo k takýmto príliš častým zmenám rozvrhu a následne k zníženej stabilite rozvrhu, a aby zároveň rozvrh vždy reflektoval na závažné zmeny vo výrobe, vznikli hybridné metódy. Sú spojením stratégií udalostí a periodickej schémy preplánovania, označované sú aj ako prístupy s kľzavým časovým horizontom. Hybridný prístup s kľzavým časovým horizontom zmierňuje nepraktickú požiadavku globálnej optimalizácie na lokálne optimálne rozvrhovanie (iba pre aktuálne vybrané časové okno). Znížením komplexnosti problému ho možno riešiť aj v reálnom čase, preto vhodný

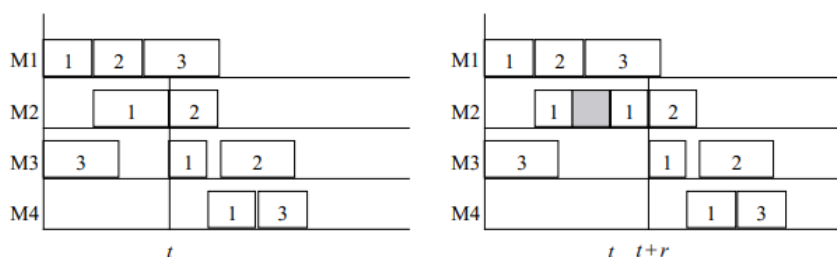
na dynamické plánovanie. Hlavnou koncepciou plánovania v kĺzavom horizonte je výber niekoľkých úloh čakajúcich na spracovanie tak, aby sa naplnilo časové okno. Pri vytváraní okna úloh je nutné určiť výberové pravidlo. Do úvahy by sa mali vziať časy príchodu úlohy a rozdiel medzi plánovaným časom ukončenia a časom splatnosti úlohy. Cez váhové koeficienty týchto parametrov môžeme určiť prioritu úlohy. Kĺzanie okna úloh sa realizuje odstránením dokončených úloh a pridaním nových úloh. Postup sa nasledovne opakuje dovtedy, kým sa neukončia všetky operácie na všetkých úlohách. Čím viac úloh je v pomyselnom okne, tým bližšie je výsledok globálnemu optimu, ale výpočtový čas je dlhší. Určenie optimálnej periódy (časového okna) rozvrhovania je náročná úloha. [41] Zhrnutie stratégií preplánovania výroby vid'. Tabuľka 5-2.

Tabuľka 5-2: Stratégie preplánovania výroby v kontexte operatívneho plánovania. [39]

Stratégie preplánovania výroby			
Dynamické (nevytvára rozvrh)	Prediktívno-reaktívne (vytvor odhad a aktualizuj) Politika iniciovania preplánovania		
Prirad'ovacie pravidlá, JIT paradigmy	Periodická	Vyvolaná udalosťou	Hybridná

5.4.3 Metódy preplánovania výroby

Metódy pre tvorbu výrobného rozvrhu zahŕňajú širokú oblasť, ich podrobný opis by presahoval potreby tejto diplomovej práce. V článku [42] je prezentovaný zástupca takýchto algoritmov na generovanie robustných rozvrhov. Je založený na genetickom algoritme, ktorý minimalizuje očakávané oneskorenie a očakávanú dĺžku trvania. Uvádzané metódy preplánovania aktualizujú rozvrhy v reakcii na abnormalitu vstupujúcu do výrobného procesu. Pre prehľadnosť uvediem len metódy: zmena posunom doprava, čiastočná zmena a úplná zmena (regenerácia rozvrhu). Zmena harmonogramu s posunom doprava odkladá každú zostávajúcu operáciu o čas potrebný na to, aby bol rozvrh realizovateľný a splňal precedenčné obmedzenia. Napríklad v Ganttovom diagrame vid'. Obr. 5.6. Ak stroj M2 zlyhá počas spracovania úlohy 1 a čas opravy si vyžaduje r časových jednotiek, potom sa čas dokončenia úlohy 1 (na stroji M2) posunie z t na $t+r$. Zostávajúce úlohy na strojoch M2, M3 a M4 sa rovnako oneskoria o r časových jednotiek. [39]



Obr. 5.6: Použitie metódy posunu doprava na preplánovanie výrobného rozvrhu. [39]

Čiastočné preplánovanie preplánuje len operácie, ktoré sú priamo alebo nepriamo ovplyvnené poruchou. Táto metóda zachováva pôvodný harmonogram v čo najväčšej možnej miere, čo sa pozitívne prejaví aj na stabilite rozvrhu. Aj preto väčšina vyvinutých heuristik uvažuje len o zmene rozvrhu ovplyvnených operácií. V článku [40] je navrhnutá stratégia preplánovania využívajúca myšlienku match-up. Myšlienka smeruje k preplánovaniu, ktoré v istom bode dosahuje rovnakého stavu s pôvodným rozvrhom, na ktorý môže nadviazať a vrátiť sa na pôvodne plánovanú trajektóriu výrobou. Po poruche stroja sa určí bod zhody pre každý stroj a preplánuje sa len tá časť rozvrhu, ktorý pokrýva časový interval medzi poruchou a časom match-up. Zmena plánovania s posunom doprava je špeciálnym prípadom tejto metódy.

Kompletná zmena (regenerácia rozvrhu) preplánuje celú množinu operácií, ktoré neboli spracované pred časom poruchy, vrátane tých, ktoré neboli ovplyvnené narušením. Hlavnou nevýhodou takéhoto prístupu je nadmerná výpočtová náročnosť. Zhrnutie metód preplánovania výroby vid'. Tabuľka 5-3.

Tabuľka 5-3: Metódy preplánovania výrobného rozvrhu. [39]

Metódy preplánovania výroby			
Tvorba rozvrhu	Úprava rozvrhu		
Implicitné rozvrhy, robustné rozvrhovacie metódy	„Right-shift“ preplánovanie	Čiastočné preplánovanie	Kompletná zmena

Predstavený koncept agentových štruktúr a architektúry CPPS poskytuje rámec na úspešnú implementáciu procesu dynamického preplánovania. Tento rámec zastupuje nielen súbor techník na vytváranie a aktualizáciu výrobných rozvrhov, ale komplexnú riadiacu stratégiu, ktorá implicitne takéto on-line plánovanie podporuje. Vychádza z princípu fungovania decentralizovaných štruktúr, v predloženej implementácii založenej na dynamickej stratégii plánovania.

5.5 Navrhnutý algoritmus organizácie výroby CPPS

Pridelovanie úloh v multiagentových systémoch je zložitý koordinačný proces, ktorý si vyžaduje inovatívne prístupy. Predstavený algoritmus vychádza z vlastností dynamickej stratégie rozvrhovania. Z povahy dopredu neznámeho, stochastického príchodu úloh implicitne nevytvára rozvrh. Takáto dynamická stratégia rozvrhovania spočíva v cyklickom preplánovaní postupne vznikajúceho rozvrhu na úrovni jednotlivých resource agentov. Úplne sa tak eliminuje centralizácia, a v dôsledku toho je rozvrhovanie vlastnej výroby plne v kompetencii inteligentného produktu (product agenta). Cieľom algoritmu je teda umožniť product agentovi nájsť vhodnú sekvenciu resource agentov, čím alokuje výrobné zdroje pre daný typ úlohy. Je nutné podotknúť, že architektúra CPPS má predpoklad implementovať len dispečerské rozvrhovanie.

Strategické a dlhodobejšie plány výrobného podniku sú plne v kompetencii systémov ERP. V kontexte kapitoly 5.4 možno tento algoritmus označiť aj ako "heuristika vkladania". Princípom takejto heuristiky je výber takého miesta (časového slotu) v rezervačných frontách resource agentov, ktoré minimalizuje nákladovú funkciu vyjadrujúcu cenu vyrobenia produktu. Následne bude táto heuristika rozšírená o prediktívny rozmer v podobe plánovania s kľzavým časovým horizontom. Stratégia plánovania sa tak zmení na prediktívno-reaktívnu. Táto forma však nie je v úplnom súlade s decentralizovaným rámcom plánovania a autonómnosti agentov, nakoľko v systéme tak vznikne jeden centrálny člen, ktorý naplánuje rozvrh všetkým produktovým agentom. Z tohto dôvodu s plánovaním v kľzavom časovom horizonte v nasledujúcej simulácii CPPS architektúry nepracujem. Formalizmus v definícií modelov agentov čerpám z [43].

5.5.1 Model výrobného zdroja (Resource agenta)

Úlohou každého resource agenta je udržiavať tento model schopností aktuálny, najmä rezervačné časové sloty, ktoré sa každou interakciou s product agentom môžu meniť. Formálny popis resource agenta v navrhnutej architektúre CPPS je:

$$RA = (X, E, T_r, Prp_p, Prp_{np}, x_0, R_T) \quad (5-5)$$

$X = \{x_0 \dots x_n\}$ množina všetkých fyzických stavov, ktoré môže RA nadobúdať.

$E = \{e_0 \dots e_n\}$ množina všetkých logistických a výrobných operácií, ktoré môžu nastať.

$T_r: X \times E \Rightarrow X$ prechodová funkcia RA.

$Prp_p: X \Rightarrow P_p$ priradzuje fyzickým stavom RA vlastnosť meniť kompozíciu výrobku tj. napríklad fyzikálna, alebo chemická zmena, montáž nejakej časti, atď.

$Prp_{np}: X \Rightarrow P_{np}$ priradzuje fyzickým stavom RA vlastnosť, ktorá neumožňuje meniť kompozíciu výrobku, tj. presun v sekvencii výrobných zdrojov, natočenie výrobku atď.

x_0 : počiatočný stav výrobného zdroja

$R_T: E \Rightarrow Sc_e$ rezervačná fronta resource agenta, priradzuje udalostiam plán vykonania.

Množina $Sc_e = \{(t_0, t_1), (t_2, t_3) \dots (t_{m-1}, t_m)\}$ udáva ohraničené časové sloty, v ktorých môže resource agent obsluhovať udalosť e_n . Pre nezápornosť plynutia času samozrejme platí:

$$t_0 \leq t_1 \leq t_2 \dots \leq t_m$$

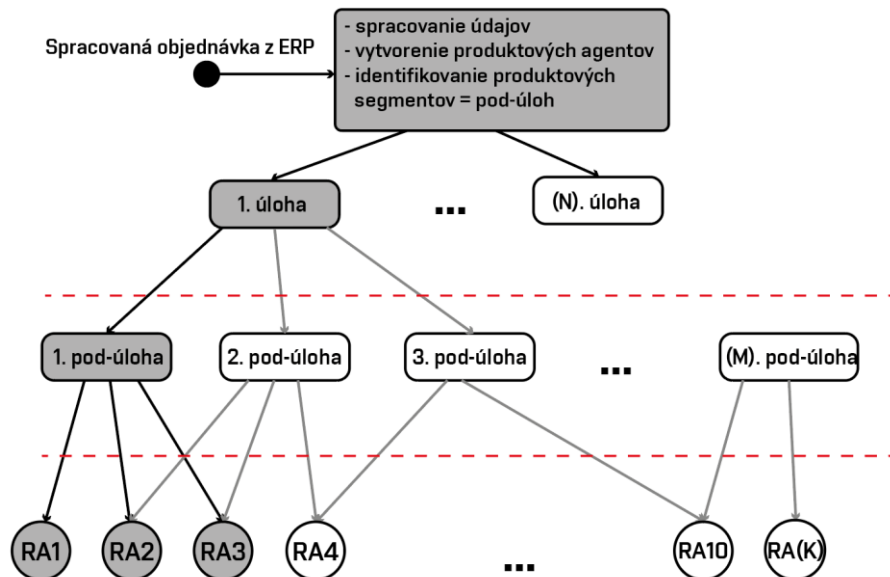
V tomto modeli neuvažujem tímovú prácu na strane resource agentov. Pri spolupráci by sa pokúsil oslovený resource agent, na základe zoznamu modelov agentov vo svojom susedstve, spoločne vyskladať kompletnú ponuku pre produktového agenta. Tento zoznam by obsahoval množinu stavov ($X_s \in X$), ktoré zdieľa s ostatnými resource agentami, (napríklad uloženie produktu na miesto, odkiaľ si ho prevezme ďalší agent). Táto kompetencia bola presunutá na produktových agentov, ktorí musia výrobnú úlohu vhodne rozložiť na pod-úlohy (produktové segmenty), ktoré reflektujú na modely schopností konkrétnych resource agentov. Uvedené zaručuje, že resource agenti budú vždy schopní poskytnúť kompletnú ponuku pre uskutočnenie výrobnéj úlohy, limitujúca bude len kapacita časových slotov.

