



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMATIZACE A INFORMATIKY

INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

Návrh řízení kombinační váhy za pomoci řídicího systému SIMOTION

Control design of multihead weigher in SIMOTION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Alexandr Capovsky

VEDOUcí PRÁCE

SUPERVISOR

Ing. Tomáš Marada, Ph.D.

BRNO 2019

Zadání diplomové práce

Ústav: Ústav automatizace a informatiky
Student: **Bc. Alexandr Capovsky**
Studijní program: Strojní inženýrství
Studijní obor: Aplikovaná informatika a řízení
Vedoucí práce: **Ing. Tomáš Marada, Ph.D.**
Akademický rok: 2018/19

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Návrh řízení kombinační váhy za pomoci řídicího systému SIMOTION

Stručná charakteristika problematiky úkolu:

Návrh řídicího algoritmu kombinační váhy. Práce popisuje funkcionalitu kombinačních vah a aplikuje různé strategie pro její optimální řízení. Cílem je realizace s využitím objektově orientovaného programování umožňující vytvoření softwarové knihovny pro strojírenskou praxi.

Cíle diplomové práce:

Prostudujte implementaci objektově orientovaného programování (OOP) v rámci systému SIMOTION. Navrhněte a implementujte vhodný řídicí algoritmus kombinačních vah s využitím OOP. Navržený systém otestujte.

Seznam doporučené literatury:

BRAUN, Michael a HORN, Wolfgang Object-Oriented Programming with SIMOTION: Fundamentals, Program Examples and Software Concepts according to IEC 61131-3. John Wiley, 2017. ISBN 978--89578-456-9.

BERETTA, Alessia. Optimal Targets Setup of a Multihead Weighing Machine. Milano, Italy, 2016. Disertační práce. Politecnico di Milano. Vedoucí práce Dr. Quirico Semeraro.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2018/19

V Brně, dne

L. S.

doc. Ing. Radomil Matoušek, Ph.D.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

ABSTRAKT

Tato diplomová práce se zabývá návrhem algoritmu pro řízení kombinační váhy za splnění předem definovaných podmínek. Dále je sestaven simulační model s proměnnými parametry a následně je otestován, jak reaguje na změny, které mohou za provozu nastat.

ABSTRACT

This diploma thesis deals with designing an algorithm for controlling multihead weigher. Also, this diploma thesis contains simulation model to test how the multihead weigher behave in different situation.

KLÍČOVÁ SLOVA

SIMOTION, kombinační váha, návrh řízení balicího stroje

KEYWORDS

SIMOTION, multihead weigher, packing machine design control

BIBLIOGRAFICKÁ CITACE

CAPOVSKY, Alexander. *Návrh řízení kombinační váhy za pomoci řídicího systému SIMOTION*, Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automatizace a informatiky.

PODĚKOVÁNÍ

Tímto bych rád poděkoval svému vedoucímu Ing. Tomáši Maradovi, Ph.D. rodině a přátelům za nadstandardní podporu při studiu a vypracování této diplomové práce.

ČESTNÉ PROHLÁŠENÍ

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. Tomáše Marady, Ph.D. a s použitím literatury uvedené v seznamu literatury.

V Brně dne 1. 3. 2019

.....
Alexandr Capovsky

OBSAH

1	ÚVOD.....	15
2	ŘÍDÍCÍ SYSTÉM SIMOTION.....	17
2.1	Obecné informace o systému SIMOTION	17
2.2	Programovací jazyky	18
2.3	SIMOTION koncepce paměti.....	19
2.4	Systém provádění úloh	20
2.4.1	Časová alokace	22
2.5	Implementace OOP.....	23
2.5.1	Nástroje OOP.....	23
3	KOMBINAČNÍ VÁHA	24
3.1	Konstrukce.....	24
3.2	Lineární váha vs Kombinační váha	25
4	NÁVRH ALGORITMU	27
4.1	Požadavky řídicího algoritmu.....	27
4.2	0/1 Problém batohu.....	28
4.3	Řešení pomocí tabulkové metody.....	28
4.3.1	Příklad 1.....	29
4.4	Problém batohu s dolní mezí	31
4.4.1	Příklad 2.....	31
4.5	Implementace algoritmu	32
4.6	Rozdílné módy pro parametr důležitost.....	33
4.6.1	Inkrementální mód.....	33
4.6.2	Hmotnostní mód	34
4.6.3	Hmotnostní mód číslo 2.....	35
4.7	Nedefinované chování	36
4.8	Zhodnocení algoritmu.....	37
5	SIMULAČNÍ MODEL	38
5.1	Popis HMI.....	38
5.2	Základní obrazovka	38
5.3	Krokový mód.....	39
5.4	Non-Stop mód.....	40
5.5	Servis a Diagnostika	40
5.6	Graf.....	42
6	VÝSLEDKY SIMULACE.....	43
6.1	Kritérium legislativní.....	43
6.2	Kritérium zrychlení dávky	47
6.2.1	Tabulka výsledků.....	47
6.3	Kritérium přesnosti	47
6.3.1	Kritérium přesnosti podle hmotnostního módu č.2	49
6.4	Ideální výsledky.....	49
6.5	Rychlost algoritmu	52
7	ZHODNOCENÍ A DISKUZE.....	53
8	ZÁVĚR	55

9	SEZNAM POUŽITÉ LITERATURY	57
10	SEZNAM OBRÁZKŮ.....	58
11	SEZNAM TABULEK	59

1 ÚVOD

Při sériové výrobě je každá nedokonalost na jednotlivém výrobku velký problém. Balící firma, která produkuje 100 000 balíků brambor denně, si nemůže dovolit příliš velkou nejistotu při vážení. Každý gram navíc představuje na konci dne velkou finanční ztrátu. Otázkou však je, jak docílit přesnosti a rychlosti navážení tak nehomogenního produktu, jako jsou například brambory. Nelze totiž zaručit přesnost váhy, tím že se budou brambory postupně přisypávat.

Řešením tohoto problému je kombinační váha. Princip tohoto stroje je v zásadě jednoduchý. Do jednotlivých kapes se dostane určitá váha produktu. Poté algoritmus spočítá, která kombinace kapes dává součet hmotnosti konvergující k cílové hmotnosti produktu. Tato kombinace kapes se následně vypustí.

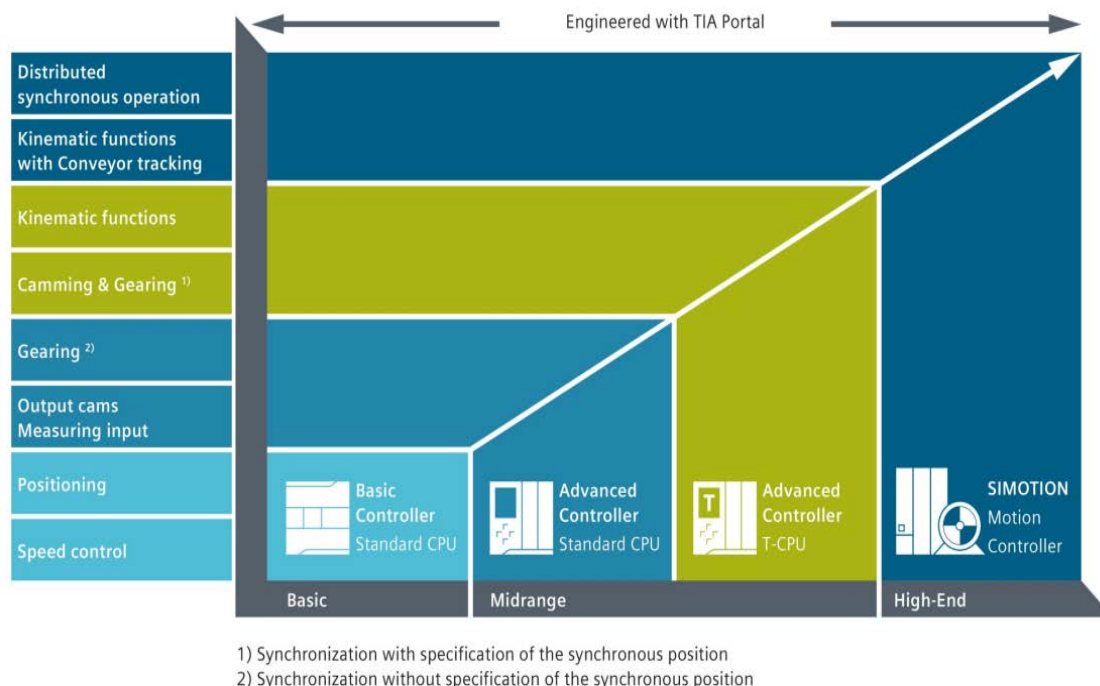
Tohle řešení je velice prosté a elegantní. Rozprostře nejistotu navážení do více kapes, které zpět zkombinuje. Vzniknou ovšem další problémy. Jak zajistit rovnoměrné využití kapes, aby se stroj nezasekl? Jak cílovou hmotnost vybrat? Jakým způsobem postupovat v případě nenalezení daného řešení? Kolik dílčích kapes je pro danou aplikaci nejvhodnější?

Tyto a další otázky jsou odpovězeny v následujících kapitolách.

2 ŘÍDÍCÍ SYSTÉM SIMOTION

Tato kapitola je věnována popisu systému SIMOTION, jeho hardwarovým konfiguracím, principu fungování a vývojových možnostech.

SIMOTION je nejvýkonnější a nejkomplexnější řídicí systém, který firma Siemens nabízí. Nejběžnější použití tohoto systému bývá při synchronizovaném řízení pohybu vícero os při krátkém (řádově stovky mikrosekund) PLC cyklu. Na následujícím obrázku je znázorněné aplikační použití systému SIMOTION.



Obr. 1: Použití systému SIMOTION [1]

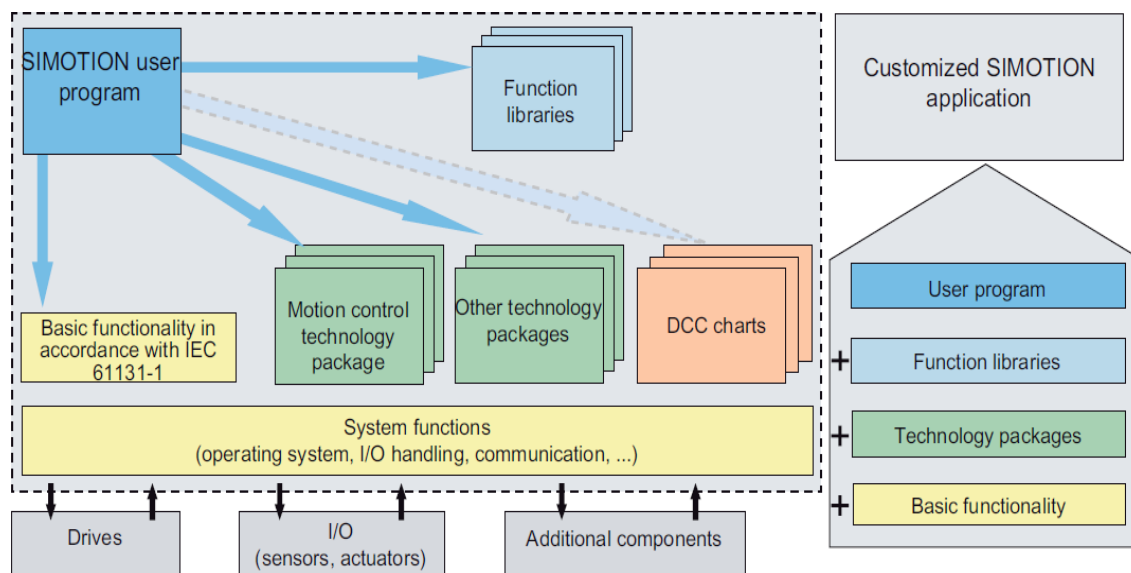
Tento systém byl vybrán z důvodu rychlé odezvy na složité matematické operace a pro možnost použití objektového orientovaného programování (OOP). Pomocí OOP lze vytvořit modulární knihovnu pro řízení kombinační váhy, která je výstupem této diplomové práce.

2.1 Obecné informace o systému SIMOTION

Základní funkcionální operační systém SIMOTION (Kernel Simotion) zahrnuje logické a aritmetické funkce a systémy s otevřenou a uzavřenou smyčkou. Programy mohou být spuštěné cyklicky, časově nebo spuštěné na základě definované události. Není tedy nutné vytvářet volání, synchronizaci a monitorování programu. Tyto funkce jsou již vytvořené.

Ve výsledku SIMOTION Kernel obsahuje veškeré funkce a odpovídá PLC podle normy IEC 61131-3 včetně systému pro kontrolu I/O. SIMOTION Kernel lze rozšířit

pomocí tzv. Technology packages. SIMOTION rovněž poskytuje systémové knihovny a knihovny pro motion control [2]. Konfigurace a programování lze provádět ve vývojovém prostředí (IDE) SIMOTION Scout nebo Scout TIA.



Obr. 2: Systémová architektura Simotion [2]

Offline/Online mód

Rozdíl mezi offline a online módem je následující:

- V offline módu mohou být tvořeny projekty a uživatelské programy. Můžete pracovat s IDE bez připojení k runtime systému. Offline mód slouží ke konfiguraci dat a systémových proměnných.
- V online módu lze nahrát projekty a data přímo do tzv. target system. Data změněná v online módu lze přehrát do offline projektu. Toto ovšem neplatí pro systémové proměnné.

