Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra systémového inženýrství

Diplomová práce

Aplikace modelu obchodního cestujícího v kamionové dopravě

Bc. Tereza Nehézová

© 2019 ČZU v Praze

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Tereza Nehézová

Systémové inženýrství

Název práce

Aplikace problému obchodního cestujícího v kamionové dopravě

Název anglicky

Travelling salesman problem application in truck transport

Cíle práce

Cílem práce je aplikace řešení okružního dopravního problému s časovými okny v mezinárodní kamionové dopravě za účelem minimalizace nákladů přepravy a času. Dílčím cílem je analýza problému za předpokladu nejistoty v ohodnocení jednotlivých tras a návrh robustních scénářů pro zvoleného dopravce.

Metodika

Diplomová práce je rozdělena na teoretickou a praktickou část. Teoretická část vychází ze studia odborné literatury a popisuje téma okružního dopravního problému a konkrétní matematické modely, které jsou pro jeho řešení využívány. Podrobněji bude popsána problematika NP-úplných problémů, problému obchodního cestujícího včetně jeho rozšíření o časová okna, robustní optimalizace a metodu větví a mezí, která bude následně využita v praktické části.

V praktické části bude konkrétně definován řešený problém. Ze získaných znalostí bude sestaven matematický model, který bude řešen pomocí metody větví a mezí za pomoci softwaru Gurobi optimizer. Výsledky budou porovnány se současným stavem a budou udělena příslušná doporučení vedoucí k finančním úsporám.

Doporučený rozsah práce

60-70

Klíčová slova

okružní dopravní problém, logistika, Np-úplné úlohy, metoda větvění a mezí

Doporučené zdroje informací

COOK, W. *Po stopách obchodního cestujícího : matematika na hranicích možností.* Praha: Dokořán, 2012. ISBN 978-80-7363-412-4.

FÁBRY, J. *Matematické modelování.* Praha: Professional Publishing, 2011. ISBN 978-80-7431-066-9.

Hlavatý, R., Brožová, H. (2017). Robust optimization approach in transportation problem. Mathematical Methods in Economics 2017.

Předběžný termín obhajoby

2018/19 LS – PEF

Vedoucí práce

Ing. Robert Hlavatý, Ph.D.

Garantující pracoviště

Katedra systémového inženýrství

Elektronicky schváleno dne 27. 2. 2019

doc. Ing. Tomáš Šubrt, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 1. 3. 2019

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 26. 03. 2019

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Aplikace modelu obchodního cestujícího v kamionové dopravě" jsem vypracoval(a) samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor(ka) uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 26. 3. 2019 _____

Poděkování

Rád(a) bych touto cestou poděkoval(a) vedoucímu práce Ing. Robertovi Hlavatému, Ph.D., za jeho odborné vedení, cenné rady a připomínky k diplomové práci.

Aplikace modelu obchodního cestujícího v kamionové dopravě

Abstrakt

Diplomová práce se věnuje aplikaci modelu obchodního cestujícího v kamionové dopravě. Je rozdělena do dvou částí. První část práce obsahuje teoretická východiska zaměřující se na vybrané oblasti. Nejprve je rozebrána problematika logistiky a dopravy. Dále se práce věnuje operačnímu výzkumu a matematickému modelování jako jeho nástroji. Následně jsou přiblíženy dopravním úlohy, problém obchodního cestujícího, jeho modifikace s časovými okny a robustní optimalizace v lineárních optimalizačních modelech. Vybraná témata jsou použita v praktické části.

Vlastní část práce je systematicky rozdělena do kapitol dle Simonovy teorie. První část se věnuje konstrukci modelu obchodního cestujícího se zahrnutím nejistoty v ohodnocení jednotlivých spojů a sestavení robustních scénářů pro vybraný okružní dopravní problém. Ve druhé části je sestrojen a vypočítám model obchodního cestujícího s časovými okny. Jako podklad pro sestrojení tohoto modelu je využit zvolený scénář z předchozí části práce. Na závěr jsou výsledky srovnány se současnou situací ve firmě a je provedeno ekonomické zhodnocení.

Klíčová slova: okružní dopravní problém, NP-úplné úlohy, logistika, metoda větvení a mezí, robustní optimalizace, dopravní úlohy, problém obchodního cestujícího, operační výzkum

Travelling salesman problem application in truck transport

Abstract

This diploma thesis deals with travelling salesman problem application in truck transport and its divided into two parts. The first part of the thesis constrains theoretical background focusing on selected topics. Firstly, the issue of transport and logistics is mentioned. Furthermore, the thesis deals with operational research and its tool, mathematical modelling. Then problem of travelling salesman is introduced as well as its time windows modification and robust approach in linear optimization models. Selected issues are used in the second part of the diploma thesis.

This part of the diploma thesis is divided into parts according to Simon's theory. First chapter deals with formulation of mathematical model of travelling salesman problem including uncertainty in the evaluation of individual connections and creating of robust scenarios for selected routing transportation problem. In the second part, the model of travelling salesman problem with time windows is constructed and calculated. The chosen robust scenario form previous part is used as the basis for this model. Finally, the results are compared with company's current situation and economic evaluation is made.

Keywords: travelling salesman problem, NP-complete problems, logistics, Branch and Bound method, robust optimization, traffic models, operational research

Obsah

1 Úvod **1**

2 Cíl práce a metodika **2**

2.1 Cíl práce 2

2.2 Metodika 2

3 Teoretická východiska **4**

3.1 Logistika 4

3.1.1 Cíle logistiky 5

3.1.2 Dopravní logistika 5

3.2 Operační výzkum 5

3.2.1 Historie operačního výzkumu 5

3.2.2 Vymezení operačního výzkumu 6

3.3 Matematické modelování 7

3.4 Celočíselné programování 9

3.4.1 Formulace úlohy celočíselného programování 10

3.4.2 Metody řešení celočíselného programování 11

3.4.3 Metoda Větvení a mezí (Branch and bound) 11

3.5 Distribuční úlohy 12

3.5.1 Jednostupňová dopravní úloha (JDÚ) 12

3.5.2 Matematický model JDÚ 13

3.6 Problém obchodního cestujícího (TSP) 13

3.6.1 Charakteristika 15

3.6.2 Statické a dynamické úlohy 15

3.6.3 Statická úloha obchodního cestujícího 16

3.6.4 Statická úloha obchodního cestujícího s časovými okny 17

3.6.5 Matematický model problému obchodního cestujícího s časovými okny 18

3.6.6 NP-úplné problémy 19

3.7 Robustní optimalizace 20

3.7.1 Matematický model 21

3.8 Softwarové řešení matematických modelů 23

3.8.1 Gurobi optimizer 23

4 Vlastní práce **26**

4.1 Charakteristika subjektu 26

4.2 Problémová situace (Fáze inteligence) 26

4.3 Robustní optimalizace 30

4.3.1 Úprava a konstrukce matematického modelu (Fáze design) 30

4.3.2 Výpočet a analýza scénářů (Fáze choice) 33
4.4 TSP s časovými okny 42
4.4.1 Konstrukce matematického modelu (Fáze design) 42
4.4.2 Výpočet (Fáze choice) 44

5 Výsledky a diskuse 46
5.1 Konstrukce robustních scénářů 46
5.2 Optimalizace v mezinárodní kamionové dopravě 48
5.3 Doporučení 49

6 Závěr 51

7 Seznam použitých zdrojů 53

8 Přílohy 57

Seznam obrázků

Obrázek 1 Supply chain 4
Obrázek 2 Průběh rozhodovacího procesu 9
Obrázek 3 - Úplná síť cest 17
Obrázek 4 - Neúplná síť cest 17
Obrázek 5 Model obchodního cestujícího upravený pro program Gurobi – 1. část 23
Obrázek 6 Model obchodního cestujícího upravený pro program Gurobi - 2. část 24
Obrázek 7 Výstupní data z programu Gurobi 25
Obrázek 8 Výsledný okruh pro $\delta ijc = 0{,}1cij$ a $\Gamma = 18$ 33
Obrázek 9 Výsledný okruh pro $\delta ijc = 0{,}1cij$ a $\Gamma = 36$ 36
Obrázek 10 Výsledný okruh pro $\delta ijc = 0{,}1cij$ a $\Gamma = 54$ 37
Obrázek 11 Výsledné pořadí míst pro $\delta ijc = 0{,}15cij$ a $\Gamma = 18$ 39
Obrázek 12 Mapa okruhu pro $\delta ijc = 0{,}15cij$ a $\Gamma = 18$ 39
Obrázek 13 Výsledný okruh pro $\delta ijc = 0{,}15cij$ a $\Gamma = 36 – 342$ 40
Obrázek 14 Výsledný okruh (se zahrnutím časových oken) 44
Obrázek 15 Optimální pořadí míst vč. hodnot Wi 45
Obrázek 16 Pořadí míst pro nejpravděpodobnější scénář ($\delta ijc = 0{,}1cij$) 46
Obrázek 17 Pořadí míst pro $\delta ijc = 0{,}1cij$ $\Gamma = 54$ 47
Obrázek 18 Optimální pořadí míst vč. hodnot Wi 48

Seznam tabulek

Tabulka 1 Seznam míst 28
Tabulka 2 Matice vzdáleností (v hodinách) 29
Tabulka 3 Výsledné hodnoty hodnot pro $\delta ijc = 0{,}1cij$ $\Gamma = 18$ 34
Tabulka 4 Výsledné hodnoty pro $\delta ijc = 0{,}1cij$ a $\Gamma = 36$ 37
Tabulka 5 Výsledné hodnoty pro $\delta ijc = 0{,}1cij$ a $\Gamma = 54$ 38
Tabulka 6 Výsledné hodnoty pro $\delta ijc = 0{,}15cij$ a $\Gamma = 18$ 40
Tabulka 7 Výsledné hodnoty pro $\delta ijc = 0{,}15cij$ a $\Gamma = 36$ - 342 41
Tabulka 8 Časová okna a doby obsluhy 43

1 Úvod

V praxi existuje mnoho případů využití dopravních modelů, především v oblasti přepravy zboží od dodavatelů k odběratelům, a to zejména takových, které minimalizují přepravní náklady. Tyto problémy jsou navíc omezeny nejen požadavky odběratelů a kapacitami dodavatelů, ale také časem, kapacitou vozidel či dalšími podmínkami, které zužují množinu přípustných řešení.

V mezinárodní kamionové dopravě je v dnešní době velký tlak na cenu, rychlost a spolehlivost. Nároky na schopnosti dispečerů stále narůstají a proto se využívá množství podpůrných nástrojů a programů, hlavně v oblasti výběru objednávek. Plánování dopravy a přijímání objednávek je však stále z největší části prováděno na základě zkušeností a schopností dispečerů. Přitom právě na činnosti dispečera nejvíce závisí, jak dobře bude doprava naplánována nejen z pohledu využití kapacit vozidel, ale také z pohledu vzdálenosti, času a jak budou efektivně vytěžováni jednotliví řidiči. Od toho se odvíjí výše nákladů a tedy i zisku společnosti.

Pro kamionovou dopravu je typický charakter úlohy obchodního cestujícího, tj. případ, kdy má vozidlo navštívit všechna místa v okruhu právě jednou a vrátit se zpět do výchozího místa co nejkratší cestou. Samotný problém obchodního cestujícího se řadí mezi tzv. NP-úplné úlohy a řadí se mezi jednu z nejznámějších kombinatorických úloh současnosti.

Diplomová práce se věnuje použití úlohy obchodního cestujícího v mezinárodní kamionové dopravě a zahrnutí hlediska nejistoty v ohodnocení jednotlivých spojů prostřednictvím robustního optimalizačního modelu. Práce je zaměřena na minimalizaci vzdálenosti a času a s tím související minimalizaci nákladů.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem práce je aplikace okružního dopravního problému s časovými okny v mezinárodní kamionové dopravě za účelem minimalizace času a nákladů přepravy.

Dílčím cílem je analýza problému za předpokladu nejistoty v ohodnocení jednotlivých tras a návrh robustních scénářů pro zvoleného dopravce.

2.2 Metodika

Diplomová práce je rozdělena na dvě části. První, teoretická část, vychází ze studia odborné literatury a vybraných článků z odborných periodik. Zabývá se zejména logistikou, operačním výzkumem a celočíselným programováním. Dále je blíže je rozebrána problematika dopravních úloh, především okružního dopravního problému a konkrétní matematické modely, které jsou pro jeho řešení využívány.

Praktická část je rozdělena do dvou částí. Nejprve je ve fázi *intelligence* podle Simonovy teorie definován řešený problém a na základě získaných informací je sestavena matice vzdáleností. Z tohoto modelu následně vychází další části práce. Nejprve je použit robustní optimalizační přístup:

1. Fáze *design* se věnuje úpravě modelu tak, aby mohlo být do modelu zahrnuto hledisko nejistoty v ohodnocení jednotlivých tras. Dále jsou stanoveny scénáře, které budou ve fázi *choice* vypočítány a porovnány.
2. Ve fázi *choice* je proveden výpočet jednotlivých scénářů a analýza výsledných okruhů. Z těchto výsledků je na základě konzultace s pověřeným pracovníkem firmy vybrán nejpravděpodobnější scénář, který je použit v další části práce.

Druhá část se věnuje zahrnutí časových oken do modelu obchodního cestujícího:

1. Fáze *design* se věnuje aplikaci modelu obchodního cestujícího s časovými okny na vybraný scénář z předchozí části práce. Jsou sestavena časová okna pro jednotlivá místa v okruhu, stanovena doba obsluhy a podle těchto informací je sestaven samotný matematický model.
2. Ve fázi *choice* je proveden výpočet sestaveného modelu a analýza výsledného okruhu.

V poslední části je výsledný model ekonomicky zhodnocen, porovnán se současným řešením firmy a jsou vyvozeny doporučení pro příslušnou firmu. Ke všem výpočtům je použit program Gurobi optimizer.

3 Teoretická východiska

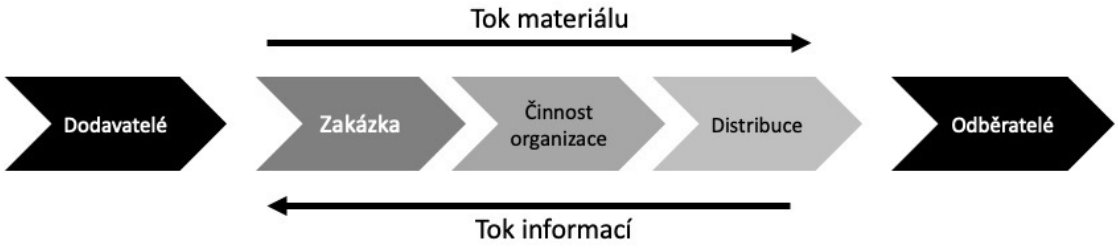
Teoretická část této práce se věnuje zejména logistice, operačnímu výzkumu a problému obchodního cestujícího a jeho variantám řešeným v praktické části. Blíže bude rozebrán robustní přístup a model obchodního cestujícího s časovými okny.

3.1 Logistika

Logistika je disciplína, která se zabývá pohybem zboží a materiálu z místa vzniku do místa spotřeby a s tím souvisejícím informačním tokem. Věnuje se všem komponentům oběhového procesu, především dopravy, řízením zásob, manipulace s matriálem, balení, distribuce a skladování. Součástí logistiky jsou také komunikační, informační a řídící systémy. Úkolem logistiky je zajistit správné zboží (materiál, službu), se správnou kvalitou, u správného zákazníka, ve správném množství, na správném místě ve správný okamžik s vynaložením přiměřených nákladů, tzv. požadavek „Sedm S“. (Drahotský, 2003; Sixta, 2005)

Kotler (2001) logistiku definuje jako proces realizace a kontroly fyzického toku materiálu a hotových výrobků od místa původu až po místo spotřeby tak, aby uspokojily potřeby zákazníka, s dosažením zisku. Kain (2018) dále uvádí, že se jedná v podstatě o integrační proces, který optimalizuje tok materiálu a zboží napříč organizací a jejích činností s cílem dostat zboží k zákazníkovi, za splnění principu „Sedm S“ (anglicky Seven R‘s“).

Obrázek 1 Supply chain



Zdroj: (Kain, 2018)

Logistika představuje poměrně novou metodologii podnikové organizace. Pohled na důležitost logistiky a její integrace do managementu podniku se měnila hlavně ve vztahu ke koncentraci výrobních kapacit, které v 90. letech předstihly možnosti do té doby dostupných možností distribuce a vzhledem ke změně orientace z trhu výrobce (omezený sortiment výrobků ve velkých výrobních sériích) k trhu zákazníka, jejímž důsledkem byla potřeba

rychlé inovace výroků s širokou paletou sortimentu. Tato změna kromě jiného vyvolala silný tlak na snižování neúměrně zvyšujících se nákladů v oblasti distribuce a to vedlo k vývoji levnějších a účinnějších metod distribuce. Větší rozmach lze zaznamenat také s dostupností technických prostředků, moderních technologií a výpočetní techniky.

Význam logistiky stále roste s rozvojem přechodu na globální trhy. Úspory nákladů dosažitelné uplatňováním logistiky jsou v zemích s fungujícím distribučním systémem odhadovány až na 10%. (Sixta, 2005)

3.1.1 **Cíle logistiky**

Hlavními kritérii, podle kterých lze dělit cíle logistiky, je oblast jejich působení (z pohledu podniku na vnější či vnitřní) a způsob měření jejich výsledků. Vnější logistické cíle se zaměřují na uspokojování přání zákazníků, kteří je uplatňují na trhu. To přispívá k udržení, případně k dalšímu rozšíření rozsahu realizovaných služeb. Mezi tyto cíle logistiky lze zařadit zvyšování objemu prodeje, zkracování dodacích lhůt, zlepšování spolehlivosti a úplnosti dodávek a zlepšování flexibility logistických služeb. Vnitřní cíle logistiky se zaměřují především na snižování nákladů. Jedná se o náklady na zásoby, dopravu, manipulaci a skladování, výrobu atd. (Sixta a Mačát, 2005)

3.1.2 **Dopravní logistika**

Dopravní logistika je jednou z aplikačních oblastí logistiky. Zabývá se řízením dopravních a přepravních procesů zejména při pohybu zásilek, rozmístěním kapacit a souvisejících pohybů prostředků a zařízení, jejichž součinnost vyžaduje uskutečnění přepravy určité zásilky či přepravy osob. Řízení dopravních procesů je zaměřeno na pohyb přepravních prostředků, manipulačních a dopravních zařízení. Součástí dopravní logistiky je i řízení činností dopravních uzlů na dopravní síti a řízení informačních toků. (Štůsek, 2005)

3.2 Operační výzkum

3.2.1 **Historie operačního výzkumu**

Počátky rozvoje operačního výzkumu spadají zejména do 30. a 40. let 20. století. Významný rozvoj této disciplíny nastal hlavně během 2. světové války, kdy v USA a Velké Británii fungovaly speciální týmy pracovníků pro analýzu strategických a taktických

vojenských problémů a operací. Po ukončení války začaly metody operačního výzkumu pronikat i do civilního průmyslu a do dalších oblastí. Významný mezník byl prudký ekonomický rozvoj v 50. letech. Vznik a vývoj operačního výzkumu je spjat se jmény jako G. B. Dantzig nebo L. Kantorovič.

Dalším důkazem o rozvoji operačního výzkumu je počet a intenzita pořádání odborných konferencí, četnost vydávání nových publikací a časopisů s odborným zaměřením na operační výzkum. Velký vliv na vývoj operačního výzkumu má jednoznačně rozvoj a využívání informačních technologií a výpočetní techniky. (Gros, 2003; Jablonský, 2002)

3.2.2 **Vymezení operačního výzkumu**

Operační výzkum či operační analýza (eng. operations research nebo management science) je soubor relativně samostatných vědních disciplín, které se zabývají analýzou různých typů rozhodovacích problémů. (Lagová, 2004)

Jak uvádí Klapka (1996), jedná se o vědeckou disciplínu, která se zabývá analýzou operací spjatých s řízením, fungováním a navrhováním složitých společensko-ekonomicko-technických systémů, tj. organizačních jednotek, v nichž spolu existují lidé a technická zařízení a v nichž existují sociální, technické a materiálové vazby. Příkladem takového systému může být průmyslový podnik, kterýkoli jeho provoz, systém veřejné dopravy apod.

Neexistuje však jedna univerzální definice operačního výzkumu a v současné době je používáno vícero označení pro tuto oblast exaktních metod. Je to způsobeno různými pohledy a vývojem operačního výzkumu. Níže je uvedeno pro srovnání několik definic.

- Dle Kaufmanna (1963):
  "Operační výzkum je vědecký přístup k rozhodování."
- Dle Winstona (1995):
  "Operační výzkum je vědecký přístup k hledání řešení, který usiluje o to, jak navrhovat a řídit systémy obvykle za podmínek vyžadujících lokalizaci omezených zdrojů"
- Dle organizace Society of Operational Research:
  „Operační výzkum je vědecký přístup k řešení problémů v řízení komplexních systémů a umožňuje rozhodovatelům dělat vědecky podložená rozhodnutí. ... Napomáhá při strategických, taktických a operativních rozhodnutích, stejně tak

pomáhá při vytváření veřejné politiky. ... Většina problémů, které operační výzkum řeší, jsou složité a často obsahují faktor nejistoty.“

Na základě uvedených definic lze říci, že operační analýza má několik typických znaků:

- zaměření na podporu rozhodování při řešení problémů s navrhováním a řešením rozsáhlých a složitých systémů v různých aplikačních oblastech,
- při řešení problémů využívá vědecký přístup spojený s aplikací modelové techniky,
- uplatnění systémového řešení,
- týmová organizace práce,
- respektování dynamiky a stochastické povahy reálných procesů.

Jak vyplývá z výše uvedených charakteristik, operační analýza je vědní disciplína, nebo spíše soubor vědních oborů, která se zabývá zkoumáním operací v rámci nějakého systému. Cílem je přitom stanovit takovou úroveň provádění těchto operací nebo jejich vzájemné vztahy tak, aby bylo zajištěno co možná nejlepší fungování systému. Pro posuzování toho, jak systém funguje, zda lépe nebo hůře, je nutné stanovit nějaké kritérium či více kritérií.

Samotné fungování systému však závisí na zdrojích, které bývají často omezené. Dále také na provádění jiných operací, které jsou mezi sebou závislé (např. používáním stejných zdrojů, případně návazností jednotlivých operací), na vnějších vlivech na chod systému atd. (Jablonský, 2002)

3.3 Matematické modelování

Základním nástrojem operační analýzy je matematické modelování. Jestliže je za pomoci operačního výzkumu analyzován nějaký systém, děje se tak prostřednictvím matematického modelu daného systému. Při analýze modelu reálného systému je třeba brát v úvahu, že model je pouze zjednodušeným obrazem reality. I přesto je matematický model nositelem celé řady výhod, díky kterým se stává často jediným prostředkem pro studium modelovaného systému.

Mezi základní výhody matematického modelování patří:

- strukturalizace systému a specifikace všech možných variant stavu systému (těch může být často neomezené množství),
- analýza chování systému ve zkráceném čase pomocí počítačové simulace - některé procesy mohou v reálném světě trvat dny, měsíce případně roky,

- snadná manipulace a provádění četných experimentů pomocí změn příslušných parametrů modelu
- nižší náklady na realizaci modelu než v případě experimentování s reálným systémem. (Jablonský, 2002)

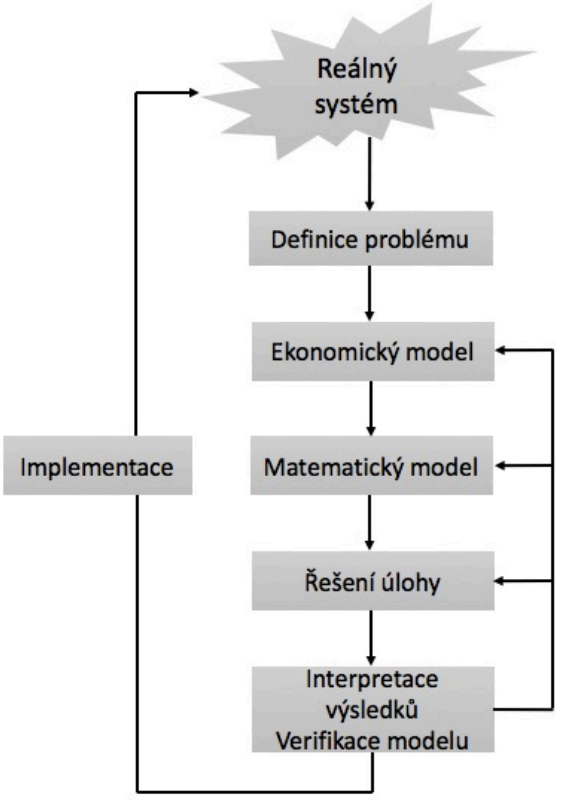
Aby byly matematické modely operačního výzkumu použitelné a poskytovaly podstatné informace, musí obsahovat srozumitelný popis všech relevantních faktorů dané situace a umožnit tak studovat všechny důležité vztahy mezi prvky zkoumaného systému. Existuje mnoho druhů matematických modelů, avšak jejich podstata je vždy stejná, představují matematické schéma (např. soustavu rovnic a funkcí) popisující daný systém.

Jako příklady některých klasických modelů operační analýzy lze zmínit:

- **Optimalizační modely** - skupina velmi obecných modelů sloužících k nalezení nejlepšího řešení problému a řešením jsou prvky konečné či nekonečné množiny. Patří sem lineární, dynamické či stochastické programování.
- **Distribuční a dopravní modely** - slouží k řešení problémů s dopravou či distribucí.
- **Modely plánování a řízení projektů** - umožňují časovou, nákladovou a zdrojovou analýzu projektů.
- **Simulační modely** - popisují a napodobují strukturu i chování zkoumaných objektů. Spadají sem modely hromadné obsluhy, zásob, obnovy a marketingové modely. (Brožová a Houška, 2002)

Při aplikaci některého z odvětví operačního výzkumu lze rozlišit několik základních na sebe navazujících fází rozhodovacího procesu (viz obrázek 2).

Obrázek 2 Průběh rozhodovacího procesu



Zdroj: Vlastní zpracování dle Fábry (2011)

3.4 Celočíselné programování

V běžné úloze lineárního programování se předpokládá, že rozhodovací proměnné mohou nabývat v optimálním řešení jakýchkoli reálných hodnot. V řadě praktických situací ale potřebujeme pouze celočíselné hodnoty rozhodovacích proměnných. Například nemůžeme dodat na trh 120,3 auta nebo přijmout ke studiu 52,6 studenta. Lze sice celočíselné hodnoty získat prostým zaokrouhlením nebo odseknutím desetinné části a v mnoha případech takto získat uspokojivé výsledky. Existují však situace, kdy tento přístup vede k velkým chybám ve srovnání s hodnotou účelové funkce celočíselného optimálního řešení, nebo nedává řešení přípustné. (Klapka, 1996)

U běžné úlohy lineárního programování se předpokládá, že optimalizované proměnné se mohou spojitě měnit. Objemy produkce mohou být vyráběny v libovolném množství, lze spojitě vynakládat finanční prostředky apod. Existují však situace, kde proměnné modelu mohou nabývat pouze předem stanovený hodnot a musí se jednat o celá čísla, např. musí být objednána pouze celá balení výrobku, automobily jsou dodávány po kusech atd. (Gros, 2003)

Celočíselné programování je speciální úlohou lineárního programování, kdy je standardní úloha LP doplněna o podmínky celočíselnosti, tj. o podmínky, pomocí kterých je zabezpečeno, aby všechny nebo jen některé proměnné úlohy nabývaly pouze celočíselných hodnot. Podmínky celočíselnosti většinou vyplývají přímo z formulace ekonomického modelu daného problému.

Kromě obecných úloh celočíselného programování se vyskytují také úlohy, kdy se mohou vyskytovat podmínky, aby proměnné nabývaly pouze hodnot 1 a 0 - takové proměnné se nazývají bivalentní. Tyto požadavky se vyskytují například u okružního dopravního problému nebo přiřazovací úlohy. Z čehož vyplývá, že okružní dopravní problém je úlohou celočíselného programování. Úlohy celočíselného programování, které obsahují pouze bivalentní proměnné se označují jako úlohy bivalentního programování. (Jablonský, 2002)

Samotný požadavek celočíselnosti snižuje počet možných variant, proto se jako řešení nabízí prostý výpočet účelové funkce pro všechny možné kombinace přípustných hodnot proměnných a nalezení takové UF, která zajišťuje požadovaný extrém. Avšak použití tohoto řešení není v praxi možné. V případě že by měl model 10 proměnných a každá z nich by měla 5 přípustných celočíselných hodnot, byl by počet variant které by bylo nutné analyzovat roven 5^{10}. V reálných situacích se může jednat o tisíce proměnných, proto by tento postup nebylo možné aplikovat v rozumném čase při rozumných nákladech. (Gros, 2003)

3.4.1 **Formulace úlohy celočíselného programování**

Obecná formulace úlohy celočíselného programování může být takováto (Hladík, 2017):

$$\text{Max } f(x)$$

$$\text{za podmínek}$$

$$g_j(x) \leq 0 \quad \forall j = 1, \dots, m \tag{3.1}$$

$$x_i \in Z \quad \forall i \in C \tag{3.2}$$

kde

$$C \subseteq (1, \dots, n) \tag{3.3}$$

V takovémto případě je podmínka celočíselnosti kladena pouze na některé proměnné, jedná se tedy o tzv. smíšenou úlohu celočíselného programování.

Pokud by se podmínka celočíselnosti týkala všech proměnných, byla by formulována takto:

$$C = (1, \dots, n) \tag{3.4}$$

V tomto případě se jedná o čistou úlohu celočíselného programování. (Hladík, 2017)

3.4.2 Metody řešení celočíselného programování

Při volbě vhodného přístupu k řešení celočíselné úlohy je třeba brát v úvahu celou řadu aspektů. Mezi nejdůležitější se řadí rozsah množiny přípustných řešení. Vzhledem k tomu, že většina celočíselných úloh má konečný počet přípustných řešení, je možné zvážit vypočet za pomocí sestavení všech možných kombinací hodnot jednotlivých proměnných, vyloučit kombinace narušující omezení a vybrat nejlepší řešení za základě srovnání hodnot účelové funkce. Tento postup je ovšem reálně použitelný pouze u úloh s malým počtem proměnných a malým počtem možných kombinací. V mnoha praktických problémech je počet přípustných řešení velký a je nutno použít jiný přístup.

