

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Modelování a simulace – dokumentace projektu

Příprava a rozvoz jídel společností Siesta pizza

Téma č. 9: Diskrétní model z oblasti služeb a dopravy

7. prosince 2020

Adam Grünwald

Dominik Nejedlý

Obsah

1	Úvod	1
1.1	Autoři, zdroje	1
1.2	Ověření validity	1
2	Rozbor tématu a použitých metod/technologií	1
2.1	Použité postupy	1
2.2	Popis původu použitých metod/technologií	2
3	Koncepce modelu	2
3.1	Popis konceptuálního modelu	3
3.1.1	Generování objednávek	3
3.1.2	Celková příprava pokrmu	4
3.1.3	Rozvoz	5
4	Architektura simulačního modelu	6
4.1	Mapování konceptuálního modelu do simulačního modelu	6
4.2	Spuštění simulačních experimentů	6
5	Podstata simulačních experimentů a jejich průběh	7
5.1	Postup experimentování	7
5.2	Experimenty	7
5.2.1	Experiment 1	7
5.2.2	Experiment 2	8
5.2.3	Experiment 3	8
5.2.4	Experiment 4	9
5.2.5	Experiment 5	10
5.3	Závěry experimentů	10
6	Shrnutí simulačních experimentů a závěr	10

1 Úvod

Obsahem této práce je sestavování modelu [2, snímek 7], který reprezentuje výrobu a distribuci pizzy po okrese Blansko společností Siesta pizza, a jeho následná simulace [2, snímek 33]. Cílem experimentů nad tímto modelem prováděných je průzkum možností dalšího růstu společnosti. Během pandemie totiž došlo k nárůstu počtu objednávek napříč rozvážkovými službami [1, Lidovky rovoz] a je tedy namístě řešit, kde je potřeba navýšit kapacity tak, aby to bylo efektivní. Bude zkoumáno, na jaké části systému [2, snímek 7] se vyplatí zaměřit a zvýšit jejich kapacity tak, aby se výrazně nezvedla doba přípravy pokrmů a dodání zákazníkům.

1.1 Autoři, zdroje

Práci vypracovali Adam Grünwald a Dominik Nejedlý při fakultě FIT VUT v Brně v rámci předmětu Modelování a simulace. Technické informace byly čerpány z materiálů k předmětu Modelování a simulace. Faktické informace týkající se problematiky byly získány od majitele pizzerie pana Kamila Brehového na osobních konzultacích.

1.2 Ověření validity

K ověření validity [2, snímek 37] nám posloužily informace od pana Kamila Brehového získané na osobních setkáních přímo v sídle firmy. Od něj jsme získali potřebná data pro sestavení modelu a ověření jeho validity. Tu jsme ověřili také pomocí experimentů a následným srovnáním s reálnými daty.

2 Rozbor tématu a použitých metod/technologií

Společnost Siesta pizza přijímá objednávky od neděle do čtvrtku od 11:00 do 21:00. V pátek a sobotu od 11:00 do 22:00. Po této době ještě dochází k doděláním již přijatých objednávek a jejich předání zákazníkům. Na provozovně pracují v současné době maximálně tři zaměstnanci, kteří se starají o výrobu pizzy. Tento počet závisí na tom, jestli se jedná o pátek či sobotu, kdy je nápor větší, nebo ostatní dny, kdy si provozovna vystačí se dvěma zaměstnanci.

Objednaná jídla poté vyrábí a dodávají svým zákazníkům buď formou výdeje v místě provozovny, nebo za pomoci svých dvou aut, která veškeré objednávky k doručení rozváží na smluvená místa. Během pracovního týdne denní počet pizz nepřesáhne 200 kusů, naopak v pátek a sobotu se tento počet pohybuje až kolem 400 pizz za jediný den. Víme také, že v průměru připadají dvě jídla na jednu objednávku, přičemž nejvíce objednávek je pouze na jedno jídlo a dále, že s rostoucím počtem pizz v objednávce klesá pravděpodobnost jejího obdržení. Jinak to bývá v případě slevových akcí X plus jedna zdarma. Tyto situace však v našem modelu neuvažujeme.

Po přijetí objednávky dochází k zahájení procesu výroby. Příprava pizzy do pece trvá pro jednu pizzu ve 40 % případů, kdy se jedná o složitější druh pizzy, v průměru 3.5 minuty. V případě lehčích pizz na přípravu, které tvoří 60 % objednávek, se tato doba pohybuje kolem dvou minut. Následně jsou tyto pizzy dány do jedné ze dvou pecí, každá s kapacitou 4 pizz, kde se pečou dvě minuty. Poté jsou zaměstnancem vyndány a zabaleny, čímž jsou připraveny k předání zákazníkovi nebo řidičovi.