2.2 Programovací jazyky

Následující programovací jazyky jsou k dispozici pro vytváření uživatelských programů.

MCC – Motion Control Chart

Je grafický programovací jazyk pro programování logiky a motion control funkcí.

ST – Structure text

Je textový programovací jazyk podle normy IEC 61131-3, který povoluje tvorbu více zdrojových souborů. Strukturovaný text od verze kernel 4.5 povoluje i objektově-orientované programování.

LAD/FBD - Ladder logic / Function Block Diagram

Grafické programovací jazyky podle normy IEC 61131-3 pro programování logiky a motion control funkcí pomocí PLCopen bloku.

DCC – Drive Control Chart

Grafický jazyk pro programování pohonů. Výhodný pro jednoduché debuggování a testování rychlosti provádění jednotlivých tasků.

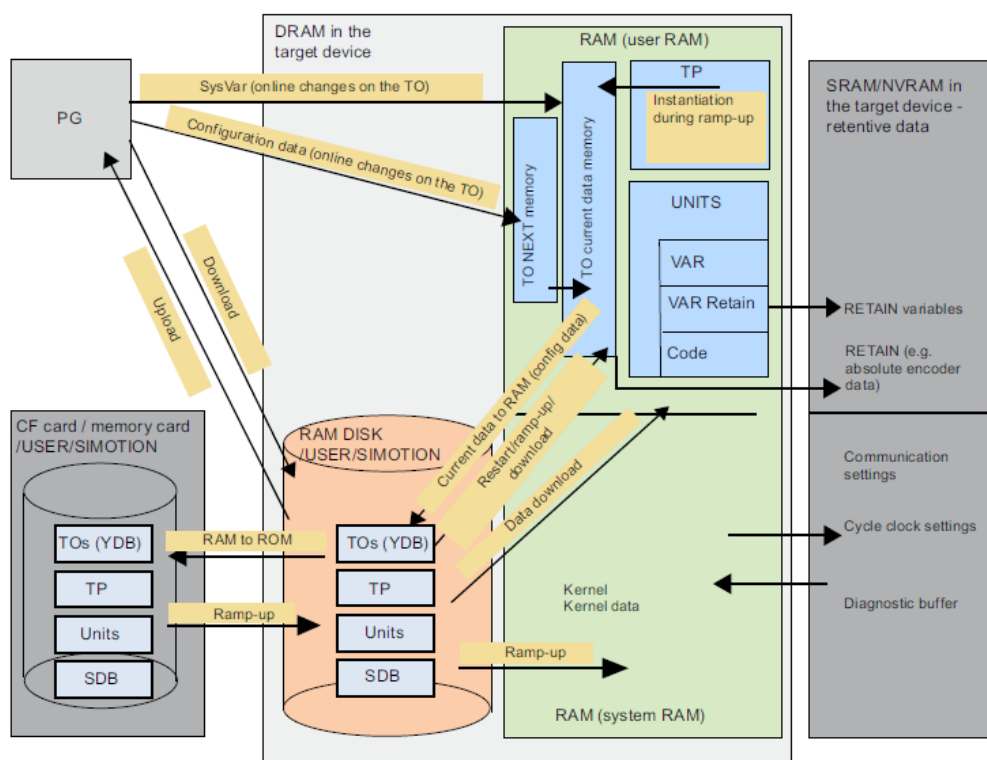
Clib Studio (C/C++ programming)

Vývojové prostředí pro Windows k programování funkcí a funkčních bloků, které jsou následně uloženy jako knihovny. V implementaci mohou být použity ve všech programovacích jazycích (MCC, ST, LAD/FBD, DCC).

2.3 SIMOTION koncepce paměti

Obecně lze koncepci paměti v PLC rozdělit do 4 bloků. Nahrávací, pracovní (uživatelská), systémová a trvalá paměť, kde se data ukládají. Každý blok využívá jiný typ paměti (RAM, SRAM/NVRAM, DRAM, ROM). V této kapitole se popíší jednotlivé paměti a jejich správu při provádění operací v IDE.

SIMOTION má konsolidovaný koncept správy paměti. Systém sám rozhodne, která data se uloží do DRAM (RAM disk, RAM) a která do SRAM/NVRAM. Paměť DRAM ztrácí svá data při odpojení přístroje od sítě. SRAM/NVRAM svá data uloží trvale.



Obr. 3: Všeobecná reprezentace konceptu paměti [2]

RAM disk

Představuje nahrávací blok paměti PLC. Jsou zde uložena všechna data, která byly staženy z počítače. Po stažení z PG RAM disk obsahuje hardwarovou konfiguraci přístroje, technologické balíčky (TP), konfigurační data technologických objektů a programové útvary (POU). S příkazem v IDE „Copy RAM to ROM“ obsah RAM disku je nahrán do paměti ROM (CF karta nebo jiná paměťová karta).

RAM

Představuje blok paměti, který je dále rozdělen na systémový a uživatelský blok paměti. Během stažení z PG jsou data získána z RAM disku. Uživatelský blok paměti obsahuje současná data o technologických objektech, proměnných a kompilované uživatelské programy. Proměnné, které jsou označeny jako trvalé, jsou uloženy v paměti SRAM/NVRAM.

Kernel data a následné nastavení jako jsou komunikační nastavení, hodiny cyklu PLC, obsah diagnostického bufferu, jsou uloženy v systémovém bloku paměti.

Využití RAM lze sledovat pomocí IDE (SIMOTION SCOUT, SCOUT-TIA) v online režimu.

ROM (paměťová karta)

Data se trvale ukládají na paměťovou kartu po provedení příkazu v IDE „Copy RAM to ROM“. Podobně jako u RAM paměti lze sledovat využitelnost ROM v IDE.

SRAM/NVRAM

Určitá data lze uložit do paměťového bloku, který při výpadku proudu data neztratí. Jedná se o data typu: ofset absolutního enkodéru, zachované proměnné označené jako VAR RETAIN, IP adresa, diagnostická vyrovnávací paměť, atd. Uživatelský program lze tyto hodnoty pomocí funkce „_savePersistentMemoryData“ uložit na paměťovou kartu. Toto zabráňuje ztrátě dat při výměně zařízení.

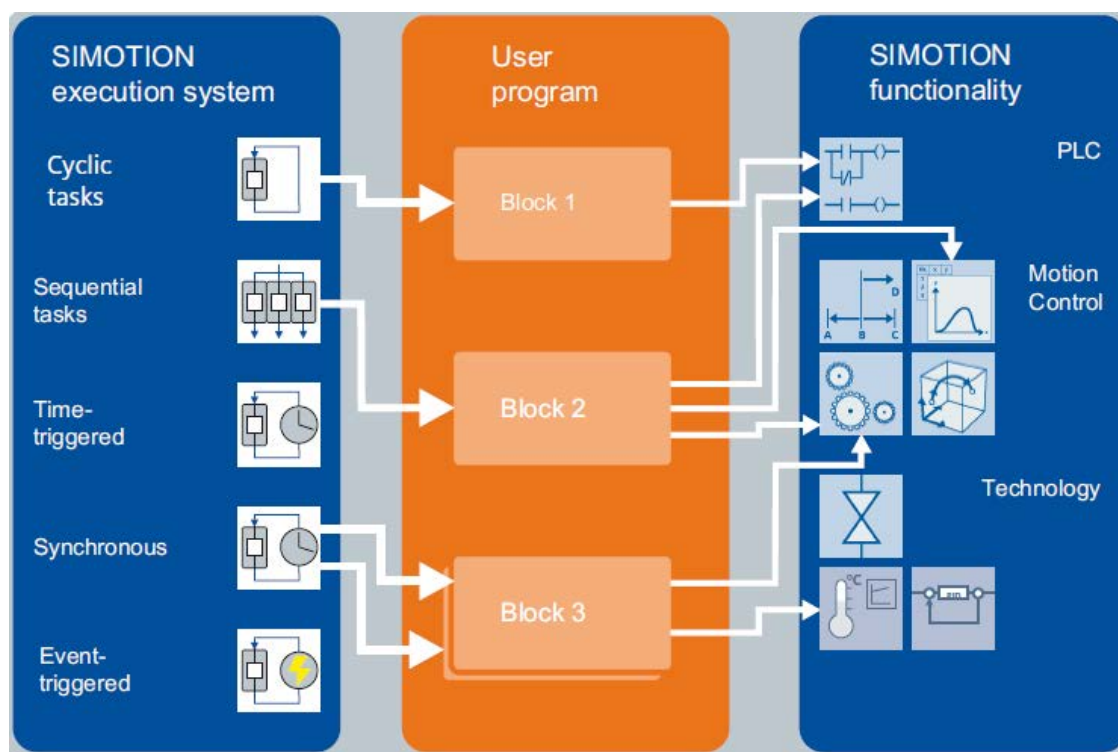
2.4 Systém provádění úloh

Systém SIMOTION poskytuje úlohy, cyklické, sekvenční, časově řízené, isochronní a spuštěné na základě události. Uživatelské programy mohou být „přiřazené“ k jednotlivým úlohám. Je zde možnost „přiřadit“ více programů k jednomu tasku. Z tohoto důvodu není nutné psát volání, koordinaci/synchronizaci programů.

Tasky mají tyto vlastnosti:

- Podmínku pro spuštění
- Prioritu (který task může být přerušen jiným taskem)

Následující tasky jsou k dispozici pro uživatelský program: Startup task, Motion task, Background task, Timer Interrupt task, Synchronous task, Systém a user interrupt task.



Obr. 4: Systém provádění úloh v SIMOTION

Pro aplikační účely jsou nejdůležitější tasky ze skupiny tzv. round-robin konkrétně Motion task a Background Task.

Motion task

Je určen pro programování sekvencí, případně pro programování motion control funkcí. Motion task nemá časové monitorování. Tedy v případě, kdy je task spuštěn, může zůstat aktivní po nespecifikované dlouhou dobu. Motion task je ukončen při splnění podmínky nebo při spuštění Shutdown task. Motion task lze pozastavit při použití příkazu WAITFORCONDITION.

Jelikož Motion task patří do skupiny round-robin, tak sdílí společný čas. Více k časové alokaci v následující podkapitole.

Background task

Je prováděn cyklicky. Po dokončení se automaticky restartuje. Background task je monitorován. Při překročení délky časového cyklu se aktivuje TimeFaultBackgroundTask, případně se systém uvede do stavu stop.

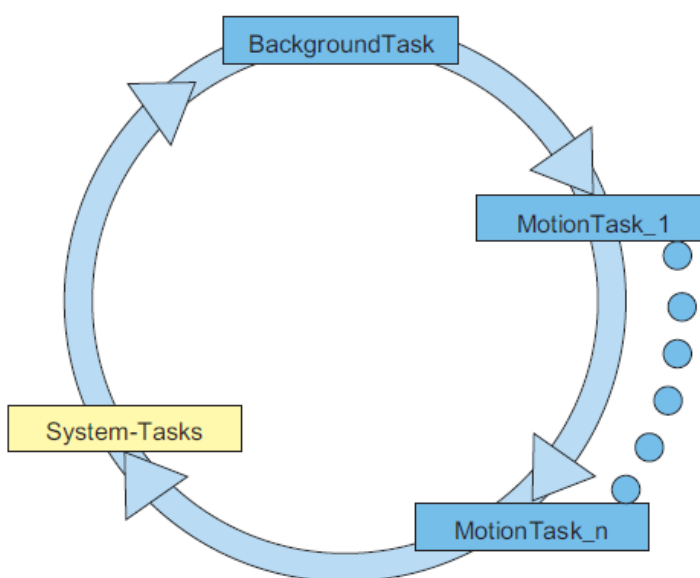
Tento task je vhodný pro matematické kalkulace, monitorování a cyklické logické funkce. Je naprosto nevhodný pro jakékoliv čekání na událost, případně na dlouhé cykly.

2.4.1 Časová alokace

Ve vývojovém prostředí SIMOTION SCOUT lze nastavit poměr časové alokace mezi Background taskem a ostatními tasky ze skupiny round robin (Motion task, Systém task)

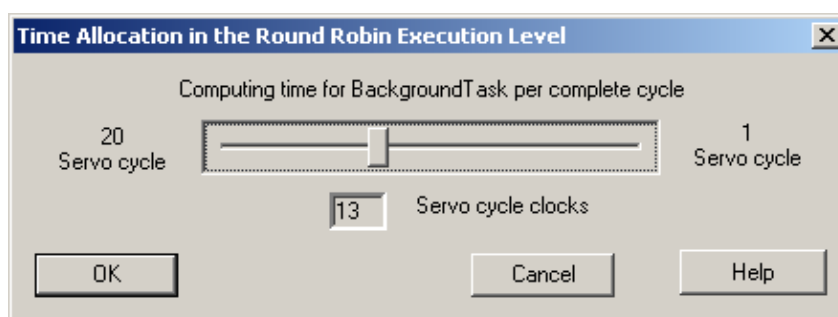
Princip Round Robin

Background task, Motion tasky a Systémové tasky jsou spouštěné sekvenčně. Díky rychlosti prováděných tasků lze považovat provádění tasku za téměř paralelní. Další task je spuštěn, jakmile předcházející task vyčerpá svůj výpočetní čas (task je dokončen, případně přešel do stavu wait). Alokační čas pro round robin cyklus je omezený. V následujícím cyklu jsou tasky prováděny od místa předchozího přerušení.