V některých případech je praktický přístup k řešení úlohy celočíselného programování skutečně takový, že se zanedbají podmínky celočíselnosti a pro následně získané optimální hodnoty rozhodovacích proměnných se provede úprava prostým zaokrouhlením na celočíselné hodnoty. Výhodou takového přístupu je úspora nákladů a času, které by bylo třeba vynaložit na formulaci a řešení celočíselného modelu. (Klapka, 1996)

Existuje celá řada metod pro řešení úloh celočíselného programování. Lze zmínit metodu sečných nadrovin, heuristické metody a kombinatorické metody, mezi něž se řadí také metoda větvení a mezí, která bude použita v praktické části práce a bude dále blíže popsána v samostatné kapitole.

3.4.3 Metoda Větvení a mezí (Branch and bound)

První myšlenka této metody byla nezávisle rozvinuta několika autory během 50. let 20. století První algoritmus zformulovali ve svém článku Land a Doig (1960),ovšem první rozumná implementace metody se objevuje až u Dakina v roce 1965. (Hladík, 2017)

Metoda se řadí, jak již bylo řečeno výše, mezi kombinatorické algoritmy. Princip

metody je natolik obecný, že ji lze po jistých úpravách použít pro celou řadu úloh celočíselného programování, například právě pro řešení úlohy obchodního cestujícího. (Jablonský, 2002)

Postup metody spočívá v systematickém procházení množiny potenciálních přípustných řešení úlohy, ve snaze vytvořit jednodušší řešitelné dílčí problémy (větve), přičemž se vyřazují podmnožiny nevhodných kandidátů.

Při výpočtu jsou využívány horní a dolní odhady hodnot (meze). V průběhu výpočtu se tyto meze zpřesňují ve snaze dostat se co nejblíže optimu. Pomocí těchto mezí je určováno, zda se v dané podmnožině nalézá lepší či horší řešení oproti dříve nalezeným. V případě horšího řešení se tato podmnožina vyřadí z celkového řešení a výpočet pokračuje větvením podmnožiny s lepší hodnotou. (Applegate, 2006,)

3.5 Distribuční úlohy

Distribuční úlohy jsou speciální skupinou úloh lineárního programování, mezi které se řadí jednostupňové a vícestupňové dopravní úlohy, přiřazovací úlohy, okružní, trasovací a mnohé další. Veškeré tyto úlohy se dají vyjádřit za pomoci lineárních modelů. Vzhledem ke specifickým vlastnostem některých ze zmíněných modelů, je možné využít specifické metody, které jsou jednodušší než simplexová metoda. U jiných naopak velikost modelů i při malé velikosti úlohy - malém počtu míst, mezi nimiž je třeba přepravu zajistit - vyžadovala výpočetní kapacitu, která neumožní efektivně nalézt jejich přesné teoretické optimum. (Šubrt a kol., 2011)

3.5.1 **Jednostupňová dopravní úloha (JDÚ)**

Jednostupňová dopravní úloha je nejjednodušším z distribučních modelů. Cílem modelu je najít takový plán dopravy mezi dodavateli a spotřebiteli, při kterém budou celkové náklady na přepravu minimální, kapacity dodavatelů budou vyčerpány a požadavky spotřebitelů budou uspokojeny. (Brožová a Houška, 2002)

Předpokládá se, že k přepravě je využíván unifikovaný druh dopravního prostředku, mezi každým dodavatelem a spotřebitelem existuje pouze jedna dopravní cesta, po které je možné přepravovat libovolné množství produktu, a náklady na přepravu jsou přímo úměrné množství přepravovaného produktu. (Šubrt a kol., 2011)

3.5.2 Matematický model JDÚ

Dopravní úloha je speciálním případem modelu lineárního programování. Její matematický model se tedy skládá ze tří částí - účelové funkce, omezujících podmínek a podmínek nezápornosti.

Hledáme minimum lineární funkce

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \rightarrow MIN \tag{3.5}$$

za podmínek

$$\sum_{j=1}^{n} x_{ij} = a_i, \qquad i = 1,2,\dots,m \tag{3.6}$$

$$\sum_{i=1}^{m} x_{ij} = b_j, \qquad j = 1,2,\dots,n \tag{3.7}$$

$$x_{ij} \geq 0 \qquad i = 1,2,\dots,m; j = 1,2,\dots,n \tag{3.8}$$

kde:
a_i = kapacity dodavatelů,

b_j = požadavky spotřebitelů,

c_{ij} = cena za přepravu jednotky,

x_{ij} = množství přepravovaného produktu. (Šubrt a kol., 2011; Fábry, 2011)

Veškeré výše popsané parametry dopravního modelu lze přehledně zapsat do tzv. dopravní tabulky, ve které se také provádí výpočet. Zdroj: (Šubrt a kol., 2011)

3.6 Problém obchodního cestujícího (TSP)

Podle Reinelta (1994) je problém obchodního cestujícího (jinak také okružní dopravní problém) jednou z největších matematických otázek současnosti a zabývá se jím mnoho odborníků z několika oborů. Jeho atraktivita spočívá ve faktu, že se jedná o poměrně jednoduše formulovatelný problém, který má však veškeré aspekty kombinatorické optimalizace a je využíván k testování pro nové algoritmy, např. pro neuronové sítě.

Mezi nejvýznamnější osobnosti, které se zasadily o pokrok v oblasti řešení problému obchodního cestujícího se řadí Dantzig, Fulkerson a Johnson. Pomocí aplikace lineárního programování se jim v roce 1954 podařilo vyřešit problém s 48 městy bez využití počítače.

Posledním dosaženým počtem míst, pro které byl problém obchodního cestujícího vyřešen, je 85900. Stalo se tak v roce 2006 a jednalo se o nalezení optimální trasy pro laser, který měl vytvořit zákazníkovi počítačový čip. (Cook, 2012)

Okružní dopravní problém je také zajímavý svým širokým využitím, a to nejen na poli dopravy, ale také v jiných odvětvích, např. jak uvádí Cook (2012), v oblasti genetického výzkumu, v pozorování planet a vesmíru (směrování teleskopů, rentgenových paprsků a laserů), kdy není fyzické navštívení místa možné a ani nutné, v ovládání průmyslových spojů, třídění dat atd. Vzhledem k množství modifikací problému obchodního cestujícího pro jeho reálné využití existuje celá řada algoritmů pro pokrytí těchto speciálních příkladů.

Některé firmy se zabývají přepravou zásilek menších rozměrů (balíčky, obálky) mezi dvěma místy a může se mnohdy jednat o velké vzdálenosti, jedná se o tzv. úlohu kurýrní služby (*messenger problem*). V tomto případě nehraje roli velikost dodávky, ovšem v jiných případech může být právě prostor vozidla stěžejní faktor. Takováto úloha je označována jako *pickup-and-delivery problem.*

Další často využívaná podoba problému obchodního cestujícího je model s časovými okny. V této úloze je zahrnuto časové hledisko v podobě intervalu, kdy má být daný uzel obsloužen. (Fiala, 2010) Z důvodu použití tohoto modelu v praktické části mu bude věnována samostatná kapitola.

Z některých dalších speciálních variant TSP lze jmenovat problém obchodního cestujícího s provázanou dvojící uzlů pro vyzvednutí a dodávku (*The traveling salesman problem with pickups, deliveries, and draft limits*), kterou se zabývá Malaguti (2018), který pro každého zákazníka definuje „pickup“ uzel a „delivery“ uzel a hledá za dodržení stanovených omezujících podmínek „nejlevnější“ cestu.

Jako další modifikaci lze zmínit „*The prize money collecting traveling sales man problem*“ (Balas, 1989), kde obchodní cestující navštěvuje města na základě odměny, kterou dostane, pokud město navštíví a „pokuty“ za každé město, které nenavštíví. Cílem této varianty TSP je minimalizovat cenu cesty a výši „pokut“ za nenavštívená místa, a zároveň získat alespoň předepsanou částku (*prize money*).

3.6.1 Charakteristika

S okružním dopravním problémem se setkáváme v případě, kde se jakékoliv produkty pravidelně rozváží z jednoho místa nebo naopak sváží na jedno místo a uspokojovat jednotlivé požadavky samostatně není příliš výhodné. Cílem je vyjet ze startovního uzlu, navštívit všechny ostatní uzly právě jednou a vrátit se zpět do výchozího místa tak, aby byla délka okruhu minimální. (Pelikán a Chýna, 2011)

Existuje mnoho typů okružních úloh a nejjednodušší z nich je jednookruhový dopravní problém. Přeprava mezi všemi obsluhovanými místy má být realizována jedním okruhem.

Dalším případem spadajícím pod okružní dopravní problémy je tzv. trasovací problém, jinak také víceokruhový dopravní problém. Zde nejčastěji časové nebo kapacitní omezující podmínky nedovolují realizovat přepravu jedním okruhem a je třeba ji rozdělit do okruhů více. (Šubrt a kol., 2011)

Z matematického hlediska patří okružní dopravní problém mezi tzv. NP-úplné problémy. Tato problematika bude více přiblížena později.

3.6.2 Statické a dynamické úlohy

Vzhledem k vysokým nárokům, které jsou v dnešní době kladeny, je důležité rozlišovat statické a dynamické úlohy. Distribuční úlohy, v nichž jsou předem, tj. předtím, než je úloha předána analytikům k řešení, známy všechny informace o zákaznících a jejich požadavcích, se nazývají statické úlohy. V reálných situacích však firma musí reagovat i na požadavky, které přicházení až po nalezení optimálního řešení, v případě okružních a rozvozních úloh optimálních tras. Předmětem dynamického přístupu se stává rozhodnutí o tom kdy a kdo (v případě použití více vozidel) nově vzniklý požadavek obslouží. (Fiala, 2010)

V dynamické úloze obchodního cestujícího může kdykoli během realizace jízdy přijít nový požadavek od zákazníka nacházejícího se kdekoli v předem určeném území, resp. od kohokoli z předem známé množiny zákazníků. V praxi se to projevuje přijetím nového požadavku dispečera a ten rozhodne o zařazení do předem naplánovaného okruhu, resp. do zbývající části okruhu, kterou vozidlo ještě musí absolvovat. (Fábry, 2006)

Práce se bude dále zabývat pouze statickou úlohou obchodního cestujícího a veškeré požadavky jednotlivých zákazníků budou známy před samotným výpočtem modelu.

3.6.3 **Statická úloha obchodního cestujícího**

V obecné formulaci je dáno n míst (uzlů) a sazba c_{ij} pro každou dvojici míst, která představuje vzdálenost, čas, případně náklady pro spojení těchto uzlů. Cílem úlohy je propojit všechna místa okružním spojením tak, aby součet sazeb pro jednotlivá spojení byl v této kombinaci minimální. Každé místo se vyskytuje právě jednou s výjimkou výchozího bodu, který uzavírá okruh. (Šubrt a kol., 2011)

Cílem je najít minimum lineární funkce

$$z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{3.9}$$

za podmínek

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1,2,\dots,n \tag{3.10}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1,2,\dots,n \tag{3.11}$$

$$u_i - u_j + n x_{ij} \leq n - 1 \qquad i = 1,2,\dots,n; j = 1,2,\dots,n; i \neq j \tag{3.12}$$

$$x_{ij} \epsilon \{0; 1\} \qquad i = 1,2,\dots,n; j = 1,2,\dots,n \tag{3.13}$$

kde

n = počet míst, které vozidlo musí projet,

c_{ij} = vzdálenost mezi místy *i* a *j,*

x_{ij} = bivalentní proměnná, nabývá hodnot 0, 1,

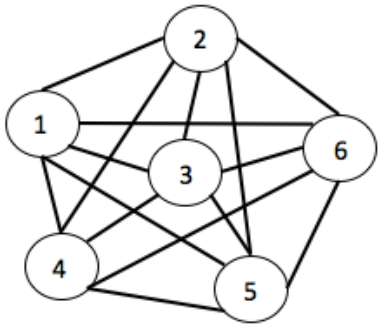
u = pomocná proměnná modelu,

podmínky (3.10) a (3.11) zajišťují, že každé místo bude navštíveno právě jednou,

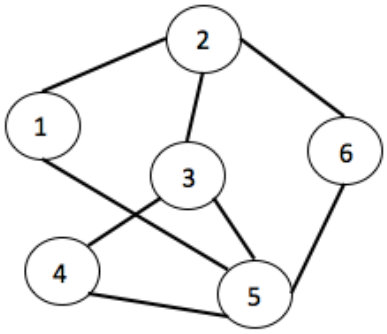
podmínka (3.12) s proměnnou u_i je opatřením proti vytváření parciálních cyklů. (Fábry, 2006)

Jednookruhové dopravní problémy se dělí na dva typy, které se odlišují charakterem spojení jednotlivých míst v dopravní síti. Jedná se o tzv. problém s úplnou sítí cest, ve kterém existuje mezi libovolnou dvojící míst přímé dopravní spojení a problém s neúplnou sítí cest, kde přímé spojení každé dvojice neexistuje.

Obrázek 3 - Úplná síť cest



Obrázek 4 - Neúplná síť cest



Zdroj: vlastní zpracování dle (Brožová a Houška, 2002)

3.6.4 Statická úloha obchodního cestujícího s časovými okny

Klasická úloha obchodního cestujícího s časovými okny (TSPTW) předpokládá podobně jako TSP znalost všech požadavků před začátkem jízdy. Pro každého zákazníka je navíc definován časový interval, ve kterém má být realizována jeho obsluha. Jedná se o standardní optimalizační okružní úlohu, ve které je cílem určit pořadí, v němž budou místa navštívena tak, aby byly splněny požadavky zákazníků a náklady spojené s rozvozem, resp.se svozem, byly minimální.

Časové okno u *i*-tého zákazníka je definováno intervalem mezi nejdříve možným začátkem obsluhy e_i a nejpozději přípustným začátkem obsluhy l_i. Okamžik, ve kterém začne reálná obsluha *i*-tého zákazníka, se značí a_i. (Fábry, 2006)

Omezení $a_i \geq e_i$, značí, že obsluha *i*-tého zákazníka nesmí začít před nejdříve možným termínem zahájení obsluhy. Pokud přijede vozidlo k zákazníkovi před nejdříve možným začátkem, musí čekat až do tohoto termínu. Pokud vozidlo nestihne přijet do nejpozději přípustného termínu, tj. pokud $a_i > l_i$ pak vzniká penále, tj. náklady spojené s nedodržením zákazníkova požadavku. Tato omezení jsou, jak uvádí Gendreau (1999) označována jako „soft“, tedy slabá omezení. V reálném světě se ale vyskytují i případy, které nepřipouštějí možnost obsluhy mimo časové stanovené okno a pro všechny zákazníky musí platit $e_i \leq a_i \leq l_i$. Tento typ omezení se označuje jako tzv. tvrdé („hard“) omezení. (Fábry, 2006)

Protože vozidlo musí respektovat časové okno zákazníka *j*, ve kterém má dojít k jeho obsluze, existují dvě strategie čekání vozidla u zákazníků:

- Vozidlo odjede k zákazníkovi j bezprostředně po dokončení obsluhy zákazníka i. Pokud přijede k zákazníkovi j před otevření časového okna, bude čekat do okamžiku e_j.
- Vozidlo zůstane u zákazníka i a k zákazníkovi j odjede v takovém okamžiku, aby k němu dorazilo přesně ve chvíli otevření jeho časového okna, tj. vyjede v okamžiku $e_j - t_{ij}$. Výhoda této strategie spočívá v tom, že během čekání vozidla může přijít další požadavek, který ovlivní dříve naplánovanou trasu, tj. vozidlo nepojede z místa i k zákazníkovi j, ale k jinému zákazníkovi tak, aby zbývající trasa byla optimální. (Fiala, 2010)

Práce v praktické části pracuje s první variantou, tj.. s odjezdu k *j*-tému zákazníkovi bezprostředně po dokončení obsluhy *i*-tého zákazníka.

3.6.5 Matematický model problému obchodního cestujícího s časovými okny

Matematický model bude podle Fábryho (2006) v případě čekání vozidla u zákazníka *j* před zahájením obsluhy vypadat takto:

$$z = \sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}x_{ij} + \sum_{j=2}^{n} W_j + \sum_{i=2}^{n} S_i \rightarrow MIN \tag{3.14}$$

za podmínek

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1,2,\dots n, \tag{3.15}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1,2,\dots n, \tag{3.16}$$

$$e_i \leq a_i \leq l_i \qquad i = 2,3,\dots,n, \tag{3.17}$$

$$a_i + S_i + t_{ij} - M(1 - x_{ij}) + W_j + v_{ij} = a_j, \quad i = 1,2,\dots n, \quad i \neq j, \quad j = 2,3,\dots n, \tag{3.18}$$

$$a_1 = 0, \tag{3.19}$$

$$a_i \geq 0, \qquad i = 2,3,\dots,n, \tag{3.20}$$

$$0 \leq v_{ij} \leq 2M(1 - x_{ij}), \quad i = 1,2,\dots n, \quad i \neq j, \quad j = 2,3,\dots n, \tag{3.21}$$

$$S_i \geq 0, \qquad i = 2,3, \dots n, \tag{3.22}$$

$$x_{ij} \in \{0,1\}, \qquad i,j, = 1,2, \dots n, \tag{3.23}$$

kde

n = počet míst, které vozidlo musí projet (vč. výchozího označeného 1),

c_{ij} = vzdálenost mezi místy i a j,

t_{ij} = doba přejezdu mezi místy i a j,

e_i = nejdříve možný termín obsluhy zákazníka *i*,

l_i = nejpozději přípustný termín obsluhy zákazníka *i*,

M = vysoká konstanta,

a_i = okamžik, ve kterém vozidlo navštíví místo *i*,

S_i = časový údaj o délce obsluhy u zákazníka *i*.

x_{ij}= bivaletní proměnná nabývající hodnoty 1, jestliže je spoj z místa *i* do místa *j* realizován. V opačném případě bude nabývat hodnoty 0.

Účelová funkce (3.14) a podmínky (3.15) - (3.16)mají stejný význam jako ve standardním modelu obchodního cestujícího. Podmínky (3.17) stanovují obsloužení zákazníka uvnitř příslušného časového okna. Omezení (3.20) zajišťuje, že časový interval mezi návštěvou zákazníka j bezprostředně po zákazníkovi i má minimálně hodnotu t_{ij}. Rovnice (3.19) definuje nulový okamžik výjezdu vozidla z výchozího místa. (Fiala,2010)

3.6.6 **NP-úplné problémy**

Problém obchodního cestujícího se řadí mezi tzv. NP- úplné problémy (NP-complete, NPC). NP-úplnost je jedna ze tříd složitosti, které poprvé definoval americký informatik Stephen Cook ve svém článku „The Complexity of Theorem-Proving Procedures“ (Cook, 1971).

Třída P je třídou složitosti, jejíž úlohy lze řešit v polynomiálním čase na deterministickém turingově stroji.Třída NP (nedeterministicky polynomiální) je množina problému, které lze řešit na v polynomiálně omezeném čase na nedeterministickém Turingově stroji, nebo lze říci, že se jedná o problémy, u kterých lze pro daný výsledek v polynomiálním čase ověřit jeho správnost. Dodnes však nebyl nalezen polynomiální algoritmus pro řešení úloh v NP a proto se všeobecně předpokládá, že $P \neq NP$. Pokud by takový algoritmus existoval, třída NP by se zhroutila do P. NP-úplné/těžké problémy jsou

takové problémy, pro které nelze pomocí nějakého algoritmu nalézt řešení v polynomiálním čase (tedy „rychle“). Jedná se tedy o nejtěžší úlohy z NP. (Arora, 2009)(Cook, 2012)

3.7 Robustní optimalizace

Klasický matematický model pracuje s předpokladem, že jsou všechny parametry přesně známy a jsou rovny nějakým nominálním hodnotám. Tento přístup však nezohledňuje vliv neurčitosti údajů na kvalitu a přípustnost modelu. Proto lze předpokládat, že bude z důvodu odlišnosti hodnot dat a nominálních hodnot narušeno několik omezení modelu a řešení získané za pomoci nominálních dat již nemusí být optimální či dokonce přípustné. Z toho vyplývá potřeba konstrukce modelů, které jsou imunní vůči nejistotě na straně dat. Prvním autorem, který se zabýval tématem robustní optimalizace, byl Soyster (1973), jenž navrhl optimalizační model lineárního programování, který poskytuje řešení odolné vůči zkreslení způsobené nejistotou dat. (Bertsimas, 2004)

Jedná se o poměrně novou disciplínu na poli matematického modelování. Po Soysterovi následovalo mnoho dalších řešitelů, zejména díky atraktivitě daného tématu z důvodu nestochastické povahy. Z novějších prací lze zmínit přínost Ben-Tal, El Ghaoui a Nemirovski (2009), kteří popsali problematiku robustní optimalizace jako celku.

Poměrně nedávno se objevil nový přístup, tzv. „multi-band“ robustní optimalizace. Ten využívá koncept klasické robustní optimalizace a přináší množství nových možností, zvláště v oblasti modelování nejistoty. Použití „multi-band“ přístupu umožňuje popsat modelovanou situaci přesněji a díky tomu získat spolehlivější řešení. Tento přístup umožňuje rozhodovateli určit nejistotu v určitých intervalech, přičemž si model stále zachovává povahu lineárního programování. (Büsing, 2012)

Samotný „multi-band“ přístup je v neustálém vývoji je dále zdokonalován. Mezi jediné autory v tomto směru se řadí zejména Büsing a D’Anderagiovanni (2013), kteří popsali teoretické základy multi-band přístupu.

3.7.1 **Matematický model**

Předpokládáme obecný model lineárního programování:

$$max \sum_{j=1}^{n} c_j x_j \tag{3.24}$$

za podmínek:

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad i = 1, \dots, m \tag{3.25}$$

$$x_j \geq 0, \qquad j = 1, \dots, n \tag{3.26}$$

V případě přítomnosti nejistoty v modelu se obvykle předpokládá, že koeficienty úlohy a_{ij} , b_j, c_i (nebo alespoň některé z nich) nejsou přesně definovány. To vede k novému problému:

$$max \sum_{j=1}^{n} (c_j + \delta_j^c) x_j \tag{3.27}$$

za podmínek

$$\sum_{j=1}^{n} (a_{ij} + \delta_{ij}^a) x_j \leq b_i + \delta_i^b \qquad i = 1, \dots, m \tag{3.28}$$

$$x_j \geq 0, \qquad j = 1, \dots, n \tag{3.29}$$

Rovnice (3.27), (3.28) a (3.29) reprezentují reformulaci (3.24), (3.25) a (3.26) s koeficienty s promítnutou nejistotou. Nejistota je vyjádřena použitím odchylky pro takové koeficienty, pro které je třeba. Předpokládáme, že jakákoli odchylka δ je libovolné reálné číslo různé od nuly. V tomto případě se jedná o lineární model s odchylkami. (Hlavatý a Brožová, 2017)

Bez újmy na obecnosti dále uvažujme, že se nejistota zachází v množině koeficientů a_{ij}. Před konstrukcí samotného robustního modelu musí být splněny některé předpoklady, jak je definovali Bertsimas a Sim (2003):

1) Pro každý koeficient a_{ij}, lze definovat deterministickou (očekávanou/obvyklou) hodnotu a maximální odchylku δ_{ij}^a od deterministické hodnoty.

2) Deterministická hodnota a_{ij} tedy patří do symetrického intervalu $\left[a_{ij} - \delta_{ij}^{a}, a_{ij} + \delta_{ij}^{a}\right]$.
3) Koeficienty nejistoty jsou stochasticky nezávislé náhodné koeficienty a každý má svou vlastní hodnotu odchylky.
4) Pro každé omezení i lze definovat maximální počet koeficientů Γ_i, který se bude zároveň odchylovat od své deterministické hodnoty v omezení i.

Poté může být sestaven robustní model, založený na odchylkách a je možné dosáhnout výsledné formy, která vychází z obecného lineárního modelu (Hlavatý a Brožová, 2017):

$$max \sum_{j=1}^{n} c_j \, x_j \tag{3.30}$$

za podmínek:

$$\sum_{j=1}^{n} a_{ij} \, x_{ij} + \Gamma_i z_i + \sum_{j \in U_i} p_{ij} \leq b_i \, , i = 1, \dots, m \tag{3.31}$$

$$z_i + p_{ij} \geq \delta_{ij}^{a} x_j, \qquad i = 1, \dots, m \, , \forall j \in U_i \tag{3.32}$$

$$z_i \geq 0, \qquad i = 1, \dots, m \tag{3.33}$$

$$p_{ij} \geq 0, \qquad i = 1, \dots, m \, , \forall j \in U_i \tag{3.34}$$

$$x_j \geq 0, \qquad j = 1, \dots, n \tag{3.35}$$

kde parametr $\Gamma_i, 0 \leq \Gamma_i \leq |U_i|, i = 1, \dots, m$, s pomocí parametru Γ lze regulovat míru nejistoty pro vybrané omezení i. Hodnota parametru Γ udává maximální počet levostranných koeficientů a_{ij}, u nichž očekáváme, že dojde k vychýlení hodnoty o nejvýše δ_{ij}^{a}, p_{ij} je pomocná proměnná pro každé a_{ij}, u kterého uvažujeme nejistotu, z_i je další pomocnou proměnnou, která realizuje provázanost mezi prvním a druhým omezením a U_i indikuje soubor indexů j pro ty a_{ij}, u nichž předpokládáme nejistotu. (Hlavatý a Brožová, 2017)

3.8 Softwarové řešení matematických modelů

3.8.1 Gurobi optimizer

Gurobi optimizer je software v širokém spektru odvětví s vysokou mírou užitku, například v oblasti výroby, nákupu, financování, distribuce, lidských zdrojů a kapitálových investic. Gurobi je osvědčený jako robustní výpočetní program, který je schopen řešit optimalizační problémy sepsané do podoby matematického modelu zahrnující miliony rozhodovacích proměnných. Pracuje s velkým množství možných řešení s cílem najít nejlepší možné za účelem napomoci firmám při jejich rozhodování.

Gurobi je placený software nabízející několik variant řešení přizpůsobené potřebám zákazníka, avšak lze využít bezplatnou akademickou verzi programu, která obsahuje vybrané moduly. Tato verze dostačuje pro potřeby této práce a bude využita v praktické části pro řešení vybraného problému. (Gurobi Optimizer, b.r.)

Vstupní data

Pro zahájení výpočtu a nalezení řešení je nutné vstupní data upravit podle příslušného matematického modelu, pro ukázku a je použit matematický model úlohy obchodního cestujícího (Obrázek 5 a Obrázek 6).

Obrázek 5 Model obchodního cestujícího upravený pro program Gurobi – 1. část

```
Minimize

6.789 x1_2 + 1.714 x1_3 + 8.26 x1_4 + 6.786 x2_1 +
5.183 x2_3 + 1.679 x2_4 + 5.221 x3_2 + 1.739 x3_1 +
6.692 x3_4 + 1.673 x4_2 + 6.661 x4_3 + 8.264 x4_1

Subject to

x1_2 + x1_3 + x1_4 = 1
x2_1 + x2_3 + x2_4 = 1
x4_2 + x4_3 + x4_1 = 1

x2_1 + x3_1 + x4_1 = 1
x1_2 + x3_2 + x4_2 = 1
x1_3 + x2_3 + x4_3 = 1
x1_4 + x2_4 + x3_4 = 1

u1 - u2 + 19 x1_2 <= 18
u1 - u3 + 19 x1_3 <= 18
u1 - u4 + 19 x1_4 <= 18

u2 - u3 + 19 x2_3 <= 18
u2 - u4 + 19 x2_4 <= 18

u3 - u2 + 19 x3_2 <= 18
u3 - u4 + 19 x3_4 <= 18

u4 - u2 + 19 x4_2 <= 18
u4 - u3 + 19 x4_3 <= 18
```

Zdroj: výstup z programu Gurobi

Obrázek 6 Model obchodního cestujícího upravený pro program Gurobi - 2. část

```
u1 >= 1
u2 >= 1
u3 >= 1
u4 >= 1

u1 <= 19
u2 <= 19
u3 <= 19
u4 <= 19


Binaries

x1_2 x1_3 x1_4
x2_1 x2_3 x2_4
x3_2 x3_1 x3_4
x4_2 x4_3 x4_1

End
```

Zdroj: výstup z programu Gurobi

Výstupní data

Po dokončení výpočtu program nabídne některý z následujících výstupů:

- Nalezne optimální řešení
- Optimální řešení není nalezeno (problém není řešitelný nebo není možné dosáhnout výsledku v uspokojivém čase).

Výstupní data pro uvedený model jsou vidět v Obrázek 7. V hlavičce je uvedena informace o použité verzi programu. Dále je vidět počet řádků a sloupců vstupního souboru. Další z důležitých informací je sloupec“gap“, kde je uvedena hodnota rozdílu mezi dolním a horním odhadem optimálního řešení. V případě ukázkového modelu je hodnota gap = 0, tzn. bylo dosaženo optimálního řešení. Hodnota účelové funkce je uvedena ve spodní části výstupu a je rovna 16,885.