5.5.2 Model inteligentného výrobku (Product agenta)

Model produktového agenta je charakterizovaný nasledujúcimi atribútmi:

- ID kód výrobku, unikátne číslo pre daný typ výrobku.
- Požiadavky na samotné výrobné zdroje (resource agent) - fyzické vybavenie, personálne kapacity a materiálové vstupy do výroby.
- Technologický plán výroby, tj. postup procesných operácií nutných pre zabezpečenie výroby fyzického produktu v zmysle zadania. Vzniká tak, že každá objednávka prichádzajúca do systému sa rozdelí na N úloh, ktoré sú definované zoznamom: $P_d = (P_1, P_2, P_3, \dots, P_N)$. Jedna úloha je zastupovaná jedným product agentom. Jednotlivé úlohy sa ďalej delia na M pod-úloh $P_{d1} \{p_{d11}, p_{d12}, \dots, p_{d1M}\}$. Toto delenie môže nadobúdať niekoľko stupňov podľa zložitosti a potreby technologicko-výrobného plánu. Zjednotením všetkých stupňov navzájom závislých pod-úloh prislúchajúcich jednej technologickej operácii vzniká koordinačná skupina, ktorá korešponduje s koordinačnou skupinou resource agenta. Ak inteligentný výrobok vyžaduje vykonanie postupnosti M pod-úloh, tak P_{NMK} (M označuje poradie pod-úlohy, K označuje číslo výrobného zdroja, na ktorom sa daná úloha vykoná) a každá pod-úloha má známy vykonávací čas T_{NMK} , potom životnosť produktového agenta T_i , sa dá vyjadriť nerovnosťou:

$$T_i \geq \sum_{MK} T_{NMK} \quad (5-6)$$



Obr. 5.7: Popisovaná štruktúra hierarchického delenia technologických operácií.

- História všetkých vykonaných operácií spolu s dátovými reportami pre účely spätného trasovania a zabezpečenia požadovanej kvality.

- Výrobná cena výrobku, ktorá reflektuje na aktuálne náklady požadovaných technologických operácií. Na tieto náklady vplývajú najmä personálna náročnosť a aktuálne ceny vstupujúcich energií do technológie výroby.
- Stavové veličiny charakterizujúce inteligentný výrobok. Patrí sem životnosť product agenta T_i , najneskorší čas dokončenia produktu d_n a požadovaná kvalita výrobku.

5.5.3 Plánovanie operácií

Plánovanie prebieha prostredníctvom protokolu založenom na spolupráci vid'. kapitola 5.3.1. Vo všeobecnosti dva najdôležitejšie typy správ na koordináciu medzi product a resource agentom sú založené na dopyte a ponuke.

- Dopyt po službe (Bid Request) - v CNP protokole tento objekt zastupuje metóda „call-for-proposal“, ktorá posiela definovanú formu žiadosti o výrobné zdroje všetkým resource agentom v systéme. Ak by sme integrovali resource agentov do koordinačných skupín podľa vzájomnej príslušnosti vid'. v kapitole 6.1, mohla by sa komunikácia zjednodušiť len na úroveň product agent vs. koordinačná skupina. Následne v rámci koordinačnej skupiny by na základe štatistík OEE a iných ukazovateľov využiteľnosti zdrojov prebehlo prerozdelenie konkrétnemu resource agentovi. Odpoveďou na dopyt je ponuka zo strany resource agenta.
- Ponuka (Bid) - definovaná forma správy od resource agenta, ktorá obsahuje informácie o poskytovaných službách spolu s časovými možnosťami v podobe rezervačných front. V predstavenom protokole CNP je táto správa zastupovaná pojmom „proposal“.

Štruktúra žiadosti o výrobné zdroje (Bid Request) obsahuje parametre:

$$B_{req} = (PA, P_{id}, d_n) \quad (5-7)$$

Prvý parameter PA zastupuje identifikáciu product agenta, ktorý posiela dopytovú správu B_{req} . Aktuálna miera rozpracovanosti označuje parameter P_{id} . Ten obsahuje množinu doposiaľ nerealizovaných technologických operácií - pod-úloh, ktoré sú dopytované od resource agenta. Posledný parameter d_n je najneskorší čas, do ktorého je nutné dopytované operácie rozvrhnúť do rezervačného plánu resorce agenta. Resource agent tak vie určiť rozpätie časového úseku svojej rezervačnej fronty, ktorý je ešte relevantný pre žiadajúceho product agenta, a ten mu poskytnúť. Konkrétne časové sloty si už vyberá product agent na základe vlastnej heuristiky výberu.

Ako odpoveď na žiadosť prichádza zo strany resource agenta ponuka. Štruktúra parametrov je nasledovná:

$$Bid = (Str_e, Str_x, Prp_p, Prp_{np}, R_T) \quad (5-8)$$

Parametre sekvencie možných udalostí - $Str_e = e_0, e_1 \dots e_n$, sekvencie možných stavov fyzického systému - $Str_x = x_0, x_1 \dots x_n$ spolu s Prp_p a Prp_{np} definujú schopnosti resource agenta v kontexte požadovaných služieb, ktoré môže alokovať. Parameter R_T je v tejto fáze veľmi dôležitý pre výber najvhodnejšieho voľného časového slotu výrobného zdroja. Následne product agent iniciuje vytvorenie rezervácie, tj. rozšírenie existujúcej množiny $Sc_e = \{(t_0, t_1), (t_2, t_3) \dots (t_{m-1}, t_m)\}$ časových úsekov v R_T o vybraný časový slot. Parametre Str_e a Str_x odosielanej ponuky získame porovnaním schopností resource agenta s tými, ktoré sú dopytované zo strany product agenta. Zo zápisu (5-9) plynie, že resource agent môže odpovedať na dopytovú požiadavku len ak nájde sekvenciu stavov X_m , ktoré mu umožnia dosiahnuť požadovaného koncového stavu vyrábaného produktu.

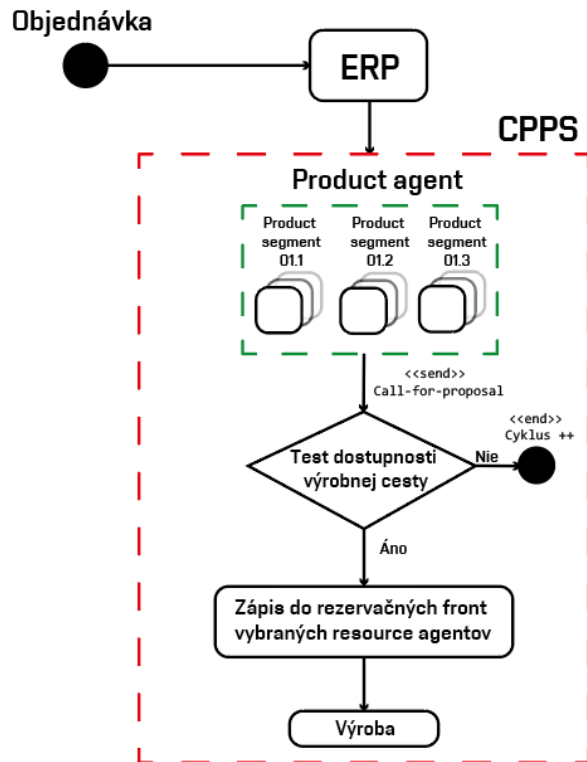
$$X_m = \{x \in X | Prp_p(x) \in P_{id} \cup Prp_{np}(x) \in P_{id}\} \cap \{x_{i+1} = T_r(x_i, e_i)\} \quad (5-9)$$

kde $0 \leq i \leq n - 1$

Po odoslání dopytovej žiadosti čaká product agent na ponuky od resource agentov. Ak ich do určitého času nedostane v požadovanom počte, alebo z nich nedokáže vyskladať model výrobného prostredia (výrobnú cestu produktu), môže modifikovať parametre žiadosti. Najčastejšie predĺžiť čas splatnosti (najneskorší čas vyrobenia), prípadne požiadať o asistenciu operátora.

V diagrame vid'. Obr. 5.8 sú zobrazené základné kroky pridelovania výrobných zdrojov jednotlivým produktovým agentom. Po prijatí objednávky systém ERP túto objednávku spracuje, určí jej výrobný postup, prioritu a rozvrhne jej vyrobenie v rámci dlhodobého strategického plánovania podniku. Po tomto kroku je objednávka posunutá do výrobnjej časti, ktorú zastupuje multiagentný systém CPPS. V prvom kroku sa vytvorí príslušný počet product agentov podľa modelu vid'. vyššie. Product agenti vzájomne medzi sebou nekomunikujú a jediná ich interakcia je priamo smerom ku konkrétnym resource agentom, alebo koordinačným skupinám resource agentov. Takéto koordinačné skupiny môžu byť zahrnuté do systému, pričom tvoria nadradeného holonického agenta nad resource agentami určitej technologickej časti. Uvádzaná integrácia resource agentov znižuje komunikačnú náročnosť a výrazne zjednodušuje proces vyjednávania. V tomto usporiadaní sú všetci zúčastnení agenti v kooperatívnom vzťahu a na vyjednanie používajú protokol CNP s potvrdzovaním.

V ďalšom kroku product agenti zozbierajú ponuky od kompetentných resource agentov, tj tých, ktorí kladne odpovedali na ich požiadavku a disponujú požadovanými schopnosťami (vybavením, materiálom a ľudskými kapacitami). Product agent sa v zmysle svojho technologického plánu všetkých výrobných operácií pokúsi zostaviť výrobnú cestu/cesty. V prípade úspechu sa pristúpi k zápisu do rozvrhov všetkých dotknutých výrobných zdrojov.



Obr. 5.8: Vývojový diagram základných častí algoritmu.

5.5.4 Heuristika vkladania úloh do rozvrhov resource agentov

Záujmom každého product agenta je minimalizovať náklady na vyrobenie výrobku, ktorého v multiagentnom systéme zastupuje. Dynamická stratégia rozvrhovania v reálnom čase a bez implicitného rozvrhu umožňuje len veľmi obmedzené možnosti optimalizácie. Produktívni agenti prichádzajú do systému stochasticky, pričom dopredu nie sú známe žiadne parametre potrebné pre tvorbu optimálneho rozvrhu. Navrhnutá stratégia vkladania úloh do rozvrhov výrobných zdrojov vychádza z princípov JIT organizácie výroby. Podľa tejto teórie sú v nákladoch na plánovanie zahrnuté výrobné náklady, náklady na vzniknuté dočasné zásoby a náklady za penalizáciu oneskorenia. Úlohy by sa mali správne prideliť jednotlivým výrobným zdrojom tak, aby sa minimalizoval ich čas rozpracovanosti a zároveň boli vyrobené čo najbližšie svojmu najneskoršiemu času vyrobenia. Všetky tieto parametre sú zahrnuté v minimalizácii nákladovej funkcie.

Uvedená optimalizácia je zavedená na globálnej úrovni nákladov celej výrobnéj cesty product agenta. Lokálna optimalizácia rozvrhu na úrovni konkrétnych resource agentov nie je zahrnutá v tejto heuristike. Z dôvodu flexibilnej povahy uvažovaného typu výroby JobShop, veľkej variability výrobkov, ale najmä precedenčných obmedzení operácií by takáto zmena v rozvrhu na jednom z resource agentov vo výrobnéj ceste produktu implikovala zmenu rozvrhov všetkých nasledujúcich operácií. Matematický popis optimalizácie výberu časového slotu:

S = Súbor strojov $\{1,2, \dots s \dots, S\}$.

M = Množina pod-úloh (produktových segmentov agenta) $\{1,2, \dots m \dots, M\}$.

$T^{(s)}$ = Dostupné pracovné hodiny stroja s v plánovacom horizonte.

$C^{(s)}$ = časovo premenlivé náklady za jednotku časového slotu stroja s .

$P^{(s)}$ = časová náročnosť výroby na stroji s , počet časových slotov.

d_m = najneskorší čas vyrobena pod-úlohy m .

δ_T^m = koeficient váhy omeškanie pod-úlohy m , (prípustné len pre poslednú technologickú operáciu p_m , inak by rozvrh porušil precedenčné obmedzenia). Pre pod-úlohy $(p_1 \dots p_{M-1})$ = dostatočne veľká hodnota.

λ = koeficient váhy pre predčasne vyrobienie.