Obr. 5: Princip round robin [2]

Jednou z funkcí v systému SIMOTION je uživatelské přidělení výpočetního času. Motion task při defaultním nastavení je prováděn 2 servo cykly. Tedy při nastavení Background tasku lze měnit délku jeho výpočetního času. V případě nastavení, které zvýhodňuje Background task (maximálně 20 Servo cyklů) se zpomalí komunikace SCOUT/OP a je tedy vhodná pouze při velmi extrémních časově kritických aplikacích.



Obr. 6: Časová alokace v SIMOTION SCOUT [2]

2.5 Implementace OOP

Objektově-orientované programování (OOP) je implantováno do systému SIMOTION Kernel verze 4.5. Tato technologie je dostupná od roku 2016. OOP je plně implementován podle standardu IEC61131-3.

Princip OOP tkví v rozdělení komplexní aplikace do jasně definovaných objektových struktur. V případě průmyslové praxe lze jednotlivé komponenty komplexního stroje definovat jako třídy. Tyto třídy lze programovat a testovat zvlášť.

Díky této technice a jasně definovaných vztazích jednotlivých tříd, lze efektivněji spravovat rozsáhlé projekty. Standardizace a znovupoužitelnost kódu je také důvodem, proč byla tato možnost programování do systému SIMOTION zavedena [5].

2.5.1 Nástroje OOP

V této podkapitole bude popisován stav pro SIMOTION Kernel 5.2. Pro většinu programovacích úloh je většina nástrojů OOP (třída, dědičnost, polymorfismus, rozhraní...) připravena k použití. V této práci se předpokládá základní znalost OOP.

Třída

Třída je definovaná klíčovým slovem CLASS. Má všechny vlastnosti třídy z jiných programovacích jazyků (C#, JAVA). Lze ji definovat jako abstraktní třídu klíčovým slovem ABSTRACT. Metody jsou definovány klíčovým slovem METHOD. V systému SIMOTION je přístupnost k jednotlivým metodám rozdělena do 3 úrovní. Klíčová slova PUBLIC, PROTECTED a PRIVATE signalizují tyto úrovně.

Dědičnost a rozhraní

Dědičnost je definována klíčovým slovem EXTENDS. Je vložena za definici třídy. Při použití dědičnosti je možnost použití všech metod předka. Pomocí klíčových slov THIS a SUPER lze zavolat zděděné metody. Vícenásobná dědičnost není povolena.

Interface neboli rozhraní je implantované klíčovým slovem INTERFACE. Je základem pro použití OOP. Pomocí něj můžeme definovat metody, které využívají více tříd. Rozhraní neobsahuje žádný program. Pomocí rozhraní lze vytvářet nezávislý software.

Reference

Definována klíčovým slovem REF. Reference je proměnná, která neobsahuje žádnou hodnotu, ale obsahuje umístění paměti proměnné nebo instance třídy. Definovaný datový typ může být přidělen k referenci. Aritmetické operace s referencemi nejsou k dispozici. Inicializační hodnota reference je NULL.

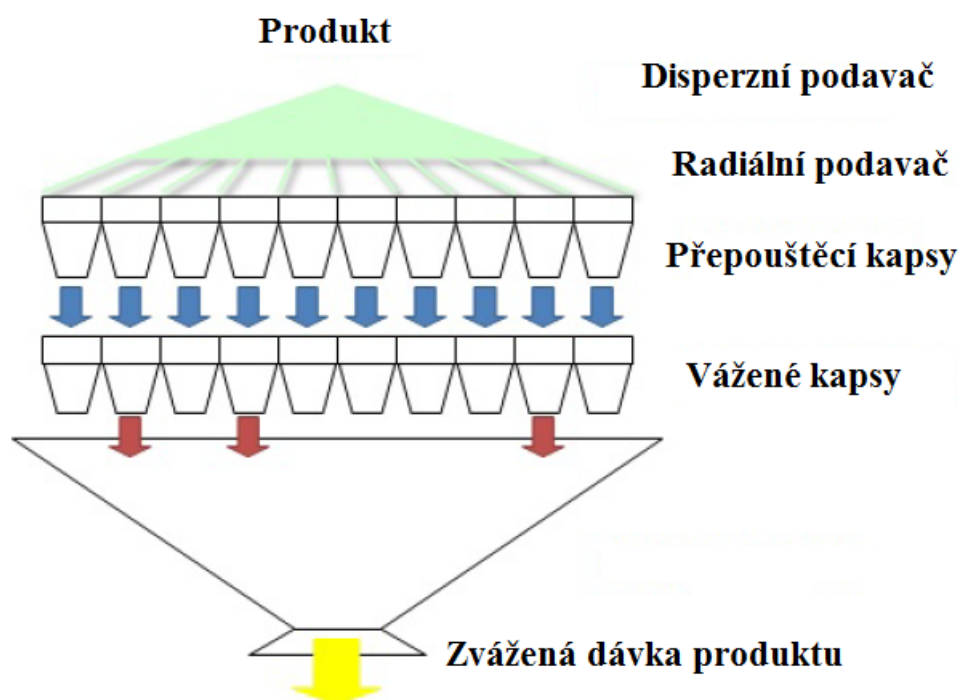
3 KOMBINAČNÍ VÁHA

Kombinační váha je průmyslové zařízení pro přesné a rychlé vážení a dávkování zboží (ovoce, zelenina, kancelářské potřeby, masné výrobky). Tento přístroj byl poprvé sestrojen v polovině 80. let 20. století. Důvod jejího použití je zejména u produktů, které nejdou navážít přesně. Jedná se zejména o produkty nestálé hmotnosti (brambory) nebo nestálého charakteru (tvaroh, želatinové bonbony).

Kombinační váha používá techniky kombinačního vážení, které jsou efektivnější než konvenční vážení k dosažení žádané váhy produktu. Tedy pro výslednou dávku použije kombinaci dílčích kapes. Cílem je, s určitou tolerancí, získat výslednou kombinaci kapes 100% váhy dané dávky. V následujících podkapitolách je popsán rozdíl mezi kombinační a lineární vahou, konstrukce kombinační váhy a způsoby řízení tohoto stroje [3].

3.1 Konstrukce

Na následujícím obrázku jsou popsány nejdůležitější komponenty kombinační váhy.



Obr. 7: Konstrukce kombinační váhy [3]

Disperzní a radiální podavač automatizují plnění produktu do přepouštěcí kapsy. Obvykle je zkonstruován ve formě pásu, kde produkt uvádí do pohybu vibrace [4].

Přepouštěcí kapsa připravuje dávku produktu do vážené kapsy. Čeká na povel pro otevření kapsy do vážené kapsy.

Vážená kapsa po zvážení dávky produktu vyšle tuto informaci do řídicího systému, který po získání hodnot všech vážených kapes rozhodne, jaká kombinace kapes bude vypuštěna. V provozu je stroj většinou kalibrován, aby jedna vážená kapsa měla kolem 33 % celkové hmotnosti. Je to kompromis mezi rychlostí navážení a přesností jedné dávky.

Zvážená dávka produktu je nejlepší kombinace námi vybraných kapes. Nejmodernější váhy disponují až 20 váženými kapsami a dokáží vyprodukovat více než 150 dávek za minutu v závislosti na váženém produktu.

3.2 Lineární váha vs Kombinační váha

Lineární váha narozdíl od kombinační váhy nemá dílčí kapsy. Produkt se posouvá díky vibracím. Pro dosažení lepší přesnosti je lineární váha vybavena jedním až třemi kaskádami pro rovnoměrné navážení produktu. Zařízení bývá použito v kombinaci s balicím strojem případně jako poloautomatický dávkovač.

Hlavní rozdíl mezi lineární a kombinační váhou je absence dílčích kapes. Tím je limitováno použití této váhy. Zvláště se nehodí pro lámavé, lepící produkty. Taktéž pro produkty nestálé hmotnosti. Lineární váhy se tedy typicky používají pro mleté koření, kávu, luštěniny. Jejich hlavní výhodou je příznivá cena [7].



Obr. 8: Lineární váha [7]

Kombinační váha je díky své konstrukci, je vhodná pro produkty, které se nedají jednoduše zvážet. V případě vážení brambor lineární vahou nelze zaručit přesnost jednotlivé dávky. Kombinační váha tuto nejistotu při vážení nahrazuje možností zkombinování různých hmotností v jednotlivých kapsách, aby výsledná hodnota dávky blížila požadované hodnotě.

Nevýhodou kombinační váhy jsou větší nároky na řízení. Výhoda je v širokém využití a balení spotřebního zboží, například masné výrobky, těstoviny, drobné pečivo, krmivo pro zvířata [7].



Obr. 9: Kombinační váha [7]

4 NÁVRH ALGORITMU

Z kapitol 3.1 a 3.2 lze vyčíst chování kombinační váhy. Správný návrh algoritmu je klíčový pro efektivní chod tohoto stroje.

Tato kapitola popisuje, jak získat optimální řešení v daném tolerančním pásmu při splnění určitých požadavků. Tyto požadavky byly shromážděny ve spolupráci s průmyslovými experty firmy Siemens. Požadavky se liší v závislosti na váženém produktu. Při splnění těchto požadavků je algoritmus konkurenceschopný a použitelný v průmyslové praxi.

Dále bude vyřešen problém využitelnosti všech kapes. Tento bod je velmi klíčový pro automatizovaný chod stroje. Pro určité průmyslové aplikace je podstatná doba, po kterou je produkt ve vážené kapse. Úkolem algoritmu je tuto dobu minimalizovat.

Chod kombinační váhy podléhá určitým regulacím [8]. Při tvorbě algoritmu byla tato skutečnost respektována.

4.1 Požadavky řídicího algoritmu

Požadavky lze shrnout do těchto bodů:

- **Výsledná zvážená hmotnost**

Pro konkurenceschopnost stroje je tento bod klíčový. Odchylku mezi žádanou hodnotou a výslednou hodnotou váhy je nutné minimalizovat.

- **Algoritmus má definovanou horní a dolní mez**

Podle vyhlášky č. 328/2000 Sb. je nutné, aby maximální odchylka hmotnosti výrobků byla 5 g nebo 5 ml. Zároveň je definováno, že hmotnost výrobků nesmí překročit 10 kg nebo 10 l [8]. Z tohoto důvodu je v algoritmu zavedena dolní mez. Horní mez je většinou požadovaná hmotnost zboží.

- **Rovnoměrné využití kapes**

Tento bod je důležitý z hlediska automatizovaného chodu stroje. Kombinační váhy fungují v autonomním režimu bez zásahu člověka. Kapsa, která zůstane dlouho nevypuštěná, představuje problém (dlouhodobé nevyužití kapsy může způsobit degradaci obsahu). Důvodem může být, že algoritmus nemůže tuto kapsu zkombinovat s jinými kapsami. V krajním případě by mohlo dojít k tomu, že všechny kapsy by byly obsazeny a došlo by k zastavení vážení. Také pro určité produkty hrozí jejich degradace.

- **Nízká výpočetní náročnost**

Tento nový algoritmus je nutné optimalizovat tak, aby jeho výpočetní náročnost nebyla exponenciální. Využívá se technik dynamického programování.

4.2 0/1 Problém batohu

V této diplomové práci byl v implementačním programu použit upravený problém batohu (anglicky Knapsack problem). Důvodem použití diskretní varianty úlohy o batohu je nemožnost dalšího dělení dílčích hmotností v jednotlivých vážících kapsách. Vážící kapsu lze vypustit pouze celou. Pro řešení problému o batohu bylo využito dynamické programování, konkrétně tabulková metoda.

Problém batohu je optimalizační metoda, kdy algoritmus maximalizuje cenu a zároveň kontroluje zda nepřekročil kapacitu batohu.

Matematický model základního 0/1 problému o batohu:

$$f(x) = \sum_{j=1}^n c_j x_j \rightarrow \max \quad (1)$$

$$\sum_{j=1}^n a_j x_j \leq b \quad (2)$$

$$x_j \in \{0,1\} \quad j = 1, \dots, n \quad (3)$$

kde n , je počet kapes, a_j je hmotnost j -té kapsy, c_j je cena j -té kapsy, b je kapacita batohu. x_j je rozhodnutí zda j -tou kapsu zahrneme do batohu nebo ne [9]. Batohem se myslí celková zvážená dávka. Cena kapsy je v této diplomové práci definována jako důležitost (anglicky importance) kapsy. Její hodnota závisí na použitém módu v řídicím programu. Podrobně bude rozebrána v kapitole 4.5.

4.3 Řešení pomocí tabulkové metody

Dynamické programování zajistí rychlejší výpočet algoritmu k nalezení optimálního řešení, než hrubá síla.