Obrázek 7 Výstupní data z programu Gurobi

```
Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/ukazkovy_model.lp
Reading time = 0.00 seconds
: 24 rows, 16 columns, 56 nonzeros
Optimize a model with 24 rows, 16 columns and 56 nonzeros
Variable types: 4 continuous, 12 integer (12 binary)
Coefficient statistics:
  Matrix range     [1e+00, 2e+01]
  Objective range  [2e+00, 8e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Found heuristic solution: objective 26.9280000
Presolve removed 8 rows and 2 columns
Presolve time: 0.00s
Presolved: 16 rows, 14 columns, 62 nonzeros
Variable types: 3 continuous, 11 integer (11 binary)

Root relaxation: objective 7.921667e+00, 12 iterations, 0.00 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0    7.92167    0    4   26.92800    7.92167  70.6%     -    0s
H    0     0                       16.8780000    7.92167  53.1%     -    0s
H    0     0                       16.8680000    7.92167  53.0%     -    0s
*    0     0               0       16.8550000   16.85500  0.00%     -    0s

Cutting planes:
  Clique: 1
  MIR: 1

Explored 1 nodes (14 simplex iterations) in 0.00 seconds
Thread count was 4 (of 4 available processors)

Solution count 4: 16.855 16.868 16.878 26.928

Optimal solution found (tolerance 1.00e-04)
Best objective 1.685500000000e+01, best bound 1.685500000000e+01, gap 0.0000%
```

Zdroj: vlastní zpracování v programu Gurobi

4 Vlastní práce

Předmětem praktické části práce je aplikace modelu obchodního cestujícího v mezinárodní kamionové dopravě. Nejprve bude provedena analýza tras za pomoci robustního přístupu, v rámci níž bude vytvořeno a vypočítáno několik scénářů. Následně bude na zvolený nejpravděpodobnější scénář aplikován model obchodního cestujícího s časovými okny.

Obě tyto části jsou rozděleny na 3 fáze rozhodovacího procesu podle Simonovy teorie. Ve fázi *intelligence* je charakterizována problémová situace a jsou sesbírána potřebná data. Tato fáze je společná pro obě části. Ve fázi *design* je na základě požadavků a omezení získaných v předchozí části vybrán a sestaven odpovídající matematický model. V poslední části (*choice*) je sestavený model vyřešen zvolenou metodou a v případě problému s časovými okny je provedena ekonomická analýza výsledného řešení a srovnání se současnou situací.

4.1 Charakteristika subjektu

Firma Josef Vondráček s. r. o. se zabývá mezinárodní kamionovou dopravou. Sídlo a distribuční sklad firmy se nachází v Hájích nedaleko Příbrami. Hlavním partnerem firmy je firma DOOSAN BOBCAT EMEA s.r.o. se sídlem v Dobříši u Příbrami a většina objednávané dopravy je realizována ve spolupráci s touto firmou. V současné době má firma přibližně 70 nákladních automobilů.

Doprava je nejčastěji směřována do Španělska, Portugalska, Francie, Německa, států Beneluxu a Velké Británie. Výjimečně směřuje doprava i do Itálie, případně do Skandinávie.

Vzhledem k zakázkám od firmy Bobcat je stanoveno několik pravidelně, s malými obměnami absolvovaných tras. Předmětem přepravy je rozvoz strojů, případně náhradních dílů k prodejcům nebo půjčovnám stavebních strojů nebo přímo ke koncovému zákazníkovi.

4.2 Problémová situace (Fáze inteligence)

Hlavní snahou je minimalizace délky trasy, protože náklady na ujetou vzdálenost tvoří největší část celkových přepravních nákladů. Tato minimalizace je žádoucí hlavně u tras, které jsou absolvovány pravidelně. Mezi náklady patří plat řidiče, který se od odvíjí od ujeté vzdálenosti. Dále jsou součástí náklady na pohonné hmoty, dálniční poplatky a

amortizace vozidla. V případě zdržení a nedodržení stanovených lhůt se mohou vyskytnout i finanční sankce.

Plánování přepravy probíhá na základě objednávek od zákazníků, kteří své poptávky zadávají do databáze celoevropského systému *RAALTRANS*. Dispečeři z tohoto systému vybírají nabídky, které jsou v souladu s aktuálními možnostmi a na základě těchto objednávek plánují jednotlivé trasy.

Většinou se jedná o jednu hlavní a několik menších objednávek, které jsou zařazeny tak, aby byla kapacita vozidla plně využita. Jednotlivé zakázky jsou řidiči přiřazeny podle míst vykládky tak, aby byly alespoň částečně umístěny na trase s co nejmenšími zajížďkami. Do trasy je zařazeno také několik míst nakládky, aby byla při zpáteční cestě využita kapacita vozidla. Veškerá místa, která má řidič navštívit, jsou známá před zahájením jízdy a nejsou v průběhu odebírána ani přidávána.

Samotná přeprava probíhá 40 tunovými nákladními automobily (čistá nosnost 24 tun) – protože se jedná o jednookruhový dopravní problém, ve kterém jsou požadavky jednotlivých zákazníků „ušity na míru“ jednotlivým kamionům, nebude v této práci požadavek jednotlivých míst ani kapacita kamionu brána v úvahu.

Trasa řešená v této práci byla vybrána po konzultaci s dispečerem firmy. Řidiči ji absolvují s mírnými obměnami pravidelně a nabízí potenciál značné úspory nákladů. Trasa také obsahuje oproti jiným hned několik kritických míst, hlavně v okolí větších měst, kde se často vyskytují dopravní zácpy a jiné komplikace, které mohou značně ovlivnit průběh a časovou náročnost cesty.

Místa, která jsou ve zvolené trase třeba navštívit, jsou zobrazena v Příloha 2 a jejich kompletní seznam je k vidění v tTabulka 1. Čísla, která jsou v tabulce přiřazená jednotlivým místům budou dále používána, a to i v matematickém modelu.

Trasa vede v okolí Paříže a směřuje primárně do přístavu v Le Havre. Právě poblíž Paříže (místa 6-10 v tTabulka 1 a zobrazení na mapě viz pPříloha 2) se velice často vyskytují dopravní komplikace – ať už spojené s hustotou silničního provozu, dopravní nehodou nebo stavebními pracemi.

Tabulka 1 Seznam míst

Číslo	Místo
1	Háje
2	Parkoviště Mulhouse
3	Hraniční přechod Rozvadov
4	Saint Vit
5	Dijon
6	Paříž – Palasieu
7	Paříž – Viroflay
8	Paříž - Saint-Germain-en-lay
9	Paříž – Andrésy
10	Paříž – Osny
11	Clermont
12	Rouen
13	Le havre – přístav 1
14	Le havre – přístav 2
15	Amiens
16	Saint - Quentin
17	Reims
18	Saaerbrucken
19	Sklad Dobříš

Výchozím místem pro všechny případy je centrální sklad firmy v Hájích u Příbrami (viz Příloha 1), kam je sváženo zboží a přerozdělováno do příslušných kamionů. Přeprava zde také končí.

Kamion vždy směřuje do přístavu v Le Havre (č. 13 a 14 na mapě, viz Příloha 2), kde probíhá vykládka i nakládka strojů Bobcat, případně součástek a náhradních dílů. Do trasy je také zařazeno jedno pravidelně navštěvované parkoviště – místo 2 (viz Příloha 2), zde řidiči absolvují povinnou devítihodinovou hodinovou pauzu. Místo 3 je zařazeno jako pomocné, aby bylo zajištěno, že vozidlo pojede přes bývalý hraniční přechod Rozvadov. Místo 19 je pravidelná vykládka při zpáteční cestě. Zbývající místa jsou logistická centra, kde se nacházejí firmy, od kterých je doprava objednávaná pravidelně. Mají být navštívena, ale přesná poloha se může měnit. V mapě jsou proto zaznamenány polohy průmyslových oblastí, kde se tato místa nacházejí.

Za použití doplňku pro Mapy.cz byla do MS Excel vyexportována matice sazeb jednotlivých tras mezi místy. Jednotlivé vzdálenosti jsou uvedeny v hodinách (z důvodu práce s časovými údaji v robustním modelu a později v modelu s časovými okny).

Tabulka 2 Matice vzdáleností (v hodinách)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	0	6.789	1.714	8.26	8.85	10.041	10.166	10.343	10.471	10.284	10.335	11.173	11.623	11.624	10.086	9.395	8.555	6.314	0.147
2	6.786	0	5.183	1.679	2.27	4.894	5.147	5.354	5.646	5.555	5.808	6.379	6.728	6.81	5.911	5.22	4.38	2.472	6.748
3	1.739	5.221	0	6.692	7.282	8.473	8.598	8.775	8.903	8.716	8.767	9.605	10.055	10.056	8.518	7.827	6.987	4.746	1.7
4	8.264	1.673	6.661	0	0.878	3.502	3.755	3.962	4.253	4.282	4.587	4.987	5.336	5.418	4.826	4.135	3.295	3.804	8.225
5	8.859	2.269	7.257	0.892	0	2.901	3.153	3.36	3.652	3.68	3.986	4.386	4.735	4.817	4.327	3.637	2.796	3.306	8.821
6	10.026	4.932	8.423	3.555	2.93	0	0.395	0.602	0.894	0.974	1.374	1.628	1.977	2.059	1.914	2.054	1.79	3.994	9.988
7	10.124	5.149	8.522	3.772	3.147	0.352	0	0.459	0.751	0.838	1.402	1.485	1.834	1.916	1.778	2.083	1.888	4.092	10.086
8	10.32	5.43	8.717	4.053	3.428	0.633	0.48	0	0.462	0.539	1.273	1.305	1.654	1.736	1.545	1.989	2.084	4.287	10.281
9	10.473	5.703	8.87	4.326	3.701	0.906	0.753	0.445	0	0.412	1.187	1.53	1.879	1.961	1.459	2.058	2.237	4.44	10.434
10	10.308	5.665	8.705	4.354	3.729	0.981	0.842	0.516	0.432	0	0.971	1.526	1.875	1.957	1.243	1.869	2.072	4.276	10.27
11	10.299	5.901	8.696	4.648	4.023	1.368	1.379	1.237	1.181	0.96	0	1.784	2.31	2.311	0.888	1.25	2.006	4.266	10.26
12	11.133	6.403	9.529	5.026	4.401	1.606	1.453	1.294	1.506	1.552	1.78	0	0.877	0.879	1.318	1.981	2.883	5.1	11.094
13	11.565	6.745	9.962	5.368	4.743	1.948	1.795	1.636	1.848	1.894	2.297	0.879	0	0.209	1.751	2.414	3.315	5.532	11.526
14	11.571	6.833	9.968	5.456	4.831	2.036	1.883	1.724	1.936	1.982	2.302	0.884	0.206	0	1.756	2.419	3.321	5.538	11.532
15	10.039	5.9	8.436	4.871	4.338	1.894	1.756	1.5	1.444	1.223	0.858	1.314	1.764	1.765	0	0.887	1.789	4.006	10
16	9.351	5.212	7.748	4.183	3.65	1.985	1.996	1.927	1.979	1.798	1.233	1.967	2.416	2.418	0.879	0	1.101	3.318	9.312
17	8.511	4.372	6.908	3.343	2.809	1.736	1.861	2.039	2.166	1.98	1.971	2.838	3.288	3.289	1.75	1.059	0	2.478	8.472
18	6.259	2.45	4.656	3.856	3.323	3.946	4.072	4.249	4.377	4.19	4.241	5.079	5.529	5.53	3.991	3.301	2.461	0	6.22
19	0.156	6.759	1.684	8.23	8.821	10.011	10.136	10.314	10.441	10.255	10.306	11.144	11.593	11.595	10.056	9.365	8.525	6.284	0

4.3 Robustní optimalizace

Ve standardním modelu obchodního cestujícího se řeší pouze minimalizace celkové ujeté vzdálenosti při projetí všech míst. Pomocí robustní optimalizace je do modelu zahrnut i faktor nejistoty (viz kapitola 3.7), který se v případě modelu obchodního cestujícího projeví v ohodnocení jednotlivých tras v účelové funkci. Tato nejistota bude zobrazena v podobě odchylek od doby jízdy mezi jednotlivými spoji. Doba od které bude vypočítána odchylka je doba získaná pomocí doplňku pro Google maps. Úprava modelu tak, aby obsahoval hledisko nejistoty, bude provedena v následujících kapitolách.

Protože nelze přesně říci, kolik spojů, případně o kolik přesně se která trasa zhorší/zlepší, bude sestaveno několik scénářů s různými hodnotami odchylek a počty dotčených tras. Na základě výsledků bude zvolen nejpravděpodobnější scénář, který bude použit pro další výpočet v modelu s časovými okny. Na základě řešených scénářů budou také určeny kritické spoje, které se mohou v rámci výsledků objevovat častěji. Tyto spoje mohou být problémové a způsobovat změny v účelové funkci.

Znalost různých scénářů při zahrnutí odchylek od očekávané doby přepravy může pomoci nejen při samotném plánování trasy, ale i v dalších oblastech, jako například zefektivnění vytěžování jednotlivých řidičů v návaznosti na jejich další cesty a celková lepší organizovanost a plynulost dopravy.

4.3.1 Úprava a konstrukce matematického modelu (Fáze design)

Na základě informací z předchozích částí budou sestaveny jednotlivé testované scénáře a robustní matematický model obchodního cestujícího.

Před konstrukcí modelu pro jednotlivé scénáře je nejprve třeba upravit model obchodního cestujícího na robustní, tj. zahrnout faktor nejistoty (viz kapitola 3.7.1.) V případě řešeném v této práci se nejistota vztahuje k časovému ohodnocení jednotlivých spojů.

Do účelové funkce je vedle prostého ohodnocení jednotlivých spojů c_{ij} třeba zahrnout také odchylku od této očekávané hodnoty. Tato změna je označena jako δ_{ij}^{c} (viz kapitola 3.7.1). Předpokládejme účelovou funkci modelu obchodního cestujícího s odchylkou v c_{ij}:

$$min \sum_{i=1}^{m} \sum_{j=1}^{n} (c_{ij} + \delta_{ij}^{c}) x_{ij} \tag{4.1}$$

Rovnici (4.1) lze vyjádřit také takto:

$$min\ E \tag{4.2}$$

za podmínek

$$\sum_{i=1}^{m} \sum_{j=1}^{n} (c_{ij} + \delta_{ij}^{c}) x_{ij} \leq E \tag{4.3}$$

(Hlavatý a Brožová, 2017)

Nyní může být model obchodního cestujícího upraven dle kapitoly 3.7.1 následovně:

$$min\ E \tag{4.4}$$

za podmínek

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} + \Gamma z + \sum_{(i,j) \in U_i} p_{ij} \leq E \tag{4.5}$$

$$z_i + p_{ij} \geq \delta_{ij}^{c} x_j, \forall (i,j) \in U_i \tag{4.6}$$

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1,2, \dots, m \tag{4.7}$$

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad j = 1,2, \dots, n \tag{4.8}$$

$$z \geq 0 \tag{4.9}$$

$$p_{ij} \geq 0 \qquad \forall (i,j) \in U_i \tag{4.10}$$

$$u_i - u_j + n x_{ij} \leq n - 1, \qquad i = 1,2, \dots, n; j = 1,2, \dots, n; i \neq j \tag{4.11}$$

$$x_{ij} \epsilon \{0; 1\} \qquad i = 1,2, \dots, n; j = 1,2, \dots, n \tag{4.12}$$

kde:

n = počet míst, které vozidlo musí projet (vč. výchozího označeného 1),

c_{ij} = vzdálenost mezi místy i a j,

u = pomocná proměnná modelu,

δ_{ij}^{c} = odchylka od hodnoty c_{ij},

Γz = umožňuje regulovat míru nejistoty pro vybrané omezení *i*. Hodnota parametru Γ udává maximální počet levostranných koeficientů c_{ij}, u nichž očekáváme, že dojde k vychýlení hodnoty o nejvýše δ_{ij}^{c},

p_{ij}= pomocná proměnná pro každé c_{ij}, které je uvažováno vychýlení nejvýše o δ_{ij}^{c},

U_i= označuje množinu těch j, pro které je uvažována odchylka δ_{ij}^{c},

x_{ij}= bivaletní proměnná nabývající hodnoty 1, jestliže je spoj z místa *i* do místa *j* realizován. V opačném případě bude nabývat hodnoty 0.

Podle takto upraveného modelu obchodního cestujícího lze sestavit robustní model pro jednotlivé scénáře. Nejprve je třeba určit potřebné hodnoty odchylek δ_{ij}^{c} a hodnotu Γ. Hodnota koeficientu Γ se odvíjí od počtu binárních proměnných v modelu. V případě řešeném v této práci je to 342. Testované scénáře budou vypočítány se zvolenými hodnotami $\Gamma \in \langle 18, 342 \rangle$, z důvodu zjednodušení a časové náročnosti jsou použity násobky 18.

Hodnoty δ_{ij}^{c} budou v jednotlivých scénářích určeny 10% – 30% změny c_{ij}. Vyšší změny jsou pravděpodobné spíše pro jednotlivé spoje a proto nebudou uvažovány. Pro scénář s uvažovanou odchylkou $\delta_{ij}^{c} = 0{,}1c_{ij}$ by omezující podmínka (4.6) pro x_{1_2}, x_{1_3} a x_{1_4} vypadaly následovně:

$$z + p_{1_2} \geq 0{,}6789 x_{1_2} \tag{4.13}$$

$$z + p_{1_3} \geq 0{,}1714 x_{1_3} \tag{4.14}$$

$$z + p_{1_4} \geq 0{,}826 x_{1_4} \tag{4.15}$$

Kompletní model vč. všech podmínek a účelové funkce upravený pro výpočet v programu Gurobi optimizer lze vidět v Příloha 4. Tento model je také výchozím pro další scénáře. Podle potřeby konkrétních scénářů bude pouze upravena hodnota Γ a omezující podmínky (4.6).

4.3.2 Výpočet a analýza scénářů (Fáze choice)

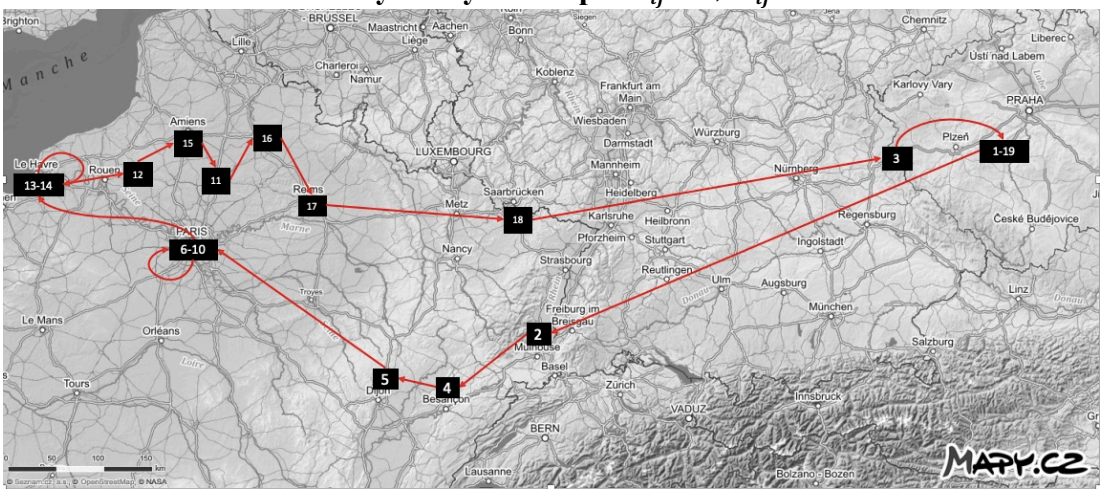
Fáze *choice* je poslední částí rozhodovacího procesu podle Simonovy teorie a bude zde proveden výpočet optimálního pořadí míst v okružním dopravním problému, určení kritických tras a nejpravděpodobnějšího scénáře, který bude použit pro výpočet v další části práce. K výpočtu byl využit software Gurobi Optimizer 8.1.0 build v8.1.0rc1 běžící na konfiguraci:

- MacBook Air (13-inch, Early 2015)
  - MacOS Mojave verze 10.14.3,
  - procesor 1.6 GHz Intel Core i5,
  - pamět 8 GB 1600 MHt DDR3.

Hodnoty účelové funkce okruhů vypočtené v této kapitole představují pouze čistou dobu jízdy, nejsou zahrnuty povinné pauzy, obsluhy v jednotlivých místech případně čekání na obsluhu. Tato hlediska budou zahrnuta až v modelu obchodního cestujícího s časovými okny. Samotná hodnosta účelové funkce reprezentuje dobu, za kterou bude okruh absolvován pokud dojde ke zdržení na všech spojích o stanovenou hodnotu δ_{ij}^{c}.

Pro $\delta_{ij}^{c} = 0{,}1c_{ij}$, $c_{ij} \in \left[c_{ij} - \delta_{ij}^{c}, c_{ij} + \delta_{ij}^{c}\right]$ a pro $\Gamma \in \langle 18, 342 \rangle$ se vyskytují 3 výsledná pořadí míst. Pro $\Gamma = 18$ vozidlo vyjíždí z výchozího místa do místa 2, dále do 4 – 5 – 6 – 7 – 8 – 9 – 10 – 13 – 14 – 12 – 11 – 15 - 16 – 17 – 18 – 3 – 19 a zpět do místa 1. Hodnota účelové funkce tohoto okruhu je 33,49 hodin, celková vzdálenost je 2666,2 km. Hodnoty pro δ_{ij}^{c} pro jednotlivé spoje jsou uvedeny v Tabulka 3.

Obrázek 8 Výsledný okruh pro $\delta_{ij}^{c} = 0{,}1c_{ij}$ a $\Gamma = 18$



Zdroj: vlastní zpracování, Mapy.cz

Tabulka 3 Výsledné hodnoty hodnot pro $\delta_{ij}^{c} = 0,1 c_{ij}$ $\Gamma = 18$

x_{ij}	δ_{ij}^{c} (hod)	c_{ij} (hod)	$\sum$ (hod)
x1_2	0,6633	6,7892	7,4525
x2_4	0,1523	1,6794	1,8317
x3_19	0,1544	1,7000	1,8544
x4_5	0,0722	0,8775	0,9497
x5_6	0,2745	2,9011	3,1756
x6_7	0,0239	0,3953	0,4192
x7_8	0,0303	0,4594	0,4897
x8_9	0,0306	0,4617	0,4923
x9_10	0,0256	0,4117	0,4373
x10_13	0,1719	1,8750	2,0469
x11_16	0,1094	1,2500	1,3594
x12_15	0,1162	1,3181	1,4343
x13_14	0,0053	0,2086	0,2139
x14_12	0,0728	0,8842	0,9570
x15_11	0,0702	0,8583	0,9285
x16_17	0,0945	1,1008	1,1953
x17_18	0,2322	2,4781	2,7103
x18_3	0,4500	4,6556	5,1056
x19_1	0,0000	0,1558	0,1558
$\sum$ (hod)	3,0393	30,4597	33,4990

Výstup z programu Gurobi pro výše uvedený testovaný scénář vypadá následovně:

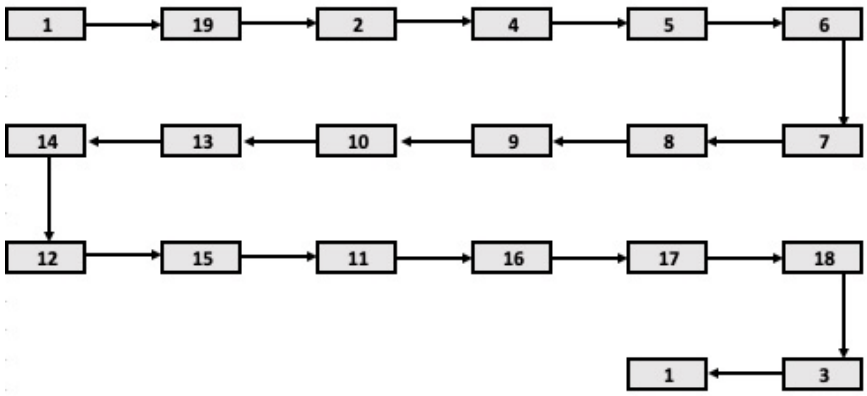
```
Academic license - for non-commercial use only
Set parameter timelimit to value 50

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/10/spravny/10_18.lp
Reading time = 0.01 seconds
: 1086 rows, 705 columns, 3749 nonzeros
Optimize a model with 1086 rows, 705 columns and 3749 nonzeros
Variable types: 363 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [1e-02, 4e+01]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Presolve removed 382 rows and 2 columns
Presolve time: 0.04s
Presolved: 704 rows, 703 columns, 5333 nonzeros
Variable types: 361 continuous, 342 integer (342 binary)
```

```
Root relaxation: objective 2.033062e+01, 131 iterations, 0.02 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

    0     0   20.33062    0   44          -   20.33062      -     -    0s
H   0     0                 67.1528000   20.33062  69.7%     -    0s
H   0     0                 50.1578000   20.33062  59.5%     -    0s
    0     0   22.58469    0   46   50.15780   22.58469  55.0%     -    0s
    0     0   22.58469    0   46   50.15780   22.58469  55.0%     -    0s
    0     0   22.94259    0   57   50.15780   22.94259  54.3%     -    0s
    0     0   23.00404    0   54   50.15780   23.00404  54.1%     -    0s
    0     0   23.00404    0   54   50.15780   23.00404  54.1%     -    0s
    0     0   23.00843    0   54   50.15780   23.00843  54.1%     -    0s
    0     0   23.00863    0   54   50.15780   23.00863  54.1%     -    0s
    0     0   23.01324    0   53   50.15780   23.01324  54.1%     -    0s
    0     0   23.01324    0   53   50.15780   23.01324  54.1%     -    0s
    0     0   23.01497    0   52   50.15780   23.01497  54.1%     -    0s
    0     0   23.01497    0   52   50.15780   23.01497  54.1%     -    0s
    0     2   23.01497    0   52   50.15780   23.01497  54.1%     -    0s
H   88    85                 44.0594000   23.31950  47.1%  20.1    0s
H  111   103                 40.3887000   23.31950  42.3%  19.1    0s
*  119   103          57     39.8299000   23.31950  41.5%  18.6    0s
*  128   101          61     33.6171000   23.32140  30.6%  18.8    0s
H  882   619                 33.5896000   24.02389  28.5%  22.5    2s
H 1478   836                 33.5060000   25.11749  25.0%  22.8    2s
 5671  3404   28.86307   40   15   33.50600   25.63532  23.5%  18.4    5s
 18369 10239   30.74118   43    8   33.50600   25.94520  22.6%  15.3   10s
 20711 11490   26.35282   30   52   33.50600   25.97869  22.5%  15.1   16s
 20754 11519   27.04685   31   76   33.50600   25.97869  22.5%  15.0   20s
 21516 11817   26.75355   39   26   33.50600   25.97869  22.5%  15.4   25s
 30347 15191   26.41254   39    6   33.50600   25.97869  22.5%  14.9   30s
 41645 18776   26.83337   37   13   33.50600   25.97869  22.5%  14.5   35s
H48566 20280                33.5059999   25.97869  22.5%  14.4   38s
 52661 21411   26.87660   55   25   33.50600   26.00676  22.4%  14.3   40s
 62915 27365   29.24033   49    9   33.50600   26.07986  22.2%  14.3   45s
 73297 33342   28.78665   51   12   33.50600   26.14092  22.0%  14.1   50s

Cutting planes:
  Learned: 5
  Gomory: 27
  Implied bound: 9
  Clique: 2
  MIR: 22
  Flow cover: 221
  Inf proof: 16
  Zero half: 8

Explored 73718 nodes (1041974 simplex iterations) in 50.01 seconds
Thread count was 4 (of 4 available processors)

Solution count 9: 33.506 33.506 33.5896 ... 67.1528

Time limit reached
Best objective 3.350599990275e+01, best bound 2.614302665259e+01, gap 21.9751%
```

Tento výstup udává veškeré informace ohledně výpočtu výchozího modelu. Program zjistil model s 1086 řádky, 705 sloupci a obsahující 342 binárních proměnných. Výsledná hodnota účelové funkce po 50 vteřinách výpočtu je 33,506, odhadovaná dolní hodnota optimálního řešení je 26,14 hodin. Hodnota ve sloupci gap značí rozdíl mezi horním a dolním odhadem optimálního řešení, v tomto případě 22 %. Pokud byl ponechán neomezený čas pro výpočet, u hodnoty účelové funkce nedocházelo po určitém čase k žádné změně, jako lze vidět v tomto případě u hodnot 2 – 50s , a výpočet nebyl ukončen ani po několika hodinách. Proto byl čas byl omezen u všech scénářů na 50 sekund.