$T_m^{(s)}$ = rozvrhovaný čas ukončenia spracovania pod-úlohy m pomocou stroja s .

$$\min \sum_{s=1}^S C^{(s)} * P^{(s)} + \sum_{m=1}^M \delta_T^m (T_m^{(s)} - d_m) + \sum_{m=1}^M \lambda (d_m - T_m^{(s)}) \quad (5-10)$$

$$T_m^{(s)} > d_m, \quad \lambda = 0;$$

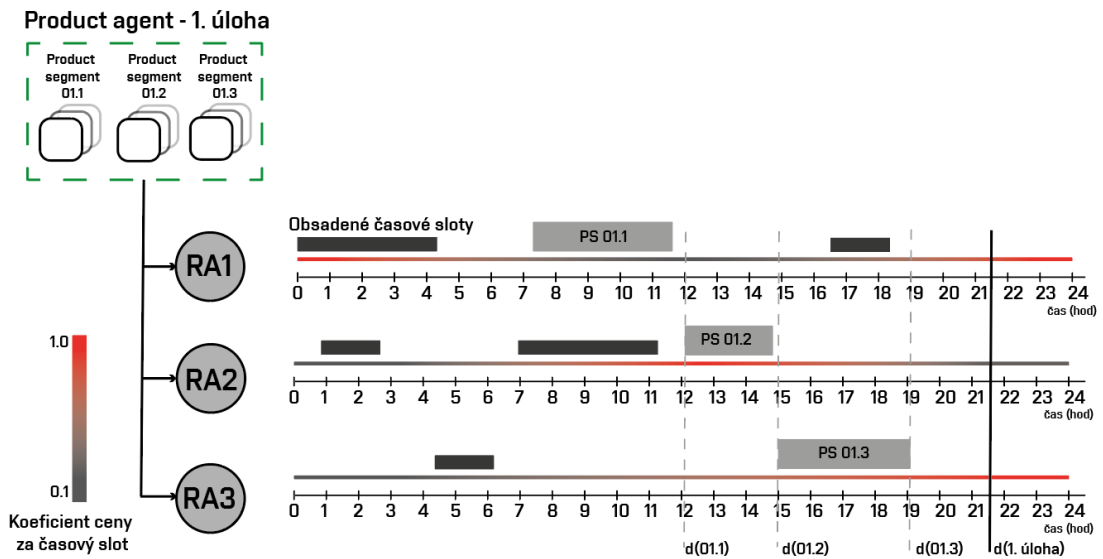
$$T_m^{(s)} < d_m, \quad \delta_T^m = 0;$$

Na prehľadovej schéme funkčnosti takejto heuristiky vid'. Obr. 5.9 je znázornené rozvrhovanie, za ktoré je zodpovedný product agent konkrétnej úlohy z pomyselnej objednávky. Product agent v správach s ponukami obdržal rezervačné fronty dotknutých resource agentov, ktorí tvoria výrobnú cestu produktu.

Cieľom je nájsť kompromis medzi čo najmenšími penalizáciami za predčasné/neskoré vyrobienie a cenou časových slotov konkrétneho výrobného zdroja. Táto cena je určená jej koeficientom, ktorý je graficky znázornený v rezervačných frontách. Cena za konkrétny časový slot sa môže odvíjať od mnohých faktorov. Vo výrobnom podniku sa za takýto faktor dá považovať cena elektrickej energie, cena za ľudskú pracovnú silu (mzdové náklady v noci sú vyššie ako cez deň), cena dopravy, materiálu, atď. V kontexte elektrickej energie môže takéto inovatívne plánovanie slúžiť ako nástroj pri regulácii energetických veličín, napríklad dodržovanie zazmluvneného štvrťhodinového maxima elektrického výkonu. Cena časového slotu výrobného zdroja by reflektovala na prognózovanú potrebu znížiť/zvýšiť odber elektrickej energie, čím by ho systémovo zrovnomerňovala. Koeficient váhy pre veľkosť penalizácie za príliš skoré alebo neskoré vyrobienie produktu nemusí byť konštanta. Je žiadateľné, aby koeficient primerane narastal s odchýlkou, a tým zvyšoval penalizáciu za veľké časové neistoty vo výrobe. V závislosti od implementácie môže koeficient rásť lineárne, prípadne po hyperbolickej funkcii.

Algoritmicky sa popisovaná heuristika dynamického rozvrhovania začína naplánovaním poslednej z pod-úloh *PS 01.3* na časové sloty s minimálnou hodnotou nákladovej funkcie. Začiatok výroby pod-úlohy *PS 01.3* tak vytvorí nový najneskorší čas

vyrobenia pre predposlednú z pod-úloh $d_{(01.2)}$ v zmysle dodržania precedenčných obmedzení. Opäť sa vypočítajú časové sloty s najnižšou hodnotou nákladov rozvrhu a pod-úloha $PS_{01.2}$ je vložená do rezervačnej fronty druhého resource agenta. V tomto kroku je zrejmé, že navrhnutá heuristika vkladania úloh minimalizuje aj čas rozpracovanosti celej úlohy - product agenta, nakoľko penalizuje každé vzdialenie sa od ideálneho priebehu výroby bez prestojov. Rekurzívne prebehne výber časových slotov pre všetky pod-úlohy (produktové segmenty). Nasleduje rezervácia časových slotov v zmysle protokolu CNP s potvrdzovaním.



Obr. 5.9: Ukážka heuristiky vkladania výrobných operácií do rozvrhov resource agentov.

5.5.5 Možné rozšírenie heuristiky vkladania

Táto heuristika môže byť rozšírená o prediktívny rozmer v podobe jednoduchého plánovania v kľzavom časovom horizonte. V dopredu určenom časovom okne by sa prichádzajúce úlohy v podobe product agentov zhromažďovali. Po uplynutí definovaného času by sa product agenti na základe priority, alebo napríklad jednoduchého dispečerskeho pravidla (najskorší termín odovzdania EDD) zoradili a v tomto poradí by pristupovali k alokácii zdrojov. Tým by sa dosiahla istá forma zvýhodnenia pre product agentov, ktorí by v úvode mali väčší výber časových slotov u resource agentov.

Avšak tak ako bolo avizované na začiatku kapitoly, táto forma oponuje decentralizovaným rámcom plánovania a autonómnosti agentov. Takto vzniknutý jeden centrálny člen, ktorý združí úlohy a už pozná ich parametre, môže použiť niektorý z algoritmov centralizovaného plánovania a vytvoriť tak optimálnejší rozvrh v porovnaní s predstavenou heuristikou vkladania.

5.5.6 Dynamické preplánovanie

Organizácia výroby CPPS poskytuje priame prepojenie medzi procesmi rozvrhovania a realitou výrobného procesu. To umožňuje, že procesy zmeny rozvrhu môžu byť riadené

reakčne, tj. hneď po identifikácii abnormálnej udalosti. Po identifikácii takejto udalosti musí prebehnúť rozpoznanie, či závažnosť dopadu na daný rozvrh spadá ešte do tolerancie, alebo nie a teda je nutné iniciovať zmenu zmeny rozvrhu. Reagovať pri každej udalosti by narúšalo stabilitu a efektívnosť rozvrhovania, vid'. kapitola 5.4.2. Na automatizáciu tohto procesu rozhodovania sú nutné definované kritériá - rozsah tolerancií, najmä pre kľúčové parametre, ktoré zabezpečujú prijateľnosť pôvodného harmonogramu bez nutnosti jeho preplánovania.

Formálne vyjadrené hľadáme maximálny interval pre parametre d_m a p_{ms} (čas trvania výroby pod-úlohy m) pre stávajúci optimálny (takmer optimálny) rozvrh π . Pričom platí, že $d_m \in [d_m; \overline{d_m}]$ a $p_{ms} \in [p_{ms}; \overline{p_{ms}}]$. Do výpočtu zahrnieme aj faktor zotrvačnosti β , ktorý vyjadruje váhu prisudzovanú stabilite systému. Vysoký β znamená, že návrh uprednostňuje vysokú zotrvačnosť, čo znamená, že len málo udalostí môže spustiť zmeny harmonogramu.

$$\max|\hat{d} - d| \tag{5-11}$$

$$L_{max}(\pi, \hat{d}) \leq L_{max}(\sigma, \hat{d}) * (1 + \beta)$$

σ - každý rozvrh kde platí $d_m \leq \hat{d}_m \leq \overline{d}_m, \beta \geq 0$

Cieľom je nájsť maximálnu toleranciu parametrov, v tomto prípade d_m a zároveň zabezpečiť, aby rozvrh π zlepšoval pôvodnú účelovú funkciu až po faktor zotrvačnosti β . Voľba β nie je ľubovoľná, ale musí byť úmerná miere zotrvačnosti daného výrobného procesu.

Ak product agent zaznamená udalosť v procese svojej výroby, ktorá naruší sledované parametre a je už mimo tolerancie, iniciuje mechanizmy preplánovania. Nad rozvrhmi dotknutých resource agentov môže použiť uvádzanú metódu - zmena posunom doprava. Pri zlyhaniach celých výrobných zdrojov si product agenti reaktívne svoju výrobu alokujú u iných agentov, ktorí ponúkajú danú službu, prípadne ak oprava nebude v medziach opisovanej tolerancie. Takáto vlastnosť automatickej rekonfigurácie a obnovenia systému po zlyhaní implicitne vychádza z vlastností heterarchickej kooperatívnej architektúry, na ktorej je CPPS založené.

6. IMPLEMENTÁCIA MAS RIADENIA VÝROBY VO VIRTUÁLNEJ REALITE

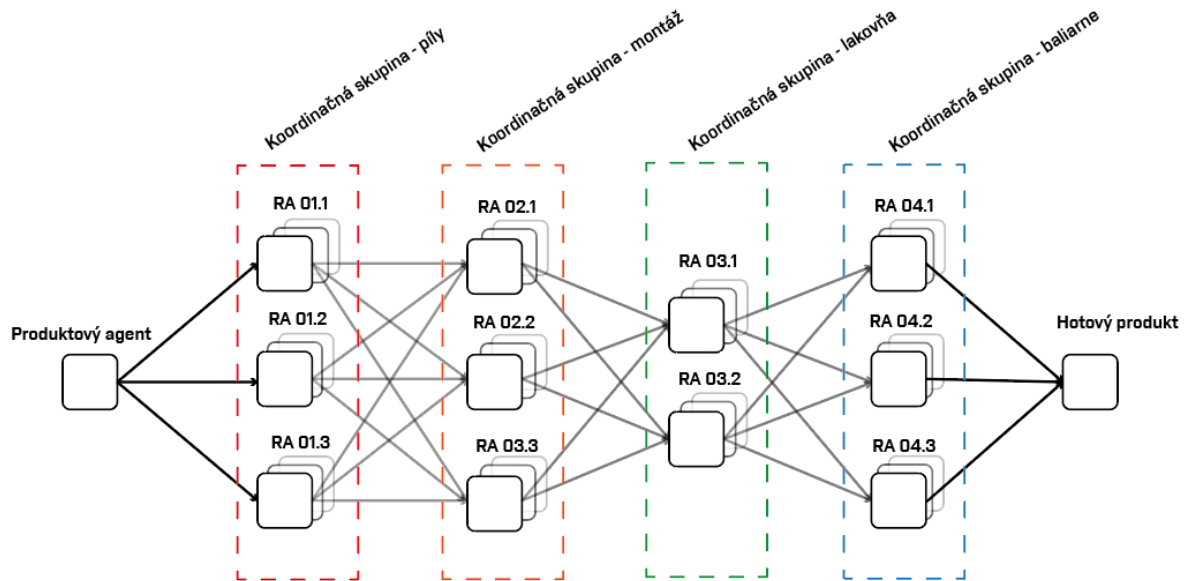
Empirické hodnotenie navrhnutého prístupu dynamického plánovania v CPPS architektúre vyžaduje definovanie konkrétnych inštancií výrobného prostredia a úloh. Použitý simulačný model slúži na zobrazenie reálneho výrobného systému a zachytáva všetky dôležité dynamické vlastnosti vrátane náhodných javov či poruchovosti výrobných zariadení. Simulácia je cieľená najmä na reprodukciu vykonávania výrobnej sekvencie použitím navrhnutých inovatívnych rozvrhovacích metód.