Úloha o batohu při řešení dynamickým programováním je n -etapový rozhodovací proces. Stav rozhodovacího procesu na začátku j -té etapy je zbývající kapacita batohu p_j . Rozhodnutí v j -té etapě se myslí určením hodnoty x_j , kdy hodnota 1 značí zahrnutou hodnotu do batohu a 0 značí nezahrnutou hodnotu. Definice stavů procesu se provádí podle následujících rovnic:

$$p_1 = b \quad (4)$$

$$p_{j+1} = p_j - a_j x_j \quad (5)$$

Přípustné rozhodnutí, což znamená rozhodnutí, kdy nedojde k překročení kapacity batohu, lze definovat pomocí následující rovnice [9].

$$x_j \in S_j(p_j) = \{x \in \{0,1\} \mid p_j - a_j x \geq 0\} \quad (6)$$

Úloha vede na systém n funkcionálních rovnic.

$$\max \sum_{j=1}^n c_j x_j = \varphi_n(b) \quad (7)$$

$$\varphi_{n-j+1}(p_j) = \max \left(c_j x_j + \varphi_{n-j}(p_j - a_j x_j) \right) \quad (8)$$

$$\varphi_1(p_n) = \max_{x_j \in S_n(p_n)} c_n x_n \quad (9)$$

V rovnici (8) jsou hodnoty $j=1, 2, \dots, n-1$. Optimální hodnota účelové funkce označena písmenem φ je zapisována do tabulky. Pro získání této optimální hodnoty je nutné nalézt řešení systému těchto funkcionálních rovnic. Systém funkcionálních rovnic je vyřešen pro $j=n, n-1, \dots, 1$ a pro hodnoty stavů p_j [10],

$$p_j \in \{\Delta, 2\Delta, \dots\} \cap \left\langle \max \left\{ 0, b - \sum_{k=1}^{j-1} a_k \right\}, b \right\rangle \quad (10)$$

kde Δ je zvolená hodnota. Symbol Δ představuje rozlišení, které dokáže vážená kapsa navážít. Při tvorbě simulačního programu byla hodnota $\Delta = 1$. V následujícím příkladě bude vysvětleno použití rovnic (1) až (10).

4.3.1 Příklad 1

Nechť jsou k dispozici 3 kapsy viz Tab.1. Kapacita batohu je 7 jednotek. Úkolem je maximalizovat cenu v batohu.

ID	j	1	2	3
cena	c_j	3	7	1
hmotnost	a_j	3	5	3

Tab. 1: Zadání příkladu 1

Sestavení matice řešení

V Tab.2 je naznačeno řešení matice. Z této matice lze získat veškerá data pro řešení problému batohu. Algoritmus, který byl odvozen z rovnic (1) až (10) lze shrnout do těchto kroků:

- 1) Začíná se Sekcí ID=3
- 2) Podle rovnice (1) a (2) byly získány hodnoty v řádcích, kde $x=0$ a $x=1$.
- 3) Poté je nutné najít maximum v každém sloupci. Toto maximum se zapíše do řádku φ_1 .
- 4) Dle toho, které z hodnot byla vyšší se zapíše rozhodnutí do řádku X3.
- 5) Řádek φ_1 se zkopíruje beze změn do sekce ID=2 pro řádek $x=0$.
- 6) Pokračuje se krokem 2 pro řádek $x=1$.
- 7) Algoritmus se ukončí po vyzkoušení všech kapes.

Akt. Kapacita	0	1	2	3	4	5	6	7
ID=3								
x=0	0	0	0	0	0	0	0	0
x=1	0	0	0	1	1	1	1	1
$\varphi_1=1$	0	0	0	1	1	1	1	1
X3	0	0	0	1	1	1	1	1
ID=2								
x=0	0	0	0	1	1	1	1	1
x=1	0	0	0	0	0	7	7	7
$\varphi_1=1$	0	0	0	1	1	7	7	7
X2	0	0	0	0	0	1	1	1
ID=1								
x=0	0	0	0	1	1	7	7	7
x=1	0	0	0	3	3	3	4	4
$\varphi_3=1$	0	0	0	3	3	7	7	7
X1	0	0	0	1	1	0	0	0

Tab. 2: Matice řešení příkladu 1

Optimální strategie

Jakmile je tato matice sestavena, je nutné najít které kapsy byly v batohu použity. Pro potřeby navrhovaného algoritmu je potřeba získat stavy při maximálním využití batohu. Konkrétně optimální hodnotu účelové funkce, optimální strategii a stavy procesu pro optimální strategii. Algoritmus pro získání použitých kapes:

- 1) Stav procesu p_1 je vždy rovna kapacitě. Podle rovnice (4).
- 2) Optimální strategie x_1 je rovna rozhodnutí X1 v sloupci p_1 . V tomto případě je rozhodnutí 0.
- 3) Z rovnice (5) vyplývá, že $p_2=7$
- 4) Optimální strategie pro x_2 je rovna rozhodnutí X2 v sloupci pro $p_2=7$. V tomto případě rozhodnutí 1.
- 5) Z rovnice (5) vyplývá, že $p_3=2$.
- 6) Optimální strategie pro x_3 je rovna rozhodnutí X3 v sloupci pro $p_3=2$. V tomto případě rozhodnutí 0.

Vektor optimální strategie se rovná (0,1,0). Z toho vyplývá, že vypuštěna byla pouze prostřední kapsa. Hodnota účelové funkce je číslo 7.

Zhodnocení algoritmu

Tento algoritmus dokáže najít optimální řešení. Je zde ale problém v tom, že není ošetřena dolní mez, což je v rozporu s požadavky algoritmu a proti zákonu [8]. V následující kapitole bude vyřešen problém s dolní mezí.

4.4 Problém batohu s dolní mezí

Modifikace oproti klasickému problému s batohem tkví v podmínce, aby hmotnost obsahu batohu neklesla pod zadanou mez. V matematickém modelu se tato skutečnost projeví modifikací rovnice (2).

$$d \leq \sum_{j=1}^n a_j x_j \leq b \quad (11)$$

Kde d , je dolní mez. Značení ostatních proměnných zůstává zachováno. Dále k stavům procesu rovnice (4) a (5) je doplněna množina přípustných stavů P_j .

$$P_{n+1} = \{0, 1, \dots, e = b - d\} \quad (12)$$

$$P_j = \{p \in \{1, \dots, b\} \mid p \in P_{j+1} \vee p - a_j \in P_{j+1}\}, j=n, n-1, \dots, 1 \quad (13)$$

Přípustná rozhodnutí:

$$x_j \in S_j(p_j) = \{x_j \in \{0, 1\} \mid p_j - a_j x_j \in P_{j+1}\} \quad (14)$$

Z rovnice (12), kde e je povolený zůstatek. Tedy rozdíl mezi kapacitou batohu b a dolní mezí d vyplývá, že každý stav, který bude mimo povolený zůstatek bude penalizován. V řídicím algoritmu byl penalizační koeficient nastaven na -10000. Tento penalizační koeficient zabráňuje porušení dolní meze [10].

4.4.1 Příklad 2

Zadání je stejné jako v příkladu 1, ale je zde doplněná dolní mez (označena písmenem d) na hodnotu 6. Penalizační koeficient je -10000. Úkolem je opět maximalizovat cenu v batohu s přihlédnutím k dolní mezi. Povolený zůstatek je značen písmenem e .

	j	1	2	3	d	b	e	Penalizace
cena	c _j	3	7	1	6	7	1	-10000
hmotnost	a _j	3	5	3				

Tab. 3: Zadání příkladu 2

Matice řešení se sestavuje podobně jako v příkladu 1 ovšem nyní je zde podmínka. Bude-li váha dávky nižší než povolený zůstatek, pak tato možnost bude penalizována. Tato skutečnost se promítne do matice řešení.

Akt. Kapacita	0	1	2	3	4	5	6	7
ID=3								
x=0	0	0	-10000	-10000	-10000	-10000	-10000	-10000
x=1	-10000	-10000	-10000	1	1	-10000	-10000	-10000
φ_1	0	0	-10000	1	1	-10000	-10000	-10000
x3	0	0	0	1	1	1	1	1
ID=2								
x=0	0	0	-10000	1	1	-10000	-10000	-10000
x=1	-10000	-10000	-10000	-10000	-10000	7	7	-9993
φ_1	0	0	-10000	1	1	7	7	-9993
x2	0	0	0	0	0	1	1	1
ID=1								
x=0	0	0	-10000	1	1	7	7	-9993
x=1	-10000	-10000	-10000	3	3	-9997	4	4
φ_3	0	0	-10000	3	3	7	7	4
x1	0	0	0	1	1	0	0	1

Tab. 4: Řešení matice příkladu 2

Algoritmus pro zjištění optimální strategie je shodný s příkladem 1. Z tohoto důvodu bude zde uvedeno pouze řešení.

Stavy procesů při optimální strategii	p_1	p_2	p_3
	7	4	4
Optimální strategie	x_1	x_2	x_3
	1	0	1

Tab. 5: Optimální strategie příkladu 2

Zhodnocení algoritmu

Optimální strategie se v tomto případě liší oproti příkladu 1. Byla zde vybrána první a poslední hodnota. Výhoda algoritmu je v nalezení optimální hodnoty v intervalu, který je ohraničen dolní mezí a kapacitou batohu. Pro správné fungování algoritmu je ovšem nutné ošetřit stav, kdy neexistuje žádná kombinace s výslednou váhou uvnitř tohoto intervalu.

4.5 Implementace algoritmu

Implementace algoritmu do kombinační váhy je následující. Celková váha batohu je výsledná zvážená dávka. Kapacita batohu je maximální hodnota, kterou je výrobce ochotný povolit. Dolní mez je definována zákonem viz. Kapitola 4.1.

Cena kapsy může být zvolena podle potřeby programátora nebo výrobce kombinačních vah. Při balení některých produktů je nutné, aby produkt byl ve vážené kapse minimální dobu. Jedná se hlavně o masné výrobky, tvaroh, atd. V tomto případě lze zvyšovat cenu kapsy s každým cyklem, kdy kapsa nebyla využita. V jiných případech nezáleží, jak dlouho bude produkt ve vážené kapse. Proto lze cenu váhy nastavit jako vzdálenost k ideální váze. Ideální váha bývá nastavena na 33 % celkové váhy. Cenu lze naprogramovat podle libovolné závislosti.

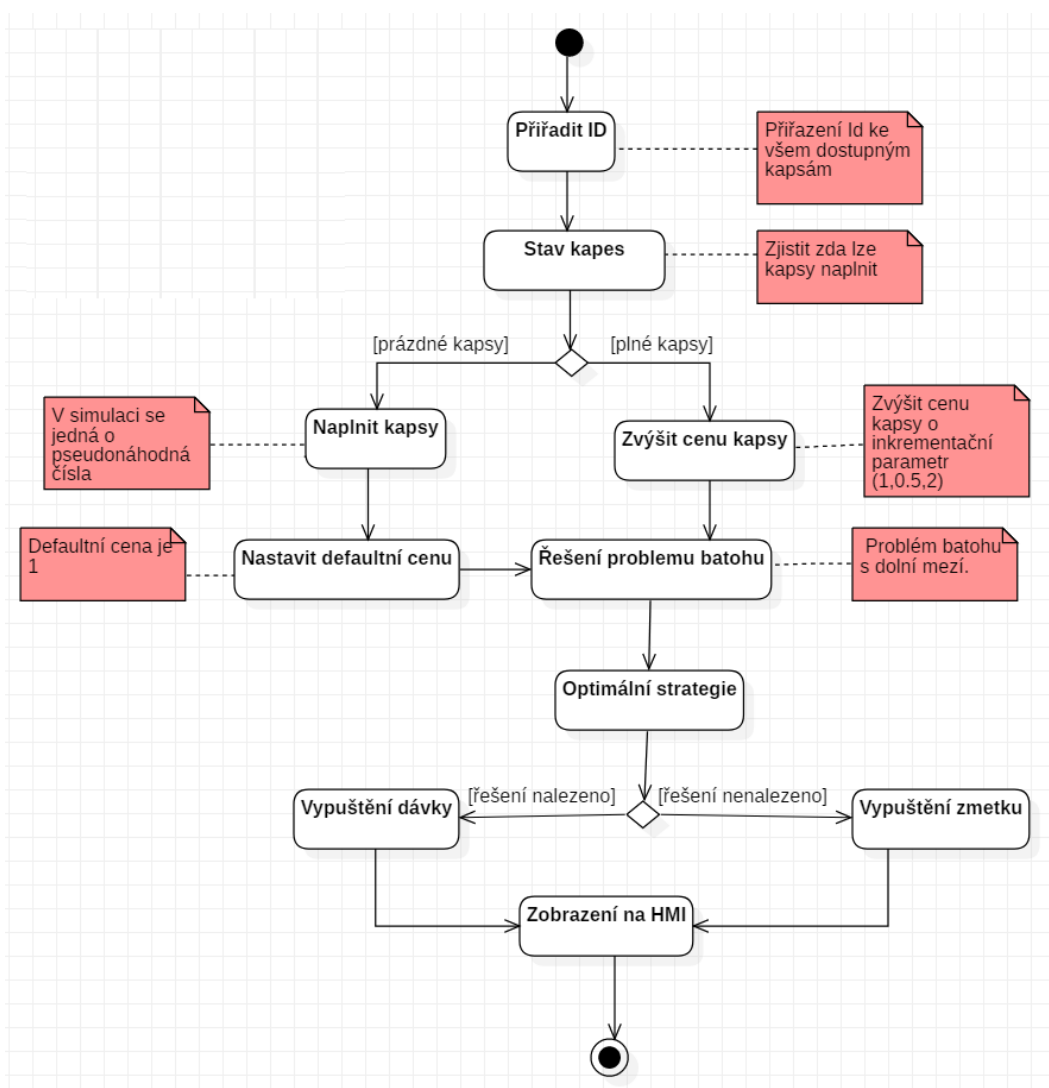
V závislosti na prioritách je nutné nastavit cenu jednotlivé kapsy. Při prioritě, kdy nevyužitelnost jednotlivých kapes ve vážicím procesu hraje vysokou roli, je potřeba cenu kapsy zvyšovat. Při prioritě zbavit se krajních hodnot je nutné tyto kapsy ohodnotit vyšší hodnotou ceny. Při prioritě získat zváženou dávku nejbliž ke kapacitě batohu je nutné maximalizovat hmotnost jednotlivých kapes. Tedy cena kapsy se rovná její hmotnosti.