Pokud u stejného modelu dojde ke změně hodnoty $\Gamma = 36,$ tak se pořadí míst v okruhu mění na 1 – 19 – 2 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 13 – 14 – 12 – 11 – 15 - 16 – 17 – 18 – 3 – 19 – 1. Dochází k zařazení místa 19 do okruhu ihned po výchozím místě a u spoje směrem z místa 19 se již vyskytuje nenulová hodnota δ_{ij}^{c}. Z toho důvodu je také vyšší hodnota účelové funkce, konkrétně 33,507 hod. Celková vzdálenost, kterou vozidlo ujede při absolvování tohoto pořadí míst je 2845 km. Hodnoty pro jednotlivé spoje a jejich příslušná zpoždění jsou vypsány v Tabulka 4. Tato varianta je však realizovatelná v malém počtu případů, protože v reálné situaci je místo 19 většinou navštěvováno jako poslední před návratem do výchozího místa

Obrázek 9 Výsledný okruh pro $\delta_{ij}^{c} = 0,1c_{ij}$ a $\Gamma = 36$

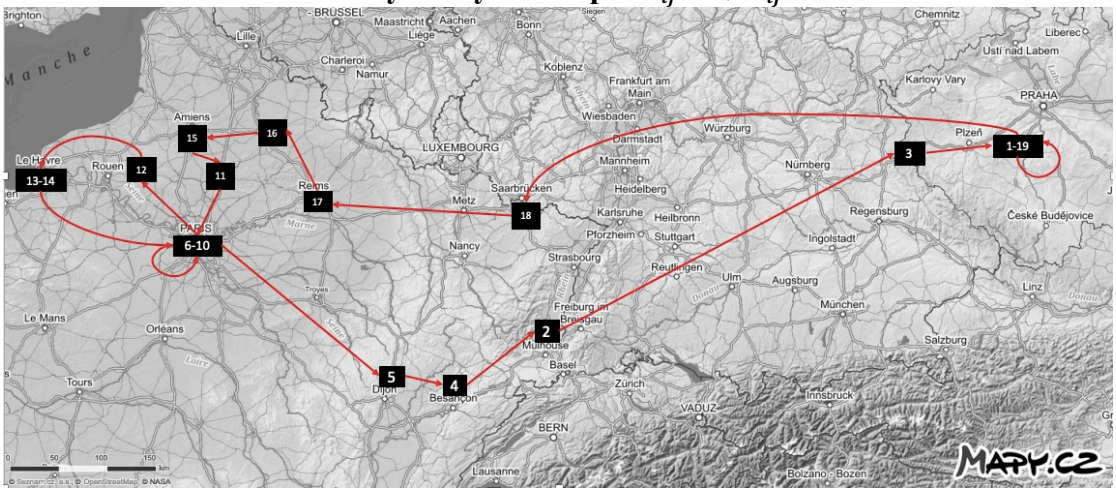


Tabulka 4 Výsledné hodnoty pro $\delta_{ij}^{c} = 0,1c_{ij}$ a $\Gamma = 36$

x_{ij}	δ_{ij}^{c} (hod)	c_{ij} (hod)	$\sum$ (hod)
x1_19	0,0147	0,1469	0,1616
x2_4	0,1679	1,6794	1,8473
x3_1	0,1739	1,7386	1,9125
x4_5	0,0878	0,8775	0,9653
x5_6	0,2901	2,9011	3,1912
x6_7	0,0395	0,3953	0,4348
x7_8	0,0459	0,4594	0,5053
x8_9	0,0462	0,4617	0,5079
x9_10	0,0412	0,4117	0,4529
x10_13	0,1875	1,8750	2,0625
x11_16	0,1250	1,2500	1,3750
x12_15	0,1318	1,3181	1,4499
x13_14	0,0209	0,2086	0,2295
x14_12	0,0884	0,8842	0,9726
x15_11	0,0858	0,8583	0,9441
x16_17	0,1101	1,1008	1,2109
x17_18	0,2478	2,4781	2,7259
x18_3	0,4656	4,6556	5,1212
x19_2	0,6759	6,7594	7,4353
$\sum$ (hod)	3,0460	30,4597	33,5057

Pro hodnotu $\Gamma = 54$ dochází k výrazné změně pořadí míst v okruhu. Z výchozího místa vozidlo jede do 18 – 17 – 16 – 15 – 11 – 10 – 9 – 8 – 12 – 14 – 13 – 7 – 6 – 5 – 4 – 3 – 19 a zpět do výchozího místa (Obrázek 10). Oproti předchozím případům jsou místa s mírnými změnami navštívena v opačném pořadí. V tomto případě je hodnota účelové funkce 33, 52 hodin čisté jízdy. (výsledné hodnoty jsou k vidění v Tabulka 5).

Obrázek 10 Výsledný okruh pro $\delta_{ij}^{c} = 0,1c_{ij}$ a $\Gamma = 54$



Zdroj: vlastní zpracování, Mapy.cz

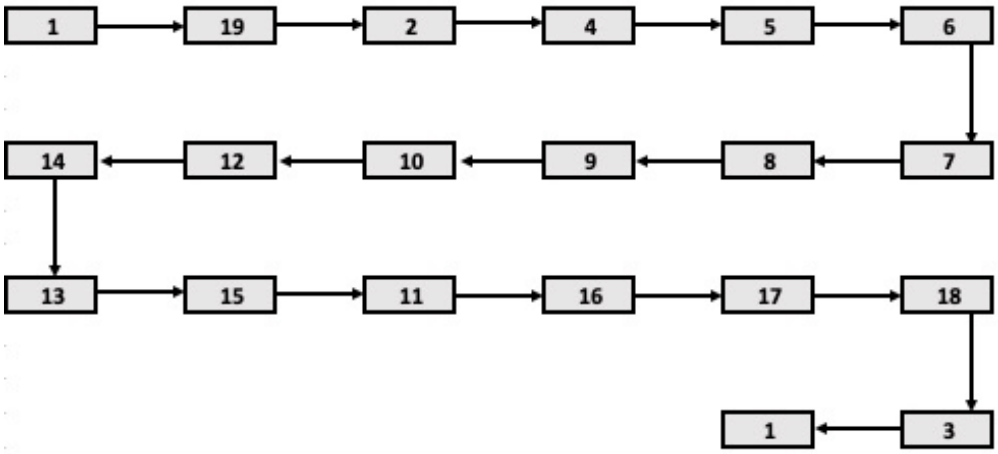
Tabulka 5 Výsledné hodnoty pro $\delta_{ij}^{c} = 0,1c_{ij}$ a $\Gamma = 54$

x_{ij}	δ_{ij}^{c} (hod)	c_{ij} (hod)	$\sum$ (hod)
x1_18	0,6314	6,3142	6,9456
x2_3	0,5183	5,1833	5,7016
x3_19	0,1700	1,7000	1,8700
x4_2	0,1673	1,6731	1,8404
x5_4	0,0892	0,8917	0,9809
x6_5	0,2930	2,9300	3,2230
x7_6	0,0352	0,3522	0,3874
x8_12	0,1305	1,3047	1,4352
x9_8	0,0445	0,4453	0,4898
x10_9	0,0432	0,4319	0,4751
x11_10	0,0960	0,9597	1,0557
x12_14	0,0879	0,8786	0,9665
x13_7	0,1795	1,7950	1,9745
x14_13	0,0206	0,2056	0,2262
x15_11	0,0858	0,8583	0,9441
x16_15	0,0879	0,8786	0,9665
x17_16	0,1059	1,0594	1,1653
x18_17	0,2461	2,4606	2,7067
x19_1	0,0156	0,1558	0,1714
$\sum$ (hod)	3,0479	30,4781	33,5260

Při dalším zvyšování hodnoty Γ se opakovaně vyskytují předchozí nalezené okruhy a pořadí míst v ani hodnota účelové funkce se nemění. Například pro hodnotu $\Gamma = 342$ vychází výsledný okruh stejně jako pro $\Gamma = 36$ (viz Obrázek 9).

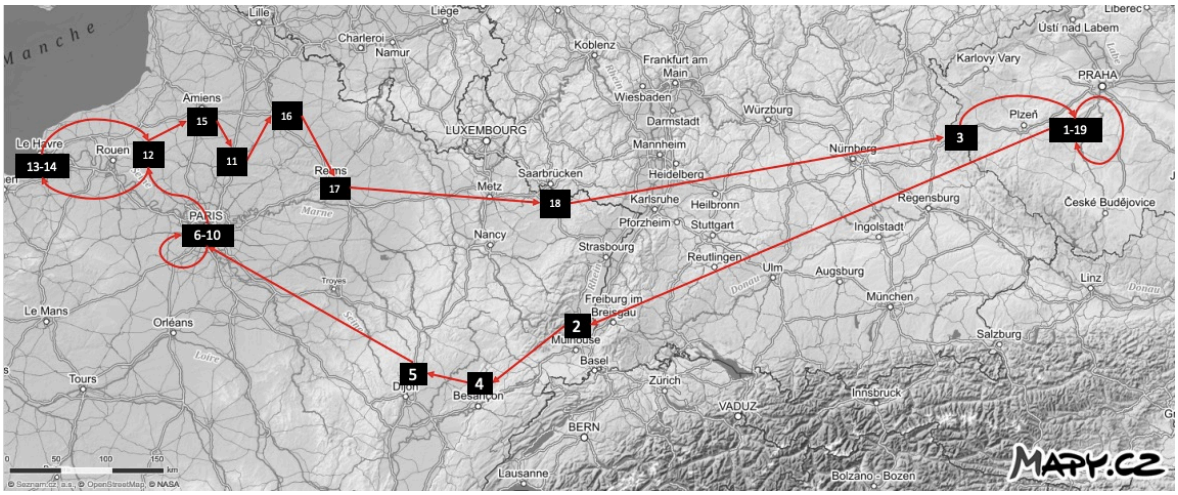
Pro další testovaný scénář je uvažováno $\delta_{ij}^{c} = 0,15c_{ij}$. Výpočet je prováděn na stejném modelu jako pro, pouze je upravená podmínka (4.6) tak, aby hodnoty δ_{ij}^{c} odpovídaly 15% z c_{ij}. Upravené podmínky jsou k vidění v přílohách. U hodnoty $\Gamma = 18$ je výsledné pořadí míst mírně jiné než v předchozích případech a je zobrazeno na následujícím schématu:

Obrázek 11 Výsledné pořadí míst pro $\delta_{ij}^{c} = 0,15c_{ij}$ a $\Gamma = 18$



Hodnota účelové funkce z důvodu nárůstu hodnoty zpoždění na jednotlivých trasách vyšší, než u varianty s $\delta_{ij}^{c} = 0,1c_{ij}$, konkrétně 34,74 hodin. Jednotlivé hodnoty c_{iJ} a δ_{ij}^{c} jsou zobrazeny v Tabulka 6.

Obrázek 12 Mapa okruhu pro $\delta_{ij}^{c} = 0,15c_{ij}$ a $\Gamma = 18$



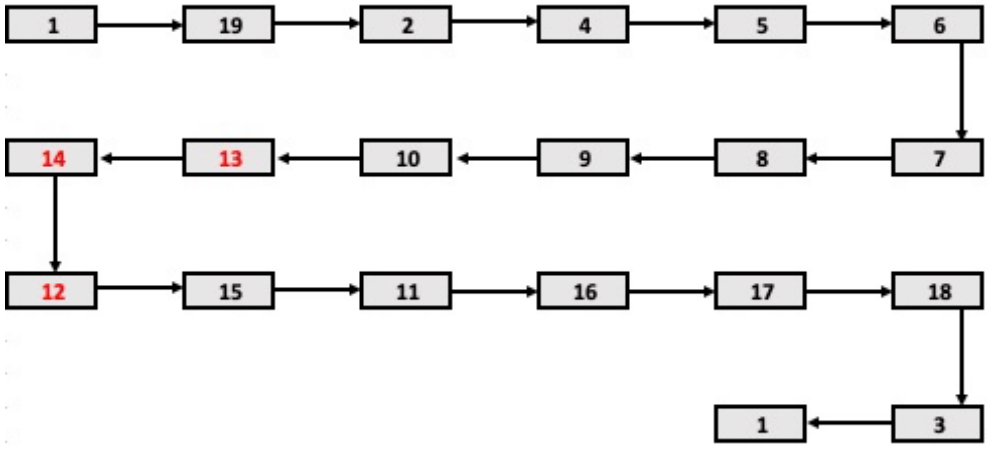
Zdroj: vlastní zpracování, Mapy.cz

Tabulka 6 Výsledné hodnoty pro $\delta_{ij}^{c} = 0,15c_{ij}$ a $\Gamma = 18$

x_{ij}	δ_{ij}^{c} (hod)	c_{ij} (hod)	$\sum$ (hod)
x1_19	0,0220	0,1469	0,1689
x2_4	0,2519	1,6794	1,9313
x3_1	0,2608	1,7386	1,9994
x4_5	0,1316	0,8775	1,0091
x5_6	0,4352	2,9011	3,3363
x6_7	0,0593	0,3953	0,4546
x7_8	0,0689	0,4594	0,5283
x8_9	0,0693	0,4617	0,5310
x9_10	0,0618	0,4117	0,4735
x10_12	0,2288	1,5256	1,7544
x11_16	0,1875	1,2500	1,4375
x12_14	0,1318	0,8786	1,0104
x13_15	0,2626	1,7506	2,0132
x14_13	0,0308	0,2056	0,2364
x15_11	0,1288	0,8583	0,9871
x16_17	0,1651	1,1008	1,2659
x17_18	0,0000	2,4781	2,4781
x18_3	0,6983	4,6556	5,3539
x19_2	1,0139	6,7594	7,7733
$\sum$ (hod)	4,2084	30,5342	34,7426

Pro hodnotu $\Gamma = 36$ dochází pouze k malé změně v pořadí, konkrétně po místě 10 následují 13 – 14 – 12 a dále je okruh stejný jako pro $\Gamma = 18$. Hodnota účelové funkce je 34,675 hodin. Stejný okruh a hodnota účelové funkce vychází i pro hodnoty $\Gamma = 56 - 342$ se nevyskytuje jiné pořadí míst ani hodnota účelové funkce. Pořadí míst je následující:

Obrázek 13 Výsledný okruh pro $\delta_{ij}^{c} = 0,15c_{ij}$ a $\Gamma = 36 - 342$



Tabulka 7 Výsledné hodnoty pro $\delta_{ij}^{c} = 0,15c_{ij}$ a $\Gamma = 36 - 342$

x_{ij}	δ_{ij}^{c} (hod)	c_{ij} (hod)	$\sum$ (hod)
x1_19	0,0220	0,1469	0,1689
x2_4	0,2519	1,6794	1,9313
x3_1	0,2608	1,7386	1,9994
x4_5	0,1316	0,8775	1,0091
x5_6	0,4352	2,9011	3,3363
x6_7	0,0593	0,3953	0,4546
x7_8	0,0689	0,4594	0,5283
x8_9	0,0693	0,4617	0,5310
x9_10	0,0618	0,4117	0,4735
x10_13	0,2813	1,8750	2,1563
x11_16	0,1875	1,2500	1,4375
x12_15	0,1977	1,3181	1,5158
x13_14	0,0313	0,2086	0,2399
x14_12	0,1326	0,8842	1,0168
x15_11	0,1288	0,8583	0,9871
x16_17	0,1651	1,1008	1,2659
x17_18	0,0000	2,4781	2,4781
x18_3	0,6983	4,6556	5,3539
x19_2	1,0139	6,7594	7,7733
$\sum$ (hod)	4,1973	30,4597	34,6570

Pro hodnoty $\delta_{ij}^{c} = 0,2c_{ij}$ až $\delta_{ij}^{c} = 0,3c_{ij}$ výpočet neposkytuje jiné okruhy než dosud uvedené, pouze dochází k zvýšení účelové funkce úměrně ke zvýšení hodnotyδ_{ij}^{c} . Ke změnám v pořadí míst dochází až pro hodnotu $\delta_{ij}^{c} = 0,4c_{ij}$ % a výše. Zvýšení o 40% je však velmi málo pravděpodobný scénář a není v této práci zahrnut.

Jako nejpravděpodobnější scénáře jsou po konzultaci s odpovědným pracovníkem firmy na základě jeho zkušeností vybrány dvě varianty, a to s hodnotami $\delta_{ij}^{c} = 0,1c_{ij}$., tzn. hodnoty c_{ij} se oproti hodnotám ve výchozí matici sazeb (viz Tabulka 2) zvýší o 10 %, a s hodnotami $\delta_{ij}^{c} = 0,3c_{ij}$, tj. dojde ke zvýšení o 30%.

Zvláště pak varianta $\delta_{ij}^{c} = 10\%$ je realističtější scénář než výpočet prostého okružního dopravního problému bez zahrnutí hlediska nejistoty v ohodnocení jednotlivých tras. Varianta s možným 30% zvýšením hodnot c_{ij} odpovídá poměrně často vyskytujícím se výkyvům v plynulosti dopravy na dané trase, způsobené hustotou provozu nebo jinými dopravními omezeními. Proto by mohla být využita právě pro období, kdy dochází

k výrazným nárůstům provozu, například prázdniny, případně při výskytu dopravních omezeních z důvodu rekonstrukce komunikací apod.

4.4 TSP s časovými okny

Na základě výsledků z předchozí kapitoly bude sestaven matematický model obchodního cestujícího s časovými okny (viz kapitola 3.6.5) pro stanovené scénáře a vypočteno optimální pořadí míst v okruhu. Následně bude provedeno srovnání se současnou situací a ekonomická analýza výsledného řešení.

4.4.1 Konstrukce matematického modelu (Fáze design)

Pro sestavení tohoto modelu je nutné určit dobu obsluhy v jednotlivých místech a časová okna, ve kterých mají být místa navštívena. Jak již bylo řečeno, model počítá s odjezdem k *j*-tému zákazníkovi bezprostředně po dokončení obsluhy *i*-tého zákazníka (viz kapitola 3.6.5).

Nejprve budou stanoveny doby obsluhy jednotlivých míst:

- 1 (počáteční místo) – 0 hodin,
- 2 (parkoviště) - 9 hodin (povinná pauza řidiče, která je zde absolvována vždy),
- 4-19 (vykládky/nakládky) – 45 minut.

Následně je třeba pro každé místo určit časová okna, která určují časové rozpětí, ve kterém má být místo obslouženo. Protože se jedná o přepravu, která probíhá ve více dnech, jsou hodnoty pro jednotlivá místa určena pro konkrétní dny od 8:00 hod do 16:00 hod. Čas, ve který řidič vyjíždí je stanoven jako $a = 0$ a od tohoto času se odvíjí hodnota všech ostatních časových oken. Při stanovení časových oken je třeba přihlédnout k době přejezdu mezi jednotlivými místy, aby byla zajištěna přípustnost modelu. Pomocí těchto hodnot lze částečně ovlivnit pořadí míst v okruhu, například místo 2 má být v první variantě navštíveno po výchozím místě. Podmínka, která v modelu zajišťuje navštívení místa v konkrétním čase vypadá například právě pro místo 2 následovně:

$$a_2 \geq 0 \tag{4.16}$$

$$a_2 \leq 16 \tag{4.17}$$

Hodnoty jsou pro ostatní místa nastaveny tak, aby obsluha daného místa probíhala od 8:00 do 16:00, tedy je zajištěno, že obsluha nebude prováděna v noci. Hodnoty a_i jsou pro jednotlivá místa sestaveny tak, aby bylo zachováno alespoň přibližně pořadí míst ze zvoleného výchozího robustního modelu. Konkrétní hodnoty časových oken a doby obsluhy pro jednotlivá místa jsou uvedeny v následující tabulce:

Tabulka 8 Časová okna a doby obsluhy

Číslo	Název	a_i	S_i
1	Háje	0	0
2	Parkoviště Mulhouse	8-24	9
3	Hraniční přechod Rozvadov	x	0,75
4	Saint Vit	24-32	0,75
5	Dijon	24-32	0,75
6	Paříž - Palasieu	48-56	0,75
7	Paříž - Viroflay	48-56	0,75
8	Paříž - Saint-Germain-en-lay	48-56	0,75
9	Paříž - Andrésy	48-56	0,75
10	Paříž - Osny	48-56	0,75
11	Clermont	72-80	0,75
12	Rouen	72-80	0,75
13	Le havre 1	72-80	0,75
14	Le havre 2	72-80	0,75
15	Amiens	72-80	0,75
16	Saint - Quentin	96-104	0,75
17	Reims	96-104	0,75
18	Saaerbrucken	x	0,75
19	Sklad Dobříš	x	0,75
		suma	21,75

V případě kamionové dopravy jsou ze zákona stanovené pravidelné pauzy řidiče, které je třeba brát v úvahu:

- řidič kamionu musí po 4,5 hodinách jízdy zařadit 45 min bezpečností pauzu,
- po 9 hodinách jízdy musí následovat 9 hodinová pauza.

Dodržování těchto přestávek je v modelu zajištěno tak, že jednotlivé úseky obsluhy v modelu jsou nastaveny na 8 – 16 hod a 9 hodinová pauza je absolvována mimo tento úsek. 45 minutové přestávky jsou v úsecích s malou vzdáleností mezi jednotlivými místy absolvovány při vykládce/nakládce. U delších úseků je přizpůsobena hodnota časového okna tak, aby při přejezdu mezi místy byla dostatečná časová rezerva. Přestávka tak může být provedena během cesty.

Dále je třeba doplnit model doplnit o podmínky zajišťující, že časový interval mezi návštěvou zákazníka j bezprostředně po zákazníkovi i má minimálně hodnotu t_{ij} (viz rovnice (3.18)). Pro x_{1_2} a x_{1_3} příslušná podmínka, upravená pro výpočet v programu Gurobi vypadá takto:

$$a_1 - a_2 + 1000000x_{1_2} + W_2 + v_{1_2} = 999983{,}532 \qquad (4.18)$$

$$a_1 - a_3 + 1000000x_{1_3} + W_3 + v_{1_3} = 999998{,}114 \qquad (4.19)$$

Tato podmínka musí být vyjádřena pro všechny spoje v modelu. Kompletní model upravený pro použití v programu Gurobi je k vidění v přílohách této práce.

4.4.2 **Výpočet (Fáze choice)**

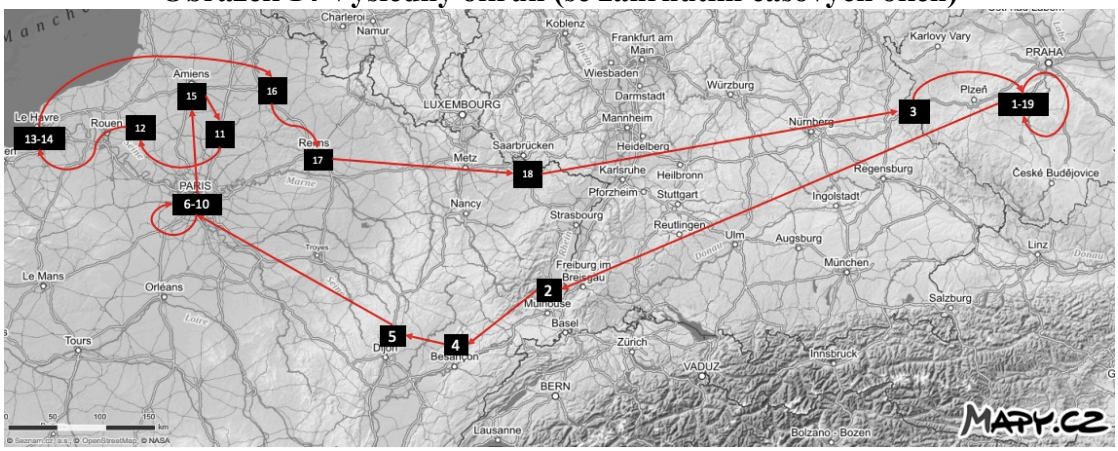
Fáze *choice* je poslední částí rozhodovacího procesu podle Simonovy teorie a bude zde proveden výpočet optimálního pořadí míst v okružním dopravním problému a analýza výsledku. K výpočtu byl opět využit software Gurobi Optimizer 8.1.0 build v8.1.0rc1 běžící na stejné konfiguraci jako v předchozím případě.

Pro výpočet platí:

- řidič vyjíždí z výchozího místa (1 na mapě) v pondělí v 8:00,
- doba obsluhy je 45 minut a řidič se jí neúčastní – může ji využít k odpočinku,
- povinná 9 hodinová pauza je , pokud je potřeba, absolvována vždy v mezi 16 hod aktuálního dne a 8 hod dne následujícího (kromě pauzy absolvované v místě 2).

Výsledný okruh pro daný model se kvůli zahrnutí omezujících podmínek, které určují časový interval navštívení místa, oproti výchozímu modelu mírně mění. Pořadí míst je 1 – 2 – 4 – 5 – 8 – 6 – 7 – 9 – 10 – 15 – 11 – 12 – 13 – 14 – 16 – 17 – 18 – 3 – 19 – 1 a na mapě vypadá následovně:

Obrázek 14 Výsledný okruh (se zahrnutím časových oken)

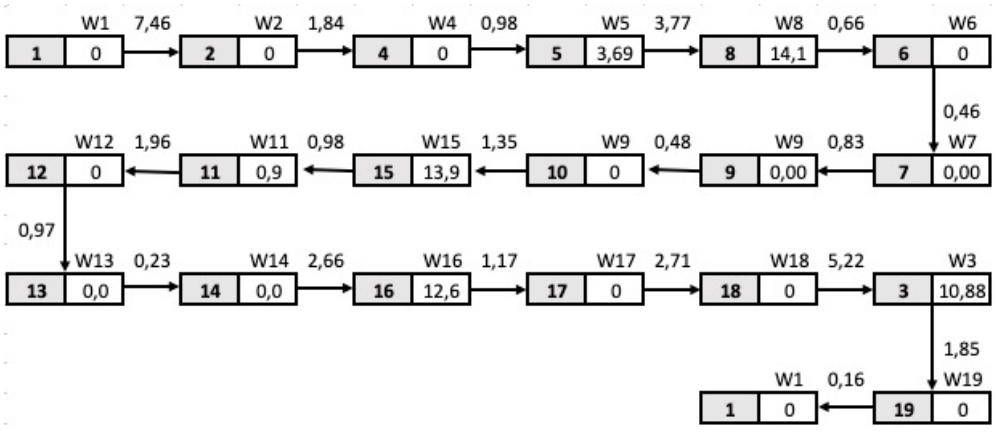


Zdroj: vlastní zpracování, Mapy.cz

Hodnota účelové funkce vč. doby obsluhy je 113, 44 hod. Výsledek modelu byl nalezen s nulovou hodnotou „gap“, tj. jedná se o řešení optimální.

V rámci okruhu dochází k čekáním v určitých místech, které mohou být využitý pro absolvování povinné pauzy. Hodnoty jednotlivých spojů v okruhu a hodnoty W_i (čekání) jsou uvedeny v následujícím schématu (v hodinách):

Obrázek 15 Optimální pořadí míst vč. hodnot W_i



Jak lze z vidět na přiloženém zobrazení trasy na mapě (), mezi místy 10 – 1 – 11 – 1 se vozidlo „vrací“. To je způsobeno tím, že model minimalizuje čekání v jednotlivých místech. Místo toho aby vozidlo v nějakém místě čekalo delší dobu, je do trasy zařazena cesta do jiného místa a čas je využit k cestě a obsluze dalších místa lepšímu využití času.

5 Výsledky a diskuse

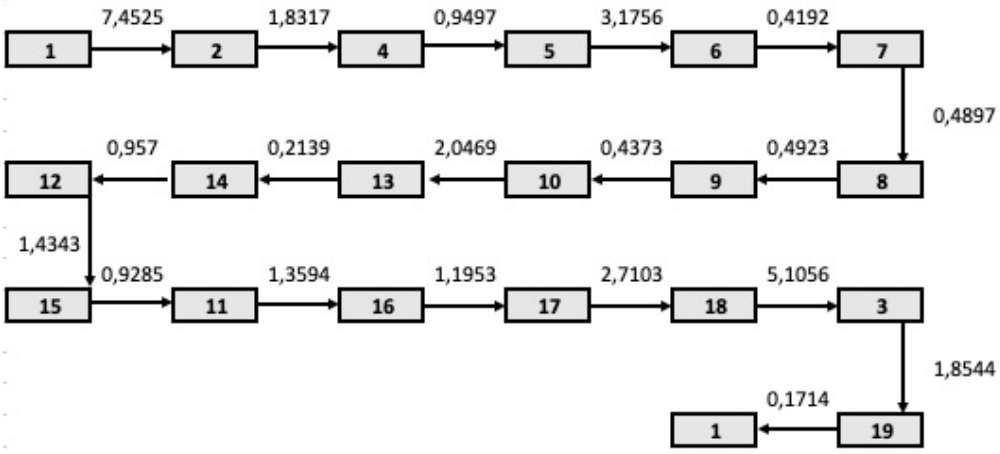
Cílem práce byla konstrukce robustních scénářů pro vybranou množinu míst a následně aplikovat model obchodního cestujícího se zahrnutím časových oken.

5.1 Konstrukce robustních scénářů

Jako první byly sestaven robustní model obchodního cestujícího. Bylo vytvořeno několik variant možných scénářů a následně byl po konzultaci s pověřeným pracovníkem firmy vybrán scénář nejpravděpodobnější. K výpočtu byl použit software Gurobi optimizer.

Konkrétní testované scénáře byly uvažovány se změnou hodnoty $\delta_{ij}^{c} = 0{,}1c_{ij}$, $\delta_{ij}^{c} = 0{,}15c_{ij}$, $\delta_{ij}^{c} = 0{,}2c_{ij}$ a $\delta_{ij}^{c} = 0{,}3c_{ij}$ v kombinaci s hodnotami $\Gamma \in \langle 18{,}342 \rangle$. Hodnoty Γ byly z důvodů časové náročnost zvoleny jako násobky 18. Jako nejpravděpodobnější scénář byl vybrán nejčastěji se vyskytující okruh pro $\delta_{ij}^{c} = 0{,}1c_{ij}$. Hodnota účelové funkce je 33,5057 hodin čisté jízdy a optimální pořadí míst je následující:

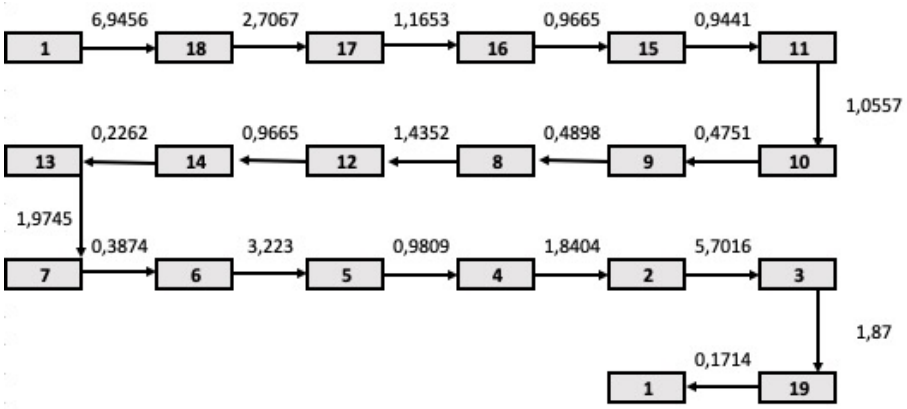
Obrázek 16 Pořadí míst pro nejpravděpodobnější scénář ($\delta_{ij}^{c} = 0{,}1c_{ij}$)



Pokud se tento scénář srovná s běžnou dobou čisté jízdy, tj. bez obsluhy a přestávek, která je dle zkušeností pracovníků firmy 38, vychází úspora 4,5 hodiny.

V pořadí míst v modelu od $\delta_{ij}^{c} = 0{,}15c_{ij}$ do $\delta_{ij}^{c} = 0{,}3c_{ij}$ v různých kombinacích s hodnotami Γ docházelo pouze k mírným změnám ve střední části okruhu, kde jsou vzdálenosti mezi jednotlivými místy poměrně krátké. Výjimku tvoří varianta $\delta_{ij}^{c} = 0{,}1c_{ij}$ a $\Gamma = 54$, kdy by měl být okruh absolvován opačně oproti většině ostatních výsledků. Přesné pořadí míst je následující:

Obrázek 17 Pořadí míst pro $\delta_{ij}^{c} = 0,1c_{ij}$ $\Gamma = 54$



Tento výsledek firma může použít při plánování dopravy v případě, že by měl být náklad vyložen nejprve v místech 18 – 17 – 16 – 15. Tato situace však nastává výjimečně, většinou kamiony míří nejdříve do míst 4 – 9.