Zákazníci si pro své pizzy chodí v 35% případů rovnou na provozovnu. V 65% případů jsou jim pak pizzy doručovány rozvozem. Ten mají na starosti dvě Škody Citygo (s roky výroby 2014 a 2016), které jsou plněny maximálně osmi kusy pizzy (pokud se nejedná o jedinou objednávku nad 8 kusů, která je výjimečná - 0.5 % objednávek), aby pizzy netrávily při více objednávkách velkou dobu na cestě a neztrácely tak na chuti a kvalitě. Auto po naplnění kapacity nebo pokud již nemá co nakládat, odjíždí k zákazníkům a vrací se v průměru za 20 minut.

2.1 Použité postupy

Model byl vytvořen v programovacím jazyce C++ s použitím simulační knihovny SIMLIB [3]. Postupy při vytváření Petriho sítě [2, snímek 123] a použití knihovny SIMLIB byly čerpány z materiálů k předmětu IMS.

2.2 Popis původu použitých metod/technologií

Pro návrh Petriho sítě bylo použito online nástroje `draw.io`¹. Při programování bylo použito tříd a funkcí jazyka C++². Překlad byl proveden nástrojem g++³. Knihovnu SIMLIB [3] jsme získali z jejích oficiálních stránek⁴.

3 Koncepce modelu

Pro konceptuální model [2, snímek 48] nad tímto systémem hromadné obsluhy [2, snímek 136] vybíráme pouze informace, které jsou pro něj podstatné. Jelikož je cílem namodelovat pracovní den s maximálním počtem objednávek, zaměříme se na průběh sobotní směny. V úvahu tak tedy bereme, že na provozovně se nachází tři zaměstnanci, dvě auta, a průměrně za den provozovna přijme objednávky celkem na 400 kusů pizz.

Denní doba v modelu uvažována není, jelikož výkyvy počtů objednávek kolem poledne, pozdního odpoledne a následně večera nejsou během sobot výrazné. Je tedy použito exponenciální rozložení [2, snímek 91] se střední dobou příchodu objednávky vypočítanou jako:

$$\text{střední doba příchodu objednávky [s]} = \frac{\text{doba přijímání objednávek [s]}}{\frac{\text{počet jídel za den}}{\text{průměrný počet jídel v objednávce}}}$$

Tedy při průměrném počtu dvou pizz v objednávce lze snadno odvodit počet objednávek z celkového počtu jídel za den a na základě toho spočítat střední dobu mezi příchody objednávek do systému.

Počet jednotlivých pizz v objednávce také není závislý na denní době, ovšem aby byla zachována určitá náhodnost a nebylo počítáno vždy jen se dvěma jídly na objednávku, jsou počty pizz v objednávce generovány náhodně pomocí rozhodovacího algoritmu. Při přijetí objednávky je vždy vygenerováno jedno jídlo a poté je šance 50 %, že se vygeneruje další, po jehož vytvoření se rozhodování na generování opakuje. Tento způsob zajišťuje, že se nejvíce objednávek bude týkat právě jednoho jídla a s rostoucím počtem pizz v objednávce bude také klesat počet takových objednávek, přičemž průměrný počet jídel v objednávce bude zachován, jelikož platí:

$$\text{průměrný počet pizz v objednávce} = \sum_{n=1}^{\infty} \frac{n}{2^n} = 2$$

V modelu není modelováno přijímání telefonických objednávek, jelikož ty může zaměstnanec přijímat během přípravy pizzy a nestráví tím tak tedy moc času.

Po přijetí objednávky na určitý počet pizz se jednotlivá jídla řadí do fronty a čekají na uvolnění zaměstnance ze skladu zaměstnanců, kterého si pizza zablokuje a zaměstnanec ji připraví. Připravené pizzy se vkládají do fronty na obsazení pece.

Pro zjednodušení modelu nejsou během přípravy a rozvozu uvažovány jednotlivé objednávky, ale konkrétní pizzy. K tomuto způsobu řešení jsme se rozhodli, jelikož nezkoumáme dobu, za kterou je objednávka doručena k zákazníkovi, ale dobu, za kterou je pizza připravená k výdeji na provozovně, nebo si ji přebírá řidič.

Čas, za který zaměstnanec připraví jednu pizzu do pece, je modelován za pomoci normálního rozložení [2, snímek 93] se středem tři a půl minuty s odchylkou deset sekund pro složitější pizzu, a středem dvě minuty s odchylkou pět sekund pro pizzu jednodušší. Pro jednoduchost je uvažována jen jedna pec s celkovou kapacitou na 8 kusů pizz, nedojde však ke snížení validity, jelikož pece není třeba rozlišovat, a tudíž se jejich kapacita může sečíst.

Po uplynutí dvou minut, po které se pizza peče, jsou hotová jídla řazena do fronty na zabalení. Jakmile se v této frontě nějaké jídlo objeví, linka (obstarávající balení) s vyšší prioritou procesu [2, snímek 139] si zabere

¹<https://app.diagrams.net/>

²<https://cppreference.com/w/cpp>

³<https://gcc.gnu.org/>

⁴<https://www.fit.vutbr.cz/~peringer/SIMLIB/>

volného zaměstnance, který všechna doposud hotová jídla čekající v této frontě zabalí. Zabalení jedné pizzy trvá rovnoměrně 10 - 15 s.