4.6 Rozdílné módy pro parametr důležitost

Parametr důležitost hraje roli při preferencích výběru kapsy. V této podkapitole bude vysvětleno, jak jednotlivé módy v simulačním modelu fungují.

4.6.1 Inkrementální mód

V Inkrementálním módu platí, že při nevyužití kapsy její důležitost vzroste o určitou hodnotu. V simulačním modelu bylo otestováno několik hodnot a jejich vliv na chod stroje. V následujícím diagramu aktivit je vysvětlen princip inkrementálního módu.



Obr. 10: Diagram aktivit inkrementálního módu

4.6.2 Hmotnostní mód

Tento mód figuruje s reálnou hmotností jednotlivých kapes. Výpočet ceny jednotlivé kapsy podle vztahu:

$$|a_j - C| = c_j \quad (15)$$

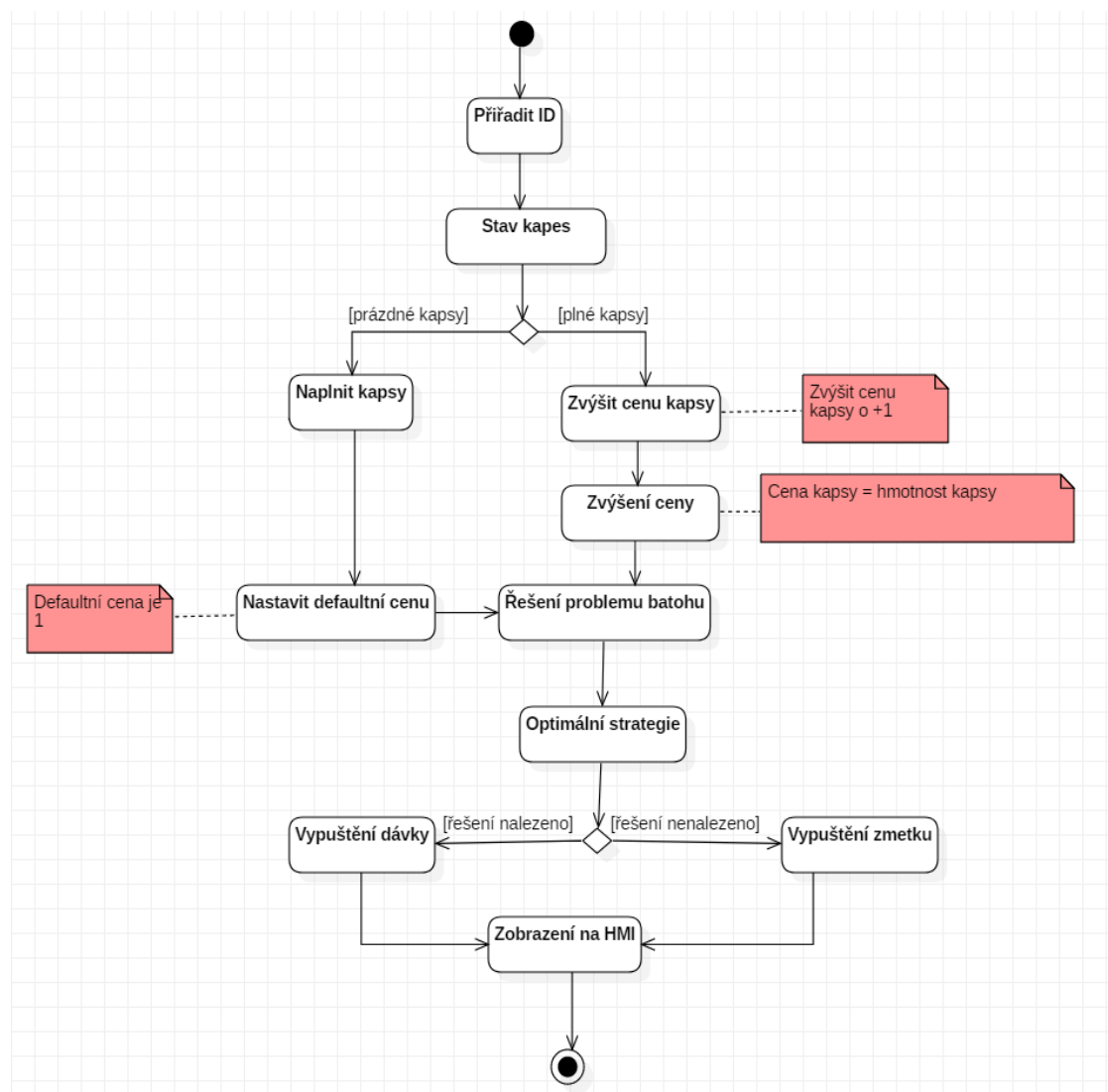
kde C značí ideální hmotnost. Ideální hmotnost je $1/3$ kapacity batohu. Cílem tohoto algoritmu je snaha eliminovat krajní hodnoty.



Obr. 11: Diagram aktivit hmotnostního módu

4.6.3 Hmotnostní mód číslo 2

Tento mód také figuruje s reálnou hmotností jednotlivých kapes. Oproti předchozímu módu se vyznačuje faktem, že parametr důležitost se rovná hmotnosti jednotlivé kapsy. To má za následek, že se algoritmus snaží maximalizovat hmotnost dávky. Tedy algoritmus inklinuje k nalezení varianty kapes s celkovou hmotností kapacity batohu. Tato vlastnost je velmi důležitá pro kritérium přesností uvedený v kapitole 6.3.



Obr. 12: Hmotnostní mód č.2

Závěrem lze říci, že parametr důležitost se v hmotnostním módu chová podle vztahu:

$$c = a \quad (16)$$

4.7 Nedefinované chování

Stav, kdy neexistuje kombinace, která by se vešla mezi dolní a horní mez. Tato možnost je zaznamenána a dojde k vypuštění dříve nedefinovaných kapes. Tento stav je nežádoucí a lze ho považovat za zmetek. Existují mnohá řešení, jak tento stav řešit.

Jedno z řešení je snížit, případně zvýšit dolní, respektive horní mez. V případě, že stále není řešení nalezeno, vypustí se pouze jedna kapsa s nejvyšší důležitostí, která je považována za zmetek. V rámci simulačního modelu byla obě řešení použita. První

řešení je výhodnější z hlediska minimalizace nedefinovaných stavů druhé z hlediska rychlosti.

4.8 Zhodnocení algoritmu

Důvody k použití tolika módů byly různé požadavky zákazníka na chování stroje. Lze říci, že pouhou změnou parametru se dokáže algoritmus přizpůsobit různým situacím.

V případě inkrementálního módu je jeho hlavní výhodou rychlost vypuštění jednotlivých kapes. Důležitost kapsy je závislá pouze na čase, kdy není využita. Kapsy se tedy využívají rovnoměrně.

Hmotnostní mód se používá pro zvýhodnění kapes, které mají extrémní hmotnost. Čím je rozdíl hmotností mezi ideální váhou a reálnou váhou kapsy vyšší, tím se zvyšuje důležitost dané kapsy.

Hmotnostní mód číslo 2 je nejspíše nejdůležitějším módem ze všech. Tento mód se nejvíce hodí pro použití, kdy klíčová vlastnost je přesnost dávky.

5 SIMULAČNÍ MODEL

Tato kapitola je věnována implementaci simulace do vývojového prostředí TIA Portál verze číslo 15. Simulační model ukazuje funkčnost kombinační váhy. Lze tedy získat informace ohledně úspěšnosti nalezení optimálního řešení, průměrný počet použitých kapes při vážení, průměrná doba, kdy určitá kapsa byla nevyužita, průměrná hodnota dávky, atd. Všechny tyto informace jsou nutné pro řízení a regulaci stroje.

5.1 Popis HMI

HMI (anglicky human machine interface) slouží k řízení stroje obsluhou. V simulačním modelu bylo vytvořeno pět obrazovek. Jedná se o základní obrazovku, krokový mód, non-stop mód, servis a diagnostiku a graf.

5.2 Základní obrazovka



Obr. 13: Základní obrazovka

Základní obrazovka slouží jako rozcestník v simulačním modelu. Je to také první obrazovka po načtení systému.

5.3 Krokový mód

SIEMENS SIMATIC HMI

5/12/2019 8:48:09 PM

START

Celkový počet kapes: +8

Pocet cyklů: +1585

Error

Pocet neoptimalnich dávek: +23

Základní diagnostika

Dolní mez: +95 Horní mez (Kapacita batohu): +100 Náhodná dolní hodnota: +20 Náhodná horní hodnota: +40

Min celková váha: 71 Aktuální celková váha: 96 Průmerná váha dávky: +97.355

ID	0	1	2	3	4	5	6	7
Váha kapsy	27	25	27	26	32	31	38	24
Důležitost kapsy	+1	+1	+4	+2	+1	+2	+1	+2
Použit/Nepoužit	●	●	●	●	●	●	●	●

Krokový mód Non-stop mod Servis/Diagnostika Graf

Obr. 14: Obrazovka krokového módu

Tento mód slouží výhradně k prezentačním účelům. Je zde zobrazeno 8 kapes. Identifikační číslo slouží k identifikaci kapsy v systému. Dále je zde zobrazena váha kapsy a její důležitost. Rozsvícení zelené kontrolky symbolizuje, zda byla kapsa vybrána do zvážené dávky. Spuštění jednoho cyklu se provádí stisknutím tlačítka START.

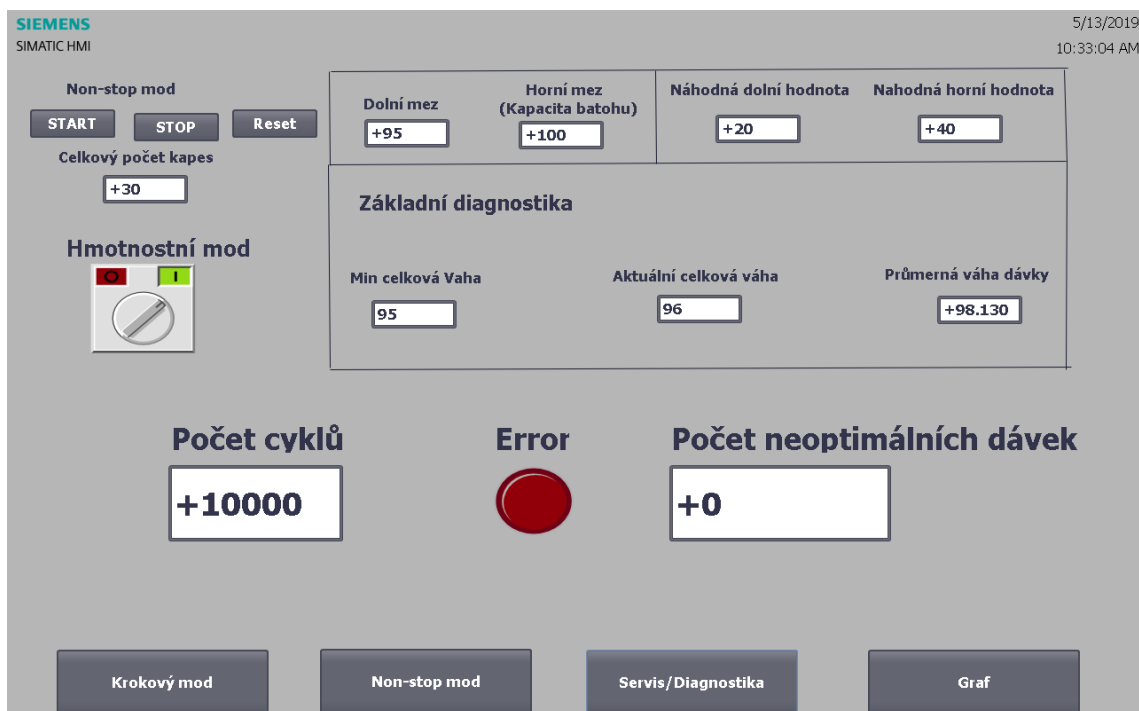
V sekci základní diagnostika lze získat informace o zvážených dávkách. Minimální celková váha zvážené dávky se počítá od doby spuštění stroje. To platí i pro průměrnou váhu. Aktuální dávka je právě zvážená dávka v probíhajícím cyklu.

Dále jsou zde základní informace o nastavení kombinační váhy. Počet kapes závisí na konstrukci stroje. V reálném provozu se tedy jedná o konstantu. Náhodná dolní a horní hodnota je pseudonáhodné číslo, která představuje hmotnost jednotlivých kapes. Pseudonáhodná vlastnost je důležitá pro opakovatelnost pokusu a vyzkoušení různého typu nastavení váhy.