Na základě výsledků testovaných scénářů lze říci, že i při malých změnách v ohodnocení tras může docházet k výrazným změnám pořadí míst v okruhu. Nelze však jednoznačně určit, k jak velké změně a na kterém spoji může v reálném případě dojít. Informace o maximální hodnotě doby strávené na trase pro jednotlivé scénáře je pro dispečery důležitým ukazatelem při dlouhodobějším vytěžování řidičů, zvláště pak s přihlédnutím k nutnosti dodržovat zákonné týdenní pauzy.

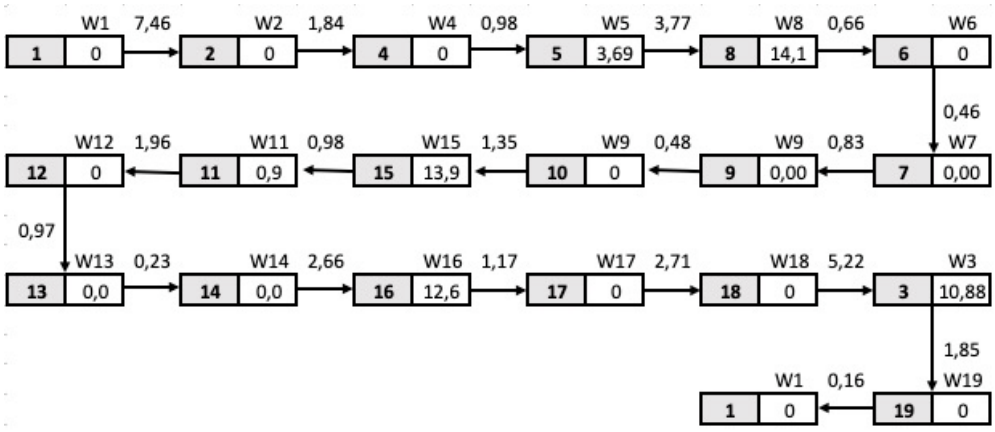
Běžná čistá doba jízdy podle informací z dopravní firmy je 38 hodin. Ve zvoleném modelu se jedná o 33,5075 hodin. To je úspora o 4,5 hodiny čisté jízdy. V celkovém součtu a v kontextu s dalším vytěžováním řidičů a povinnosti konat i týdenní a 14 denní pauzy znamená významnou úsporu. Dále se nabízí možnost využívat řidiče více pro rozvoz a svoz nákladů do distribučního centra firmy v Hájích u Příbrami a skladu firmy Bobcat, pokud dojde lepšímu naplánování trasy a získání rezervy v rámci povinných dlouhodobých přestávek.

Znalost doby jízdy mezi jednotlivými spoji včetně jejich změn může mít významný vliv při plánování dopravy. Zdržení mohou být způsobeny například dlouhodobou uzavírkou či dopravním omezením. Na základě informace o optimálním pořadí míst s danou hodnotou zdržení lze přizpůsobit plánování dopravy a vyhnout se případným komplikacím vhodnou kombinací objednávek.

5.2 Optimalizace v mezinárodní kamionové dopravě

Ze sestavených robustních scénářů byla pro sestrojení modelu obchodního cestujícího s časovými okny vybrána nejčastěji se vyskytující varianta pro $\delta_{ij}^{c} = 0{,}1c_{ij}$. Pro daná místa byly sestaveny tzv. časová okna, která udávají interval, ve kterém má být provedena obsluha konkrétního místa. Zároveň má každé místo stanovenou dobu obsluhy. Posléze byl sestaven odpovídající model a pomocí programu Gurobi optimizer vypočítán. Výsledné optimální pořadí míst vč. hodnot c_{ij} a W_i je následující:

Obrázek 18 Optimální pořadí míst vč. hodnot W_i



Hodnota účelové funkce vč. všech čekání a obsluhy míst je 113,44 hodin. Po přenesení jednotlivých spojů do matice vzdáleností v km je výsledná trasa dlouhá 2988,8 km (matice vzdáleností v km je uvedena v přílohách této práce). Průměrná vzdálenost řešeného okruhu, kterou řidiči absolvují je 3700 km. Vypočítaný model nabízí značně nižší ujetou vzdálenost a tím i úsporu nákladů. Náklad na 1 km firma vyčíslila na 28 Kč a v této částce je zahrnut plat řidiče, silniční poplatky a cena pohonných hmot.. To znamená, že za současné situace daný okruh stojí 107800,- a náklady na okruh, který je výsledkem modelu v této práci je 80676,-. Rozdíl je tedy 27127,-. To je značná úspora nákladů, zvláště přihlédne-li se k faktu, že je okruh absolvován s každý týden 1 - 2x, jedná se o nezanedbatelnou roční úsporu nákladů přibližně 1 300 000,- Kč.

5.3 Doporučení

Vzhledem k faktu, že dispečeři nevyužívají žádný systém, který by jim umožňoval nějakou formu optimalizace vzdáleností a času, a spoléhají se pouze na informace ohledně dopravní situace, aplikaci GPS a hlavně vlastní zkušenosti, mohlo by zavedení podobného programu přispět v celkovém měřítku ke značným úsporám. Software RAALTRANS, který firma používá v současné době, nabízí pouze funkce související s přiřazováním objednávek jednotlivým řidičům/kamionům, kilometrovník a výpočet nákladů na základě ručně zadané trasy a nenabízí žádnou možnost optimalizace vzdálenosti nebo času.

Pokud by byl uvažován software využitý k výpočtům v této práci, tedy Gurobi optimizer, bylo by jeho zavedení a využívání v podniku v jeho ryzí formě prakticky nereálné. Program vyžaduje znalosti konkrétních matematických modelů a jejich konstrukce. Program vyžaduje vstupy v podobě textového formátu a i na samotném modelu je třeba provést drobné úpravy. Většina modelů je navíc velmi rozsáhlých a v takovémto měřítku se velice snadno vyskytne chyba, kterou je velmi složité a zdlouhavé najít. Dalším faktorem, který znemožňuje využití programu Gurobi je fakt, že konstrukce modelu pro problém řešený v této práci je poměrně časově náročná a dispečer se musí věnovat i dalším řidičům a objednávkám. Model je i pro takovýto problém značně rozsáhlý a práce s ním je v kontextu dopravní firmy neefektivní. Program Gurobi lze propojit s MS Excel, což se jeví jako možné řešení, ovšem značné omezené a stále vyžadující určité znalosti v oblasti matematického modelování.

Lze doporučit využití programu, konkrétně modelově orientovaného systému pro podporu rozhodování, který umožňuje optimalizaci dopravních tras v podmínkách spediční firmy. Důležitý je přívětivější uživatelský interface pro osobu, která není seznámena s konstrukcí matematických modelů. Výhodou ne-li požadavkem je možnost vkládat vstupní data pomocí standardizovaného vstupního formuláře, např. informace o místech, která mají být obsloužena pomocí adresy nebo importem přímo z objednávkového systému. Samozřejmostí je kontrola správnosti dat. Součástí systému by měl být kilometrovník, informace o dopravní situaci (např. na základě informací z Google maps), sazby mýtného, databáze automobilů (sledování kapacity) a měl by umožňovat modifikaci již zadaných dat – např. za pomoci využití modelů a metod pro řešení dynamického modelu obchodního cestujícího. Jako výstup by měl systém poskytovat návrh několika tras s možností výběru, zda je cílem minimalizovat vzdálenost nebo například přepravní náklady (cena mýtného).

Velkou výhodou systému by bylo možnost udržovat databázi řidičů s přehledem čerpání týdenních/čtrnáctidenních limitů omezených legislativou. Otázkou je, zda takový program je v dnešní době na trhu a zda je finančně dostupný pro firmu takovéto velikosti.

6 Závěr

Cílem diplomové práce bylo analyzovat vybraný dopravní problém a následně na něj aplikovat model obchodního cestujícího s časovými okny. Dílčím cílem byla konstrukce robustních scénářů. Oba cíle diplomové práce byly splněny.

Teoretická část práce se věnovala v první části nejprve logistice, jejím cílům a použitím v oblasti dopravy. V další části byl charakterizován operačním výzkum a matematické modelování jako jeho nástroj a byl pospán průběhu konstrukce matematického modelu. Samostatná část byla věnována celočíselnému programování, které je konkrétně v podobě binárního programování použito v praktické části této práce. Celočíselný model byl charakterizován a byly uvedeny metody pro řešení celočíselných úloh. Dála byla uvedena oblast dopravních úloh, zejména problému obchodního cestujícího, vč. jeho modifikace se zahrnutím časových oken. Poslední část se věnovala robustní optimalizaci a úpravě lineárního optimalizačního modelu na robustní. Všechna uvedená teoretická východiska byla zpracována na základě syntézy odborných zdrojů, které byly prostudovány před zpracováním vlastní části práce.

Praktická část práce byla postupně zpracována na základě rozdělení podle Simonovy teorie. Nejdříve byl ve fázi *inteligence* identifikován problém, veškeré jeho podstatné části a na základě informací o rozložení míst byla vytvořena matice sazeb. Protože je v modelu pracováno s časovými údaji, je matice uvedena v hodinách. První kapitola vlastní části práce se věnovala konstrukci několika variant robustních scénářů pro zvolený dopravní problém. Ve fázi *design* byl upraven obecný matematický model obchodního cestujícího tak, aby zahrnoval hledisko nejistoty v ohodnocení vzdáleností mezi jednotlivými místy a pro jednotlivé scénáře byly sestaveny konkrétní omezující podmínky. Ve fázi *choice* byl proveden výpočet pomocí programu Gurobi optimizer, analýza výsledných scénářů a výběr nejpravděpodobnějšího scénáře, který byl použit v další části.

Druhá kapitola vlastní části práce navazuje na předchozí. Jako východisko pro konstrukci matematického modelu byl použit zvolený scénář vypočítaný v rámci robustní optimalizace dopravního problému. V této části bylo opět použito rozdělení podle Simonovy teorie. Ve fázi *design* byly určeny hodnoty časových oken a doby obsluhy jednotlivých míst. V návaznosti byl sestaven matematický model obchodního cestujícího zahrnující časová okna. Ve fázi *choice* byl proveden výpočet pomocí programu Gurobi optimizer,. Následně

bylo provedena ekonomická analýza výsledků a porovnání se současným řešením ve firmě. Z výsledků byly vyvozeny závěry a doporučení.

7 Seznam použitých zdrojů

APPLEGATE, David L. c2006. *The traveling salesman problem: a computational study*. Princeton: Princeton University Press. ISBN 978-0-691-12993-8.

ARORA, Sanjeev a Boaz BARAK, 2009. *Computational complexity: a modern approach*. New York: Cambridge University Press. ISBN 978-0521424264.

BALAS, Egon, 1989. The prize collecting traveling salesman problem. *Networks* [online]. Carnegie Mellon University, Pittsburgh, Pennsylvania, United States, Říjen 1989, **16**(6), 621-636 [cit. 2018-10-26]. ISSN 00283045. Dostupné z: https://doi-org.infozdroje.czu.cz/10.1002/net.3230190602

BERTSIMAS, D. and SIM, M.: Robust discrete optimization and network flows. Mathematical Programming, 98 (2003), 49-71.

BÜSING, Cristina a Fabio D'ANDREAGIOVANNI, 2012. New results under multi-band uncerainty in robust optimization. *Proc. Of the 11th symposium on Experimental Algorithms - SEA 2012* [online]. Heildelberg, 2012, 63-74 [cit. 2018-12-05].

BÜSING, C. and D'ANDREAGIOVANNI, F.: Robust optimization under Multi-band uncertainty – Part I: Theory, 2013.

BERTSIMAS, Dimitris a Melvyn SIM, 2004. The Price of Robustness. *Operations Research* [online]. **52**(1), 35-53 [cit. 2018-12-05]. DOI: 10.1287/opre.1030.0065. ISSN 0030-364X. Dostupné z: http://pubsonline.informs.org/doi/abs/10.1287/opre.1030.0065

BROŽOVÁ, Helena a Milan HOUŠKA, 2002. *Základní metody operační analýzy*. Praha: Česká zemědělská univerzita v Praze, Provozně ekonomická fakulta ve vydavatelství Credit. ISBN 9788021309517.

COOK, Stephen A., 1971. The complexity of theorem-proving procedures. *Proceedings of the third annual ACM symposium on Theory of computing - STOC '71*. New York, New

York, USA: ACM Press, 1971, 151-158. DOI: 10.1145/800157.805047. Dostupné také z: http://portal.acm.org/citation.cfm?doid=800157.805047

COOK, William, 2012. *Po stopách obchodního cestujícího: matematika na hranicích možností*. Praha: Argo. Zip (Argo: Dokořán): Dokořán). ISBN 978-80-7363-412-4.

DRAHOTSKÝ, Ivo a Bohumil ŘEZNÍČEK, 2003. *Logistika - procesy a jejich řízení*. Brno: Computer Press. Praxe manažera (Computer Press). ISBN 80-7226-521-0.

FÁBRY, Jan, 2011. *Matematické modelování*. Praha: Professional Publishing. ISBN 978-80-7431-066-9.

FIALA, Petr, 2010. Operační výzkum: nové trendy. Praha: Professional Publishing. ISBN 978-80-7431-036-2.

Gurobi Optimizer: State of the Art *Mathematical Programming Solver, b.r. Gurobi Optimization: An easier way to make better decisions* [online]. [cit. 2018-11-17]. Dostupné z: http://www.gurobi.com/products/gurobi-optimizer

GROS, Ivan, 2003. *Kvantitativní metody v manažerském rozhodování*. Praha: Grada. Expert (Grada). ISBN 8024704218.

HLADÍK, Milan, 2017. *Celočíselné programování*: text k přednášce. Praha, 90 s. Dostupné také z: https://kam.mff.cuni.cz/~hladik/CP/text_cp.pdf. Text k přednášce. Univerzita Karlova v Praze.

HLAVATÝ, Robert a Helena BROŽOVÁ, 2017. *Mathematical Methods in Economics 2017: Robust optimization approach in transportation problem* [online]. 1-6 [cit. 2018-11-17]. Dostupné z: https://www.researchgate.net/publication/319670419_Robust_optimization_approach_in_transportation_problem

JABLONSKÝ, Josef, 2002. *Operační výzkum: kvantitativní modely pro ekonomické rozhodování*. 2. vyd. Praha: Professional Publishing. ISBN 80-86419-42-8.

KAIN, Ravi a Ajay VERMA, 2018. Logistics Management in Supply Chain – An Overview. *Materials Today* [online]. 24. 3. 2018, (5), 3811-3816 [cit. 2018-10-25]. ISSN 2214-7853. Dostupné z: https://www.sciencedirect.com/science/article/pii/S2214785317329073

KAUFMAN, Arnold. Methods and models of operations research. Englewood Cliffs, N.J.: Prentice-Hall, 1963. ISBN 978-0135794098.

KLAPKA, Jindřich, Jiří DVOŘÁK a Pavel POPELA. *Metody operačního výzkumu*. Brno: Vysoké učení technické, 1996. ISBN 80-214-0817-0.

KOTLER, Philip, c2001. *Marketing management*. 10. rozš. vyd., 1. vyd. [Grada Publishing]. Praha: Grada. Profesionál. ISBN 80-247-0016-6.

LAGOVÁ, Milada a Josef JABLONSKÝ. *Lineární modely: Operační výzkum – kvantitativní modely pro ekonomické rozhodování*. Praha: Oeconomica, 2004. ISBN 80-245-0816-8.

MALAGUTI, Enrico, Silvano MARTELLO a Alberto SANTINI, 2018. The traveling salesman problem with pickups, deliveries, and draft limits. *Omega* [online]. 24. 1. 2017, (74), 50-58 [cit. 2018-10-26]. ISSN 0305-0483. Dostupné z: https://doi.org/10.1016/j.omega.2017.01.005

REINELT, G., c1994. *The traveling salesman: computational solutions for TSP applications*. New York: Springer-Verlag. ISBN 0-387-58334-3.

SIXTA, Josef a Václav MAČÁT, 2005. *Logistika: teorie a praxe*. Brno: CP Books. Business books (CP Books). ISBN 80-251-0573-3.

ŠTŮSEK, Jaromír. 2005. Logistický management. V Praze: Česká zemědělská univerzita, Provozně ekonomická fakulta, Katedra řízení. ISBN 80-213-1259-9.

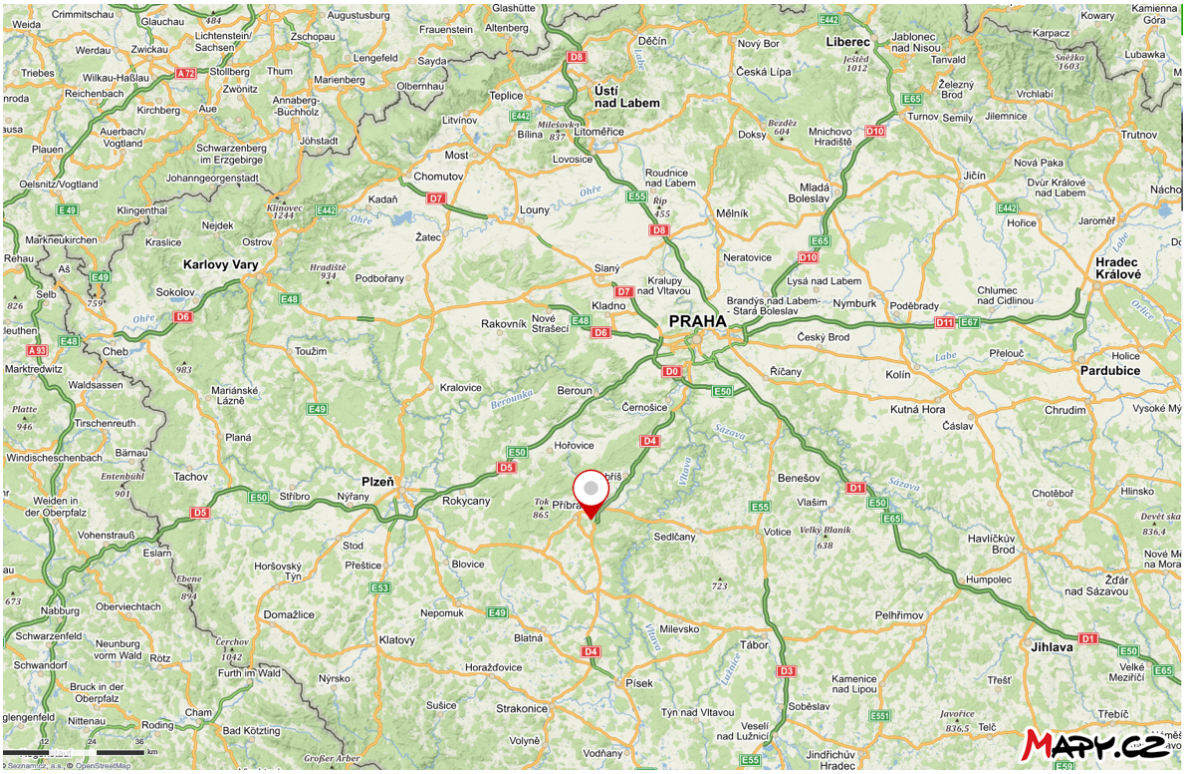
WAYNE L. WINSTON., 1997. Operations research: applications and algorithms. 3. ed. Belmont, Calif: Duxbury Press. ISBN 978-0534520205.

What is Operational Research?. The Operational Research Society [online]. [cit. 2019-03-25]. Dostupné z: https://www.theorsociety.com/about-or/

8 Přílohy

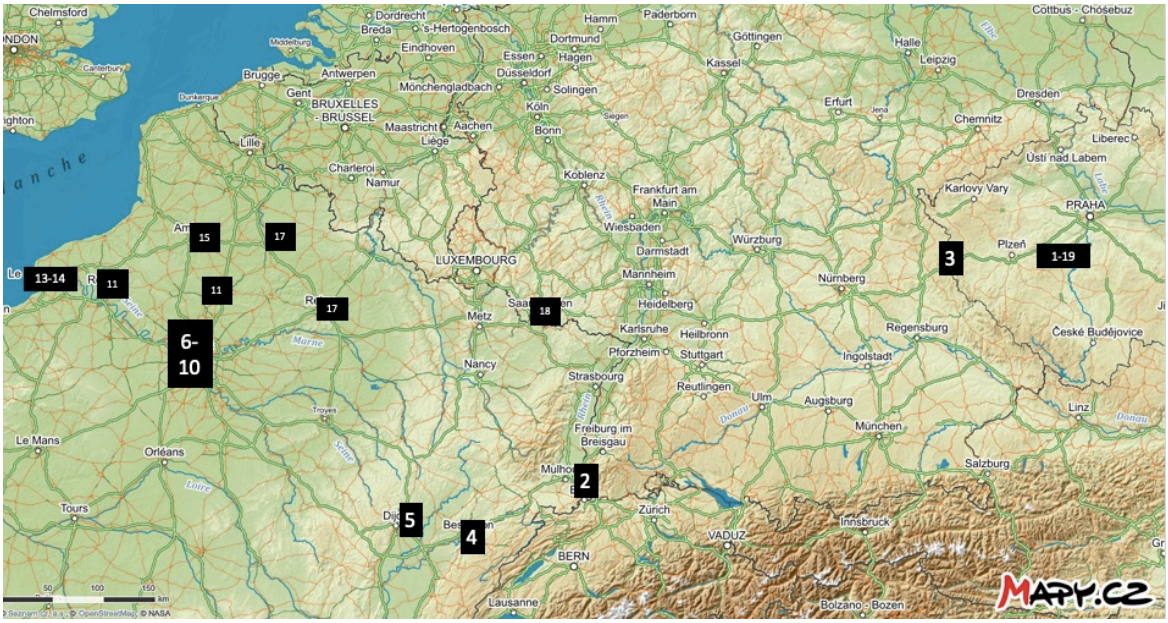
Příloha 1 Sídlo firmy Josef Vondráček s. r. o. 58
Příloha 2 Místa na trase 58
Příloha 3 Matice vzdáleností (v km) 59
Příloha 4 Robustní model $\delta ijc = 0{,}1cij$ Γz $= 18$ 60
Příloha 5 Omezující podmínka pro (4.6) $\delta ijc = 0{,}15cij$ 73
Příloha 6 Podmínka (4.6) pro $\delta ijc = 0{,}2cij$ 77
Příloha 7 Výstup z Gurobi pro $\delta ijc = 10\%$ Γ $= 18$ 81
Příloha 8 Výstup z Gurobi pro $\delta ijc = 10\%$ Γ $= 3$ 83
Příloha 9 Výstup z Gurobi pro $\delta ijc = 10\%$ Γ=56 84
Příloha 10 Výstup z Gurobi $\delta ijc = 15\%$ Γz $= 18$ 86
Příloha 11 Výstup z Gurobi pro $\delta ijc = 15\%$, Γ $= 36$ 88
Příloha 12 Matematický model TSP s časovými okny 90
Příloha 13 Výstup z programu Gurobi pro TSP s časovými okny 109

Příloha 1 Sídlo firmy Josef Vondráček s. r. o.



Zdroj: Mapy.cz

Příloha 2 Místa na trase



Zdroj: Mapy.cz

Příloha 3 Matice vzdáleností (v km)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	0,0	682,8	146,8	831,6	901,8	1027,1	1034,1	1045,4	1049,6	1046,9	999,5	1159,2	1214,2	1214,5	1045,8	968,3	869,8	625,1	6,0
2	682,2	0,0	536,8	163,1	233,3	533,6	552,0	566,2	586,7	529,5	561,6	682,8	725,5	734,5	634,9	557,3	458,8	233,6	679,8
3	146,6	537,4	0,0	686,2	756,4	881,7	888,7	900,0	904,2	901,6	854,2	1013,9	1068,8	1069,1	900,5	822,9	724,4	479,7	144,2
4	831,2	162,6	685,8	0,0	79,2	379,5	397,9	412,1	432,6	434,7	470,3	528,7	571,4	580,4	535,5	457,9	359,4	393,9	828,9
5	901,9	233,3	756,5	79,4	0,0	300,5	318,9	333,1	353,6	355,7	391,3	449,7	492,4	501,4	479,4	401,9	303,4	337,8	899,5
6	1029,1	535,5	883,7	381,7	302,4	0,0	17,4	31,6	52,1	64,7	104,0	148,3	190,9	199,9	165,3	187,2	162,5	410,7	1026,8
7	1033,6	550,3	888,2	396,4	317,2	17,2	0,0	16,5	37,0	50,4	101,8	133,1	175,8	184,8	151,1	185,0	166,9	415,2	1031,2
8	1046,0	565,8	900,6	411,9	332,7	32,8	16,7	0,0	16,8	25,5	102,3	119,2	161,9	170,9	133,5	185,2	179,3	427,6	1043,7
9	1049,6	585,3	904,2	431,5	352,2	52,3	36,2	16,7	0,0	11,8	94,9	125,6	168,3	177,3	126,1	180,1	182,9	431,2	1047,2
10	1047,3	531,7	901,9	435,1	355,9	64,0	48,9	24,5	12,3	0,0	77,2	124,5	167,1	176,1	108,4	181,1	180,6	428,9	1044,9
11	999,2	561,5	853,8	467,8	388,6	98,7	98,2	100,7	93,6	75,5	0,0	119,2	192,0	192,3	80,8	115,4	132,5	380,8	996,8
12	1159,3	681,8	1014,0	527,9	448,7	148,7	132,6	118,7	125,9	125,5	118,4	0,0	79,6	79,9	120,2	196,9	292,5	540,9	1157,0
13	1213,4	724,3	1068,0	570,5	491,2	191,3	175,2	161,3	168,5	168,0	192,2	80,7	0,0	8,9	174,3	250,9	346,6	595,0	1211,0
14	1213,7	732,1	1068,3	578,2	499,0	199,0	183,0	169,0	176,3	175,8	192,6	81,0	8,7	0,0	174,6	251,2	346,9	595,3	1211,4
15	1045,3	634,7	899,9	536,5	478,7	164,7	149,6	132,9	125,9	107,8	81,2	119,5	174,5	174,8	0,0	82,8	178,5	426,9	1042,9
16	968,7	558,1	823,3	459,9	402,1	183,8	183,3	183,5	181,4	178,8	115,6	195,9	250,8	251,1	82,5	0,0	101,8	350,3	966,3
17	870,2	459,6	724,8	361,4	303,6	159,8	166,8	178,2	182,3	179,7	132,5	291,3	346,2	346,5	177,9	100,3	0,0	251,8	867,8
18	624,8	234,8	479,4	396,6	338,8	409,3	416,3	427,7	431,8	429,2	381,8	541,5	596,4	596,7	428,1	350,5	252,0	0,0	622,4
19	6,0	680,4	144,4	829,2	899,4	1024,7	1031,7	1043,0	1047,2	1044,6	997,2	1156,9	1211,8	1212,1	1043,5	965,9	867,4	622,8	0,0