Po zabalení je pizza buď vydána zákazníkovi na prodejně a opouští systém, nebo si stoupá do fronty na rozvoz. Procesy aut s řidiči čekají až budou mít co rozvézt. Jakmile se objeví pizzy ve frontě na rozvoz, jsou naloženy do auta. Doba nakládky se řídí rovnoměrným rozložením [2, 89] v intervalu od jedné do tří minut. Do auta může být současně dáno maximálně 8 pizz. K většímu naložení dochází jen ve výjimečných situacích, kdy přijde veliká objednávka na osm a více kusů. Toto přeložení jindy nenastane a tedy je v našem modelu dále neuvažujeme. Pizzy připravené na rozvoz nad počet osmi čekají dále ve frontě na další auto.

Doba řidiče na cestě je modelována normálním rozložením se středem 20 minut a odchylkou 3 minuty. Po této době se auto s řidičem vrací zpět a opět čeká na pizzy připravené na rozvoz. Všechna jídla, která auto veze, jsou považována za doručená ve chvíli, kdy se auto vrátí z dané rozvážky, jelikož nelze přesně určit, kdy byla jednotlivá jídla předána zákazníkům.

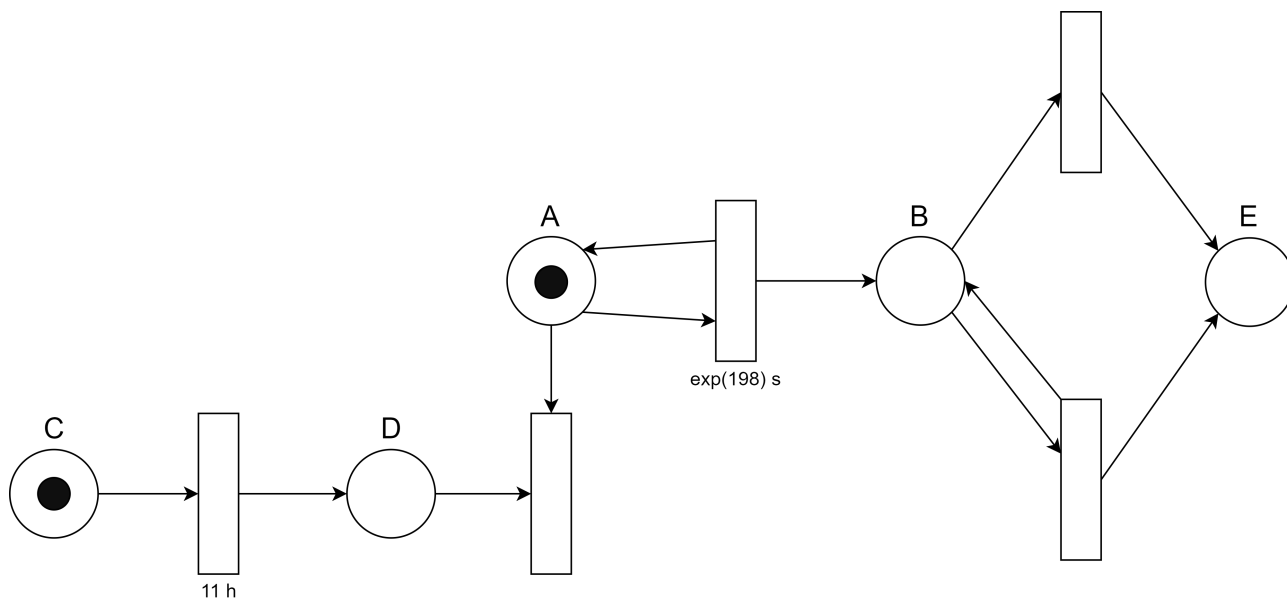
V modelu se nachází také časový interval směny, pomocí kterého zkoumáme, jestli po ukončení přijímání objednávek jsou všechny pizzy předány zákazníkům a auta jsou zpět na provozovně do konce pracovní doby. Po uplynutí tohoto intervalu se simulace ukončí a kontroluje se, jestli se vše opravdu podařilo.

3.1 Popis konceptuálního modelu

Model je reprezentován pomocí Petriho sítě a je rozdělen do tří částí.

3.1.1 Generování objednávek

Tato část modeluje generování objednávek, následné generování jídel ke každé z nich a současně časovač doby, po kterou jsou objednávky přijímány. Vygenerovaná jídla dále postupují do druhé části systému 3.1.2, kde probíhá jejich příprava.