Pro nastavení výpočtového algoritmu je klíčové nastavení hodnoty dolní a horní meze. Lze je individuálně upravovat. V případě, že neexistuje kombinace hmotnosti jednotlivých kapes, která by dávala celkovou dávku v daných mezích, rozsvítí se kontrolka ERROR. Počet neoptimálních dávek tuto skutečnost zaznamenává.

5.4 Non-Stop mód

Tento mód je simulací reálného chování stroje. Kombinační váha funguje v provozu zcela samostatně bez zásahu obsluhy. Na HMI obrazovce byl přidán tlačítko STOP pro zastavení vážení.



Obr. 15: Non-stop mód

Zvýrazněn je počet cyklů, po který program běží. Dále je zvýrazněn počet neoptimálních dávek, jelikož to je nutné sledovat při běhu programu. V případě, že v programu začne počet neoptimálních dávek exponenciálně stoupat, program přejde do stavu STOP.

Hmotnostní mód lze zapnout přepínačem umístěným vlevo nahoře. Tento mód lze za běhu programu libovolně přepínat.

5.5 Servis a Diagnostika

Na této obrazovce se sbírají veškeré statistická data o stroji. Výsledek těchto dat je využit v kapitole 6. Tato data jsou důležitá pro správnou regulaci podavače.



Obr. 16: Obrazovka Servisu a Diagnostiky

V servisní obrazovce jsou zobrazeny všechny statistické údaje o systému. Aktuální celková důležitost je výsledek 0/1 problému batohu. Tento výsledek je zprůměrován za celou dobu běhu stroje a zapsán v položce průměrná důležitost dávky. Max důležitost jedné kapsy je důležitost jedné kapsy, za celou dobu běhu programu.

Maximální počet cyklů nevyužitelnosti jedné kapsy a její průměr jsou klíčové hodnoty pro porovnání Inkrementálního a hmotnostního módu. Na této obrazovce lze vidět grafické zobrazení sledované veličiny v čase. V defaultním nastavení je sledována veličina aktuální váha.

5.6 Graf



Obr. 17: Obrazovka grafu

Na této obrazovce lze vidět grafické zobrazení sledované veličiny v čase. V defaultním nastavení je sledována veličina aktuální váha. Pro ilustrační účely se sledována veličina aktualizuje každou vteřinu.

6 VÝSLEDKY SIMULACE

Simulační model slouží k porovnání statistických metod při různém nastavení systému. Výsledkem této simulace je zjištění, za jakých podmínek a s jakou přesností dokáže kombinační váha zvážit vybranou dávku. Ohodnotit, jak dobře tento úkol algoritmus zvládá, lze pomocí několika kritérií. Počet kapes pro testování určitých kritérií bude vždy 8, 16, 24. Při každém testu bude provedeno 10 000 cyklů.

6.1 Kritérium legislativní

Podle zákona je nutné se vejít do vymezené meze [8]. Pokud je to pro výrobce prioritou a nejde mu o skutečnou přesnost, pak lze systém nastavit následovně.

Dolní mez je nastavena na zákonnou hodnotu. Horní mez je dávka, která je psána na obalu. Systém tedy bude hledat řešení v tomto intervalu. Očekávané chování je takové, že výsledná dávka bude v zákonném intervalu nebo bude vyhodnocena jako zmetek.

Byly zde použity různé rozsahy pro náhodnou dolní a horní hodnotu. Tyhle hodnoty představují podavač, jak plní kapsy produktem. Z průmyslové praxe jsou rozsahy hmotnosti jednotlivých kapes vybrány kolem 33 % hodnoty celkové dávky. Kapacita batohu je 100 a dolní mez je 95. Úspěšnost nalezení optimálního řešení představuje procentuální úspěšnost, kdy hmotnost dávky je v zákonném intervalu.

Varianta 8 kapes

Rozsah podavače 28-38 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	96,604	96,868	96,940
Průměrný počet použitých kapes	2,928	2,936	2,938
Průměrný počet cyklů nevyužitých kapes	3,946	4,165	3,819
Maximální počet cyklů nevyužitelnosti 1 kapsy	14	14	12
Počet neoptimálních řešení	733	647	630
Úspěšnost nalezení optimálního řešení	92,67 %	93,53 %	93,70 %

Tab. 6: Legislativní kritérium při 8 kapsách v rozsahu podavače 28-38

Hmotnostní mód číslo 2 podle očekávání maximalizuje hmotnost zvýšené dávky, a i v ostatních parametrech získává nejlepší výsledky. Úspěšnost nalezení optimálního řešení je limitována podavačem, který neposkytuje potřebnou variabilitu k nalezení optimálního řešení.

Rozsah podavače 20-40 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	97,396	97,33	97,507
Průměrný počet použitých kapes	3,247	3,54	3,251
Průměrný počet cyklů nevyužitých kapes	3,351	3,803	3,334
Maximální počet cyklů nevyužitelnosti 1 kapsy	8	18	8
Počet neoptimálních řešení	127	81	110
Úspěšnost nalezení optimálního řešení	98,73 %	99,19 %	98,90 %

Tab. 7: Legislativní kritérium při 8 kapsách v rozsahu podavače 20-40

Z důvodu zlepšení nastavení podavače se i úspěšnost nalezení optimálního řešení zlepšilo. Ostatní parametry se rovněž zlepšily.

Rozsah podavače 5-35 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	97,769	97,769	97,700
Průměrný počet použitých kapes	4,886	4,853	4,883
Průměrný počet cyklů nevyužitých kapes	2,385	2,938	2,217
Maximální počet cyklů nevyužitelnosti 1 kapsy	5	11	5
Počet neoptimálních řešení	10	24	11
Úspěšnost nalezení optimálního řešení	99,90 %	99,76 %	99,89 %

Tab. 8: Legislativní kritérium při 8 kapsách v rozsahu podavače 5-35

Z výsledků simulace je vidět, že velmi záleží na kvalitě podavače. Pokud podavač poskytuje dostatečnou variabilitu hmotnostních kapes, tak je stroj schopen téměř se stoprocentní úspěšností nalezení optimálního řešení. Hmotnostní mód číslo 2 dosahuje podobných nebo lepších výsledků než ostatní módy. Z hlediska využití je tedy tento mód nejvhodnější k použití.

Varianta 16 kapes

Vyšší počet kapes zastupuje potřebnou variabilitu u podavače. Očekává se zvýšení procentuální úspěšnosti na úkor maximálního počtu cyklů nevyužitelnosti jedné kapsy a samozřejmě průměrný počet cyklů, kdy kapsa nebude využita. Opět zde bude zkoumáno, který mód bude nejvhodnější k použití

Rozsah podavače 28-38 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	98,611	98,656	98,715
Průměrný počet cyklů nevyužitých kapes	7,552	8,256	7,259
Maximální počet cyklů nevyužitelnosti 1 kapsy	20	19	18
Počet neoptimálních řešení	117	103	85
Úspěšnost nalezení optimálního řešení	98,83 %	98,97 %	99,15 %

Tab. 9: Legislativní kritérium při 16 kapsách v rozsahu podavače 28-38

Rozsah podavače 20-40 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	98,019	97,944	98,081
Průměrný počet cyklů nevyužitých kapes	5,998	8,55	5,778
Maximální počet cyklů nevyužitelnosti 1 kapsy	9	30	10
Počet neoptimálních řešení	0	0	0
Úspěšnost nalezení optimálního řešení	100 %	100 %	100 %

Tab. 10: Legislativní kritérium při 16 kapsách v rozsahu podavače 20-40

Rozsah podavače 20-60 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	97,766	97,724	97,823
Průměrný počet cyklů nevyužitých kapes	9,860	9,323	8,967
Maximální počet cyklů nevyužitelnosti 1 kapsy	32	60	26
Počet neoptimálních řešení	84	31	56
Úspěšnost nalezení optimálního řešení	99,16 %	99,97 %	99,44 %

Tab. 11: Legislativní kritérium při 16 kapsách v rozsahu podavače 20-60

Se zvýšením počtu kapes lze pozorovat výrazné zlepšení v úspěšnosti nalezení optimálního řešení.

Varianta 24 kapes

Rozsah podavače 28-38 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	98,937	98,948	98,947
Průměrný počet cyklů nevyužitých kapes	10,492	12,195	8,967
Maximální počet cyklů nevyužitelnosti 1 kapsy	24	28	26
Počet neoptimálních řešení	17	14	14
Úspěšnost nalezení optimálního řešení	99,83 %	99,86 %	99,86 %

Tab. 12: Legislativní kritérium při 24 kapsách v rozsahu podavač 28-38

Rozsah podavače 20-40 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	98,306	97,961	98,299
Průměrný počet cyklů nevyužitých kapes	8,704	12,313	8,233
Maximální počet cyklů nevyužitelnosti 1 kapsy	13	41	12
Počet neoptimálních řešení	0	0	0
Úspěšnost nalezení optimálního řešení	100 %	100 %	100 %

Tab. 13: Legislativní kritérium při 24 kapsách v rozsahu podavače 20-40

Rozsah mezi 20-60 jednotek	Inkrementální mód	Hmotnostní mód	Hmotnostní mód č.2
Průměrná váha dávky	97,991	97,844	97,995
Průměrný počet cyklů nevyužitých kapes	13,915	13,595	12,495
Maximální počet cyklů nevyužitelnosti 1 kapsy	37	78	36
Počet neoptimálních řešení	5	2	1
Úspěšnost nalezení optimálního řešení	99,95 %	99,98 %	99,99 %

Tab. 14: Legislativní kritérium při 24 kapsách a rozsahu podavač 20-60

Při 24 kapsách nastavený trend dále pokračuje. Je zde ještě vyšší úspěšnost nalezení optimálního řešení, přestože podavač poskytuje velmi variabilní hodnoty. Přesnost při 24 kapsách ovšem doprovází nepříznivý jev, kdy počet cyklů nevyužitelnosti

jednotlivých kapes roste. Tento fakt lze velmi těžko mírnit. Jedno z řešení je nastíněno v následující kapitole.

6.2 Kritérium zrychlení dávky

Při preferenci zrychlit dávku je nutné zvýšit důležitost kapsy. Jestliže se kapsa nevyužije v daném cyklu, tak pro další cyklus se její důležitost zvýší o parametr p . Parametr $p=1$ již byl vyzkoušen v kritériu legislativním. Zvýšení parametru p se provede lineárně a exponenciálně. Jedná se o tedy o funkční závislost. Rozsah hodnot, které bude poskytovat podavač bude v rozmezí 20-40 jednotek.

6.2.1 Tabulka výsledků

Počet kapes = 8	$c(t)=2$	$c(t)=2*p$	$c(t)=p^2$
Průměrný počet cyklů nevyužitých kapes	3.341	3.328	3.349
Maximální počet cyklů nevyužitelnosti 1 kapsy	7	7	24
Počet kapes = 16	$c(t)=2$	$c(t)=2*p$	$c(t)=p^2$
Průměrný počet cyklů nevyužitých kapes	5.971	5.829	7.981
Maximální počet cyklů nevyužitelnosti 1 kapsy	9	8	83
Počet kapes = 24	$c(t)=2$	$c(t)=2*p$	$c(t)=p^2$
Průměrný počet cyklů nevyužitých kapes	8.665	8.280	15.605
Maximální počet cyklů nevyužitelnosti 1 kapsy	13	10	1020

Tab. 15: Tabulka výsledků při různých parametrech p

Z výsledků lze pozorovat, že počet cyklů t , kdy kapsa čeká na využití, je nejmenší při nastavení parametru p podle vztahu:

$$c(t) = 2 * p \quad (17)$$

6.3 Kritérium přesnosti

Přesnost bývá nejdůležitějším parametrem pro výrobce váhy. Se zvyšující tolerancí a větším počtem kapes se minimalizuje počet neoptimálních řešení. Cílová zvážená dávka je 100, ale pro jeho dosažení je kapacita batohu navýšená. Podavač je nastaven na dolní náhodnou hodnotu 20 a horní náhodnou hodnotu 40. V této kapitole jsou ukázány i módy, které nepracují ideálně (inkrementální a hmotnostní mód číslo 1). Tento experiment bylo nutné provést, aby bylo experimentálně dokázáno, který mód je nejoptimálnější.

Varianta 8 kapes

Dolní mez- 97, Horní mez-103	Inkrementální mód	Hmotnostní mód
Průměrná váha dávky	99,958	99,376
Průměrný počet použitých kapes	3,332	3,313
Průměrný počet cyklů nevyužitých kapes	3,27	3,786
Maximální počet cyklů nevyužitelnosti 1 kapsy	7	14
Počet neoptimálních řešení	103	109
Úspěšnost nalezení optimálního řešení	98,97 %	98,91 %

Tab. 16: Kritérium přesnosti při 8 kapsách a intervalu výsledků 97-103

Nepřesnost je stále vysoká a chybovost relativně nízká. Se snížením intervalu mezi horní a dolní mezí dojde k zpřesnění výsledné dávky, nicméně chybovost vzroste. Maximální počet cyklů nevyužitelnosti některé z kapes je pořád relativně nízký.