6.1 Simulovaný model výrobného podniku

Pre potreby overenia teoretických výstupov diplomovej práce a navrhnutého algoritmu bol nasimulovaný výrobný podnik na výrobu nábytku. Podnik nemá konsolidovaný stály odber, prevažuje nepredvídateľný tok objednávok bez možnosti akejkoľvek predikcie. Preto uplatňuje výrobnú stratégiu make-to-order, pričom nemá možnosť držať skladové zásoby. Z tohto dôvodu je pri rozvrhovaní výroby kladený dôraz na predstavenú metodiku JIT. Použitá ťažná stratégia výroby umožňuje široké portfólio možných produktov s individuálnymi vlastnosťami, čo vyžaduje flexibilitu možných výrobných trás podľa konkrétnej špecifikácie. Každý produkt má individuálne technologické operácie, ktoré sú viazané precedenčnými obmedzeniami. Uvedené skutočnosti vedú na rozvrhovací problém typu Job-Shop, ktorého riešením je použitie navrhnutých agentovo orientovaných algoritmov. Bližšie predpoklady na použitie MAS v organizácii výroby vid' kapitola 4.3.

Model obsahuje fixnú štruktúru výrobnéj linky, organizáciu a technologické procesy výroby, ktoré sú jasne dopredu definované. Usporiadanie výroby je vyobrazené vid' Obr. 6.1. Ilustrácia približuje rozvrhnutie jednotlivých resource agentov do koordinačných skupín podľa príslušnosti k určitému typu ponúkanej služby. Agenti v týchto skupinách sa od seba odlišujú špecifikáciami ich vlastností vo vzťahu k výrobe. Pri definícii týchto vlastností sme postupovali v zmysle normy ISA 95, ktorá špecifikuje objektové modely výrobného procesu s minimálnym súborom atribútov. Použité objektové modely pri špecifikácii agentov sú model materiálu, model vybavenia a model personálu. Každý z týchto modelov ďalej pozostáva z „class“ a „definition“ vlastností. Všetci resource agenti sú v simulácii zastupovaní svojimi procesnými segmentami, ktoré sú definované pomocou takéhoto objektového modelu. Počet a typ rôznych procesných segmentov určuje, aké technologické operácie, s akými parametrami a v akej kvalite dokáže resource agent vykonať. Zároveň pri každom resource agentovi môžeme uvažovať s definovanou mierou paralelného spracovávania prípadne redundancie. Z obrázka je tiež zrejma úplná prepojenosť resource agentov navzájom, čo je nutný predpoklad flexibilnej výrobnej trasy. V realite by mohla byť tvorená najčastejšie dopravníkmi, alebo iným typom

manipulátorov. V simulácii presun výrobku medzi výrobnými zdrojmi neuvažujem. Ich doplnenie by vďaka škálovateľnosti multiagentnej architektúry znamenalo jednoduché pridanie resource agentov manipulátorov, prípadne ich koordinačnej skupiny.



Obr. 6.1: Rozvrhnutie výrobných zdrojov v simulovanom podniku

Surový materiál pre výrobu dreveného nábytku je k dispozícii v štandardných rozmeroch a kvalite, pričom pre zjednodušenie tieto parametre dostačujú na všetky možné varianty produktov. Vo výrobnom podniku sa nachádzajú pracoviská píly s rôznym zameraním. V simulovanom scenári sa neráta so skladovými kapacitami, preto je surový materiál na výrobu umiestnený podľa špecifikácie na pracoviskách píly. Následne sa nachádza v možnej trase produktu cez výrobu montážna dielňa s rôznym zameraním na konkrétnu ponúkanú technologickú operáciu. Ďalšie pracoviská sú zadefinované ako lakovne s možnou špecializáciou na konkrétne typy lakovania a rôzne farby. Posledný z ponúkaných výrobných zdrojov sú pracoviská baliarní. Svojimi kapacitami zodpovedajú rozsahu veľkostí ponúkaných produktov. Na ilustrácii vid' Obr. 6.2 je príklad definície objektového modelu resource agenta.

```
yield return new ResourceAgent(
new List<ProcessSegment>()
{
    new ProcessSegment("peter[stolár]", "okružná_pila_1[pila]", "dub[drevo],buk[drevo],jaseň[drevo]"),
    new ProcessSegment("jan[stolár]", "okružná_pila_2[pila]", "borovica[drevo]"),
    new ProcessSegment("juraj[stolár]", "priamočiará_pila_1[pila]", "smrek[drevo]"),
}, "Pila_RA01.1");
```

Obr. 6.2: Výpis z kódu konštruktora resource agenta - definícia procesných segmentov.

Po zvolení výrobnej časti simulácie pristúpime k definovaniu samotných vyrábaných produktov. Každý produkt je zastupovaný produktovým agentom, ktorý v multiagentovom systéme zastupuje jeho záujmy - najmä cieľ byť vyrobený. Agent inteligentného výrobku je rovnako ako resource agent rozdelený na segmenty a tie v zmysle objektového modelu definujú požiadavky jednotlivých technologických

operácií. S výhodou bol využitý tento model ďalej v algoritme pre hľadanie a priradzovanie úloh správnym výrobným zdrojom, ktoré spĺňajú požadované vlastnosti. Ďalej produkt agent zapuzdruje parametre potrebné pre rozvrhovanie, najmä čas trvania výrobenia, a čas odovzdania (splatnosti), vid'. Obr. 6.3. Pri základnom overovaní činnosti simulácie boli zvolené produkty, ktoré vyžadujú rôznu trasu cez výrobné zariadenia. Simulovaný podnik vyrába nábytok, preto boli zvolené výrobky: stolička, polička a skriňa. Produktovým agentom zastupujúcim tieto výrobky boli nadefinované simulované parametre technologických operácií a boli vložené do multiagentného prostredia.

```
return new DistributedLib.Agents.SimpleAgents.ProductAgent(
    new List<ProductionSegment>()
    {
        new ProductionSegment(1, "[stolár]", "[pila]", "dub[drevo],buk[drevo]", new Dictionary<string, dynamic>() { { "production_time", 16 } }},
        new ProductionSegment(2, "[montáž]", "[ponk]", "", new Dictionary<string, dynamic>() { { "production_time", 15 } }},
        new ProductionSegment(3, "[lakovník]", "[striekajúca_pištoľ]", "fermež[farba]", new Dictionary<string, dynamic>() { { "production_time", 12 }, { "farba", "červená" } }},
        new ProductionSegment(4, "[baliarne]", "[ručné_balenie]", "folia[material]", new Dictionary<string, dynamic>() { { "production_time", 12 }, { "väzba", "pevna" } }},
    }, dueTime);
```

Obr. 6.3: Výpis z kódu konštruktora product agenta - definícia produktových segmentov.

6.2 Validácia simulácie riadiacej logiky

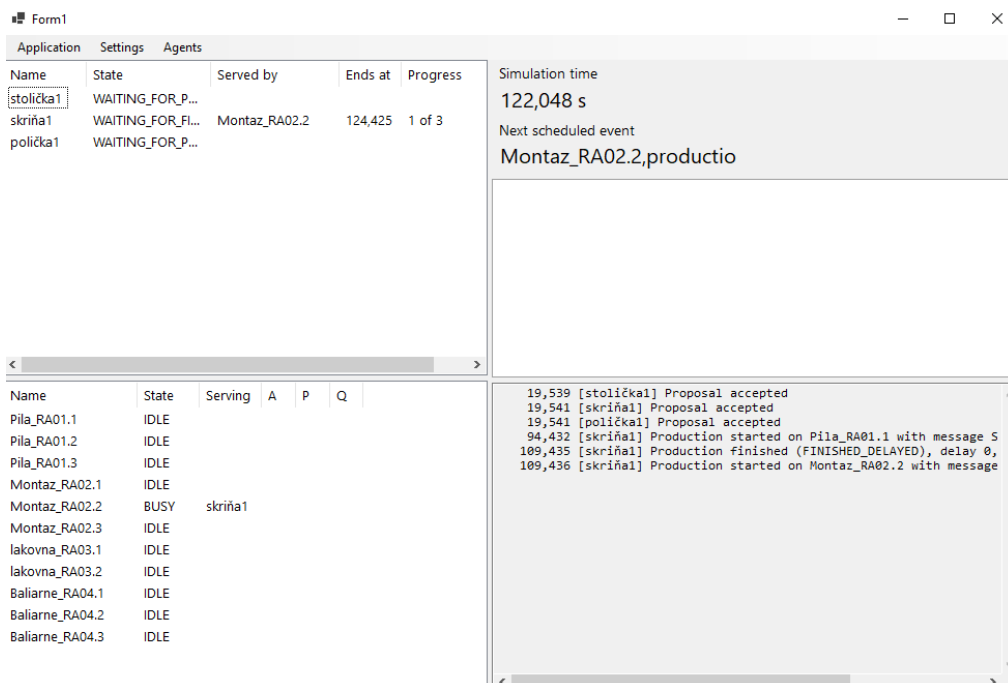
V tejto kapitole stručne opíšem priebeh simulácie. Zvolený bol jednoduchý príklad výroby troch produktov, ktorý zachytáva všetky dôležité vlastnosti predstaveného konceptu organizácie výroby v zmysle MAS. Analogicky je nutné uviesť, že tento model simulovaného výrobného podniku je navrhnutý s veľkou mierou abstrakcie a zanedbania parametrov plynúcich z reálnej výroby, napríklad vôbec neuvažuje pohyb výrobku medzi výrobnými strojmi a z toho plynúce dopravné časy.

Na sledovanie priebehu simulácie bolo vytvorené jednoduché grafické okno vid'. Obr. 6.4. V ľavej časti je zobrazený prehľad vložených product agentov. Pri každom z nich sú pre informovanie užívateľa k dispozícii aj sledované parametre - aktuálny stav, rozpracovanosť produktu, či aktuálny resource agent, na ktorom prebieha výroba spolu s časom ukončenia konkrétneho produktového segmentu. V dolnej časti vľavo vidíme prehľad modelu výrobnéj linky, ktorá sa skladá z resource agentov v zmysle obrázku vid'. Obr. 6.1. Pri každom z nich je uvedené, ktorého z produktových agentov aktuálne obsluhujú, v okne sú tiež uvedené štatistické údaje pre meranie produktivity výrobného zdroja OEE. Informácie o priebehu simulácie sa nachádzajú v pravej časti okna. Dôležitý parameter je čas simulácie, ktorý je možné zvoleným koeficientom zrýchľovať alebo spomaľovať. To je potrebná vlastnosť najmä pre to, aby simulátor mohol slúžiť ako nástroj na ladenie a optimalizáciu organizácie výroby založenej na MAS. Vpravo dole sú logy komunikácie medzi agentami vo výrobnom prostredí.

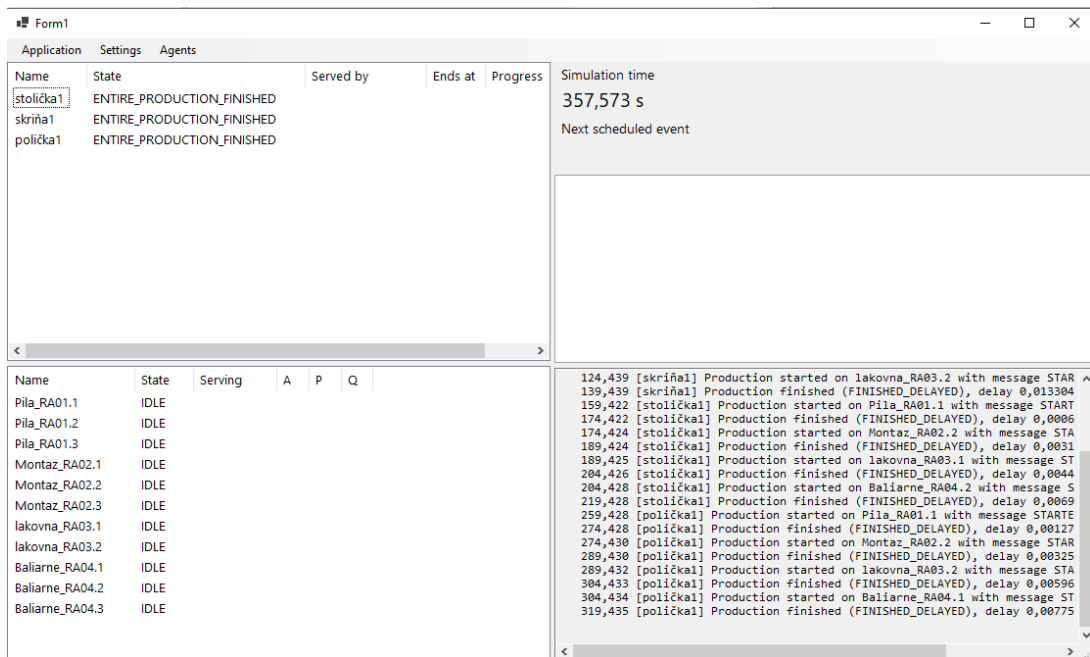
Na uvedenom obrázku Obr. 6.4 je zobrazený začiatok simulovanej úlohy. V logoch komunikácie sú zobrazené prvé správy o akceptácii ponúk od resource agentov. Následne každý product agent vo svojej rézii použitím predstavenej heuristiky vid'. kapitola 5.5 naplánoval a zarezvoval svoju vlastnú výrobu. V okne simulácie Obr. 6.4 je zobrazená začatá výroba prvého z troch product agentov, ktorý mal nastavený najskorší čas

odovzdania - 140s. Z ďalšieho obrázka zo záveru simulácie viď. Obr. 6.5 je prvý product agent úspešne vyrobený a v logoch komunikácie je vidieť, že algoritmus správne naplánoval výrobu jednotlivých produktových segmentov v čase tak, aby minimalizoval nákladovú funkciu rozvrhu. Minimalizoval čas rozpracovanosti a zároveň dodržal všetky princípy výroby JIT. V uvedenom simulačnom príklade bol koeficient ceny za časové sloty konštanta a časy dodania sú pre druhý produkt 220s a pre tretí 320s. V okne simulátora viď. Obr. 6.5 je celá simulovaná úloha ukončená. Všetky produkty boli úspešne vyrobené. Uvedený jednoduchý scenár simulácie výroby poskytuje overenie správnosti navrhutej topológie, organizácie a rozvrhovania výroby v multiagentnom prostredí.