Příloha 4 Robustní model $\delta_{ij}^{c} = 0,1c_{ij}$ Γz = 18

Minimize

E

Subject to

6.789 x1_2 + 1.714 x1_3 + 8.26 x1_4 + 8.85 x1_5 + 10.041 x1_6 + 10.166 x1_7 + 10.343 x1_8 + 10.471 x1_9 + 10.284 x1_10 + 10.335 x1_11 + 11.173 x1_12 + 11.623 x1_13 + 11.624 x1_14 + 10.086 x1_15 + 9.395 x1_16 + 8.555 x1_17 + 6.314 x1_18 + 0.147 x1_19 + 6.786 x2_1 + 5.183 x2_3 + 1.679 x2_4 + 2.27 x2_5 + 4.894 x2_6 + 5.147 x2_7 + 5.354 x2_8 + 5.646 x2_9 + 5.555 x2_10 + 5.808 x2_11 + 6.379 x2_12 + 6.728 x2_13 + 6.81 x2_14 + 5.911 x2_15 + 5.22 x2_16 + 4.38 x2_17 + 2.472 x2_18 + 6.748 x2_19 + 5.221 x3_2 + 1.739 x3_1 + 6.692 x3_4 + 7.282 x3_5 + 8.473 x3_6 + 8.598 x3_7 + 8.775 x3_8 + 8.903 x3_9 + 8.716 x3_10 + 8.767 x3_11 + 9.605 x3_12 + 10.055 x3_13 + 10.056 x3_14 + 8.518 x3_15 + 7.827 x3_16 + 6.987 x3_17 + 4.746 x3_18 + 1.7 x3_19 + 1.673 x4_2 + 6.661 x4_3 + 8.264 x4_1 + 0.878 x4_5 + 3.502 x4_6 + 3.755 x4_7 + 3.962 x4_8 + 4.253 x4_9 + 4.282 x4_10 + 4.587 x4_11 + 4.987 x4_12 + 5.336 x4_13 + 5.418 x4_14 + 4.826 x4_15 + 4.135 x4_16 + 3.295 x4_17 + 3.804 x4_18 + 8.225 x4_19 + 2.269 x5_2 + 7.257 x5_3 +0.892 x5_4 +8.859 x5_1 + 2.901 x5_6 + 3.153 x5_7 + 3.36 x5_8 + 3.652 x5_9 + 3.68 x5_10 + 3.986 x5_11 + 4.386 x5_12 + 4.735 x5_13 + 4.817 x5_14 + 4.327 x5_15 + 3.637 x5_16 + 2.796 x5_17 + 3.306 x5_18 + 8.821 x5_19 + 4.932 x6_2 + 8.423 x6_3 + 3.555 x6_4 + 2.93 x6_5 + 10.026 x6_1 + 0.395 x6_7 + 0.602 x6_8 + 0.894 x6_9 + 0.974 x6_10 + 1.374 x6_11 + 1.628 x6_12 + 1.977 x6_13 + 2.059 x6_14 + 1.914 x6_15 + 2.054 x6_16 + 1.79 x6_17 + 3.994 x6_18 + 9.988 x6_19 + 5.149 x7_2 + 8.522 x7_3 + 3.772 x7_4 + 3.147 x7_5 + 0.352 x7_6 + 10.124 x7_1 + 0.459 x7_8 + 0.751 x7_9 + 0.838 x7_10 + 1.402 x7_11 + 1.485 x7_12 + 1.834 x7_13 + 1.916 x7_14 + 1.778 x7_15 + 2.083 x7_16 + 1.888 x7_17 + 4.092 x7_18 + 10.086 x7_19 + 5.43 x8_2 + 8.717 x8_3 + 4.053 x8_4 + 3.428 x8_5 + 0.633 x8_6 + 0.48 x8_7 + 10.32 x8_1 + 0.462 x8_9 + 0.539 x8_10 + 1.273 x8_11 + 1.305 x8_12 + 1.654 x8_13 + 1.736 x8_14 + 1.545 x8_15 + 1.989 x8_16 + 2.084 x8_17 + 4.287 x8_18 + 10.281 x8_19 + 5.703 x9_2 + 8.87 x9_3 + 4.326 x9_4 + 3.701 x9_5 + 0.906 x9_6 + 0.753 x9_7 + 0.445 x9_8 + 10.473 x9_1 + 0.412 x9_10 + 1.187 x9_11 + 1.53 x9_12 + 1.879 x9_13 + 1.961 x9_14 + 1.459 x9_15 + 2.058 x9_16 + 2.237 x9_17 + 4.44 x9_18 + 10.434 x9_19 + 5.665 x10_2 + 8.705 x10_3 + 4.354 x10_4 + 3.729 x10_5 + 0.981 x10_6 + 0.842 x10_7 + 0.516 x10_8 + 0.432 x10_9 + 10.308 x10_1 + 0.971 x10_11 + 1.526 x10_12 + 1.875 x10_13 + 1.957 x10_14 + 1.243 x10_15 + 1.869 x10_16 + 2.072 x10_17 + 4.276 x10_18 + 10.27 x10_19 + 5.901 x11_2 + 8.696 x11_3 + 4.648 x11_4 + 4.023 x11_5 + 1.368 x11_6 + 1.379 x11_7 + 1.237 x11_8 + 1.181 x11_9 + 0.96 x11_10 + 10.299 x11_1 + 1.784 x11_12 + 2.31 x11_13 + 2.311 x11_14 + 0.888 x11_15 + 1.25 x11_16 + 2.006 x11_17 + 4.266 x11_18 + 10.26 x11_19 + 6.403 x12_2 + 9.529 x12_3 + 5.026 x12_4 + 4.401 x12_5 + 1.606 x12_6 + 1.453 x12_7 + 1.294 x12_8 + 1.506 x12_9 + 1.552 x12_10 + 1.78 x12_11 + 11.133 x12_1 + 0.877 x12_13 + 0.879 x12_14 + 1.318 x12_15 + 1.981 x12_16 + 2.883 x12_17 + 5.1 x12_18 + 11.094 x12_19 + 6.745 x13_2 + 9.962 x13_3 + 5.368 x13_4 + 4.743 x13_5 + 1.948 x13_6 + 1.795 x13_7 + 1.636 x13_8 + 1.848 x13_9 + 1.894 x13_10 + 2.297 x13_11 + 0.879 x13_12 + 11.565 x13_1 + 0.209 x13_14 + 1.751 x13_15 + 2.414 x13_16 + 3.315 x13_17 + 5.532 x13_18 + 11.526 x13_19 + 6.833 x14_2 + 9.968 x14_3 + 5.456 x14_4 + 4.831 x14_5 + 2.036 x14_6 + 1.883 x14_7 + 1.724 x14_8 + 1.936 x14_9 + 1.982 x14_10 + 2.302 x14_11 +0.884 x14_12 + 0.206 x14_13 + 11.571 x14_1 + 1.756 x14_15 + 2.419 x14_16 + 3.321 x14_17 + 5.538 x14_18 + 11.532 x14_19 + 5.9 x15_2 + 8.436 x15_3 + 4.871 x15_4 + 4.338 x15_5 + 1.894 x15_6 + 1.756 x15_7 + 1.5 x15_8 + 1.444 x15_9 + 1.223 x15_10 + 0.858 x15_11 + 1.314 x15_12 + 1.764 x15_13 + 1.765 x15_14 + 10.039 x15_1 + 0.887 x15_16 + 1.789 x15_17 + 4.006 x15_18 + 10 x15_19 + 5.212 x16_2 + 7.748 x16_3 + 4.183 x16_4 + 3.65 x16_5 + 1.985 x16_6 + 1.996 x16_7 + 1.927 x16_8 + 1.979 x16_9 + 1.798 x16_10 + 1.233 x16_11 + 1.967 x16_12 + 2.416 x16_13 + 2.418 x16_14 + 0.879 x16_15 + 9.351 x16_1 + 1.101 x16_17 + 3.318 x16_18 + 9.312 x16_19 + 4.372 x17_2 + 6.908 x17_3 + 3.343 x17_4 + 2.809 x17_5 + 1.736 x17_6 + 1.861 x17_7 + 2.039 x17_8 + 2.166 x17_9 + 1.98 x17_10 + 1.971 x17_11 + 2.838 x17_12 + 3.288 x17_13 + 3.289 x17_14 + 1.75 x17_15 + 1.059 x17_16 + 8.511 x17_1 + 2.478 x17_18 + 8.472 x17_19 + 2.45 x18_2 + 4.656 x18_3 + 3.856 x18_4 + 3.323 x18_5 + 3.946 x18_6 + 4.072 x18_7 + 4.249 x18_8 + 4.377 x18_9 + 4.19 x18_10 + 4.241 x18_11 + 5.079 x18_12 + 5.529 x18_13 + 5.53 x18_14 + 3.991 x18_15 + 3.301 x18_16 + 2.461 x18_17 + 6.259 x18_1 + 6.22 x18_19 + 6.759 x19_2 + 1.684 x19_3 + 8.23 x19_4 + 8.821 x19_5 + 10.011 x19_6 + 10.136 x19_7 + 10.314 x19_8 + 10.441 x19_9 + 10.255 x19_10 + 10.306 x19_11 + 11.144 x19_12 + 11.593 x19_13 + 11.595 x19_14 + 10.056 x19_15 + 9.365 x19_16 + 8.525 x19_17 + 6.284 x19_18 + 0.156 x19_1 + 18 z + p1_2 + p1_3 + p1_4 + p1_5 + p1_6 + p1_7 + p1_8 + p1_9 + p1_10 + p1_11 + p1_12 + p1_13 + p1_14 + p1_15 + p1_16 + p1_17 + p1_18 + p1_19 + p2_1 + p2_3 + p2_4 + p2_5 + p2_6 + p2_7 + p2_8 + p2_9 + p2_10 + p2_11 + p2_12 + p2_13 + p2_14 + p2_15 + p2_16 + p2_17 + p2_18 + p2_19 + p3_1 + p3_2 + p3_4 + p3_5 + p3_6 + p3_7 + p3_8 +

p3_9 + p3_10 + p3_11 + p3_12 + p3_13 + p3_14 + p3_15 + p3_16 + p3_17 + p3_18 + p3_19 + p4_1 + p4_2 + p4_3 + p4_5 + p4_6 + p4_7 + p4_8 + p4_9 + p4_10 + p4_11 + p4_12 + p4_13 + p4_14 + p4_15 + p4_16 + p4_17 + p4_18 + p4_19 + p5_1 + p5_2 + p5_3 + p5_4 + p5_6 + p5_7 + p5_8 + p5_9 + p5_10 + p5_11 + p5_12 + p5_13 + p5_14 + p5_15 + p5_16 + p5_17 + p5_18 + p5_19 + p6_1 + p6_2 + p6_3 + p6_4 + p6_5 + p6_7 + p6_8 + p6_9 + p6_10 + p6_11 + p6_12 + p6_13 + p6_14 + p6_15 + p6_16 + p6_17 + p6_18 + p6_19 + p7_1 + p7_2 + p7_3 + p7_4 + p7_5 + p7_6 + p7_8 + p7_9 + p7_10 + p7_11 + p7_12 + p7_13 + p7_14 + p7_15 + p7_16 + p7_17 + p7_18 + p7_19 + p8_1 + p8_2 + p8_3 + p8_4 + p8_5 + p8_6 + p8_7 + p8_9 + p8_10 + p8_11 + p8_12 + p8_13 + p8_14 + p8_15 + p8_16 + p8_17 + p8_18 + p8_19 + p9_1 + p9_2 + p9_3 + p9_4 + p9_5 + p9_6 + p9_7 + p9_8 + p9_10 + p9_11 + p9_12 + p9_13 + p9_14 + p9_15 + p9_16 + p9_17 + p9_18 + p9_19 + p10_1 + p10_2 + p10_3 + p10_4 + p10_5 + p10_6 + p10_7 + p10_8 + p10_9 + p10_11 + p10_12 + p10_13 + p10_14 + p10_15 + p10_16 + p10_17 + p10_18 + p10_19 + p11_1 + p11_2 + p11_3 + p11_4 + p11_5 + p11_6 + p11_7 + p11_8 + p11_9 + p11_10 + p11_12 + p11_13 + p11_14 + p11_15 + p11_16 + p11_17 + p11_18 + p11_19 + p12_1 + p12_2 + p12_3 + p12_4 + p12_5 + p12_6 + p12_7 + p12_8 + p12_9 + p12_10 + p12_11 + p12_13 + p12_14 + p12_15 + p12_16 + p12_17 + p12_18 + p12_19 + p13_1 + p13_2 + p13_3 + p13_4 + p13_5 + p13_6 + p13_7 + p13_8 + p13_9 + p13_10 + p13_11 + p13_12 + p13_14 + p13_15 + p13_16 + p13_17 + p13_18 + p13_19 + p14_1 + p14_2 + p14_3 + p14_4 + p14_5 + p14_6 + p14_7 + p14_8 + p14_9 + p14_10 + p14_11 + p14_12 + p14_13 + p14_15 + p14_16 + p14_17 + p14_18 + p14_19 + p15_1 + p15_2 + p15_3 + p15_4 + p15_5 + p15_6 + p15_7 + p15_8 + p15_9 + p15_10 + p15_11 + p15_12 + p15_13 + p15_14 + p15_16 + p15_17 + p15_18 + p15_19 + p16_1 + p16_2 + p16_3 + p16_4 + p16_5 + p16_6 + p16_7 + p16_8 + p16_9 + p16_10 + p16_11 + p16_12 + p16_13 + p16_14 + p16_15 + p16_17 + p16_18 + p16_19 + p17_1 + p17_2 + p17_3 + p17_4 + p17_5 + p17_6 + p17_7 + p17_8 + p17_9 + p17_10 + p17_11 + p17_12 + p17_13 + p17_14 + p17_15 + p17_16 + p17_18 + p17_19 + p18_1 + p18_2 + p18_3 + p18_4 + p18_5 + p18_6 + p18_7 + p18_8 + p18_9 + p18_10 + p18_11 + p18_12 + p18_13 + p18_14 + p18_15 + p18_16 + p18_17 + p18_19 + p19_1 + p19_2 + p19_3 + p19_4 + p19_5 + p19_6 + p19_7 + p19_8 + p19_9 + p19_10 + p19_11 + p19_12 + p19_13 + p19_14 + p19_15 + p19_16 + p19_17 + p19_18 - E <= 0

z + p1_2 - 0.6789 x1_2 >= 0
z + p1_3 - 0.1714 x1_3 >= 0
z + p1_4 - 0.826 x1_4 >= 0
z + p1_5 - 0.885 x1_5 >= 0
z + p1_6 - 1.0041 x1_6 >= 0
z + p1_7 - 1.0166 x1_7 >= 0
z + p1_8 - 1.0343 x1_8 >= 0
z + p1_9 - 1.0471 x1_9 >= 0
z + p1_10 - 1.0284 x1_10 >= 0
z + p1_11 - 1.0335 x1_11 >= 0
z + p1_12 - 1.1173 x1_12 >= 0
z + p1_13 - 1.1623 x1_13 >= 0
z + p1_14 - 1.1624 x1_14 >= 0
z + p1_15 - 1.0086 x1_15 >= 0
z + p1_16 - 0.9395 x1_16 >= 0
z + p1_17 - 0.8555 x1_17 >= 0
z + p1_18 - 0.6314 x1_18 >= 0
z + p1_19 - 0.0147 x1_19 >= 1

z + p2_1 - 0.6786 x2_1 >= 0
z + p2_3 - 0.5183 x2_3 >= 0
z + p2_4 - 0.1679 x2_4 >= 0
z + p2_5 - 0.227 x2_5 >= 0
z + p2_6 - 0.4894 x2_6 >= 0
z + p2_7 - 0.5147 x2_7 >= 0
z + p2_8 - 0.5354 x2_8 >= 0
z + p2_9 - 0.5646 x2_9 >= 0
z + p2_10 - 0.5555 x2_10 >= 0
z + p2_11 - 0.5808 x2_11 >= 0
z + p2_12 - 0.6379 x2_12 >= 0
z + p2_13 - 0.6728 x2_13 >= 0
z + p2_14 - 0.681 x2_14 >= 0
z + p2_15 - 0.5911 x2_15 >= 0
z + p2_16 - 0.522 x2_16 >= 0
z + p2_17 - 0.438 x2_17 >= 0
z + p2_18 - 0.2472 x2_18 >= 0
z + p2_19 - 0.6748 x2_19 >= 1

z + p3_1 - 0.1739 x3_1 >= 0
z + p3_2 - 0.5221 x3_2 >= 0
z + p3_4 - 0.6692 x3_4 >= 0
z + p3_5 - 0.7282 x3_5 >= 0
z + p3_6 - 0.8473 x3_6 >= 0
z + p3_7 - 0.8598 x3_7 >= 0
z + p3_8 - 0.8775 x3_8 >= 0
z + p3_9 - 0.8903 x3_9 >= 0
z + p3_10 - 0.8716 x3_10 >= 0
z + p3_11 - 0.8767 x3_11 >= 0
z + p3_12 - 0.9605 x3_12 >= 0
z + p3_13 - 1.0055 x3_13 >= 0
z + p3_14 - 1.0056 x3_14 >= 0
z + p3_15 - 0.8518 x3_15 >= 0
z + p3_16 - 0.7827 x3_16 >= 0
z + p3_17 - 0.6987 x3_17 >= 0
z + p3_18 - 0.4746 x3_18 >= 0
z + p3_19 - 0.17 x3_19 >= 1

z + p4_1 - 0.8264 x4_1 >= 0
z + p4_2 - 0.1673 x4_2 >= 0
z + p4_3 - 0.6661 x4_3 >= 0
z + p4_5 - 0.0878 x4_5 >= 0
z + p4_6 - 0.3502 x4_6 >= 0
z + p4_7 - 0.3755 x4_7 >= 0
z + p4_8 - 0.3962 x4_8 >= 0

z + p4_9 - 0.4253 x4_9 >= 0
z + p4_10 - 0.4282 x4_10 >= 0
z + p4_11 - 0.4587 x4_11 >= 0
z + p4_12 - 0.4987 x4_12 >= 0
z + p4_13 - 0.5336 x4_13 >= 0
z + p4_14 - 0.5418 x4_14 >= 0
z + p4_15 - 0.4826 x4_15 >= 0
z + p4_16 - 0.4135 x4_16 >= 0
z + p4_17 - 0.3295 x4_17 >= 0
z + p4_18 - 0.3804 x4_18 >= 0
z + p4_19 - 0.8225 x4_19 >= 1

z + p5_1 - 0.8859 x5_1 >= 0
z + p5_2 - 0.2269 x5_2 >= 0
z + p5_3 - 0.7257 x5_3 >= 0
z + p5_4 - 0.0892 x5_4 >= 0
z + p5_6 - 0.2901 x5_6 >= 0
z + p5_7 - 0.3153 x5_7 >= 0
z + p5_8 - 0.336 x5_8 >= 0
z + p5_9 - 0.3652 x5_9 >= 0
z + p5_10 - 0.368 x5_10 >= 0
z + p5_11 - 0.3986 x5_11 >= 0
z + p5_12 - 0.4386 x5_12 >= 0
z + p5_13 - 0.4735 x5_13 >= 0
z + p5_14 - 0.4817 x5_14 >= 0
z + p5_15 - 0.4327 x5_15 >= 0
z + p5_16 - 0.3637 x5_16 >= 0
z + p5_17 - 0.2796 x5_17 >= 0
z + p5_18 - 0.3306 x5_18 >= 0
z + p5_19 - 0.8821 x5_19 >= 1

z + p6_1 - 1.0026 x6_1 >= 0
z + p6_2 - 0.4932 x6_2 >= 0
z + p6_3 - 0.8423 x6_3 >= 0
z + p6_4 - 0.3555 x6_4 >= 0
z + p6_5 - 0.293 x6_5 >= 0
z + p6_7 - 0.0395 x6_7 >= 0
z + p6_8 - 0.0602 x6_8 >= 0
z + p6_9 - 0.0894 x6_9 >= 0
z + p6_10 - 0.0974 x6_10 >= 0
z + p6_11 - 0.1374 x6_11 >= 0
z + p6_12 - 0.1628 x6_12 >= 0
z + p6_13 - 0.1977 x6_13 >= 0
z + p6_14 - 0.2059 x6_14 >= 0
z + p6_15 - 0.1914 x6_15 >= 0
z + p6_16 - 0.2054 x6_16 >= 0
z + p6_17 - 0.179 x6_17 >= 0
z + p6_18 - 0.3994 x6_18 >= 0
z + p6_19 - 0.9988 x6_19 >= 1

z + p7_1 - 1.0124 x7_1 >= 0
z + p7_2 - 0.5149 x7_2 >= 0
z + p7_3 - 0.8522 x7_3 >= 0
z + p7_4 - 0.3772 x7_4 >= 0
z + p7_5 - 0.3147 x7_5 >= 0
z + p7_6 - 0.0352 x7_6 >= 0
z + p7_8 - 0.0459 x7_8 >= 0
z + p7_9 - 0.0751 x7_9 >= 0
z + p7_10 - 0.0838 x7_10 >= 0
z + p7_11 - 0.1402 x7_11 >= 0
z + p7_12 - 0.1485 x7_12 >= 0
z + p7_13 - 0.1834 x7_13 >= 0
z + p7_14 - 0.1916 x7_14 >= 0
z + p7_15 - 0.1778 x7_15 >= 0
z + p7_16 - 0.2083 x7_16 >= 0
z + p7_17 - 0.1888 x7_17 >= 0
z + p7_18 - 0.4092 x7_18 >= 0
z + p7_19 - 1.0086 x7_19 >= 1

z + p8_1 - 1.032 x8_1 >= 0
z + p8_2 - 0.543 x8_2 >= 0
z + p8_3 - 0.8717 x8_3 >= 0
z + p8_4 - 0.4053 x8_4 >= 0
z + p8_5 - 0.3428 x8_5 >= 0
z + p8_6 - 0.0633 x8_6 >= 0
z + p8_7 - 0.048 x8_7 >= 0
z + p8_9 - 0.0462 x8_9 >= 0
z + p8_10 - 0.0539 x8_10 >= 0
z + p8_11 - 0.1273 x8_11 >= 0
z + p8_12 - 0.1305 x8_12 >= 0
z + p8_13 - 0.1654 x8_13 >= 0
z + p8_14 - 0.1736 x8_14 >= 0
z + p8_15 - 0.1545 x8_15 >= 0
z + p8_16 - 0.1989 x8_16 >= 0
z + p8_17 - 0.2084 x8_17 >= 0
z + p8_18 - 0.4287 x8_18 >= 0
z + p8_19 - 1.0281 x8_19 >= 0

z + p9_1 - 1.0473 x9_1 >= 0
z + p9_2 - 0.5703 x9_2 >= 0
z + p9_3 - 0.887 x9_3 >= 0
z + p9_4 - 0.4326 x9_4 >= 0
z + p9_5 - 0.3701 x9_5 >= 0
z + p9_6 - 0.0906 x9_6 >= 0
z + p9_7 - 0.0753 x9_7 >= 0
z + p9_8 - 0.0445 x9_8 >= 0
z + p9_10 - 0.0412 x9_10 >= 0
z + p9_11 - 0.1187 x9_11 >= 0
z + p9_12 - 0.153 x9_12 >= 0
z + p9_13 - 0.1879 x9_13 >= 0
z + p9_14 - 0.1961 x9_14 >= 0
z + p9_15 - 0.1459 x9_15 >= 0
z + p9_16 - 0.2058 x9_16 >= 0
z + p9_17 - 0.2237 x9_17 >= 0
z + p9_18 - 0.444 x9_18 >= 0
z + p9_19 - 1.0434 x9_19 >= 0

z + p10_1 - 1.0308 x10_1 >= 0
z + p10_2 - 0.5665 x10_2 >= 0
z + p10_3 - 0.8705 x10_3 >= 0
z + p10_4 - 0.4354 x10_4 >= 0
z + p10_5 - 0.3729 x10_5 >= 0
z + p10_6 - 0.0981 x10_6 >= 0
z + p10_7 - 0.0842 x10_7 >= 0

z + p10_8 - 0.0516 x10_8 >= 0
z + p10_9 - 0.0432 x10_9 >= 0
z + p10_11 - 0.0971 x10_11 >= 0
z + p10_12 - 0.1526 x10_12 >= 0
z + p10_13 - 0.1875 x10_13 >= 0
z + p10_14 - 0.1957 x10_14 >= 0
z + p10_15 - 0.1243 x10_15 >= 0
z + p10_16 - 0.1869 x10_16 >= 0
z + p10_17 - 0.2072 x10_17 >= 0
z + p10_18 - 0.4276 x10_18 >= 0
z + p10_19 - 1.027 x10_19 >= 0

z + p11_1 - 1.0299 x11_1 >= 0
z + p11_2 - 0.5901 x11_2 >= 0
z + p11_3 - 0.8696 x11_3 >= 0
z + p11_4 - 0.4648 x11_4 >= 0
z + p11_5 - 0.4023 x11_5 >= 0
z + p11_6 - 0.1368 x11_6 >= 0
z + p11_7 - 0.1379 x11_7 >= 0
z + p11_8 - 0.1237 x11_8 >= 0
z + p11_9 - 0.1181 x11_9 >= 0
z + p11_10 - 0.096 x11_10 >= 0
z + p11_12 - 0.1784 x11_12 >= 0
z + p11_13 - 0.231 x11_13 >= 0
z + p11_14 - 0.2311 x11_14 >= 0
z + p11_15 - 0.0888 x11_15 >= 0
z + p11_16 - 0.125 x11_16 >= 0
z + p11_17 - 0.2006 x11_17 >= 0
z + p11_18 - 0.4266 x11_18 >= 0
z + p11_19 - 1.026 x11_19 >= 0

z + p12_1 - 1.1133 x12_1 >= 0
z + p12_2 - 0.6403 x12_2 >= 0
z + p12_3 - 0.9529 x12_3 >= 0
z + p12_4 - 0.5026 x12_4 >= 0
z + p12_5 - 0.4401 x12_5 >= 0
z + p12_6 - 0.1606 x12_6 >= 0
z + p12_7 - 0.1453 x12_7 >= 0
z + p12_8 - 0.1294 x12_8 >= 0
z + p12_9 - 0.1506 x12_9 >= 0
z + p12_10 - 0.1552 x12_10 >= 0
z + p12_11 - 0.178 x12_11 >= 0
z + p12_13 - 0.0877 x12_13 >= 0
z + p12_14 - 0.0879 x12_14 >= 0
z + p12_15 - 0.1318 x12_15 >= 0
z + p12_16 - 0.1981 x12_16 >= 0
z + p12_17 - 0.2883 x12_17 >= 0
z + p12_18 - 0.51 x12_18 >= 0
z + p12_19 - 1.1094 x12_19 >= 0

z + p13_1 - 1.1565 x13_1 >= 0
z + p13_2 - 0.6745 x13_2 >= 0
z + p13_3 - 0.9962 x13_3 >= 0
z + p13_4 - 0.5368 x13_4 >= 0
z + p13_5 - 0.4743 x13_5 >= 0
z + p13_6 - 0.1948 x13_6 >= 0
z + p13_7 - 0.1795 x13_7 >= 0
z + p13_8 - 0.1636 x13_8 >= 0
z + p13_9 - 0.1848 x13_9 >= 0
z + p13_10 - 0.1894 x13_10 >= 0
z + p13_11 - 0.2297 x13_11 >= 0
z + p13_12 - 0.0879 x13_12 >= 0
z + p13_14 - 0.0209 x13_14 >= 0
z + p13_15 - 0.1751 x13_15 >= 0
z + p13_16 - 0.2414 x13_16 >= 0
z + p13_17 - 0.3315 x13_17 >= 0
z + p13_18 - 0.5532 x13_18 >= 0
z + p13_19 - 1.1526 x13_19 >= 0

z + p14_1 - 1.1571 x14_1 >= 0
z + p14_2 - 0.6833 x14_2 >= 0
z + p14_3 - 0.9968 x14_3 >= 0
z + p14_4 - 0.5456 x14_4 >= 0
z + p14_5 - 0.4831 x14_5 >= 0
z + p14_6 - 0.2036 x14_6 >= 0
z + p14_7 - 0.1883 x14_7 >= 0
z + p14_8 - 0.1724 x14_8 >= 0
z + p14_9 - 0.1936 x14_9 >= 0
z + p14_10 - 0.1982 x14_10 >= 0
z + p14_11 - 0.2302 x14_11 >= 0
z + p14_12 - 0.0884 x14_12 >= 0
z + p14_13 - 0.0206 x14_13 >= 0
z + p14_15 - 0.1756 x14_15 >= 0
z + p14_16 - 0.2419 x14_16 >= 0
z + p14_17 - 0.3321 x14_17 >= 0
z + p14_18 - 0.5538 x14_18 >= 0
z + p14_19 - 1.1532 x14_19 >= 0

z + p15_1 - 1.0039 x15_1 >= 0
z + p15_2 - 0.59 x15_2 >= 0
z + p15_3 - 0.8436 x15_3 >= 0
z + p15_4 - 0.4871 x15_4 >= 0
z + p15_5 - 0.4338 x15_5 >= 0
z + p15_6 - 0.1894 x15_6 >= 0
z + p15_7 - 0.1756 x15_7 >= 0
z + p15_8 - 0.15 x15_8 >= 0
z + p15_9 - 0.1444 x15_9 >= 0
z + p15_10 - 0.1223 x15_10 >= 0
z + p15_11 - 0.0858 x15_11 >= 0
z + p15_12 - 0.1314 x15_12 >= 0
z + p15_13 - 0.1764 x15_13 >= 0
z + p15_14 - 0.1765 x15_14 >= 0
z + p15_16 - 0.0887 x15_16 >= 0
z + p15_17 - 0.1789 x15_17 >= 0
z + p15_18 - 0.4006 x15_18 >= 0
z + p15_19 - 1 x15_19 >= 0

z + p16_1 - 0.9351 x16_1 >= 0
z + p16_2 - 0.5212 x16_2 >= 0
z + p16_3 - 0.7748 x16_3 >= 0
z + p16_4 - 0.4183 x16_4 >= 0
z + p16_5 - 0.365 x16_5 >= 0
z + p16_6 - 0.1985 x16_6 >= 0
z + p16_7 - 0.1996 x16_7 >= 0

z + p16_8 - 0.1927 x16_8 >= 0
z + p16_9 - 0.1979 x16_9 >= 0
z + p16_10 - 0.1798 x16_10 >= 0
z + p16_11 - 0.1233 x16_11 >= 0
z + p16_12 - 0.1967 x16_12 >= 0
z + p16_13 - 0.2416 x16_13 >= 0
z + p16_14 - 0.2418 x16_14 >= 0
z + p16_15 - 0.0879 x16_15 >= 0
z + p16_17 - 0.1101 x16_17 >= 0
z + p16_18 - 0.3318 x16_18 >= 0
z + p16_19 - 0.9312 x16_19 >= 0

z + p17_1 - 0.8511 x17_1 >= 0
z + p17_2 - 0.4372 x17_2 >= 0
z + p17_3 - 0.6908 x17_3 >= 0
z + p17_4 - 0.3343 x17_4 >= 0
z + p17_5 - 0.2809 x17_5 >= 0
z + p17_6 - 0.1736 x17_6 >= 0
z + p17_7 - 0.1861 x17_7 >= 0
z + p17_8 - 0.2039 x17_8 >= 0
z + p17_9 - 0.2166 x17_9 >= 0
z + p17_10 - 0.198 x17_10 >= 0
z + p17_11 - 0.1971 x17_11 >= 0
z + p17_12 - 0.2838 x17_12 >= 0
z + p17_13 - 0.3288 x17_13 >= 0
z + p17_14 - 0.3289 x17_14 >= 0
z + p17_15 - 0.175 x17_15 >= 0
z + p17_16 - 0.1059 x17_16 >= 0
z + p17_18 - 0.2478 x17_18 >= 0
z + p17_19 - 0.8472 x17_19 >= 0

z + p18_1 - 0.6259 x18_1 >= 0
z + p18_2 - 0.245 x18_2 >= 0
z + p18_3 - 0.4656 x18_3 >= 0
z + p18_4 - 0.3856 x18_4 >= 0
z + p18_5 - 0.3323 x18_5 >= 0
z + p18_6 - 0.3946 x18_6 >= 0
z + p18_7 - 0.4072 x18_7 >= 0
z + p18_8 - 0.4249 x18_8 >= 0
z + p18_9 - 0.4377 x18_9 >= 0
z + p18_10 - 0.419 x18_10 >= 0
z + p18_11 - 0.4241 x18_11 >= 0
z + p18_12 - 0.5079 x18_12 >= 0
z + p18_13 - 0.5529 x18_13 >= 0
z + p18_14 - 0.553 x18_14 >= 0
z + p18_15 - 0.3991 x18_15 >= 0
z + p18_16 - 0.3301 x18_16 >= 0
z + p18_17 - 0.2461 x18_17 >= 0
z + p18_19 - 0.622 x18_19 >= 0