Dolní mez- 98, Horní mez-102	Inkrementální mód	Hmotnostní mód
Průměrná váha dávky	99,721	99,312
Průměrný počet použitých kapes	3,324	3,311
Průměrný počet cyklů nevyužitých kapes	3,328	3,788
Maximální počet cyklů nevyužitelnosti 1 kapsy	9	12
Počet neoptimálních řešení	220	264
Úspěšnost nalezení optimálního řešení	97,80 %	97,36 %

Tab. 17: Kritérium přesnosti při 8 kapsách a intervalu výsledků 98-102

Úspěšnost podle předpokladu klesá nicméně průměrná váha se včetně zmetků zlepšuje.

Dolní mez 99, Horní mez-101	Inkrementální mód	Hmotnostní mód
Průměrná váha dávky	99,38	99,003
Průměrný počet použitých kapes	3,313	3,294
Průměrný počet cyklů nevyužitých kapes	3,584	3,87
Maximální počet cyklů nevyužitelnosti 1 kapsy	11	14
Počet neoptimálních řešení	477	661
Úspěšnost nalezení optimálního řešení	95,53 %	93,39 %

Tab. 18: Kritérium přesnosti při 8 kapsách a intervalu výsledků 99-101

V tabulce je vidět, že při přísnějších dolních a horních mezí dochází k velmi časté chybovosti. Více než 1 %. Při zapnutém hmotnostním módu se výsledky zhoršují.

Varianta 16 kapes

Při vyšších kapsách se obecně počítá se zlepšením úspěšnosti nalezení optimálního řešení na úkor doby, kdy budou jednotlivé kapsy čekat na využití.

Dolní mez-99, Horní mez-101	Inkrementální mód	Hmotnostní mód
Průměrná váha dávky	100,07	99,997
Průměrný počet použitých kapes	3,336	3,333
Průměrný počet cyklů nevyužitých kapes	6,211	8,058
Maximální počet cyklů nevyužitelnosti 1 kapsy	12	28
Počet neoptimálních řešení	2	8
Úspěšnost nalezení optimálního řešení	99,98 %	99,92 %

Tab. 19: Kritérium přesnosti při 16 kapsách a intervalu výsledků 99-101

I při nastavení velmi malé tolerance úspěšnost nalezení optimálního řešení dosahuje téměř 100 %. Z toho důvodu je zde řešení pouze této varianty.

Varianta 24 kapes

Dolní mez-99, Horní mez-101	Inkrementální mód	Hmotnostní mód
Průměrná váha dávky	100,145	99,997
Průměrný počet použitých kapes	3,338	3,333
Průměrný počet cyklů nevyužitých kapes	8,823	8,058
Maximální počet cyklů nevyužitelnosti 1 kapsy	14	28
Počet neoptimálních řešení	0	8
Úspěšnost nalezení optimálního řešení	100 %	99,92 %

Tab. 20: Kritérium přesnosti při 24 kapsách a intervalu výsledků 99-101

Při použití inkrementálního nebo hmotnostního módu číslo 1 je průměrná váha zvážené dávky velmi přesná. Jediná nevýhoda je v tom, že cíleně překračují cílovou hodnotu.

6.3.1 Kritérium přesnosti podle hmotnostního módu č.2

Jelikož hmotnostní mód číslo 2 se vždy snaží maximalizovat hmotnost dané dávky, není zde nutné nastavit kapacitu batohu přesahující cílovou hodnotu. Dolní mez se nastaví minimálně na zákonnou hodnotu [8]. Pokud ovšem úspěšnost nalezení optimálního řešení je kolem 99 % je žádoucí dolní mez zvýšit. To má za následek zlepšení průměrné zvážené dávky.

6.4 Ideální výsledky

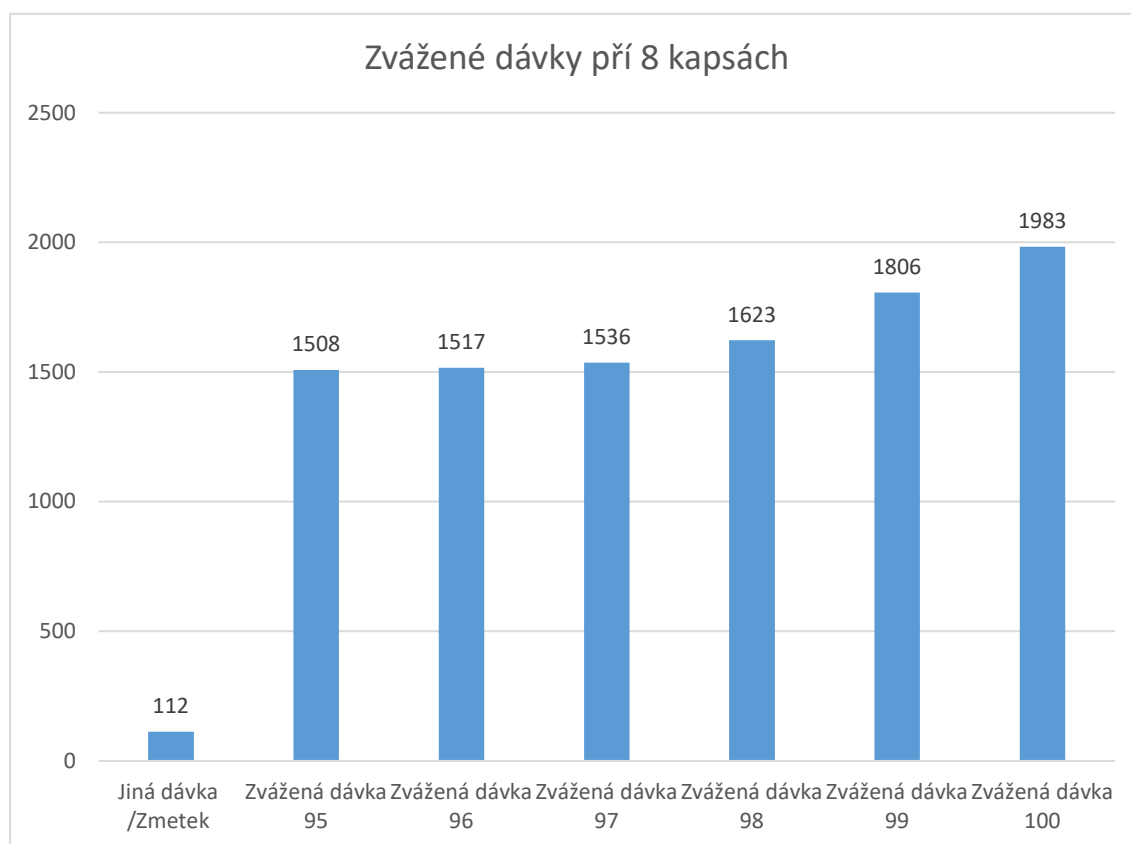
V kapitole 6.3.1 bylo popsáno, jak lze tyto optimální výsledky získat. V této kapitole budou představeny tabulky a grafy výsledky při nastavení podavače v mezích 20 až 40. Simulačně bylo provedeno 10 000 cyklů. Cílová hodnota dávky je 100 jednotek.

Varianta 8 kapes

Dolní mez-95, Horní mez-100	Hmotnostní mód č.2
Průměrná váha dávky	97,507
Průměrný počet použitých kapes	3,251
Průměrný počet cyklů nevyužitých kapes	3,344
Maximální počet cyklů nevyužitelnosti 1 kapsy	8
Počet neoptimálních řešení	110
Úspěšnost nalezení optimálního řešení	98,90 %

Tab. 21: Výsledky ideálního nastavení pro 8 kapes

Průměrná váha dávky je s odchylkou 0.1 jednotky maximální počet cyklů nevyužitelnosti jedné kapsy je velmi nízký stejně jako její průměr. Při porovnání s variantou inkrementálního módu lze usoudit, že úspěšnost nalezení je vyšší. Průměrná hodnota zvážené dávky je nižší, ale to jen z toho důvodu, že nedochází k překročení cílové váhy.



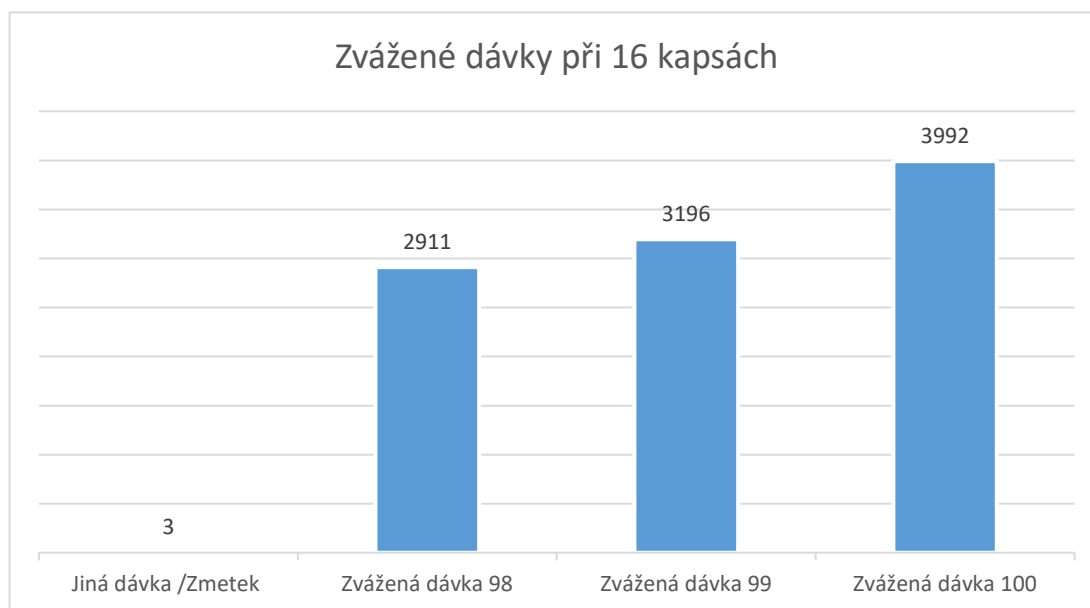
Obr. 18: Zvážené dávky pro 8 kapes

Varianta 16 kapes

Dolní mez-98, Horní mez-100	Hmotnostní mód č.2
Průměrná váha dávky	99,103
Průměrný počet použitých kapes	3,304
Průměrný počet cyklů nevyužitých kapes	6,079
Maximální počet cyklů nevyužitelnosti 1 kapsy	14
Počet neoptimálních řešení	3
Úspěšnost nalezení optimálního řešení	99,97 %

Tab. 22: Výsledky ideálního nastavení pro 16 kapes

Průměrná váha je horší než u varianty s 8 kapsami ovšem z hlediska snížení horní meze, kdy algoritmus dovoluje pouze výsledky s maximální váhou 100 jde o zlepšení. Ostatní parametry zůstaly v rozumných mezích. I v porovnání s inkrementálním módem jsou hodnoty velmi podobné. Ovšem v tomto případě nedošlo k překročení cílové váhy.

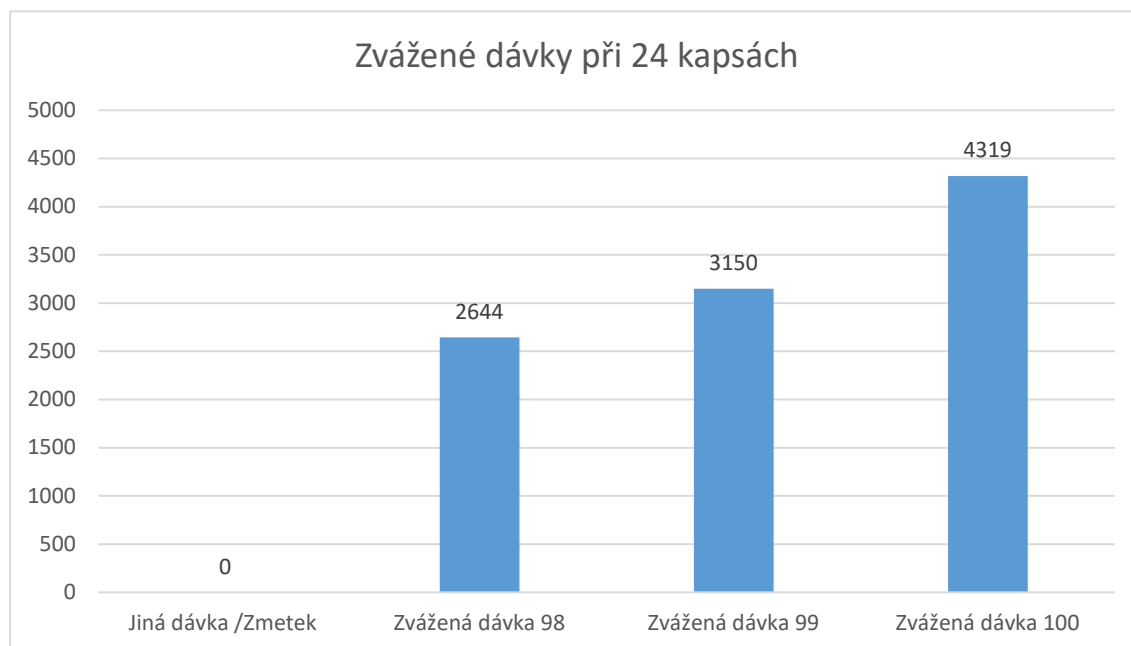


Obr. 19: Zvážené dávky pro 16 kapes

Varianta 24 kapes

Dolní mez-98, Horní mez-100	Hmotnostní mód č.2
Průměrná váha dávky	99,163
Průměrný počet použitých kapes	3,306
Průměrný počet cyklů nevyužitých kapes	8,494
Maximální počet cyklů nevyužitelnosti 1 kapsy	15
Počet neoptimálních řešení	0
Úspěšnost nalezení optimálního řešení	100 %

Tab. 23: Výsledky ideálního nastavení pro 24 kapes



Obr. 20: Zvážené dávky pro 24 kapes

Průměrná váha zvážené dávky navzdory zvýšení počtu kapes dosahuje podobné hodnoty jako u varianty s 16 kapsami. Ostatní parametry zůstávají podobné. Z toho lze usoudit, že není nutné investovat do varianty se 24 kapsami. Zlepšení není příliš významné.