Vytvorený simulátor samozrejme predpokladá komplexnejšie overovanie algoritmov organizácie distribuovanej výroby. Umožňuje simulovať oneskorenie nad niektorým z rozvrhovaných produktových segmentov s príslušným štatistickým rozdelením. Taktiež prípadné vyhodnocovanie OEE parametrov poskytuje možnosť riadiť rovnomerné využitie redundantných výrobných zdrojov - v kompetencii koordinačných skupín. Na prezentovanie takýchto komplexnejších výstupov simulácie distribuovanej výroby bude nutné implementovať pokročilejšie grafické vizualizácie najmä Ganttové diagramy rezervačných front jednotlivých resource agentov. Otvorenosť použitej platformy na implementáciu simulátora umožňuje jeho rozširovanie v rôznych smeroch, podľa aktuálnej potreby testovania.



Obr. 6.4: Grafické okno simulátora - začiatok simulovanej výroby s architektúrou CPPS.



Obr. 6.5: Grafické okno simulátora - koniec simulovanej výroby s architektúrou CPPS.

6.2.1 Použité metódy dynamického rozvrhovania

Pri implementácii simulovanej výroby boli empiricky porovnané dve metódy vkladania nových úloh do rezervačných front výrobných zdrojov.

- Vloženie novej úlohy vždy na poslednú pozíciu rezervačnej fronty (koniec aktuálnej výrobnéj sekvencie) resource agenta.
- Vloženie novej úlohy na najvýhodnejšiu pozíciu použitím predstavenéj heuristiky vkladania - vypočíta cenu vloženia novej úlohy do všetkých možných pozícií rezervačnej fronty, vloží ju na časový slot, ktorý vedie k minimalizácii nákladovej funkcie.

Jednoduchá metóda vkladania úlohy vždy na koniec rezervačnej fronty neposkytuje žiadnu možnosť ovplyvniť čas reálneho vyrobenia produktu, tj. neberie do úvahy najneskorší čas vyrobenia. Dôsledkom sú zvýšené náklady na skladovanie pri príliš skorom vyrobení produktu a taktiež určitá pravdepodobnosť nedodržania termínu. Úspešnosť celého systému rozvrhovania je tak veľmi citlivá na poradie príchodu jednotlivých product agentov do systému alokácie zdrojov. Navyše, ak niektorý z produktov vyžaduje vykonanie technologickej operácie, ktorá je precedenčne závislá v procese výroby viackrát, tak vytvorí nevyužitú časovú okná v rezervačnej fronte výrobného zdroja. Metóda je dostačujúca pre typ výroby Flow shop, ktorá je obmedzená len na sekvenčný tok výroby bez vzájomnej prepojenosti výrobných trás. Pre tento typ výroby je navrhnutá architektúra multiagentnej organizácie výroby nevhodná.

Princíp fungovania, predpoklady použitia a výhody navrhnutéj heuristiky vkladania úloh sú už podrobne rozobraté vid'. kapitola 5.5.

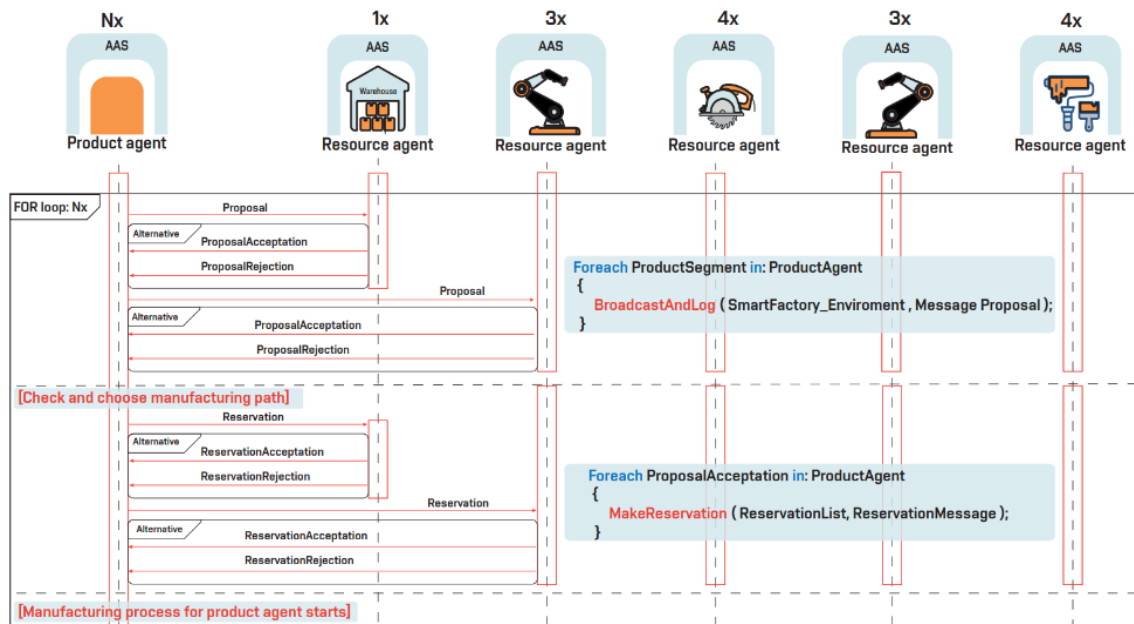
6.3 Opis implementácie simulačného prostredia

Programové prostredie C# a NET framework pre tvorbu modelu a algoritmu rozvrhovania bolo zvolené z dôvodu nezávislosti na licenčných obmedzeniach programov tretích strán a platforme operačného systému. Zámernom tohto rozhodnutia bolo implementačne sa priblížiť aktuálnym riešeniam a tým zjednodušiť možnú integráciu podobného algoritmu napríklad do funkčných blokov a knižníc systémov priemyselnej informatiky.

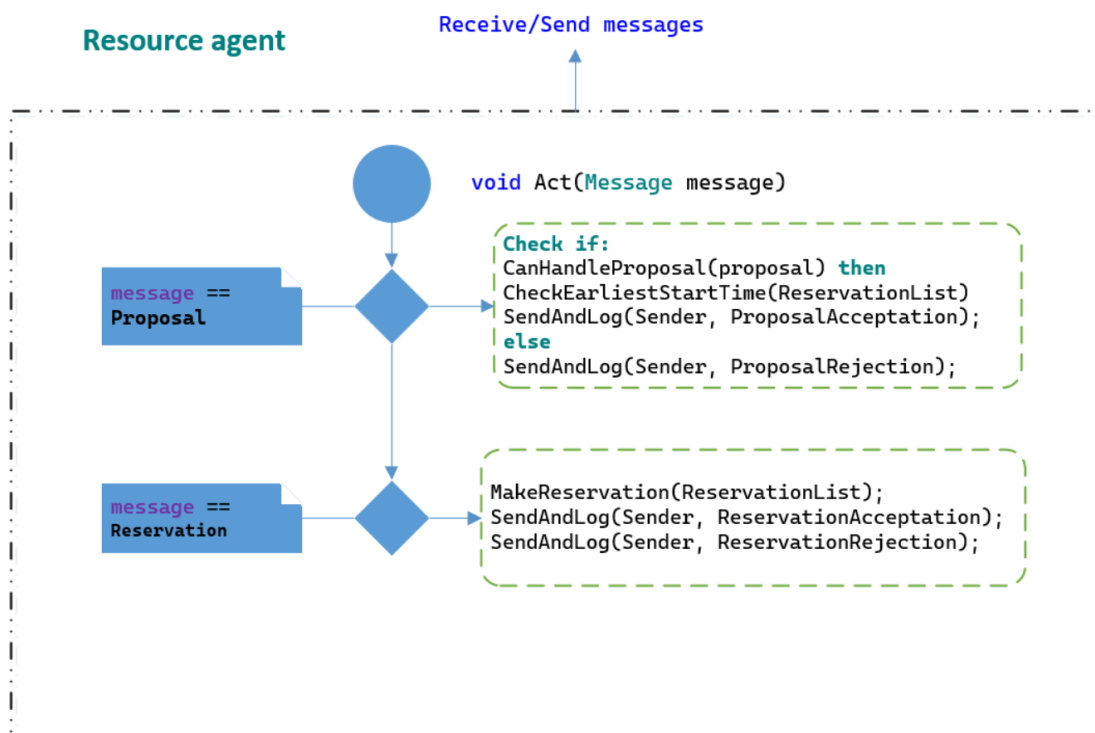
V nasledujúcich statiach stručne uvádzam štruktúru implementácie softvéru simulátora v podobe UML sekvenčných, vývojových a class diagramov. Detailnejšia analýza a popis naprogramovaného softvéru nie sú obsahom tejto práce. V simulátore je použitý CNP protokol s potvrdzovaním, takže resource agenti môžu odpovedať na výzvy (posielať ponuky) a až po potvrdení zo strany product agenta alokovať časové sloty. Na obrázkoch vid'. Obr. 6.6, Obr. 6.7, Obr. 6.8,

Obr. 6.9. sa detailnejšie uvádza znázornená štruktúra interakcií medzi agentami pomocou CNP protokolu. Na ďalších obrázkoch, vid'. Obr. 6.10 - 6.13 sú zobrazené class-diagramy celého programového vybavenia uvedeného simulátora distribuovanej výroby.

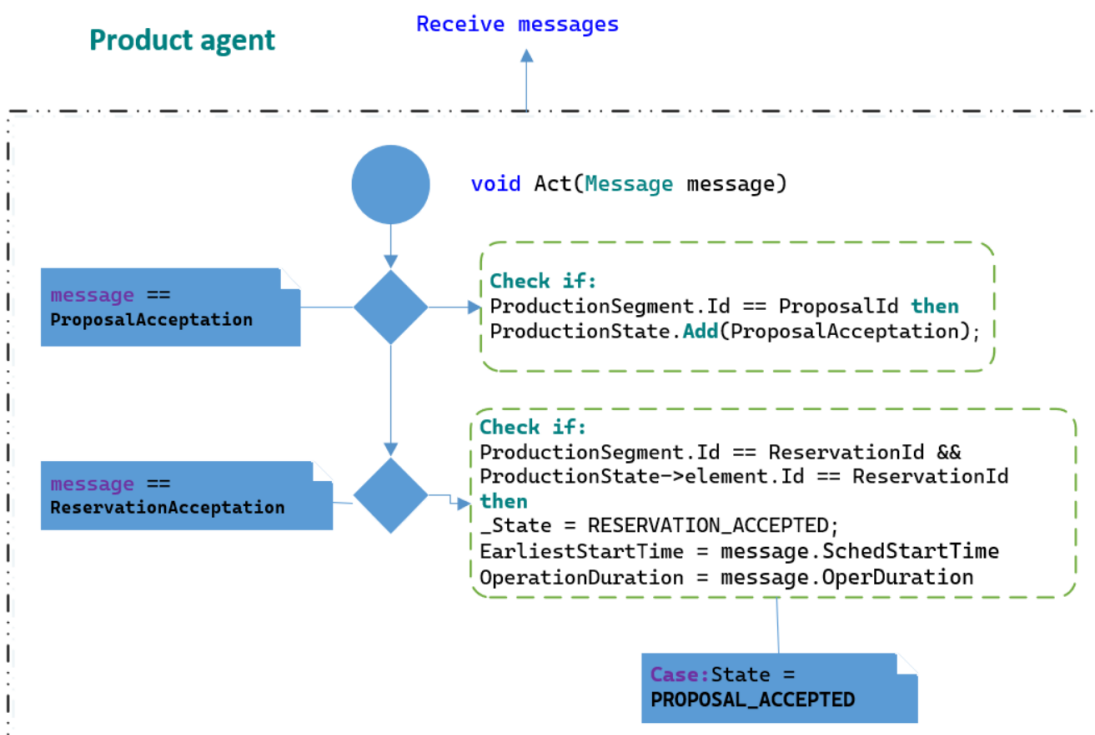
Celú implementáciu simulátora navrhol a odborne zastrešoval vedúci diplomovej práce Ing. Václav Kaczmarczyk, Ph.D