z + p19_1 - 0.0156 x19_1 >= 0
z + p19_2 - 0.6759 x19_2 >= 0
z + p19_3 - 0.1684 x19_3 >= 0
z + p19_4 - 0.823 x19_4 >= 0
z + p19_5 - 0.8821 x19_5 >= 0
z + p19_6 - 1.0011 x19_6 >= 0
z + p19_7 - 1.0136 x19_7 >= 0
z + p19_8 - 1.0314 x19_8 >= 0
z + p19_9 - 1.0441 x19_9 >= 0
z + p19_10 - 1.0255 x19_10 >= 0
z + p19_11 - 1.0306 x19_11 >= 0
z + p19_12 - 1.1144 x19_12 >= 0
z + p19_13 - 1.1593 x19_13 >= 0
z + p19_14 - 1.1595 x19_14 >= 0
z + p19_15 - 1.0056 x19_15 >= 0
z + p19_16 - 0.9365 x19_16 >= 0
z + p19_17 - 0.8525 x19_17 >= 0
z + p19_18 - 0.6284 x19_18 >= 0

p1_2 >= 0
p1_3 >= 0
p1_4 >= 0
p1_5 >= 0
p1_6 >= 0
p1_7 >= 0
p1_8 >= 0
p1_9 >= 0
p1_10 >= 0
p1_11 >= 0
p1_12 >= 0
p1_13 >= 0
p1_14 >= 0
p1_15 >= 0
p1_16 >= 0
p1_17 >= 0
p1_18 >= 0
p1_19 >= 0

p2_1 >= 0
p2_3 >= 0
p2_4 >= 0
p2_5 >= 0
p2_6 >= 0
p2_7 >= 0
p2_8 >= 0
p2_9 >= 0
p2_10 >= 0
p2_11 >= 0
p2_12 >= 0
p2_13 >= 0
p2_14 >= 0
p2_15 >= 0
p2_16 >= 0
p2_17 >= 0
p2_18 >= 0
p2_19 >= 0

p3_1 >= 0
p3_2 >= 0
p3_4 >= 0
p3_5 >= 0
p3_6 >= 0
p3_7 >= 0
p3_8 >= 0

p3_9 >= 0
p3_10 >= 0
p3_11 >= 0
p3_12 >= 0
p3_13 >= 0
p3_14 >= 0
p3_15 >= 0
p3_16 >= 0
p3_17 >= 0
p3_18 >= 0
p3_19 >= 0

p4_1 >= 0
p4_2 >= 0
p4_3 >= 0
p4_5 >= 0
p4_6 >= 0
p4_7 >= 0
p4_8 >= 0
p4_9 >= 0
p4_10 >= 0
p4_11 >= 0
p4_12 >= 0
p4_13 >= 0
p4_14 >= 0
p4_15 >= 0
p4_16 >= 0
p4_17 >= 0
p4_18 >= 0
p4_19 >= 0

p5_1 >= 0
p5_2 >= 0
p5_3 >= 0
p5_4 >= 0
p5_6 >= 0
p5_7 >= 0
p5_8 >= 0
p5_9 >= 0
p5_10 >= 0
p5_11 >= 0
p5_12 >= 0
p5_13 >= 0
p5_14 >= 0
p5_15 >= 0
p5_16 >= 0
p5_17 >= 0
p5_18 >= 0
p5_19 >= 0

p6_1 >= 0
p6_2 >= 0
p6_3 >= 0
p6_4 >= 0
p6_5 >= 0
p6_7 >= 0
p6_8 >= 0
p6_9 >= 0
p6_10 >= 0
p6_11 >= 0
p6_12 >= 0
p6_13 >= 0
p6_14 >= 0
p6_15 >= 0
p6_16 >= 0
p6_17 >= 0
p6_18 >= 0
p6_19 >= 0

p7_1 >= 0
p7_2 >= 0
p7_3 >= 0
p7_4 >= 0
p7_5 >= 0
p7_6 >= 0
p7_8 >= 0
p7_9 >= 0
p7_10 >= 0
p7_11 >= 0
p7_12 >= 0
p7_13 >= 0
p7_14 >= 0
p7_15 >= 0
p7_16 >= 0
p7_17 >= 0
p7_18 >= 0
p7_19 >= 0

p8_1 >= 0
p8_2 >= 0
p8_3 >= 0
p8_4 >= 0
p8_5 >= 0
p8_6 >= 0
p8_7 >= 0
p8_9 >= 0
p8_10 >= 0
p8_11 >= 0
p8_12 >= 0
p8_13 >= 0
p8_14 >= 0
p8_15 >= 0
p8_16 >= 0
p8_17 >= 0
p8_18 >= 0
p8_19 >= 0

p9_1 >= 0
p9_2 >= 0
p9_3 >= 0
p9_4 >= 0
p9_5 >= 0
p9_6 >= 0
p9_7 >= 0

p9_8 >= 0
p9_10 >= 0
p9_11 >= 0
p9_12 >= 0
p9_13 >= 0
p9_14 >= 0
p9_15 >= 0
p9_16 >= 0
p9_17 >= 0
p9_18 >= 0
p9_19 >= 0

p10_1 >= 0
p10_2 >= 0
p10_3 >= 0
p10_4 >= 0
p10_5 >= 0
p10_6 >= 0
p10_7 >= 0
p10_8 >= 0
p10_9 >= 0
p10_11 >= 0
p10_12 >= 0
p10_13 >= 0
p10_14 >= 0
p10_15 >= 0
p10_16 >= 0
p10_17 >= 0
p10_18 >= 0
p10_19 >= 0

p11_1 >= 0
p11_2 >= 0
p11_3 >= 0
p11_4 >= 0
p11_5 >= 0
p11_6 >= 0
p11_7 >= 0
p11_8 >= 0
p11_9 >= 0
p11_10 >= 0
p11_12 >= 0
p11_13 >= 0
p11_14 >= 0
p11_15 >= 0
p11_16 >= 0
p11_17 >= 0
p11_18 >= 0
p11_19 >= 0

p12_1 >= 0
p12_2 >= 0
p12_3 >= 0
p12_4 >= 0
p12_5 >= 0
p12_6 >= 0
p12_7 >= 0

p12_8 >= 0
p12_9 >= 0
p12_10 >= 0
p12_11 >= 0
p12_13 >= 0
p12_14 >= 0
p12_15 >= 0
p12_16 >= 0
p12_17 >= 0
p12_18 >= 0
p12_19 >= 0

p13_1 >= 0
p13_2 >= 0
p13_3 >= 0
p13_4 >= 0
p13_5 >= 0
p13_6 >= 0
p13_7 >= 0
p13_8 >= 0
p13_9 >= 0
p13_10 >= 0
p13_11 >= 0
p13_12 >= 0
p13_14 >= 0
p13_15 >= 0
p13_16 >= 0
p13_17 >= 0
p13_18 >= 0
p13_19 >= 0

p14_1 >= 0
p14_2 >= 0
p14_3 >= 0
p14_4 >= 0
p14_5 >= 0
p14_6 >= 0
p14_7 >= 0
p14_8 >= 0
p14_9 >= 0
p14_10 >= 0
p14_11 >= 0
p14_12 >= 0
p14_13 >= 0
p14_15 >= 0
p14_16 >= 0
p14_17 >= 0
p14_18 >= 0
p14_19 >= 0

p15_1 >= 0
p15_2 >= 0
p15_3 >= 0
p15_4 >= 0
p15_5 >= 0
p15_6 >= 0
p15_7 >= 0

p15_8 >= 0
p15_9 >= 0
p15_10 >= 0
p15_11 >= 0
p15_12 >= 0
p15_13 >= 0
p15_14 >= 0
p15_16 >= 0
p15_17 >= 0
p15_18 >= 0
p15_19 >= 0

p16_1 >= 0
p16_2 >= 0
p16_3 >= 0
p16_4 >= 0
p16_5 >= 0
p16_6 >= 0
p16_7 >= 0
p16_8 >= 0
p16_9 >= 0
p16_10 >= 0
p16_11 >= 0
p16_12 >= 0
p16_13 >= 0
p16_14 >= 0
p16_15 >= 0
p16_17 >= 0
p16_18 >= 0
p16_19 >= 0

p17_1 >= 0
p17_2 >= 0
p17_3 >= 0
p17_4 >= 0
p17_5 >= 0
p17_6 >= 0
p17_7 >= 0
p17_8 >= 0
p17_9 >= 0
p17_10 >= 0
p17_11 >= 0
p17_12 >= 0
p17_13 >= 0
p17_14 >= 0
p17_15 >= 0
p17_16 >= 0
p17_18 >= 0
p17_19 >= 0

p18_1 >= 0
p18_2 >= 0
p18_3 >= 0
p18_4 >= 0
p18_5 >= 0
p18_6 >= 0
p18_7 >= 0
p18_8 >= 0
p18_9 >= 0
p18_10 >= 0
p18_11 >= 0
p18_12 >= 0
p18_13 >= 0
p18_14 >= 0
p18_15 >= 0
p18_16 >= 0
p18_17 >= 0
p18_19 >= 0

p19_1 >= 0
p19_2 >= 0
p19_3 >= 0
p19_4 >= 0
p19_5 >= 0
p19_6 >= 0
p19_7 >= 0
p19_8 >= 0
p19_9 >= 0
p19_10 >= 0
p19_11 >= 0
p19_12 >= 0
p19_13 >= 0
p19_14 >= 0
p19_15 >= 0
p19_16 >= 0
p19_17 >= 0
p19_18 >= 0

x1_2 + x1_3 + x1_4 + x1_5 + x1_6 + x1_7 + x1_8 + x1_9 + x1_10 + x1_11 + x1_12 + x1_13 + x1_14 + x1_15 + x1_16 + x1_17 + x1_18 + x1_19 = 1
x2_1 + x2_3 + x2_4 + x2_5 + x2_6 + x2_7 + x2_8 + x2_9 + x2_10 + x2_11 + x2_12 + x2_13 + x2_14 + x2_15 + x2_16 + x2_17 + x2_18 + x2_19 = 1
x3_2 + x3_1 + x3_4 + x3_5 + x3_6 + x3_7 + x3_8 + x3_9 + x3_10 + x3_11 + x3_12 + x3_13 + x3_14 + x3_15 + x3_16 + x3_17 + x3_18 + x3_19 = 1
x4_2 + x4_3 + x4_1 + x4_5 + x4_6 + x4_7 + x4_8 + x4_9 + x4_10 + x4_11 + x4_12 + x4_13 + x4_14 + x4_15 + x4_16 + x4_17 + x4_18 + x4_19 = 1
x5_2 + x5_3 + x5_4 + x5_1 + x5_6 + x5_7 + x5_8 + x5_9 + x5_10 + x5_11 + x5_12 + x5_13 + x5_14 + x5_15 + x5_16 + x5_17 + x5_18 + x5_19 = 1
x6_2 + x6_3 + x6_4 + x6_5 + x6_1 + x6_7 + x6_8 + x6_9 + x6_10 + x6_11 + x6_12 + x6_13 + x6_14 + x6_15 + x6_16 + x6_17 + x6_18 + x6_19 = 1

x7_2 + x7_3 + x7_4 + x7_5 + x7_6 + x7_1 + x7_8 + x7_9 + x7_10 + x7_11 + x7_12 + x7_13 + x7_14 + x7_15 + x7_16 + x7_17 + x7_18 + x7_19 = 1
x8_2 + x8_3 + x8_4 + x8_5 + x8_6 + x8_7 + x8_1 + x8_9 + x8_10 + x8_11 + x8_12 + x8_13 + x8_14 + x8_15 + x8_16 + x8_17 + x8_18 + x8_19 = 1
x9_2 + x9_3 + x9_4 + x9_5 + x9_6 + x9_7 + x9_8 + x9_1 + x9_10 + x9_11 + x9_12 + x9_13 + x9_14 + x9_15 + x9_16 + x9_17 + x9_18 + x9_19 = 1
x10_1 + x10_2 + x10_3 + x10_4 + x10_5 + x10_6 + x10_7 + x10_8 + x10_9 + x10_11 + x10_12 + x10_13 + x10_14 + x10_15 + x10_16 + x10_17 + x10_18 + x10_19 = 1
x11_1 + x11_2 + x11_3 + x11_4 + x11_5 + x11_6 + x11_7 + x11_8 + x11_9 + x11_10 + x11_12 + x11_13 + x11_14 + x11_15 + x11_16 + x11_17 + x11_18 + x11_19 = 1
x12_1 + x12_2 + x12_3 + x12_4 + x12_5 + x12_6 + x12_7 + x12_8 + x12_9 + x12_11 + x12_10 + x12_13 + x12_14 + x12_15 + x12_16 + x12_17 + x12_18 + x12_19 = 1
x13_1 + x13_2 + x13_3 + x13_4 + x13_5 + x13_6 + x13_7 + x13_8 + x13_9 + x13_11 + x13_12 + x13_10 + x13_14 + x13_15 + x13_16 + x13_17 + x13_18 + x13_19 = 1
x14_1 + x14_2 + x14_3 + x14_4 + x14_5 + x14_6 + x14_7 + x14_8 + x14_9 + x14_11 + x14_12 + x14_13 + x14_10 + x14_15 + x14_16 + x14_17 + x14_18 + x14_19 = 1
x15_1 + x15_2 + x15_3 + x15_4 + x15_5 + x15_6 + x15_7 + x15_8 + x15_9 + x15_11 + x15_12 + x15_13 + x15_14 + x15_10 + x15_16 + x15_17 + x15_18 + x15_19 = 1
x16_1 + x16_2 + x16_3 + x16_4 + x16_5 + x16_6 + x16_7 + x16_8 + x16_9 + x16_11 + x16_12 + x16_13 + x16_14 + x16_15 + x16_10 + x16_17 + x16_18 + x16_19 = 1
x17_1 + x17_2 + x17_3 + x17_4 + x17_5 + x17_6 + x17_7 + x17_8 + x17_9 + x17_11 + x17_12 + x17_13 + x17_14 + x17_15 + x17_16 + x17_10 + x17_18 + x17_19 = 1
x18_1 + x18_2 + x18_3 + x18_4 + x18_5 + x18_6 + x18_7 + x18_8 + x18_9 + x18_11 + x18_12 + x18_13 + x18_14 + x18_15 + x18_16 + x18_17 + x18_10 + x18_19 = 1
x19_1 + x19_2 + x19_3 + x19_4 + x19_5 + x19_6 + x19_7 + x19_8 + x19_9 + x19_11 + x19_12 + x19_13 + x19_14 + x19_15 + x19_16 + x19_17 + x19_18 + x19_10 = 1

x2_1 + x3_1 + x4_1 + x5_1 + x6_1 + x7_1 + x8_1 + x9_1 + x10_1 + x11_1 + x12_1 + x13_1 + x14_1 + x15_1 + x16_1 + x17_1 + x18_1 + x19_1 = 1
x1_2 + x3_2 + x4_2 + x5_2 + x6_2 + x7_2 + x8_2 + x9_2 + x10_2 + x11_2 + x12_2 + x13_2 + x14_2 + x15_2 + x16_2 + x17_2 + x18_2 + x19_2 = 1
x1_3 + x2_3 + x4_3 + x5_3 + x6_3 + x7_3 + x8_3 + x9_3 + x10_3 + x11_3 + x12_3 + x13_3 + x14_3 + x15_3 + x16_3 + x17_3 + x18_3 + x19_3 = 1
x1_4 + x2_4 + x3_4 + x5_4 + x6_4 + x7_4 + x8_4 + x9_4 + x10_4 + x11_4 + x12_4 + x13_4 + x14_4 + x15_4 + x16_4 + x17_4 + x18_4 + x19_4 = 1
x1_5 + x2_5 + x3_5 + x4_5 + x6_5 + x7_5 + x8_5 + x9_5 + x10_5 + x11_5 + x12_5 + x13_5 + x14_5 + x15_5 + x16_5 + x17_5 + x18_5 + x19_5 = 1
x1_6 + x2_6 + x3_6 + x4_6 + x5_6 + x7_6 + x8_6 + x9_6 + x10_6 + x11_6 + x12_6 + x13_6 + x14_6 + x15_6 + x16_6 + x17_6 + x18_6 + x19_6 = 1
x1_7 + x2_7 + x3_7 + x4_7 + x5_7 + x6_7 + x8_7 + x9_7 + x10_7 + x11_7 + x12_7 + x13_7 + x14_7 + x15_7 + x16_7 + x17_7 + x18_7 + x19_7 = 1
x1_8 + x2_8 + x3_8 + x4_8 + x5_8 + x6_8 + x7_8 + x9_8 + x10_8 + x11_8 + x12_8 + x13_8 + x14_8 + x15_8 + x16_8 + x17_8 + x18_8 + x19_8 = 1
x1_9 + x2_9 + x3_9 + x4_9 + x5_9 + x6_9 + x7_9 + x8_9 + x10_9 + x11_9 + x12_9 + x13_9 + x14_9 + x15_9 + x16_9 + x17_9 + x18_9 + x19_9 = 1
x1_10 + x2_10 + x3_10 + x4_10 + x5_10 + x6_10 + x7_10 + x8_10 + x9_10 + x11_10 + x12_10 + x13_10 + x14_10 + x15_10 + x16_10 + x17_10 + x18_10 + x19_10 = 1
x1_11 + x2_11 + x3_11 + x4_11 + x5_11 + x6_11 + x7_11 + x8_11 + x9_11 + x10_11 + x12_11 + x13_11 + x14_11 + x15_11 + x16_11 + x17_11 + x18_11 + x19_11 = 1
x1_12 + x2_12 + x3_12 + x4_12 + x5_12 + x6_12 + x7_12 + x8_12 + x9_12 + x10_12 + x11_12 + x13_12 + x14_12 + x15_12 + x16_12 + x17_12 + x18_12 + x19_12 = 1
x1_13 + x2_13 + x3_13 + x4_13 + x5_13 + x6_13 + x7_13 + x8_13 + x9_13 + x10_13 + x11_13 + x12_13 + x14_13 + x15_13 + x16_13 + x17_13 + x18_13 + x19_13 = 1
x1_14 + x2_14 + x3_14 + x4_14 + x5_14 + x6_14 + x7_14 + x8_14 + x9_14 + x10_14 + x11_14 + x12_14 + x13_14 + x15_14 + x16_14 + x17_14 + x18_14 + x19_14 = 1
x1_15 + x2_15 + x3_15 + x4_15 + x5_15 + x6_15 + x7_15 + x8_15 + x9_15 + x10_15 + x11_15 + x12_15 + x13_15 + x14_15 + x16_15 + x17_15 + x18_15 + x19_15 = 1

x1_16 + x2_16 + x3_16 + x4_16 + x5_16 + x6_16 + x7_16 + x8_16 + x9_16 + x10_16 + x11_16 + x12_16 + x13_16 + x14_16 + x15_16 + x17_16 + x18_16 + x19_16 = 1
x1_17 + x2_17 + x3_17 + x4_17 + x5_17 + x6_17 + x7_17 + x8_17 + x9_17 + x10_17 + x11_17 + x12_17 + x13_17 + x14_17 + x15_17 + x16_17 + x18_17 + x19_17 = 1
x1_18 + x2_18 + x3_18 + x4_18 + x5_18 + x6_18 + x7_18 + x8_18 + x9_18 + x10_18 + x11_18 + x12_18 + x13_18 + x14_18 + x15_18 + x16_18 + x17_18 + x19_18 = 1
x1_19 + x2_19 + x3_19 + x4_19 + x5_19 + x6_19 + x7_19 + x8_19 + x9_19 + x10_19 + x11_19 + x12_19 + x13_19 + x14_19 + x15_19 + x16_19 + x17_19 + x18_19 = 1

u1 - u2 + 19 x1_2 <= 18
u1 - u3 + 19 x1_3 <= 18
u1 - u4 + 19 x1_4 <= 18
u1 - u5 + 19 x1_5 <= 18
u1 - u6 + 19 x1_6 <= 18
u1 - u7 + 19 x1_7 <= 18
u1 - u8 + 19 x1_8 <= 18
u1 - u9 + 19 x1_9 <= 18
u1 - u10 + 19 x1_10 <= 18
u1 - u11 + 19 x1_11 <= 18
u1 - u12 + 19 x1_12 <= 18
u1 - u13 + 19 x1_13 <= 18
u1 - u14 + 19 x1_14 <= 18
u1 - u15 + 19 x1_15 <= 18
u1 - u16 + 19 x1_16 <= 18
u1 - u17 + 19 x1_17 <= 18
u1 - u18 + 19 x1_18 <= 18
u1 - u19 + 19 x1_19 <= 18

u2 - u3 + 19 x2_3 <= 18
u2 - u4 + 19 x2_4 <= 18
u2 - u5 + 19 x2_5 <= 18
u2 - u6 + 19 x2_6 <= 18
u2 - u7 + 19 x2_7 <= 18
u2 - u8 + 19 x2_8 <= 18
u2 - u9 + 19 x2_9 <= 18
u2 - u10 + 19 x2_10 <= 18
u2 - u11 + 19 x2_11 <= 18
u2 - u12 + 19 x2_12 <= 18
u2 - u13 + 19 x2_13 <= 18
u2 - u14 + 19 x2_14 <= 18
u2 - u15 + 19 x2_15 <= 18
u2 - u16 + 19 x2_16 <= 18
u2 - u17 + 19 x2_17 <= 18
u2 - u18 + 19 x2_18 <= 18
u2 - u19 + 19 x2_19 <= 18

u3 - u2 + 19 x3_2 <= 18
u3 - u4 + 19 x3_4 <= 18
u3 - u5 + 19 x3_5 <= 18
u3 - u6 + 19 x3_6 <= 18
u3 - u7 + 19 x3_7 <= 18
u3 - u8 + 19 x3_8 <= 18
u3 - u9 + 19 x3_9 <= 18
u3 - u10 + 19 x3_10 <= 18
u3 - u11 + 19 x3_11 <= 18
u3 - u12 + 19 x3_12 <= 18
u3 - u13 + 19 x3_13 <= 18
u3 - u14 + 19 x3_14 <= 18
u3 - u15 + 19 x3_15 <= 18
u3 - u16 + 19 x3_16 <= 18
u3 - u17 + 19 x3_17 <= 18
u3 - u18 + 19 x3_18 <= 18
u3 - u19 + 19 x3_19 <= 18

u4 - u2 + 19 x4_2 <= 18
u4 - u3 + 19 x4_3 <= 18
u4 - u5 + 19 x4_5 <= 18
u4 - u6 + 19 x4_6 <= 18
u4 - u7 + 19 x4_7 <= 18
u4 - u8 + 19 x4_8 <= 18
u4 - u9 + 19 x4_9 <= 18
u4 - u10 + 19 x4_10 <= 18
u4 - u11 + 19 x4_11 <= 18
u4 - u12 + 19 x4_12 <= 18
u4 - u13 + 19 x4_13 <= 18
u4 - u14 + 19 x4_14 <= 18
u4 - u15 + 19 x4_15 <= 18
u4 - u16 + 19 x4_16 <= 18
u4 - u17 + 19 x4_17 <= 18
u4 - u18 + 19 x4_18 <= 18
u4 - u19 + 19 x4_19 <= 18

u5 - u2 + 19 x5_2 <= 18
u5 - u3 + 19 x5_3 <= 18
u5 - u4 + 19 x5_4 <= 18
u5 - u6 + 19 x5_6 <= 18
u5 - u7 + 19 x5_7 <= 18
u5 - u8 + 19 x5_8 <= 18
u5 - u9 + 19 x5_9 <= 18
u5 - u10 + 19 x5_10 <= 18
u5 - u11 + 19 x5_11 <= 18
u5 - u12 + 19 x5_12 <= 18
u5 - u13 + 19 x5_13 <= 18
u5 - u14 + 19 x5_14 <= 18
u5 - u15 + 19 x5_15 <= 18
u5 - u16 + 19 x5_16 <= 18
u5 - u17 + 19 x5_17 <= 18
u5 - u18 + 19 x5_18 <= 18
u5 - u19 + 19 x5_19 <= 18

u6 - u2 + 19 x6_2 <= 18
u6 - u3 + 19 x6_3 <= 18
u6 - u4 + 19 x6_4 <= 18
u6 - u5 + 19 x6_5 <= 18
u6 - u7 + 19 x6_7 <= 18
u6 - u8 + 19 x6_8 <= 18

u6 - u9 + 19 x6_9 <= 18
u6 - u10 + 19 x6_10 <= 18
u6 - u11 + 19 x6_11 <= 18
u6 - u12 + 19 x6_12 <= 18
u6 - u13 + 19 x6_13 <= 18
u6 - u14 + 19 x6_14 <= 18
u6 - u15 + 19 x6_15 <= 18
u6 - u16 + 19 x6_16 <= 18
u6 - u17 + 19 x6_17 <= 18
u6 - u18 + 19 x6_18 <= 18
u6 - u19 + 19 x6_19 <= 18

u7 - u2 + 19 x7_2 <= 18
u7 - u3 + 19 x7_3 <= 18
u7 - u4 + 19 x7_4 <= 18
u7 - u5 + 19 x7_5 <= 18
u7 - u6 + 19 x7_6 <= 18
u7 - u8 + 19 x7_8 <= 18
u7 - u9 + 19 x7_9 <= 18
u7 - u10 + 19 x7_10 <= 18
u7 - u11 + 19 x7_11 <= 18
u7 - u12 + 19 x7_12 <= 18
u7 - u13 + 19 x7_13 <= 18
u7 - u14 + 19 x7_14 <= 18
u7 - u15 + 19 x7_15 <= 18
u7 - u16 + 19 x7_16 <= 18
u7 - u17 + 19 x7_17 <= 18
u7 - u18 + 19 x7_18 <= 18
u7 - u19 + 19 x7_19 <= 18

u8 - u2 + 19 x8_2 <= 18
u8 - u3 + 19 x8_3 <= 18
u8 - u4 + 19 x8_4 <= 18
u8 - u5 + 19 x8_5 <= 18
u8 - u6 + 19 x8_6 <= 18
u8 - u7 + 19 x8_7 <= 18
u8 - u9 + 19 x8_9 <= 18
u8 - u10 + 19 x8_10 <= 18
u8 - u11 + 19 x8_11 <= 18
u8 - u12 + 19 x8_12 <= 18
u8 - u13 + 19 x8_13 <= 18
u8 - u14 + 19 x8_14 <= 18
u8 - u15 + 19 x8_15 <= 18
u8 - u16 + 19 x8_16 <= 18
u8 - u17 + 19 x8_17 <= 18
u8 - u18 + 19 x8_18 <= 18
u8 - u19 + 19 x8_19 <= 18

u9 - u2 + 19 x9_2 <= 18
u9 - u3 + 19 x9_3 <= 18
u9 - u4 + 19 x9_4 <= 18
u9 - u5 + 19 x9_5 <= 18
u9 - u6 + 19 x9_6 <= 18
u9 - u7 + 19 x9_7 <= 18
u9 - u8 + 19 x9_8 <= 18
u9 - u10 + 19 x9_10 <= 18
u9 - u11 + 19 x9_11 <= 18
u9 - u12 + 19 x9_12 <= 18
u9 - u13 + 19 x9_13 <= 18
u9 - u14 + 19 x9_14 <= 18
u9 - u15 + 19 x9_15 <= 18
u9 - u16 + 19 x9_16 <= 18
u9 - u17 + 19 x9_17 <= 18
u9 - u18 + 19 x9_18 <= 18
u9 - u19 + 19 x9_19 <= 18

u10 - u2 + 19 x10_2 <= 18
u10 - u3 + 19 x10_3 <= 18
u10 - u4 + 19 x10_4 <= 18
u10 - u5 + 19 x10_5 <= 18
u10 - u6 + 19 x10_6 <= 18
u10 - u7 + 19 x10_7 <= 18
u10 - u8 + 19 x10_8 <= 18
u10 - u9 + 19 x10_9 <= 18
u10 - u11 + 19 x10_11 <= 18
u10 - u12 + 19 x10_12 <= 18
u10 - u13 + 19 x10_13 <= 18
u10 - u14 + 19 x10_14 <= 18
u10 - u15 + 19 x10_15 <= 18
u10 - u16 + 19 x10_16 <= 18
u10 - u17 + 19 x10_17 <= 18
u10 - u18 + 19 x10_18 <= 18
u10 - u19 + 19 x10_19 <= 18

u11 - u2 + 19 x11_2 <= 18
u11 - u3 + 19 x11_3 <= 18
u11 - u4 + 19 x11_4 <= 18
u11 - u5 + 19 x11_5 <= 18
u11 - u6 + 19 x11_6 <= 18
u11 - u7 + 19 x11_7 <= 18
u11 - u8 + 19 x11_8 <= 18
u11 - u9 + 19 x11_9 <= 18
u11 - u10 + 19 x11_10 <= 18
u11 - u12 + 19 x11_12 <= 18
u11 - u13 + 19 x11_13 <= 18
u11 - u14 + 19 x11_14 <= 18
u11 - u15 + 19 x11_15 <= 18
u11 - u16 + 19 x11_16 <= 18
u11 - u17 + 19 x11_17 <= 18
u11 - u18 + 19 x11_18 <= 18
u11 - u19 + 19 x11_19 <= 18

u12 - u2 + 19 x12_2 <= 18
u12 - u3 + 19 x12_3 <= 18
u12 - u4 + 19 x12_4 <= 18
u12 - u5 + 19 x12_5 <= 18
u12 - u6 + 19 x12_6 <= 18
u12 - u7 + 19 x12_7 <= 18
u12 - u8 + 19 x12_8 <= 18
u12 - u9 + 19 x12_9 <= 18
u12 - u10 + 19 x12_10 <= 18
u12 - u11 + 19 x12_11 <= 18
u12 - u13 + 19 x12_13 <= 18
u12 - u14 + 19 x12_14 <= 18

u12 - u15 + 19 x12_15 <= 18
u12 - u16 + 19 x12_16 <= 18
u12 - u17 + 19 x12_17 <= 18
u12 - u18 + 19 x12_18 <= 18
u12 - u19 + 19 x12_19 <= 18