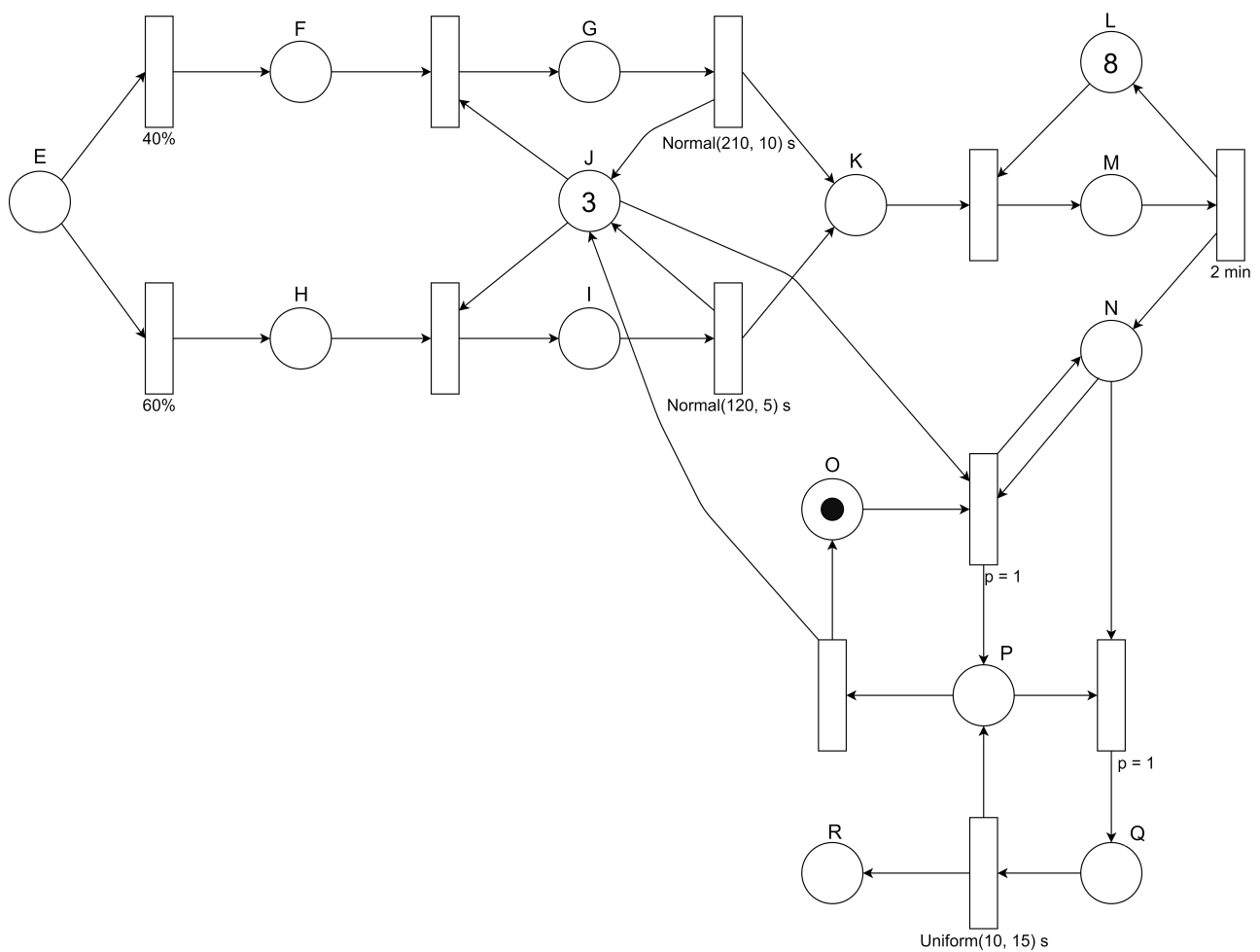
Obrázek 1: Petriho síť - příchod objednávek do systému

- **A** generování objednávek
- **B** generování jídel k objednávce

- **C** zahájení příjmu objednávek
- **D** konec přijímání objednávek
- **E** jednotlivá jídla k přípravě

3.1.2 Celková příprava pokrmu

Zde je řešena příprava jednotlivých pizz, jejich pečení v peci a následné balení. Je zde modelována rozdílná náročnost pizz na přípravu, která se odráží v délce jejich přípravy. O tu se starají tři zaměstnanci. Připravené pizy následně obsadí místo v peci, kde se pečou dvě minuty a poté čekají na zabalení. Následně putují do další části systému 3.1.3, kde si je buď přebírají zákazníci nebo řidiči.