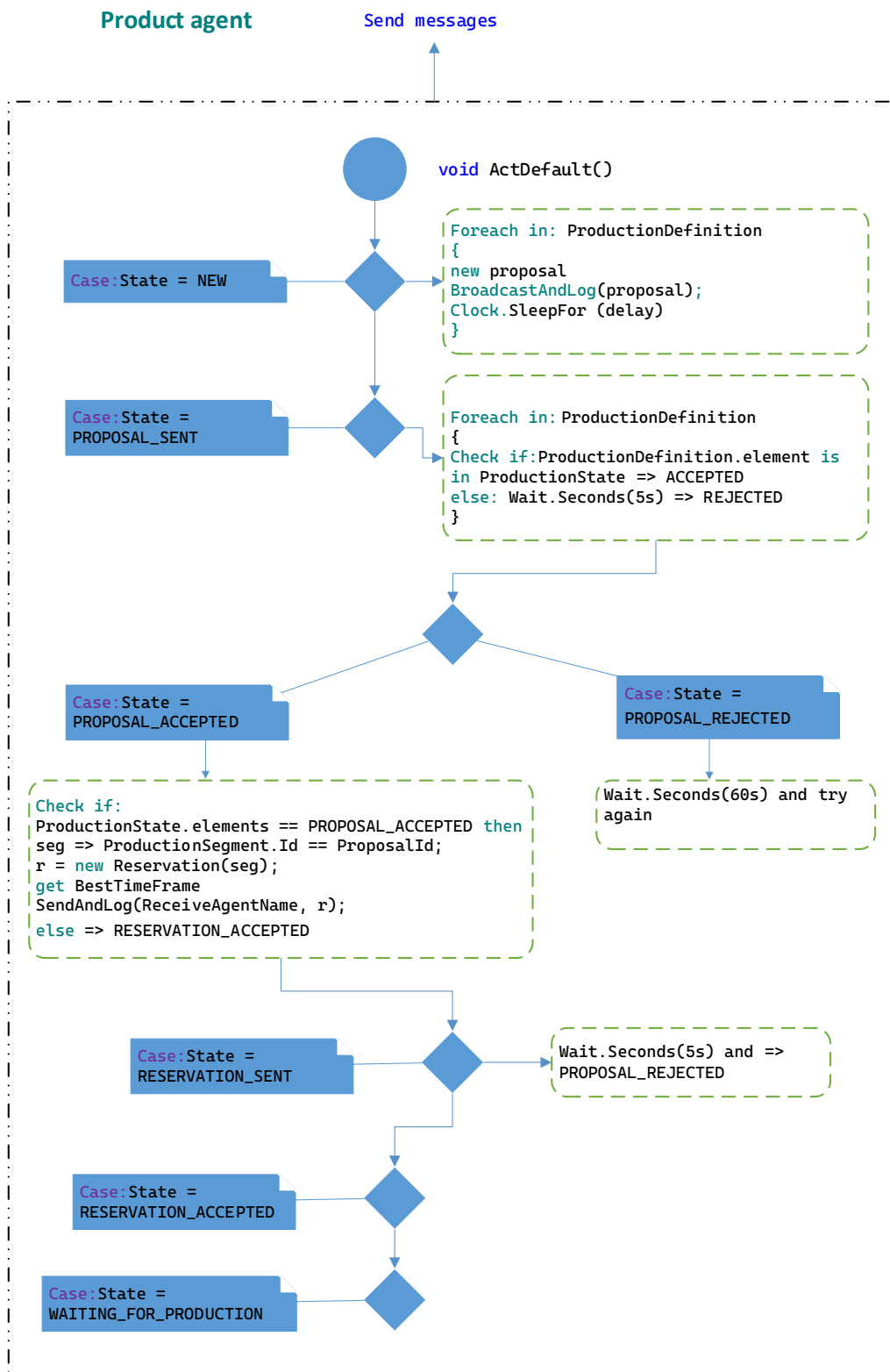
Obr. 6.6: Priama komunikácia medzi agentami (CNP - Contract Net Protocol).



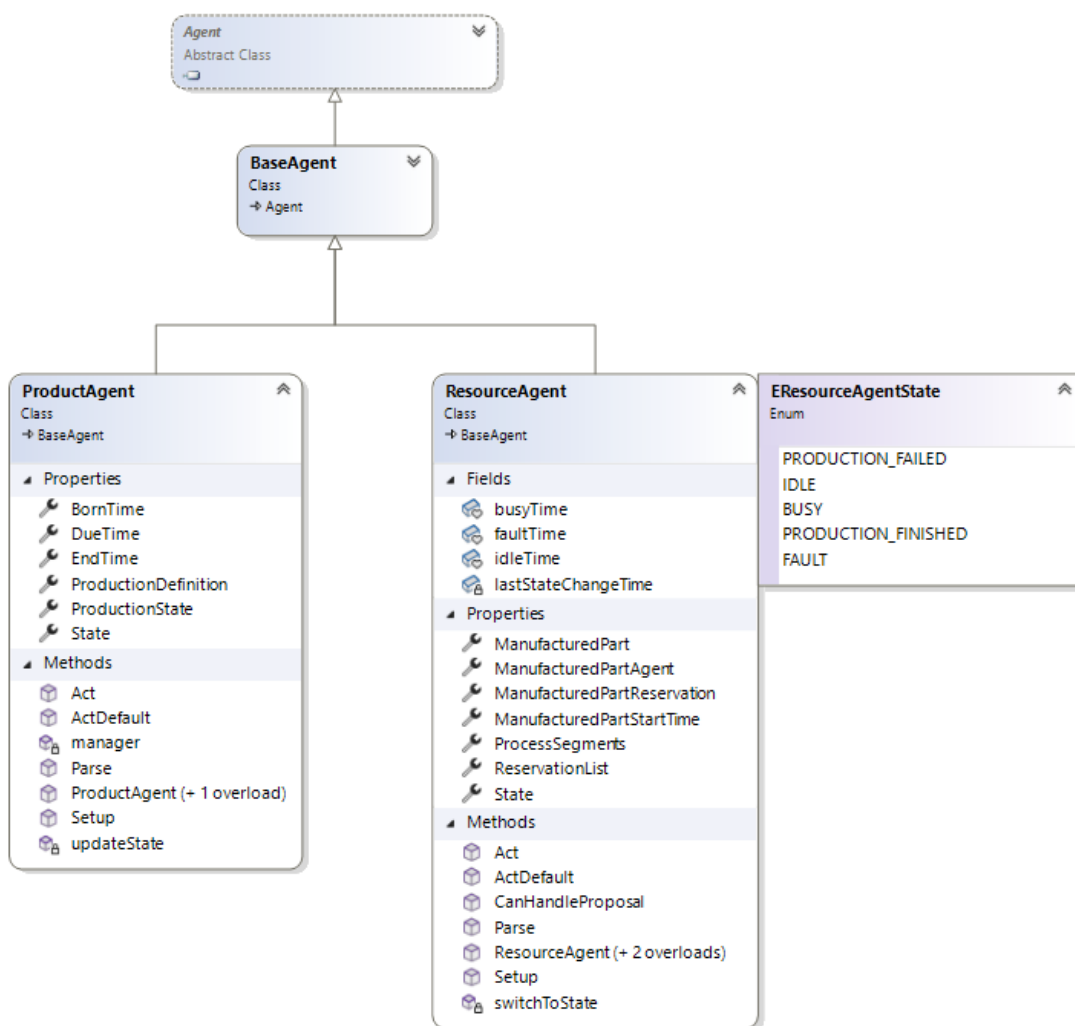
Obr. 6.7: Prehľadový diagram priebehu rezervácie na strane resource agenta, prijímanie správ.



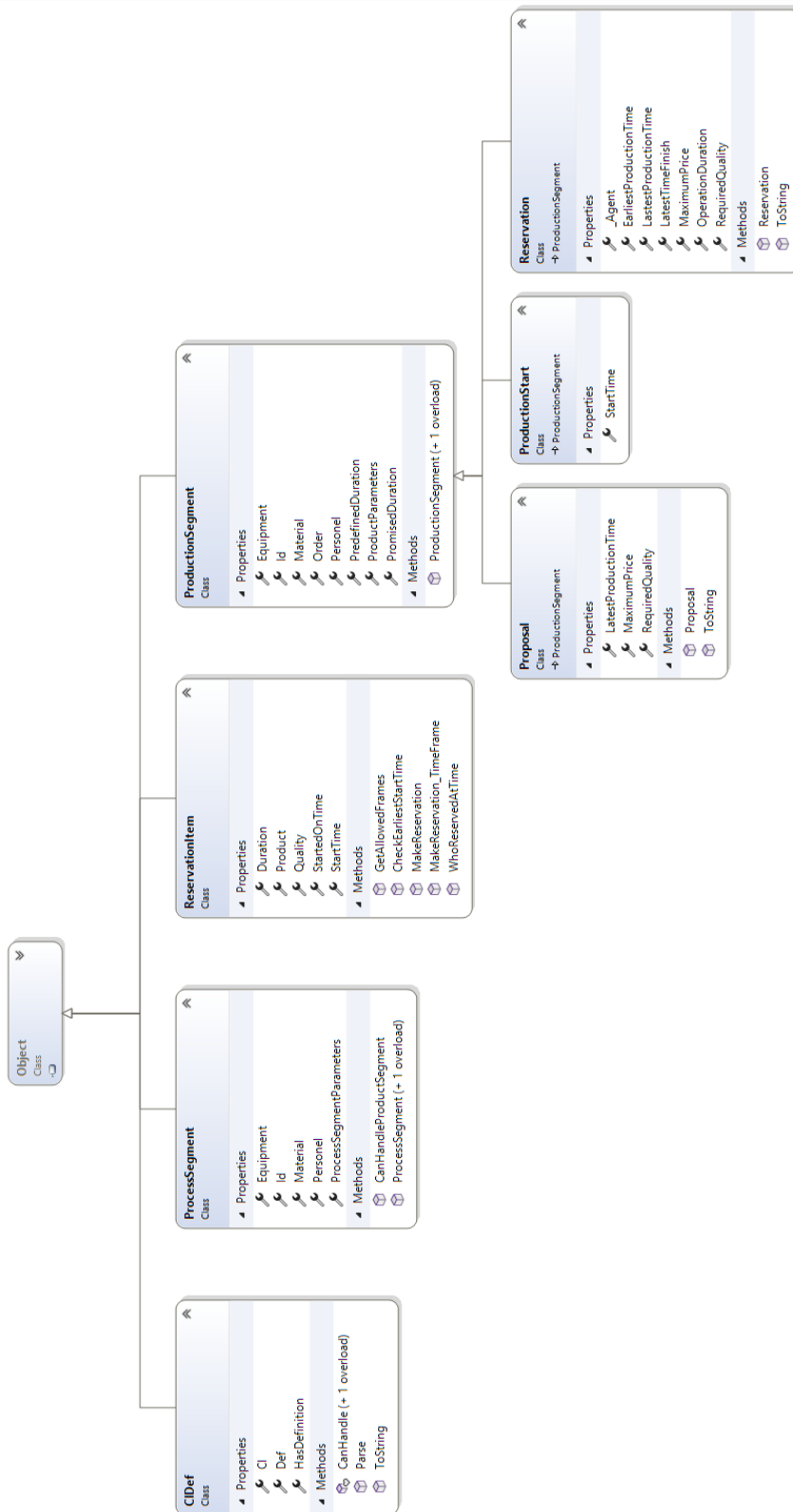
Obr. 6.8: Prehľadový diagram priebehu rezervácie na strane product agenta - prijímanie správ.