6.5 Rychlost algoritmu

Byl použit systém SIMOTION, který se vyznačuje velkou rychlostí. Celý tento systém bude řídit nejen tuto rozhodovací část, které kapsy se vysypou, ale i celý stroj. Je tedy nutné zajistit, aby rychlost provádění byla vysoká.

Výsledek rychlosti provádění je takový, že i při 24 kapsách dojde k rozhodnutí do 4 milisekund. Pro testovací účely byl sestaven nereálný případ 800 kapes. Při této konfiguraci bylo dosaženo rychlosti 16 milisekund na jeden cyklus.

Závěrem lze říci, že systém SIMOTION je více než dostačující, co se týče rychlosti.

7 ZHODNOCENÍ A DISKUZE

Řízení kombinační váhy za pomoci algoritmu 0/1 problému batohu s dolní mezí je vyhovující. Z výsledků simulace vychází, že hmotnostní mód číslo 2 je neoptimálnější. Tudíž cena kapsy se rovná její hmotnosti. Při nevyužití kapsy v daném cyklu se cena kapsy zvýší o hodnotu +1.

Toto nastavení je kompromisem rychlého a rovnoměrného vypuštění kapes a zároveň maximalizace celkové zvážené dávky v rámci daného intervalu. Jistě lze upřednostnit rychlost vypuštění kapes, případně průměrné přesnosti celkově zvážené dávky. Z výsledků lze pozorovat, že stroj dosahuje stanovené přesnosti, a i ostatní parametry jsou uspokojivé. Tento algoritmus je tedy použitelný v průmyslové praxi.

Ve vlastním řešení této diplomové práce není vyřešen případ nedefinovaného chování. V případě, že se nenajde řešení, vyhodnotí se dávka jako zmetek. Tento stav nebyl ošetřen z důvodu demonstrace úspěšnosti algoritmu v jeho základní podobě.

Jedno z řešení, jak nedefinovaný stav vyřešit, je dočasné zvýšení horní meze. Horní mez by tedy byla dynamická proměnná, která se v případě nenalezení optimálního řešení zvýší a algoritmus se provede znovu.

Pro robustnost aplikace je třeba vyřešit problém jistoty vypuštění všech kapes v dohledné době. I v případě postupného zvyšování ceny kapsy není zaručeno, že kapsa bude vypuštěna. Z výsledků simulace je vidět, že tento jev nenastává při správném nastavení podavače. Ovšem v případě, že kapsa bude mít hmotnost přesahující celkovou váhu dávky, pak k jejímu vypuštění nikdy nedojde.

Dále je zde problém, kdy zákazník požaduje vypuštění kapsy do určitého počtu cyklů. Tento stav může být vyřešen tak, že daná kapsa bude vypuštěna navzdory její ceně a o její hmotnost se horní mez sníží. Poté se provede algoritmus s modifikovanou horní mezí. Opět tato modifikace nebyla naprogramována z důvodu demonstrace algoritmu v základní podobě.

Celkové zhodnocení algoritmu na základě výsledků simulace lze hodnotit jako uspokojivé. Všechny požadavky byly splněny. Robustnost algoritmu byla ukázaná při různorodém nastavení podavače. Ve všech případech dosahovaly výsledky simulace velmi dobrých výsledků. Algoritmus je deterministický. Tato vlastnost je důležitá pro průmysl. Je totiž důležité, aby stroj v každém stavu věděl, jak se má zachovat.

8 ZÁVĚR

Předložena diplomová práce pojednává o řízení kombinační váhy. Úkolem bylo sestavit algoritmus, který dokáže vybrat kapsy způsobem, že jejich celková hmotnost konverguje k cílené váze a zároveň využitelnost kapes zůstane rovnoměrná. Tento hlavní cíl byl splněn.

V první části práce je popsán řídicí systém SIMOTION, na kterém je simulace spuštěna. Hlavní výhodou tohoto systému je ve vysokém výpočetním výkonu a zároveň v komfortu při psaní kódu. Je zde totiž možnost použití OOP, což zvyšuje znovupoužitelnost programu.

Druhá část práce je věnována konstrukci kombinační váhy. Tato kapitola je důležitá pro pochopení mechaniky stroje a jeho limitů. Algoritmus vychází z těchto poznatků.

Ve třetí části je popsán samotný algoritmus. Jde o jednu stěžejních částí práce. Jsou popsány požadavky na řízení stroje, hlavní idea a matematický popis algoritmu. Konkrétně se jedná o problém batohu s definovanou dolní mezí. Je zde popsáno, kdy algoritmus dosáhne nedefinovaného chování.

Ve čtvrté části je popsán vzhled HMI, které bylo v rámci demonstračních důvodů naprogramováno. Jsou zde popsány obrazovky a jejich funkčnost. Kód, který tohle vše řídí, je k dispozici v příloze.

Pátá část se věnuje výsledkům simulace. Dojde zde ke zhodnocení a závěru, který mód je neoptimálnější pro řízení. Jsou zde tabulky popisující jednotlivé statistické parametry.

V poslední části je popsáno zhodnocení algoritmu a návrhy na jeho další vývoj. Jsou zde popsány způsoby, jak vyřešit nedefinované chování a dále jsou diskutována další různá vylepšení podle požadavků zákazníka.

9 SEZNAM POUŽITÉ LITERATURY

- [1] Siemens Global Website. *SIMOTION – the high-end motion control system* [online]. Berlín, Německo: Siemens, c1996-2018 [cit. 2018-03-09]. Dostupné z: <https://www.siemens.com/simotion>
- [2] *SIMOTION Runtime Basic function*. Erlangen Německo, 2017. Dostupné také z: <https://support.industry.siemens.com/cs/document/109749942/motion-control-basic-functions?dti=0&pnid=14505&lc=en-WW>
- [3] BERETTA, Alessia. Optimal Targets Setup of a Multihead Weighing Machine. Milano, Italy, 2016. Disertační práce. Politecnico di Milano. Vedoucí práce Dr. Quirico Semeraro.
- [4] GARCÍA-DÍAZ, J.C. and Pulido-Rojano, A., Monitoring and control of the multihead weighing process through a modified control chart. *DYNA* 84(200), pp.135-142, 2017.
- [5] *SIMOTION V4.5 from Siemens* [online]. 23. 11. 2016 [cit. 2019-05-04]. Dostupné z: https://www.youtube.com/watch?v=aHdfaeI9E_k
- [6] ST-Structure text. *Industry Support Siemens* [online]. Erlangen: Siemens, 2018 [cit. 2019-05-04]. Dostupné z: <https://support.industry.siemens.com/cs/document/109757326/st-structured-text?dti=0&pnid=14505&lc=en-CZ>
- [7] LINEÁRNÍ VÁHA Appec Astro. *Appec Astro* [online]. Vlašim: Appec Group, 2019 [cit. 2019-05-05]. Dostupné z: <https://www.appecastro.cz/cz/linearni-vaha-ev/>
- [8] ČESKO. § 1 písm. a) bod 3. vyhlášky č. 328/2000 Sb., Ministerstva průmyslu a obchodu o způsobu zhotovení některých druhů hotově baleného zboží, jehož množství se vyjadřuje v jednotkách hmotnosti nebo objemu. In: *Zákony pro lidi.cz* [online]. © AION CS 2010-2019 [cit. 5. 5. 2019]. Dostupné z: <https://www.zakonyprolidi.cz/cs/2000-328#p1-a-3>
- [9] KLAPKA, J.; DVOŘÁK, J. Úvod do operační a systémové analýsy. Učební texty vysokých škol. Učební texty vysokých škol. Brno: VUT v Brně, Fakulta strojního inženýrství, 2002. 92 s. ISBN: 80-214-1839- 7.
- [10] DVOŘÁK, Jiří. Dynamické programování: Brno, 2019. Dostupné také z: https://moodle.vutbr.cz/pluginfile.php/482135/mód_resource/content/2/DP/DP.pdf
- [11] BRAUN, Michael a Wolfgang HORN. Object-Oriented Programming with SIMOTION: Fundamentals, Program Examples and Software Concepts according to IEC 61131-3. John Wiley, 2017. ISBN 978-3-89578-456-9.
- [12] MICHAEL, Garey a Johnson DAVID. Computers and Intractability: A Guide to the Theory of NP-completeness. Ilustrované vydání, dotisk. New York: W. H. Freeman, 1979. ISBN 0716710447.
- [13] MICHAEL, Garey a Johnson DAVID. Computers and Intractability: A Guide to the Theory of NP-completeness. Ilustrované vydání, dotisk. New York: W. H. Freeman, 1979. ISBN 0716710447.
- [14] Ishida Europe. Ishida Europe [online]. Vlašim [cit. 2019-05-23]. Dostupné z: https://www.ishidaeurope.cz/eu/cz/vyrobky/multihead_weighing_machines/multi_head_weighers/rv_multihead_weigher.cfm
- [15] Hristakeva, Maya & Shrestha, Dipti [online]. (2019). Different Approaches to Solve the 0/1 Knapsack Problem. [cit. 2019-05-23]. Dostupné z: http://www.micsymposium.org/mics_2005/papers/paper102.pdf

10 SEZNAM OBRÁZKŮ

Obr. 1: Použití systému SIMOTION [1]	17
Obr. 2: Systémová architektura Simotion [2].....	18
Obr. 3: Všeobecná reprezentace konceptu paměti [2].....	19
Obr. 4: Systém provádění úloh v SIMOTION	21
Obr. 5: Princip round robin [2].....	22
Obr. 6: Časová alokace v SIMOTION SCOUT [2]	22
Obr. 7: Konstrukce kombinační váhy [3]	24
Obr. 8: Lineární váha [7]	25
Obr. 9: Kombinační váha [7].....	26
Obr. 10: Diagram aktivit inkrementálního módu	34
Obr. 11: Diagram aktivit hmotnostního módu	35
Obr. 12: Hmotnostní mód č.2	36
Obr. 13: Základní obrazovka.....	38
Obr. 14: Obrazovka krokového módu	39
Obr. 15: Non-stop mód.....	40
Obr. 16: Obrazovka Servisu a Diagnostiky	41
Obr. 17: Obrazovka grafu.....	42
Obr. 18: Zvážené dávky pro 8 kapes	50
Obr. 19: Zvážené dávky pro 16 kapes	51
Obr. 20: Zvážené dávky pro 24 kapes	52

11 SEZNAM TABULEK

Tab. 1: Zadání příkladu 1	29
Tab. 2: Matice řešení příkladu 1	30
Tab. 3: Zadání příkladu 2	31
Tab. 4: Řešení matice příkladu 2	32
Tab. 5: Optimální strategie příkladu 2	32
Tab. 6: Legislativní kritérium při 8 kapsách v rozsahu podavače 28-38	43
Tab. 7: Legislativní kritérium při 8 kapsách v rozsahu podavače 20-40	44
Tab. 8: Legislativní kritérium při 8 kapsách v rozsahu podavače 5-35	44
Tab. 9: Legislativní kritérium při 16 kapsách v rozsahu podavače 28-38	45
Tab. 10: Legislativní kritérium při 16 kapsách v rozsahu podavače 20-40	45
Tab. 11: Legislativní kritérium při 16 kapsách v rozsahu podavače 20-60	45
Tab. 12: Legislativní kritérium při 24 kapsách v rozsahu podavač 28-38	46
Tab. 13: Legislativní kritérium při 24 kapsách v rozsahu podavače 20-40	46
Tab. 14: Legislativní kritérium při 24 kapsách a rozsahu podavač 20-60	46
Tab. 15: Tabulka výsledků při různých parametrech p	47
Tab. 16: Kritérium přesnosti při 8 kapsách a intervalu výsledků 97-103	48
Tab. 17: Kritérium přesnosti při 8 kapsách a intervalu výsledků 98-102	48
Tab. 18: Kritérium přesnosti při 8 kapsách a intervalu výsledků 99-101	48
Tab. 19: Kritérium přesnosti při 16 kapsách a intervalu výsledků 99-101	49
Tab. 20: Kritérium přesnosti při 24 kapsách a intervalu výsledků 99-101	49
Tab. 21: Výsledky ideálního nastavení pro 8 kapes.....	50
Tab. 22: Výsledky ideálního nastavení pro 16 kapes.....	51
Tab. 23: Výsledky ideálního nastavení pro 24 kapes.....	51

OBSAH PŘILOŽENÉHO CD

Název	Popis
2019_DP_Capovsky_Alexandr_171227.pdf	Textová část diplomové práce
SimulationModel3.0.zap15	Simulační model v TIA Portálu 15.1