Obrázek 2: Petriho síť - výroba pizz

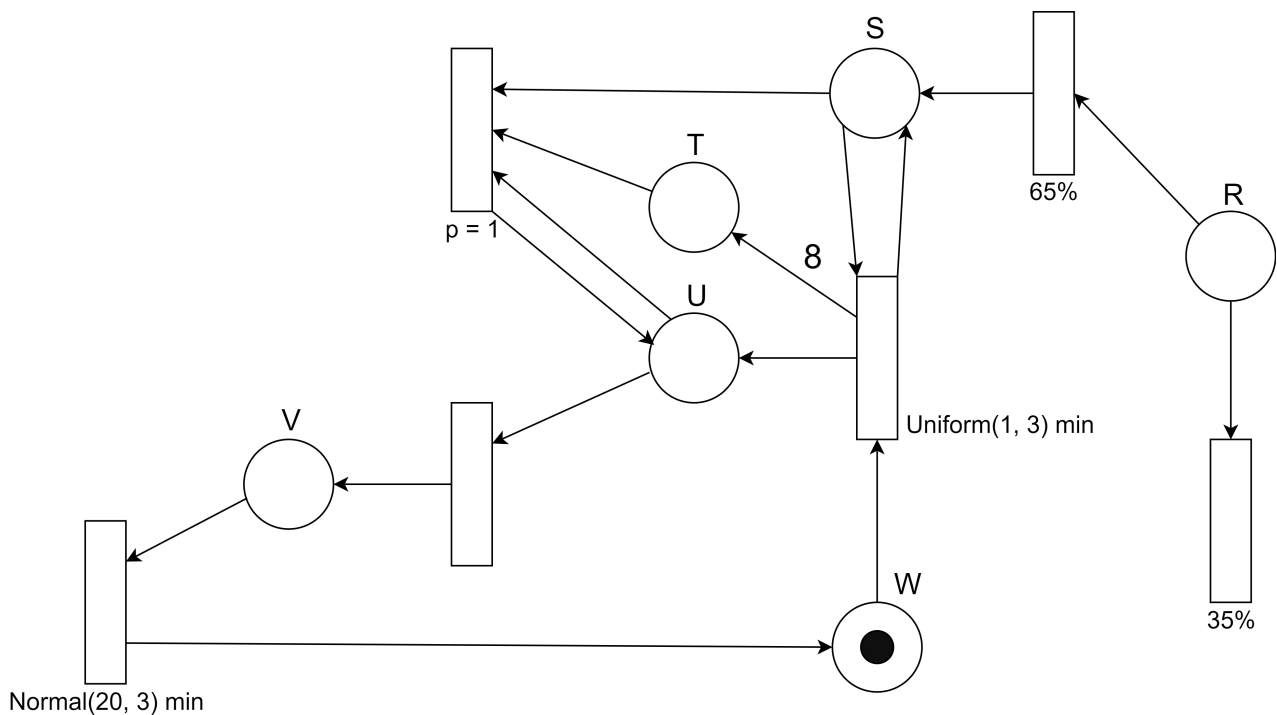
- **E** jednotlivá jídla k přípravě
- **F** složitější pokrmy čekající na přípravu
- **G** příprava složitějšího jídla

- **H** jednodušší pokrmy čekající na přípravu
- **I** příprava jednoduššího jídla
- **J** volní zaměstnanci
- **K** jídla připravená k pečení
- **L** volná místa v peci
- **M** pečení jídla
- **N** jídla připravená k balení
- **O** identifikátor, zdali někdo obstarává balení jídel
- **P** zaměstnanec balící připravená jídla
- **Q** balení jídla
- **R** jídla připravená k výdeji

3.1.3 Rozvoz

V této části Petriho sítě jednotlivé pizzy buď vystupují ze systému, jelikož si je přeberou zákazníci, nebo putují dále do fronty pizz určených k rozvozu. Odtud si je vybírají řidiči, kteří se snaží zaplnit kapacitu aut, avšak pokud nemají co naložit, odjíždí s jídly k zákazníkům a následně přijíždí zpět.

Níže znázorněná Petriho síť představuje linku jednoho auta. V případě více aut má každé z nich tuhle samostatnou linku. Pokud má tedy náš systém ve výchozím stavu auta dvě, potom se v celkové Petriho síti objeví dvakrát i tato linka.



Obrázek 3: Petriho síť - výdej a rozvoz pizz

- **R** jídla připravení k výdeji
- **S** jídla určená k rozvozu
- **T** volná místa v autě
- **U** nakládání auta
- **V** auto rozváží jídla zákazníkům
- **W** identifikátor, zdali je auto dostupné

4 Architektura simulačního modelu

Spuštěním simulačního modelu dojde k 1000 běhům simulačních experimentů, kdy každý z nich reprezentuje jeden pracovní den - sobotu. Před každým z běhů dojde k nové inicializaci modelového času, front a statistik pro jednotlivé běhy. Jednotka modelového času odpovídá jedné vteřině. Po ukončení každého z běhů lze z vypsaných statistik zjistit, jestli se do konce pracovní směny podařilo doručit všechna jídla, a zdali se auta stihla vrátit zpět. Ze statistik lze také vyčíst průměrné vytížení zaměstnanců na provozovně, peci a aut s řidiči. Dále pak minimální, průměrný a maximální čas, za který byla pizza připravena k předání zákazníkovi, předána řidiči a jejich společnou statistiku. Po skončení všech běhů se vypíše jejich souhrnné statistiky.