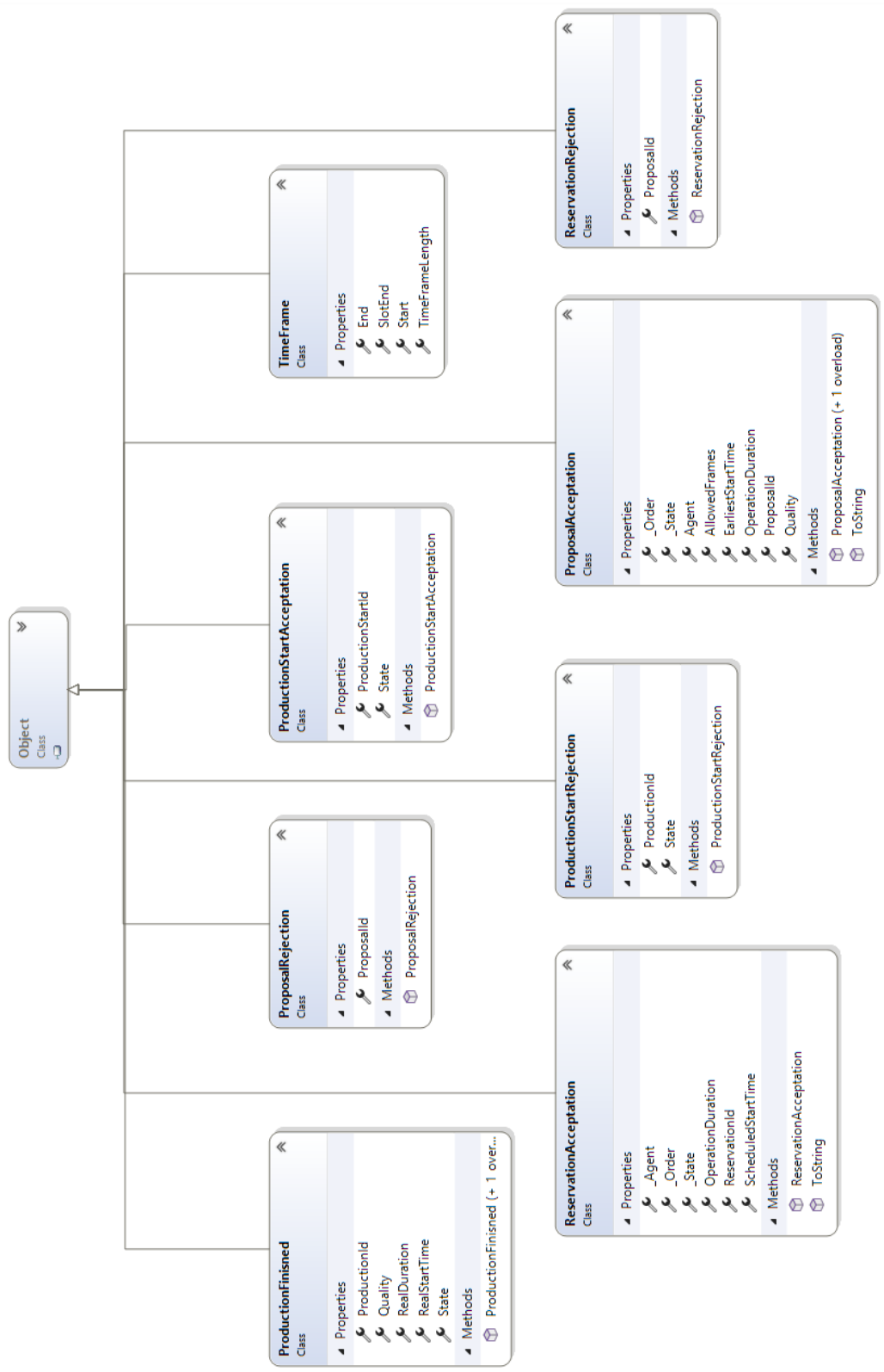
Obr. 6.9: Prehľadový diagram priebehu rezervácie na strane product agenta - odosielanie správ.



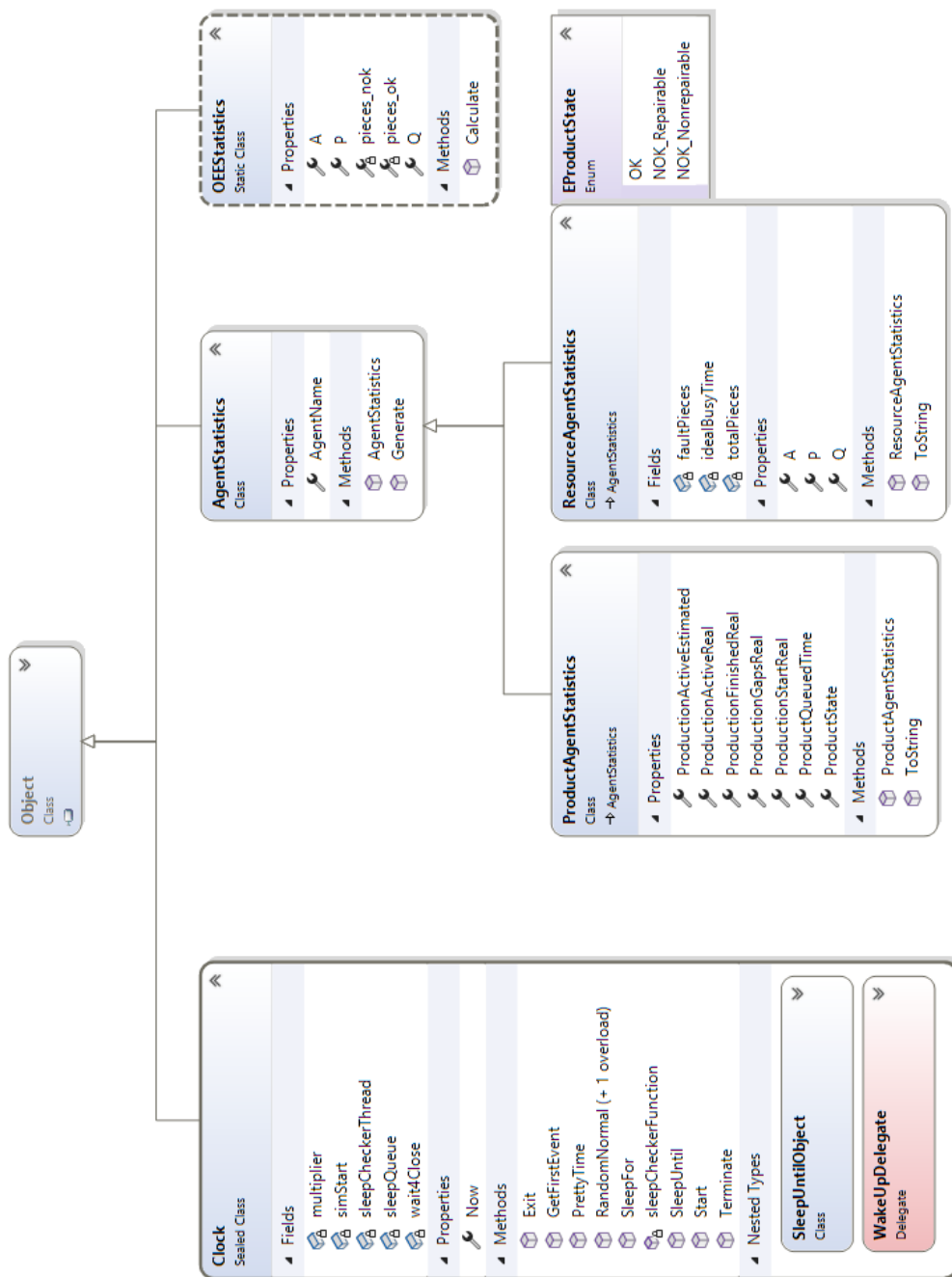
Obr. 6.10: Class-diagram - simulátor distribuuovanej výroby (1).



Obr. 6.11 Class-diagram - simulátor distribovanej výroby (2).



Obr. 6.12: Class-diagram - simulátor distribovanej výroby (3).



Obr. 6.13: Class-diagram - simulátor distribovanej výroby (4).

7. ZÁVER

Diskutovaná problematika diplomovej práce je inšpirovaná koncepciou Priemysel 4.0 najmä v kontexte zavádzania inovatívnych technológií a koncepčných zmien na zlepšovanie organizácie rozhodovacích procesov vo výrobe.

V prvej časti diplomovej práce sú uvádzané teoretické informácie a zdroje, týkajúce sa oblasti plánovania, riadenia a organizácie výrobného procesu s akcentom na analýzu súčasného stavu v danej problematike plánovacích systémov. V ďalších statiach teoretickej časti práce sa moja pozornosť upriamuje na popis multiagentovej štruktúry ako moderného a efektívneho nástroja v oblasti riadenia a organizácie výrobného sféry. Poskytujem terminológiu a teóriu na realizáciu organizácií v multiagentových systémoch.

Hlavným predmetom riešenia tejto práce je návrh a implementácia multiagentového systému pre riadenie organizácie výroby, ktorý vychádza z bázy teoretických znalostí obsiahnutých v jednotlivých kapitolách práce. Navrhnutá architektúra, vzťahy a topológia autonómnych agentov v istej miere abstrakcie simuluje distribuovanú inteligenciu naprieč výrobným podnikom.

Ďalšia časť predloženej práce sa venuje algoritmom plánovania a rozvrhovania úloh, ktoré vychádzajú z decentralizovaného rámca paradigmy multiagentných systémov. Implicitne poskytujú riešenie dynamickej povahy reálneho plánovania výroby. Bol navrhnutý algoritmus dynamického rozvrhovania, ktorý poskytuje heuristiku vkladania úloh do rozvrhov inteligentných výrobných zdrojov. Jeho prínos oproti jednoduchým dispečerským pravidlám spočíva v možnosti adaptívne meniť cenu jednotlivých časových slotov výrobného zdroja spolu s dodržaním JIT princípov štíhlej výroby.

Navrhnutý prístup k rozvrhovaniu a organizácii výroby bol validovaný na nasimulovanom výrobnom podniku s nábytkom. Pre ilustráciu funkčnosti boli vyšpecifikované produkty aj výrobné operácie, nad ktorými je definovaná multiagentná architektúra CPPS. Pre účel simulácie bolo v súlade s teoretickými výstupmi práce navrhnuté a naprogramované simulačné prostredie. Jednou z hlavných pridaných hodnôt diplomovej práce je implementácia komplexného simulátora v jazyku C#. Uvádzané individuálne riešenie simulácie nezávislej na používanej platforme operačného systému, alebo programov tretích strán, poskytuje východiskový rámec pre ďalší výskumnú činnosť problematiky MAS v organizácií výroby.

Iný vedecký problém, ktorý z problematiky vyplýva, spočíva v identifikácii a prepojení už existujúcich sústav technológií vo výrobe tak, aby výsledkom bola realizácia popisovaného hetararchického systému. V teoretickej rovine z multiagentového prístupu vo výrobe rezultuje autonómnosť, modularita a otvorenosť výrobného procesu. V súčasnom vysoko konkurenčnom prostredí je reálny predpoklad, že uvádzané inovatívne riadenie výroby vedie k výraznej konkurenčnej výhode.