4.1 Mapování konceptuálního modelu do simulačního modelu

Časovač ukončující přijímání objednávek je implementován jako proces s názvem `StopOrderGenerator`, který na začátku směny naplánuje sám sebe na dobu, kdy má dojít ke konci přijímání objednávek zajišťovaných událostí `OrderGenerator`. V momentě, kdy je `StopOrderGenerator` znovu aktivován, zruší proces `OrderGenerator` a vytvoří proces `ShiftEnd`. Ten čeká, až jsou předány všechna jídla a auta jsou zpět na provozovně.

Událost `OrderGenerator` vytváří nové procesy `Meal` pro danou objednávku. Počet jídel v objednávce se náhodně generuje algoritmem níže 1. Jednotlivá jídla poté zabírají při jejich přípravě jednoho zaměstnance na dobu, která je určena obtížností přípravy jídla. Po této době je zabráno místo v peci, kde se pizza peče dvě minuty. Po upečení pizza vstupuje do fronty `readyMealsForPacking`, odkud jsou vybírány procesem `Packing`, který prioritně zabírá zaměstnance. Po zabalení je pizza buď připravená k vydání zákazníkovi, nebo vstupuje do fronty `readyMealsForDelivery`, odkud je vybírá jeden z procesů `Car`, reprezentující auto, starající se o doručování jídel zákazníkům. Procesy reprezentující auta jsou zahájeny na začátku každého simulačního běhu.

```
do
| vytvoření procesu Meal
while  $Uniform(0,100) \leq 50$ ;
```

Algoritmus 1: Generování jídel v objednávce

4.2 Spuštění simulačních experimentů

Program je nejprve nutné přeložit pomocí příkazu `make`. Poté je možné spustit simulaci příkazem:

```
./ims_proj [-r runs] [-m meals] [-e employees] [-f furnace_places] [-c cars]
[-s shift_len] [-a acceptance_time]
```


Případně je možné simulaci spustit s výchozími vstupními hodnotami příkazem `make run`.

Vstupní parametry:

- `-r runs` počet simulačních běhů (výchozí - 1000)
- `-m meals` průměrný počet jídel za směnu (výchozí - 400)
- `-e employees` počet zaměstnanců (výchozí - 3)
- `-f furnace_places` počet míst v peci (výchozí - 8)
- `-c cars` počet aut (výchozí - 2)
- `-s shift_len` délka směny v minutách (výchozí - 720)
- `-a acceptance_time` doba, po kterou jsou přijímány objednávky, v minutách (výchozí - 660)

Pro výpis nápovědy je možné použít parametr `-h`. V tomto případě program vytiskne nápovědu a úspěšně skončí.

5 Podstata simulačních experimentů a jejich průběh

V první fázi bylo nutné ověřit, že je náš navržený model validní. Po provedení prvního experimentu jsme se o tom přesvědčili, neboť výstupní hodnoty odpovídaly informacím od majitele pizzerie. V dalších experimentech jsme si dali za cíl návrh nejlepšího řešení vedoucího k navýšení kapacity doručených jídel tak, aby nedošlo k markantnímu zvýšení doby dodávky a s tím spojenému poklesu kvality.

5.1 Postup experimentování

Naše experimentování probíhalo tak, že jsme pro každou konfiguraci spustili 1000 simulačních běhů, které jsme sdružili a některé hodnoty zprůměrovali, abychom dostali vypovídající statistiku o tom, jak je daná konfigurace vyhovující. Souhrnné výstupní hodnoty byly poté posuzovány jako celek, kdy jsme kontrolovali průměrnou a maximální dobu předání pizzy zákazníkovi či kurýrovi. Tato doba by neměla přesáhnout hodinu a čtvrt, aby nedocházelo k poklesu spokojenosti zákazníků. Dále jsme se zaměřili na vytíženost zaměstnanců na prodejně, peci a řidičů s cílem optimalizace využití kapacity. Jako hlavní určující parametr bylo také řádné ukončení směny v daném časovém limitu, který by měl být podle majitele pizzerie pana Brehového trvat maximálně hodinu po konci přijímání objednávek. Samozřejmě se může stát, že některý den vzhledem k dalším okolnostem bude vymykat průměru, nicméně by tato „úspěšnost“ měla dosahovat minimálně 95 %.

5.2 Experimenty

Každý z experimentů má za cíl zjistit, zdali je daná konfigurace pro daný počet jídel vyhovující. To je možné určit prostřednictvím sesbíraných údajů jako je záznam o úspěšném ukončení směny v daném časovém limitu, průměrná a maximální doba expedice pizzy a vytížení zaměstnanců, pecí a aut.

5.2.1 Experiment 1

První experiment spočívá ve validaci modelu. V úvahu se bere sobota s průměrným počtem jídel za směnu rovným 400, třemi zaměstnanci na provozovně, pecí, která je schopna pojmout až 8 pizz současně, a dvěma rozvoznými auty. Výsledky experimentu jsou v tabulce 1.

Počet běhů [ks]	1000
Počet jídel [ks]	400
Počet zaměstnanců [ks]	3
Průměrné vytížení zaměstnanců [%]	52.235
Kapacita pece [ks]	8
Průměrné vytížení pece [%]	13.949
Počet aut [ks]	2
Průměrné vytížení aut [%]	91.575
Průměrná doba k předání [min]	12
Maximální doba k předání [min]	66
Průměrná úspěšnost směny [%]	100

Tabulka 1: Experiment 1

Z této tabulky (1) lze vyčíst, že prvotní experimenty prokázaly validitu námi navrženého modelu. Všechny směny stihly předat a rozvézt všechno jídlo před koncem směny. Je také patrné, že vytíženost pece je relativně nízká a její kapacita je tak dostačující. Naopak vytížení aut s řidiči se pohybuje těsně pod maximem. Při dalším experimentování tudíž bude vhodné se na tento parametr zaměřit.

5.2.2 Experiment 2

V tomto experimentu zkoumáme, jestli a jak by systém dokázal obstát v momentě, kdy by se průměrný počet jídel za den zvedl o 25 % na 500 jídel za den. Zajímá nás také, ve které oblasti modelu vzroste vytížení nejvíce.

Počet běhů [ks]	1000
Počet jídel [ks]	500
Počet zaměstnanců [ks]	3
Průměrné vytížení zaměstnanců [%]	65.2
Kapacita pece [ks]	8
Průměrné vytížení pece [%]	17,4
Počet aut [ks]	2
Průměrné vytížení aut [%]	93.53
Průměrná doba k předání [min]	14
Maximální doba k předání [min]	84
Průměrná úspěšnost směny [%]	99.6

Tabulka 2: Experiment 2

Z tabulky (2) je patrné, že vytížení zaměstnanců vzrostlo o 13 % a že vytížení aut se ještě více přiblížilo maximu. V dalších fázích bude tedy namísto zaměřit se na zvyšování počtů právě v těchto oblastech. Oproti tomu průměrné vytížení pecí se stále drží pod dvaceti procenty a dá se předpokládat, že tento parametr nebude mít zásadní roli na prodloužení délky vyřizování objednávek.

Z tabulky lze také vyčíst, že maximální doba nutná k předání pizzy zákazníkovi nebo řidiči již stoupla nad stanovený limit hodiny a čtvrt. I když tedy téměř všechny směny skončily úspěchem, bude nutné při navýšení počtu objednávek zvýšit i počet zaměstnanců nebo aut za předpokladu, že má být limit doručení dodržen.

5.2.3 Experiment 3

V tomto experimentu přidáme jedno auto s řidičem při průměrných 500 jídlech za den. Slibujeme si od toho, že by měla klesnout vytíženost řidičů a doba do předání jídla.

Počet běhů [ks]	1000
Počet jídel [ks]	500
Počet zaměstnanců [ks]	3
Průměrné vytížení zaměstnanců [%]	64.95
Kapacita pece [ks]	8
Průměrné vytížení pece [%]	17.35
Počet aut [ks]	3
Průměrné vytížení aut [%]	90.41
Průměrná doba k předání [min]	12
Maximální doba k předání [min]	60
Průměrná úspěšnost směny [%]	100

Tabulka 3: Experiment 3

Třetí experiment nám ukázal, že při zvýšení počtu aut o jedno se maximální doba předání nedostane přes hodinu a čtvrt. Lze tedy toto řešení použít v případě nárůstu počtu denních objednávek. Vytížení pecí i zaměstnanců zůstaly na přibližně stejných hodnotách, což se dalo předpokládat. Vytížení aut se snížilo jen nepatrně, tedy i přes přidání jednoho auta, zůstává dopravní složka modelu nejvytíženější.

5.2.4 Experiment 4

V tomto experimentu přidáme oproti normálu jednoho zaměstnance při průměrných 500 jídlech za den s cílem zmenšení vytíženosti zaměstnanců a zrychlení přípravy a balení pizzy. Také chceme porovnat, jaký je rozdíl mezi přidáním jednoho auta a jednoho zaměstnance a případně, která z těchto změn vede k lepším výsledkům.

Počet běhů [ks]	1000
Počet jídel [ks]	500
Počet zaměstnanců [ks]	4
Průměrné vytížení zaměstnanců [%]	48.85
Kapacita pece [ks]	8
Průměrné vytížení pece [%]	17.38
Počet aut [ks]	2
Průměrné vytížení aut [%]	93.12
Průměrná doba k předání [min]	12
Maximální doba k předání [min]	62
Průměrná úspěšnost směny [%]	99.6

Tabulka 4: Experiment 4

Z výsledků čtvrtého experimentu v tabulce (4) je patrné, že navýšení počtu zaměstnanců o jednoho člověka vede také k návratu maximální doby předání jídla pod požadovanou hodinu a čtvrt. Průměrné i maximální časy předání se pohybují kolem stejných hodnot jako v případě předchozího experimentu 5.2.3. Celková úspěšnost směny se nepatrně snížila oproti předchozímu experimentu, nicméně se jedná o tak zanedbatelný rozdíl, že ve výsledku by nehrál téměř žádnou roli.