LITERATÚRA

- [1] GRZNÁR, Miroslav, Ľuboslav SZABO a Jaroslav JANÍK. *Teória a prax podnikového plánovania v podmienkach globalizácie*. 1. vyd. Bratislava: Ekonóm, 2009. ISBN 9788022526999.
- [2] MAJTÁN, Miroslav, Miroslav GRZNÁR, Anna KACHAŇÁKOVÁ, Štefan SLÁVIK, Ľuboslav SZABO, Miroslava SZARKOVÁ a Elena THOMASOVÁ. *Manažment*. Bratislava: Sprint vfra, 2003, 429 s. Nová ekonómia. ISBN 80-89085-17-2.
- [3] MANLIG, František a František KOBLASA. *Plánování a rozvrhování výroby (vybrané kapitoly)* [online]. 52 [cit. 2022-10-23]. Dostupné z: https://www.unipranet.zcu.cz/doc/15_Planovani%20a%20rozvrhovani%20vyroby%20Manlig%20&%20Koblasa%20TUL.pdf
- [4] PARALIČ, Ján. *ROZVRHOVANIE A LOGISTIKA* [online]. Košice: Equilibria, s.r.o, 2010 [cit. 2022-10-24]. Dostupné z: <https://jan.paralic.website.tuke.sk/ral.html>. Technická univerzita v Košiciach (TUKE).
- [5] SODOMKA, Peter. Pokročilé plánování a řízení výroby: Velký přehled plánovacích a řídicích metod v informačních systémech. In: *SystemOnline* [online]. [cit. 2022-10-23]. Dostupné z: <https://www.systemonline.cz/rizeni-vyroby/pokrocile-planovani-a-rizeni-vyroby.htm>
- [6] KOBLASA, František. *Uplatnění heuristických optimalizačních metod v oblasti rozvrhování strojírenské výroby středních a malých podniků* [online]. Liberec, 2013 [cit. 2022-12-23]. Dostupné z: <https://knihovna-opac.tul.cz/records/4dd2f0be-e47f-474b-b219-adf8b39fcd4b>. Dizertačná práca. Technická univerzita v Liberci. Vedoucí práce František Manlig.
- [7] ŠULOVÁ, Dagmar. *Metody plánování a řízení výroby v podnikových informačních systémech a jejich uplatnění při řízení výrobního procesu* [online]. Zlín, 2009 [cit. 2022-12-23]. Dostupné z: <https://digilib.k.utb.cz/handle/10563/7915?locale-attribute=cs>. Dizertačná práca. Univerzita Tomáše Bati ve Zlíně Fakulta managementu a ekonomiky. Vedoucí práce Petr Sodomka.
- [8] PINEDO, Michael L. *Scheduling Theory, Algorithms and Systems* [online]. Fifth Edition. NYU Stern School of Business New York, NY, USA [cit. 2022-11-13]. ISBN 978-3-319-26580-3.
- [9] MILOSLAV, STIBOR a Kutil MICHAL. *TORSCHES SCHEDULING TOOLBOX: LIST SCHEDULING: 7 th International Scientific – Technical*

- Conference* [online]. 1-8 [cit. 2022-11-13]. Dostupné z:
<https://rtime.felk.cvut.cz/scheduling-toolbox/download/articles/2006/process-control-list-schedulig.pdf>
- [10] KACZMARCZYK, Václav. *Optimální metody řízení energetické spotřeby budov* [online]. Brno, 2015 [cit. 2022-10-25]. Dostupné z:
<http://hdl.handle.net/11012/42679>. Disertační práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav automatizace a měřicí techniky. Vedoucí práce František Zezulka.
- [11] PARALIČ, Ján. *Riešenie úloh rozvrhovania logickým programovaním ohraničení* [online]. Košice, 1997 [cit. 2022-12-19]. Dostupné z:
<https://jan.paralic.website.tuke.sk/knihy/RozvrhovanieCLP.pdf>. Dizertačná práca. Fakulta elektrotechniky a informatiky Technická univerzita v Košiciach.
- [12] BAŠTINEC, Jaromír. *Statistika, stochastické procesy, operační výzkum* [online]. Ústav matematiky FEKT VUT v Brně, 2014 [cit. 2022-12-19]. Dostupné z:
http://matika.umat.feec.vutbr.cz/inovace/texty/DMA1/CZ/DMA1_plna_verze_CZ.pdf
- [13] BRUCKER, Peter. *Scheduling Algorithms* [online]. Fifth Edition. Osnabruck, Nemecko, 2006 [cit. 2022-11-20]. ISBN 978-3-540-69515-8. Dostupné z: <https://archive.org/details/schedulingalgori0000bruc>
- [14] SZABÓ, Adam. *Využitie branch and bound prístupu pre parametrické intervalové lineárne sústavy* [online]. Praha, 2018 [cit. 2022-12-22]. Dostupné z:
<https://dspace.cuni.cz/bitstream/handle/20.500.11956/103478/130239471.pdf?sequence=1&isAllowed=y>. Bakalárska práca. Univerzita Karlova Katedra aplikované matematiky. Vedoucí práce Jaroslav Horáček.
- [15] KVASNIČKA, V., J. POSPÍCHAL a P. TIŇO. *Evolučné algoritmy* [online]. Bratislava, 2000 [cit. 2022-12-22]. Dostupné z:
http://www2.fiit.stuba.sk/~kvasnicka/Modelovanie_simluacia/Lecture_01/chapter_01B.pdf. Slovenská technická univerzita v Bratislave.
- [16] ŠVANČARA, Juraj. *SIMULAČNÁ ANALÝZA MNOHOVÝROBKOVÉHO VÝROBNÉHO SYSTÉMU* [online]. Bratislava, 2012 [cit. 2022-11-20]. Dostupné z:
<https://opac.crzp.sk/?fn=detailBiblioForm&sid=7ECB8D2670C8FCA5B8DFD6521760>. Dizertačná práca. Slovenská technická univerzita v Bratislave. Vedoucí práce Zdenka Králová.

- [17] PARALIČ, Ján. *Rozvrhovanie a logistika* [online]. In: . [cit. 2022-11-20]. Dostupné z: <https://jan.paralic.website.tuke.sk/ral.html>
- [18] LUKÁŠ, Ďurica. *Multiagentový logistický systém s implementáciou vo virtuálnej realite* [online]. 2016 [cit. 2022-10-25]. Dostupné z: <https://opac.crzp.sk/?fn=detailBiblioForm&sid=F2FAE62FE51E8196D0A7712DB6AB>. Dizertačná práca. Žilinská univerzita v Žiline 02000 02150. Vedoucí práce Bubeník, Peter.
- [19] DILTS, D.M., N.P. BOYD a H.H. WHORMS. The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems* [online]. **1991**, 79-93 [cit. 2022-11-07]. Dostupné z: doi:0278-6125
- [20] DUFFIE, Neil A. *Synthesis of Heterarchical manufacturing systems: Computers in Industry* [online]. 167-174 [cit. 2022-11-07]. Dostupné z: doi:0166-3615
- [21] DUFFIE, Neil A. *An Approach to the Design of Distributed Machinery Control Systems* [online]. IEEE, 435-442 [cit. 2022-11-07]. Dostupné z: doi:10.1109/TIA.1982.4504105
- [22] CRUZ SALAZAR, Luis a Birgit VOGEL-HEUSER. *Industry 4.0 and Cyber Physical Production Systems (CPPS)* [online]. In: . [cit. 2023-01-19]. Dostupné z: <https://www.researchgate.net/project/Industry-40-and-Cyber-Physical-Production-Systems-CPPS>
- [23] KUBÍK, Aleš. *Inteligentní agenty: Tvorba aplikačného software na bázi multiagentových systémů*. 1rd edition. Brno: Computer Press, Computer Press. ISBN 8025103234.
- [24] PRYMEK, Miroslav a Aleš HORÁK. *Multi-Agent Framework for Power Systems Simulation and Monitoring* [online]. [cit. 2023-02-03]. Dostupné z: https://nlp.fi.muni.cz/publications/icict2005_hales_prymek/icict2005_xprymek_hales.pdf
- [25] FININ, Tim a Richard FRITZSON. *KQML as an Agent Communication Language: Proceedings of the third international conference on Information and knowledge management* [online]. [cit. 2023-01-31]. Dostupné z: <https://dl.acm.org/doi/abs/10.1145/191246.191322>
- [26] JÜRGEN MÜLLER, H. *Towards agent systems engineering* [online]. [cit. 2023-04-16]. Dostupné z: doi:10.1016/S0169-023X(97)00013-X
- [27] WOOLDRIDGE, M. a N.R. JENNINGS. *Software engineering with agents: pitfalls and pratfalls* [online]. [cit. 2023-04-29]. Dostupné z: doi:10.1109/4236.769419

- [28] SHEN, Weiming. *Distributed Manufacturing Scheduling Using Intelligent Agents* [online]. 1-7 [cit. 2023-04-29]. Dostupné z: doi:10.1109/5254.988492
- [29] KRAUS, Sarit. *Negotiation and cooperation in multi-agent environments* [online]. 1-20 [cit. 2023-04-29]. Dostupné z: [https://doi.org/10.1016/S0004-3702\(97\)00025-8](https://doi.org/10.1016/S0004-3702(97)00025-8)
- [30] ROSSIT, Daniel, Fernando TOHMÉ a Mariano FRUTOS. Production planning and scheduling in Cyber-Physical Production Systems. *International Journal of Computer Integrated Manufacturing* [online]. 385-395 [cit. 2023-05-05]. Dostupné z: doi:10.1080/0951192X.2019.1605199
- [31] WANG, Lihui a Xi WANG. *Cloud-Based Cyber-Physical Systems in Manufacturing* [online]. 1rd edition. Springer Cham, 2018 [cit. 2023-05-05]. ISBN 978-3-319-67692-0. Dostupné z: <https://link.springer.com/book/10.1007/978-3-319-67693-7>
- [32] SEITZ, Matthias, Felix GEHLHOF, Luis CRUZ SALAZAR, Alexander FAY a Birgit VOGEL-HEUSER. Automation platform independent multi-agent system for robust networks of production resources in industry 4.0. *Journal of Intelligent Manufacturing* [online]. [cit. 2022-12-27]. Dostupné z: <https://link.springer.com/article/10.1007/s10845-021-01759-2>
- [33] PAOLUCCI, Massimo a Robert SACILE. *Agent-Based Manufacturing and Control Systems: New Agile Manufacturing Solutions for Achieving Peak Performance* [online]. [cit. 2023-04-16]. Dostupné z: doi:<https://doi.org/10.1201/9780203492666>
- [34] HEINRICH, Steffen a Holger DÜRR. *An Agent-based Manufacturing Management System for Production and Logistics within Cross-Company Regional and National Production Networks* [online]. 01-08 [cit. 2023-04-16]. Dostupné z: https://www.researchgate.net/publication/220484025_An_Agent-based_Manufacturing_Management_System_for_Production_and_Logistics_within_Cross-Company_Regional_and_National_Production_Networks
- [35] BURIAN, Pavel. Multiagentné systémy príčinou vyššej konkurencieschopnosti výrobných a logistických podnikových procesov (2). *AT&P journal* [online]. Bratislava, 2006(12) [cit. 2023-05-06]. Dostupné z: <https://www.atpjournals.sk/buxus/docs/atp-2006-12-65.pdf>

- [36] NEUMAN, Clifford. Scale in Distributed Systems. *Readings in Distributed Computing Systems: IEEE Computer Society Press* [online]. Los Alamitos, 463–489 [cit. 2023-05-11]. Dostupné z: <https://www.semanticscholar.org/paper/Scale-in-Distributed-Systems-Neuman/60bea4373bf486faedfa43ac12ac90bed26c87f5>
- [37] SCHILLO, Michael. *Multiagent Robustness: Autonomy vs. Organisation* [online]. 2004 [cit. 2023-05-07]. Dostupné z: https://www.researchgate.net/publication/235424370_Multiagent_Robustness_Autonomy_vs_Organisation. Dizertačná práca. Naturwissenschaftlich-Technischen Fakultaten der Universität des Saarlandes. Vedoucí práce Hans-Peter Lenhof.
- [38] FRANKOVIČ, Baltazár, Ivana BUDINSKÁ a Dang T-TUNG. AGENTS COALITION IN COORDINATION PROCESS. *IFAC Proceedings Volumes* [online]. **2002**(35), 259-264 [cit. 2023-04-12]. Dostupné z: doi:10.3182/20020721-6-ES-1901.00045
- [39] ERNANI VIEIRA, Guilherme, Jeffrey HERRMANN a Edward LIN. *Rescheduling Manufacturing Systems: A Framework of Strategies, Policies, and Methods* [online]. [cit. 2023-05-03]. Dostupné z: doi:10.1023/A:1022235519958
- [40] GORGULU, Elif a M.Selim AKTURK. Match-up scheduling under a machine breakdown. *European Journal of Operational Research* [online]. **1999**(1121), 81-97 [cit. 2023-05-04]. ISSN 0377-2217. Dostupné z: [https://doi.org/10.1016/S0377-2217\(97\)00396-2](https://doi.org/10.1016/S0377-2217(97)00396-2)
- [41] FANG, Jian a Yugeng XI. A rolling horizon job shop rescheduling strategy in the dynamic environment. *Int J Adv Manuf Technol* [online]. 227–232 [cit. 2023-05-04]. Dostupné z: <https://doi.org/10.1007/BF01305874>
- [42] BYEON, Eui-Seok, S. WU a Robert STORER. *Decomposition Heuristics for Robust Job-Shop Scheduling* [online]. [cit. 2023-05-04]. Dostupné z: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=681248>
- [43] KOVALENKO, Ilya, Daria RYASHENTSEVA, Birgit VOGEL-HEUSER, Dawn TILBURY a Kira BARTON. Dynamic Resource Task Negotiation to Enable Product Agent Exploration in Multi-Agent Manufacturing Systems. *IEEE ROBOTICS AND AUTOMATION LETTERS* [online]. **32019**(4), 2854-2861 [cit. 2023-05-12]. Dostupné z: doi:10.1109/LRA.2019.2921947

ZOZNAM PRÍLOH

PRÍLOHA A: Simulačný model – projekt MAS_production.zip