Z nasimulovaných hodnot tohoto experimentu a experimentu předchozího 5.2.3 vede k nepatrně lepším výsledkům provozu právě přidání jednoho auta, avšak jedná se o tak malý rozdíl, že s přihlédnutím k nákladům na samostatného zaměstnance v porovnání s náklady na řidiče a nové vozidlo, se jako lepší volba jeví spíše právě přidání zaměstnance.

Co se týká kapacita pece, tak ta stále zůstává více než dostačující a tedy navýšení její kapacity nutné není.

5.2.5 Experiment 5

V tomto experimentu přidáme jednoho zaměstnance a jedno auto oproti stávajícímu stavu při průměrných 500 jídlech za den s cílem zmenšení vytíženosti zaměstnanců a řidičů a zrychlení přípravy a balení pizzy a zkrácení doby dodávky. Také nás zajímá, zdali sloučení dvou předešlých experimentů 5.2.3 a 5.2.4 přinese další zlepšení, jelikož jejich samostatné testování vedlo k velmi podobným výsledkům.

Počet běhů [ks]	1000
Počet jídel [ks]	500
Počet zaměstnanců [ks]	4
Průměrné vytížení zaměstnanců [%]	49.18
Kapacita pece [ks]	8
Průměrné vytížení pece [%]	17.50
Počet aut [ks]	2
Průměrné vytížení aut [%]	89.6
Průměrná doba k předání [min]	9.5
Maximální doba k předání [min]	43
Průměrná úspěšnost směny [%]	100

Tabulka 5: Experiment 5

Z pátého experimentu a jeho výsledků (5) jsme se dozvěděli, že pokud by byl navýšen jak počet zaměstnanců na provozovně, tak počet aut s řidiči a nárůstu počtu objednávek by byl roven 25 %, průměrné a maximální časy potřebné k předání jídel by dokonce klesly oproti normálu, došlo by tudíž ke zkvalitnění služeb a ke zvýšení spokojenosti zákazníků.

5.3 Závěry experimentů

Z námi prováděných pěti experimentů bylo zjištěno, že námi navržený model je validní a koresponduje tak se skutečností. Získali jsme také nové informace o chování systému, které mohou být využity v případě růstu zájmu o pizzy od firmy Siesta pizza [4].

V rámci dalších experimentů by bylo možné díky námi navrženému modelu zjistit, jak by se systém choval, kdyby došlo ještě k výraznějšímu nárůstu počtu objednaných jídel, které stavy by bylo nejvhodnější při jakých nárůstech navýšit, případně pokud by se rozhodli pro výraznou reorganizaci firmy, bylo by možné si její fungování odsimulovat apod.

6 Shrnutí simulačních experimentů a závěr

Na základě prvního experimentu bylo prokázáno, že náš model je validní, jelikož výsledky z něj získané odpovídaly informacím od majitele pizzerie pana Brehového.

Z dalších experimentů bylo zjištěno, že v případě nárůstu průměrného počtu denních objednávek o 25 % by společnost již nezvládala za stávajícího stavu zákazníky obsluhovat tak, aby maximální doba od přijmutí objednávky do předání jídla zákazníkovi nebo řidiči nestoupla nad hodinu a čtvrt. Při navýšení počtu zaměstnanců nebo aut o jednoho by však společnost opět stíhala svá jídla předávat v požadovaném čase. Pokud by došlo k navýšení obou těchto kapacit, průměrné i maximální časy by se dokonce snížily oproti stávající situaci.

Na co jsme dále přišli bylo také to, že současná kapacita pecí je více než dostačující a není tudíž potřeba její kapacity navyšovat.

Závěry a nová zjištění vycházející z těchto experimentů budou poskytnuty panu Kamilu Brehovému. Na něm pak bude rozhodnutí, ke kterým krokům přistoupí v případě nárůstu počtu objednávaných jídel.

Literatura

- [1] Jančarová, L.: Máme hodně objednávek, nestíháme, tvrdí rozvozce pizzy. Jídlo přiveze kurýr s rouškou a v rukavicích. [online], 15. března 2020, [vid. 2020-12-05].
URL https://www.lidovky.cz/domov/provoz-restauraci-omezen-jidlo-priveze-kuryr-s-rouskou-a-v-rukavicich.A200314_173945_ln_domov_tmr
- [2] Peringer, P.; Hrubý, M.: Modelování a simulace, Text k přednáškám kursu Modelování a simulace na FIT VUT v Brně. [online], 22. listopad 2018, [vid. 2018-12-05].
URL <https://www.fit.vutbr.cz/study/courses/IMS/public/prednasky/IMS.pdf>
- [3] Peringer, P.; Leska, D.; Martinek, D.: SIMLIB/C++ (SIMulation LIBrary for C++). [online], 19. září 2018, [vid. 2018-12-05].
URL <http://www.fit.vutbr.cz/~peringer/SIMLIB>
- [4] SIESTA: SIESTA. [online], 2020, [vid. 2020-12-05].
URL <http://www.siestapizza.cz>