u13 - u2 + 19 x13_2 <= 18
u13 - u3 + 19 x13_3 <= 18
u13 - u4 + 19 x13_4 <= 18
u13 - u5 + 19 x13_5 <= 18
u13 - u6 + 19 x13_6 <= 18
u13 - u7 + 19 x13_7 <= 18
u13 - u8 + 19 x13_8 <= 18
u13 - u9 + 19 x13_9 <= 18
u13 - u10 + 19 x13_10 <= 18
u13 - u11 + 19 x13_11 <= 18
u13 - u12 + 19 x13_12 <= 18
u13 - u14 + 19 x13_14 <= 18
u13 - u15 + 19 x13_15 <= 18
u13 - u16 + 19 x13_16 <= 18
u13 - u17 + 19 x13_17 <= 18
u13 - u18 + 19 x13_18 <= 18
u13 - u19 + 19 x13_19 <= 18

u14 - u2 + 19 x14_2 <= 18
u14 - u3 + 19 x14_3 <= 18
u14 - u4 + 19 x14_4 <= 18
u14 - u5 + 19 x14_5 <= 18
u14 - u6 + 19 x14_6 <= 18
u14 - u7 + 19 x14_7 <= 18
u14 - u8 + 19 x14_8 <= 18
u14 - u9 + 19 x14_9 <= 18
u14 - u10 + 19 x14_10 <= 18
u14 - u11 + 19 x14_11 <= 18
u14 - u12 + 19 x14_12 <= 18
u14 - u13 + 19 x14_13 <= 18
u14 - u15 + 19 x14_15 <= 18
u14 - u16 + 19 x14_16 <= 18
u14 - u17 + 19 x14_17 <= 18
u14 - u18 + 19 x14_18 <= 18
u14 - u19 + 19 x14_19 <= 18

u15 - u2 + 19 x15_2 <= 18
u15 - u3 + 19 x15_3 <= 18
u15 - u4 + 19 x15_4 <= 18
u15 - u5 + 19 x15_5 <= 18
u15 - u6 + 19 x15_6 <= 18
u15 - u7 + 19 x15_7 <= 18
u15 - u8 + 19 x15_8 <= 18
u15 - u9 + 19 x15_9 <= 18
u15 - u10 + 19 x15_10 <= 18
u15 - u11 + 19 x15_11 <= 18
u15 - u12 + 19 x15_12 <= 18
u15 - u13 + 19 x15_13 <= 18
u15 - u14 + 19 x15_14 <= 18
u15 - u16 + 19 x15_16 <= 18
u15 - u17 + 19 x15_17 <= 18
u15 - u18 + 19 x15_18 <= 18
u15 - u19 + 19 x15_19 <= 18

u16 - u2 + 19 x16_2 <= 18
u16 - u3 + 19 x16_3 <= 18
u16 - u4 + 19 x16_4 <= 18
u16 - u5 + 19 x16_5 <= 18
u16 - u6 + 19 x16_6 <= 18
u16 - u7 + 19 x16_7 <= 18
u16 - u8 + 19 x16_8 <= 18
u16 - u9 + 19 x16_9 <= 18
u16 - u10 + 19 x16_10 <= 18
u16 - u11 + 19 x16_11 <= 18
u16 - u12 + 19 x16_12 <= 18
u16 - u13 + 19 x16_13 <= 18
u16 - u14 + 19 x16_14 <= 18
u16 - u15 + 19 x16_15 <= 18
u16 - u17 + 19 x16_17 <= 18
u16 - u18 + 19 x16_18 <= 18
u16 - u19 + 19 x16_19 <= 18

u17 - u2 + 19 x17_2 <= 18
u17 - u3 + 19 x17_3 <= 18
u17 - u4 + 19 x17_4 <= 18
u17 - u5 + 19 x17_5 <= 18
u17 - u6 + 19 x17_6 <= 18
u17 - u7 + 19 x17_7 <= 18
u17 - u8 + 19 x17_8 <= 18
u17 - u9 + 19 x17_9 <= 18
u17 - u10 + 19 x17_10 <= 18
u17 - u11 + 19 x17_11 <= 18
u17 - u12 + 19 x17_12 <= 18
u17 - u13 + 19 x17_13 <= 18
u17 - u14 + 19 x17_14 <= 18
u17 - u15 + 19 x17_15 <= 18
u17 - u16 + 19 x17_16 <= 18
u17 - u18 + 19 x17_18 <= 18
u17 - u19 + 19 x17_19 <= 18

u18 - u2 + 19 x18_2 <= 18
u18 - u3 + 19 x18_3 <= 18
u18 - u4 + 19 x18_4 <= 18
u18 - u5 + 19 x18_5 <= 18
u18 - u6 + 19 x18_6 <= 18
u18 - u7 + 19 x18_7 <= 18
u18 - u8 + 19 x18_8 <= 18
u18 - u9 + 19 x18_9 <= 18
u18 - u10 + 19 x18_10 <= 18
u18 - u11 + 19 x18_11 <= 18
u18 - u12 + 19 x18_12 <= 18
u18 - u13 + 19 x18_13 <= 18
u18 - u14 + 19 x18_14 <= 18
u18 - u15 + 19 x18_15 <= 18
u18 - u16 + 19 x18_16 <= 18
u18 - u17 + 19 x18_17 <= 18
u18 - u19 + 19 x18_19 <= 18

u19 - u2 + 19 x19_2 <= 18
u19 - u3 + 19 x19_3 <= 18
u19 - u4 + 19 x19_4 <= 18
u19 - u5 + 19 x19_5 <= 18
u19 - u6 + 19 x19_6 <= 18
u19 - u7 + 19 x19_7 <= 18
u19 - u8 + 19 x19_8 <= 18
u19 - u9 + 19 x19_9 <= 18
u19 - u10 + 19 x19_10 <= 18
u19 - u11 + 19 x19_11 <= 18
u19 - u12 + 19 x19_12 <= 18
u19 - u13 + 19 x19_13 <= 18
u19 - u14 + 19 x19_14 <= 18
u19 - u15 + 19 x19_15 <= 18
u19 - u16 + 19 x19_16 <= 18
u19 - u17 + 19 x19_17 <= 18
u19 - u18 + 19 x19_18 <= 18

u1 >= 1
u2 >= 1
u3 >= 1
u4 >= 1
u5 >= 1
u6 >= 1
u7 >= 1
u8 >= 1
u9 >= 1
u10 >= 1
u11 >= 1
u12 >= 1
u13 >= 1
u14 >= 1
u15 >= 1
u16 >= 1
u17 >= 1
u18 >= 1
u19 >= 1

u1 <= 19
u2 <= 19
u3 <= 19
u4 <= 19
u5 <= 19
u6 <= 19
u7 <= 19
u8 <= 19
u9 <= 19
u10 <= 19
u11 <= 19
u12 <= 19
u13 <= 19
u14 <= 19
u15 <= 19
u16 <= 19
u17 <= 19
u18 <= 19
u19 <= 19

Binaries

x1_2 x1_3 x1_4 x1_5 x1_6 x1_7 x1_8 x1_9 x1_10 x1_11 x1_12 x1_13 x1_14 x1_15 x1_16 x1_17 x1_18 x1_19
x2_1 x2_3 x2_4 x2_5 x2_6 x2_7 x2_8 x2_9 x2_10 x2_11 x2_12 x2_13 x2_14 x2_15 x2_16 x2_17 x2_18 x2_19
x3_2 x3_1 x3_4 x3_5 x3_6 x3_7 x3_8 x3_9 x3_10 x3_11 x3_12 x3_13 x3_14 x3_15 x3_16 x3_17 x3_18 x3_19
x4_2 x4_3 x4_1 x4_5 x4_6 x4_7 x4_8 x4_9 x4_10 x4_11 x4_12 x4_13 x4_14 x4_15 x4_16 x4_17 x4_18 x4_19
x5_2 x5_3 x5_4 x5_1 x5_6 x5_7 x5_8 x5_9 x5_10 x5_11 x5_12 x5_13 x5_14 x5_15 x5_16 x5_17 x5_18 x5_19
x6_2 x6_3 x6_4 x6_5 x6_1 x6_7 x6_8 x6_9 x6_10 x6_11 x6_12 x6_13 x6_14 x6_15 x6_16 x6_17 x6_18 x6_19
x7_2 x7_3 x7_4 x7_5 x7_6 x7_1 x7_8 x7_9 x7_10 x7_11 x7_12 x7_13 x7_14 x7_15 x7_16 x7_17 x7_18 x7_19
x8_2 x8_3 x8_4 x8_5 x8_6 x8_7 x8_1 x8_9 x8_10 x8_11 x8_12 x8_13 x8_14 x8_15 x8_16 x8_17 x8_18 x8_19
x9_2 x9_3 x9_4 x9_5 x9_6 x9_7 x9_8 x9_1 x9_10 x9_11 x9_12 x9_13 x9_14 x9_15 x9_16 x9_17 x9_18 x9_19
x10_2 x10_3 x10_4 x10_5 x10_6 x10_7 x10_8 x10_9 x10_1 x10_11 x10_12 x10_13 x10_14 x10_15 x10_16 x10_17 x10_18 x10_19
x11_2 x11_3 x11_4 x11_5 x11_6 x11_7 x11_8 x11_9 x11_10 x11_1 x11_12 x11_13 x11_14 x11_15 x11_16 x11_17 x11_18 x11_19
x12_2 x12_3 x12_4 x12_5 x12_6 x12_7 x12_8 x12_9 x12_10 x12_11 x12_1 x12_13 x12_14 x12_15 x12_16 x12_17 x12_18 x12_19
x13_2 x13_3 x13_4 x13_5 x13_6 x13_7 x13_8 x13_9 x13_10 x13_11 x13_12 x13_1 x13_14 x13_15 x13_16 x13_17 x13_18 x13_19
x14_2 x14_3 x14_4 x14_5 x14_6 x14_7 x14_8 x14_9 x14_10 x14_11 x14_12 x14_13 x14_1 x14_15 x14_16 x14_17 x14_18 x14_19
x15_2 x15_3 x15_4 x15_5 x15_6 x15_7 x15_8 x15_9 x15_10 x15_11 x15_12 x15_13 x15_14 x15_1 x15_16 x15_17 x15_18 x15_19
x16_2 x16_3 x16_4 x16_5 x16_6 x16_7 x16_8 x16_9 x16_10 x16_11 x16_12 x16_13 x16_14 x16_15 x16_1 x16_17 x16_18 x16_19
x17_2 x17_3 x17_4 x17_5 x17_6 x17_7 x17_8 x17_9 x17_10 x17_11 x17_12 x17_13 x17_14 x17_15 x17_16 x17_1 x17_18 x17_19

x18_2 x18_3 x18_4 x18_5 x18_6 x18_7 x18_8 x18_9 x18_10 x18_11 x18_12 x18_13 x18_14 x18_15 x18_16 x18_17 x18_1 x18_19
x19_2 x19_3 x19_4 x19_5 x19_6 x19_7 x19_8 x19_9 x19_10 x19_11 x19_12 x19_13 x19_14 x19_15 x19_16 x19_17 x19_18 x19_1

End

Příloha 5 Omezující podmínka (4.6) pro $\delta_{ij}^{c} = 0,15c_{ij}$

z + p1_2 - 1.0184 x1_2 >= 0
z + p1_3 - 0.2571 x1_3 >= 0
z + p1_4 - 1.239 x1_4 >= 0
z + p1_5 - 1.3275 x1_5 >= 0
z + p1_6 - 1.5061 x1_6 >= 0
z + p1_7 - 1.5248 x1_7 >= 0
z + p1_8 - 1.5515 x1_8 >= 0
z + p1_9 - 1.5706 x1_9 >= 0
z + p1_10 - 1.5427 x1_10 >= 0
z + p1_11 - 1.5503 x1_11 >= 0
z + p1_12 - 1.676 x1_12 >= 0
z + p1_13 - 1.7435 x1_13 >= 0
z + p1_14 - 1.7437 x1_14 >= 0
z + p1_15 - 1.5128 x1_15 >= 0
z + p1_16 - 1.4093 x1_16 >= 0
z + p1_17 - 1.2832 x1_17 >= 0
z + p1_18 - 0.9471 x1_18 >= 0
z + p1_19 - 0.022 x1_19 >= 0

z + p2_1 - 1.0179 x2_1 >= 0
z + p2_3 - 0.7775 x2_3 >= 0
z + p2_4 - 0.2519 x2_4 >= 0
z + p2_5 - 0.3405 x2_5 >= 0
z + p2_6 - 0.7342 x2_6 >= 0
z + p2_7 - 0.772 x2_7 >= 0
z + p2_8 - 0.8031 x2_8 >= 0
z + p2_9 - 0.8468 x2_9 >= 0
z + p2_10 - 0.8333 x2_10 >= 0
z + p2_11 - 0.8713 x2_11 >= 0
z + p2_12 - 0.9568 x2_12 >= 0
z + p2_13 - 1.0093 x2_13 >= 0
z + p2_14 - 1.0215 x2_14 >= 0
z + p2_15 - 0.8866 x2_15 >= 0
z + p2_16 - 0.783 x2_16 >= 0
z + p2_17 - 0.657 x2_17 >= 0
z + p2_18 - 0.3708 x2_18 >= 0
z + p2_19 - 1.0121 x2_19 >= 0

z + p3_1 - 0.2608 x3_1 >= 0
z + p3_2 - 0.7832 x3_2 >= 0
z + p3_4 - 1.0038 x3_4 >= 0
z + p3_5 - 1.0923 x3_5 >= 0
z + p3_6 - 1.2709 x3_6 >= 0

z + p3_7 - 1.2896 x3_7 >= 0
z + p3_8 - 1.3163 x3_8 >= 0
z + p3_9 - 1.3354 x3_9 >= 0
z + p3_10 - 1.3075 x3_10 >= 0
z + p3_11 - 1.315 x3_11 >= 0
z + p3_12 - 1.4408 x3_12 >= 0
z + p3_13 - 1.5083 x3_13 >= 0
z + p3_14 - 1.5085 x3_14 >= 0
z + p3_15 - 1.2776 x3_15 >= 0
z + p3_16 - 1.174 x3_16 >= 0
z + p3_17 - 1.048 x3_17 >= 0
z + p3_18 - 0.7119 x3_18 >= 0
z + p3_19 - 0.255 x3_19 >= 0

z + p4_1 - 1.2396 x4_1 >= 0
z + p4_2 - 0.251 x4_2 >= 0
z + p4_3 - 0.9992 x4_3 >= 0
z + p4_5 - 0.1316 x4_5 >= 0
z + p4_6 - 0.5253 x4_6 >= 0
z + p4_7 - 0.5632 x4_7 >= 0
z + p4_8 - 0.5943 x4_8 >= 0
z + p4_9 - 0.638 x4_9 >= 0
z + p4_10 - 0.6423 x4_10 >= 0
z + p4_11 - 0.688 x4_11 >= 0
z + p4_12 - 0.748 x4_12 >= 0
z + p4_13 - 0.8004 x4_13 >= 0
z + p4_14 - 0.8127 x4_14 >= 0
z + p4_15 - 0.7239 x4_15 >= 0
z + p4_16 - 0.6203 x4_16 >= 0
z + p4_17 - 0.4943 x4_17 >= 0
z + p4_18 - 0.5707 x4_18 >= 0
z + p4_19 - 1.2338 x4_19 >= 0

z + p5_1 - 1.3289 x5_1 >= 0
z + p5_2 - 0.3403 x5_2 >= 0
z + p5_3 - 1.0885 x5_3 >= 0
z + p5_4 - 0.1338 x5_4 >= 0
z + p5_6 - 0.4352 x5_6 >= 0
z + p5_7 - 0.473 x5_7 >= 0
z + p5_8 - 0.504 x5_8 >= 0
z + p5_9 - 0.5478 x5_9 >= 0
z + p5_10 - 0.552 x5_10 >= 0
z + p5_11 - 0.5979 x5_11 >= 0
z + p5_12 - 0.6578 x5_12 >= 0
z + p5_13 - 0.7102 x5_13 >= 0
z + p5_14 - 0.7225 x5_14 >= 0
z + p5_15 - 0.649 x5_15 >= 0
z + p5_16 - 0.5455 x5_16 >= 0
z + p5_17 - 0.4195 x5_17 >= 0
z + p5_18 - 0.4959 x5_18 >= 0
z + p5_19 - 1.3231 x5_19 >= 0

z + p6_1 - 1.504 x6_1 >= 0
z + p6_2 - 0.7398 x6_2 >= 0
z + p6_3 - 1.2635 x6_3 >= 0
z + p6_4 - 0.5332 x6_4 >= 0
z + p6_5 - 0.4395 x6_5 >= 0
z + p6_7 - 0.0593 x6_7 >= 0
z + p6_8 - 0.0903 x6_8 >= 0
z + p6_9 - 0.1341 x6_9 >= 0
z + p6_10 - 0.1462 x6_10 >= 0
z + p6_11 - 0.206 x6_11 >= 0
z + p6_12 - 0.2441 x6_12 >= 0
z + p6_13 - 0.2965 x6_13 >= 0
z + p6_14 - 0.3088 x6_14 >= 0
z + p6_15 - 0.287 x6_15 >= 0
z + p6_16 - 0.3082 x6_16 >= 0
z + p6_17 - 0.2685 x6_17 >= 0
z + p6_18 - 0.599 x6_18 >= 0
z + p6_19 - 1.4981 x6_19 >= 0

z + p7_1 - 1.5187 x7_1 >= 0
z + p7_2 - 0.7723 x7_2 >= 0
z + p7_3 - 1.2783 x7_3 >= 0
z + p7_4 - 0.5658 x7_4 >= 0
z + p7_5 - 0.4721 x7_5 >= 0
z + p7_6 - 0.0528 x7_6 >= 0
z + p7_8 - 0.0689 x7_8 >= 0
z + p7_9 - 0.1127 x7_9 >= 0
z + p7_10 - 0.1258 x7_10 >= 0
z + p7_11 - 0.2103 x7_11 >= 0
z + p7_12 - 0.2227 x7_12 >= 0
z + p7_13 - 0.2751 x7_13 >= 0
z + p7_14 - 0.2874 x7_14 >= 0
z + p7_15 - 0.2667 x7_15 >= 0
z + p7_16 - 0.3124 x7_16 >= 0
z + p7_17 - 0.2833 x7_17 >= 0
z + p7_18 - 0.6138 x7_18 >= 0
z + p7_19 - 1.5129 x7_19 >= 0

z + p8_1 - 1.548 x8_1 >= 0
z + p8_2 - 0.8145 x8_2 >= 0
z + p8_3 - 1.3075 x8_3 >= 0
z + p8_4 - 0.6079 x8_4 >= 0
z + p8_5 - 0.5142 x8_5 >= 0
z + p8_6 - 0.095 x8_6 >= 0
z + p8_7 - 0.072 x8_7 >= 0
z + p8_9 - 0.0693 x8_9 >= 0
z + p8_10 - 0.0808 x8_10 >= 0
z + p8_11 - 0.191 x8_11 >= 0
z + p8_12 - 0.1957 x8_12 >= 0
z + p8_13 - 0.2481 x8_13 >= 0
z + p8_14 - 0.2604 x8_14 >= 0
z + p8_15 - 0.2318 x8_15 >= 0
z + p8_16 - 0.2984 x8_16 >= 0
z + p8_17 - 0.3125 x8_17 >= 0
z + p8_18 - 0.643 x8_18 >= 0
z + p8_19 - 1.5422 x8_19 >= 0

z + p9_1 - 1.5709 x9_1 >= 0
z + p9_2 - 0.8554 x9_2 >= 0
z + p9_3 - 1.3305 x9_3 >= 0
z + p9_4 - 0.6489 x9_4 >= 0
z + p9_5 - 0.5551 x9_5 >= 0
z + p9_6 - 0.1359 x9_6 >= 0
z + p9_7 - 0.113 x9_7 >= 0
z + p9_8 - 0.0668 x9_8 >= 0
z + p9_10 - 0.0618 x9_10 >= 0
z + p9_11 - 0.178 x9_11 >= 0
z + p9_12 - 0.2295 x9_12 >= 0
z + p9_13 - 0.2819 x9_13 >= 0
z + p9_14 - 0.2942 x9_14 >= 0
z + p9_15 - 0.2188 x9_15 >= 0
z + p9_16 - 0.3086 x9_16 >= 0
z + p9_17 - 0.3355 x9_17 >= 0
z + p9_18 - 0.666 x9_18 >= 0
z + p9_19 - 1.5651 x9_19 >= 0

z + p10_1 - 1.5463 x10_1 >= 0
z + p10_2 - 0.8498 x10_2 >= 0
z + p10_3 - 1.3058 x10_3 >= 0
z + p10_4 - 0.6531 x10_4 >= 0
z + p10_5 - 0.5594 x10_5 >= 0
z + p10_6 - 0.1471 x10_6 >= 0
z + p10_7 - 0.1263 x10_7 >= 0
z + p10_8 - 0.0774 x10_8 >= 0
z + p10_9 - 0.0648 x10_9 >= 0
z + p10_11 - 0.1456 x10_11 >= 0
z + p10_12 - 0.2288 x10_12 >= 0
z + p10_13 - 0.2813 x10_13 >= 0
z + p10_14 - 0.2935 x10_14 >= 0
z + p10_15 - 0.1865 x10_15 >= 0
z + p10_16 - 0.2804 x10_16 >= 0
z + p10_17 - 0.3108 x10_17 >= 0
z + p10_18 - 0.6413 x10_18 >= 0
z + p10_19 - 1.5405 x10_19 >= 0

z + p11_1 - 1.5448 x11_1 >= 0
z + p11_2 - 0.8852 x11_2 >= 0
z + p11_3 - 1.3044 x11_3 >= 0
z + p11_4 - 0.6972 x11_4 >= 0
z + p11_5 - 0.6034 x11_5 >= 0
z + p11_6 - 0.2052 x11_6 >= 0
z + p11_7 - 0.2068 x11_7 >= 0
z + p11_8 - 0.1855 x11_8 >= 0
z + p11_9 - 0.1771 x11_9 >= 0
z + p11_10 - 0.144 x11_10 >= 0
z + p11_12 - 0.2676 x11_12 >= 0
z + p11_13 - 0.3465 x11_13 >= 0
z + p11_14 - 0.3467 x11_14 >= 0
z + p11_15 - 0.1331 x11_15 >= 0
z + p11_16 - 0.1875 x11_16 >= 0
z + p11_17 - 0.3008 x11_17 >= 0
z + p11_18 - 0.6399 x11_18 >= 0
z + p11_19 - 1.539 x11_19 >= 0

z + p12_1 - 1.6699 x12_1 >= 0
z + p12_2 - 0.9604 x12_2 >= 0
z + p12_3 - 1.4294 x12_3 >= 0
z + p12_4 - 0.7538 x12_4 >= 0
z + p12_5 - 0.6601 x12_5 >= 0
z + p12_6 - 0.2409 x12_6 >= 0
z + p12_7 - 0.2179 x12_7 >= 0
z + p12_8 - 0.1941 x12_8 >= 0
z + p12_9 - 0.2259 x12_9 >= 0
z + p12_10 - 0.2328 x12_10 >= 0
z + p12_11 - 0.267 x12_11 >= 0
z + p12_13 - 0.1316 x12_13 >= 0
z + p12_14 - 0.1318 x12_14 >= 0
z + p12_15 - 0.1977 x12_15 >= 0
z + p12_16 - 0.2972 x12_16 >= 0
z + p12_17 - 0.4324 x12_17 >= 0
z + p12_18 - 0.765 x12_18 >= 0
z + p12_19 - 1.664 x12_19 >= 0

z + p13_1 - 1.7348 x13_1 >= 0
z + p13_2 - 1.0117 x13_2 >= 0
z + p13_3 - 1.4943 x13_3 >= 0
z + p13_4 - 0.8052 x13_4 >= 0
z + p13_5 - 0.7114 x13_5 >= 0
z + p13_6 - 0.2922 x13_6 >= 0
z + p13_7 - 0.2693 x13_7 >= 0
z + p13_8 - 0.2454 x13_8 >= 0
z + p13_9 - 0.2773 x13_9 >= 0
z + p13_10 - 0.284 x13_10 >= 0
z + p13_11 - 0.3445 x13_11 >= 0
z + p13_12 - 0.1318 x13_12 >= 0
z + p13_14 - 0.0313 x13_14 >= 0
z + p13_15 - 0.2626 x13_15 >= 0
z + p13_16 - 0.362 x13_16 >= 0
z + p13_17 - 0.4973 x13_17 >= 0
z + p13_18 - 0.8298 x13_18 >= 0
z + p13_19 - 1.729 x13_19 >= 0

z + p14_1 - 1.7356 x14_1 >= 0
z + p14_2 - 1.025 x14_2 >= 0
z + p14_3 - 1.4951 x14_3 >= 0
z + p14_4 - 0.8184 x14_4 >= 0
z + p14_5 - 0.7247 x14_5 >= 0
z + p14_6 - 0.3054 x14_6 >= 0
z + p14_7 - 0.2825 x14_7 >= 0
z + p14_8 - 0.2586 x14_8 >= 0
z + p14_9 - 0.2905 x14_9 >= 0
z + p14_10 - 0.2973 x14_10 >= 0
z + p14_11 - 0.3453 x14_11 >= 0
z + p14_12 - 0.1326 x14_12 >= 0
z + p14_13 - 0.0308 x14_13 >= 0
z + p14_15 - 0.2634 x14_15 >= 0
z + p14_16 - 0.3628 x14_16 >= 0
z + p14_17 - 0.4981 x14_17 >= 0
z + p14_18 - 0.8307 x14_18 >= 0
z + p14_19 - 1.7298 x14_19 >= 0

z + p15_1 - 1.5058 x15_1 >= 0
z + p15_2 - 0.885 x15_2 >= 0
z + p15_3 - 1.2654 x15_3 >= 0
z + p15_4 - 0.7306 x15_4 >= 0
z + p15_5 - 0.6507 x15_5 >= 0
z + p15_6 - 0.2841 x15_6 >= 0
z + p15_7 - 0.2633 x15_7 >= 0
z + p15_8 - 0.225 x15_8 >= 0
z + p15_9 - 0.2165 x15_9 >= 0
z + p15_10 - 0.1835 x15_10 >= 0
z + p15_11 - 0.1288 x15_11 >= 0
z + p15_12 - 0.1971 x15_12 >= 0
z + p15_13 - 0.2645 x15_13 >= 0
z + p15_14 - 0.2648 x15_14 >= 0
z + p15_16 - 0.1331 x15_16 >= 0
z + p15_17 - 0.2683 x15_17 >= 0
z + p15_18 - 0.6009 x15_18 >= 0
z + p15_19 - 1.5 x15_19 >= 0

z + p16_1 - 1.4026 x16_1 >= 0
z + p16_2 - 0.7818 x16_2 >= 0
z + p16_3 - 1.1622 x16_3 >= 0
z + p16_4 - 0.6274 x16_4 >= 0
z + p16_5 - 0.5475 x16_5 >= 0
z + p16_6 - 0.2978 x16_6 >= 0
z + p16_7 - 0.2993 x16_7 >= 0
z + p16_8 - 0.289 x16_8 >= 0
z + p16_9 - 0.2969 x16_9 >= 0
z + p16_10 - 0.2698 x16_10 >= 0
z + p16_11 - 0.185 x16_11 >= 0
z + p16_12 - 0.295 x16_12 >= 0
z + p16_13 - 0.3625 x16_13 >= 0
z + p16_14 - 0.3627 x16_14 >= 0
z + p16_15 - 0.1318 x16_15 >= 0
z + p16_17 - 0.1651 x16_17 >= 0
z + p16_18 - 0.4977 x16_18 >= 0
z + p16_19 - 1.3968 x16_19 >= 0

z + p17_1 - 1.2766 x17_1 >= 0
z + p17_2 - 0.6558 x17_2 >= 0
z + p17_3 - 1.0362 x17_3 >= 0
z + p17_4 - 0.5014 x17_4 >= 0
z + p17_5 - 0.4214 x17_5 >= 0
z + p17_6 - 0.2604 x17_6 >= 0
z + p17_7 - 0.2792 x17_7 >= 0
z + p17_8 - 0.3058 x17_8 >= 0
z + p17_9 - 0.3249 x17_9 >= 0
z + p17_10 - 0.297 x17_10 >= 0
z + p17_11 - 0.2957 x17_11 >= 0
z + p17_12 - 0.4257 x17_12 >= 0
z + p17_13 - 0.4932 x17_13 >= 0
z + p17_14 - 0.4934 x17_14 >= 0
z + p17_15 - 0.2625 x17_15 >= 0
z + p17_16 - 0.1589 x17_16 >= 0
z + p17_18 - 0 x17_18 >= 0
z + p17_19 - 1.2708 x17_19 >= 0

z + p18_1 - 0.9388 x18_1 >= 0
z + p18_2 - 0.3675 x18_2 >= 0
z + p18_3 - 0.6983 x18_3 >= 0
z + p18_4 - 0.5784 x18_4 >= 0
z + p18_5 - 0.4984 x18_5 >= 0
z + p18_6 - 0.592 x18_6 >= 0
z + p18_7 - 0.6108 x18_7 >= 0
z + p18_8 - 0.6374 x18_8 >= 0
z + p18_9 - 0.6565 x18_9 >= 0
z + p18_10 - 0.6285 x18_10 >= 0
z + p18_11 - 0.6361 x18_11 >= 0
z + p18_12 - 0.7619 x18_12 >= 0
z + p18_13 - 0.8293 x18_13 >= 0
z + p18_14 - 0.8295 x18_14 >= 0
z + p18_15 - 0.5987 x18_15 >= 0
z + p18_16 - 0.4951 x18_16 >= 0
z + p18_17 - 0.3691 x18_17 >= 0
z + p18_19 - 0.933 x18_19 >= 0

z + p19_1 - 0.0234 x19_1 >= 0
z + p19_2 - 1.0139 x19_2 >= 0
z + p19_3 - 0.2527 x19_3 >= 0
z + p19_4 - 1.2345 x19_4 >= 0
z + p19_5 - 1.3231 x19_5 >= 0
z + p19_6 - 1.5016 x19_6 >= 0
z + p19_7 - 1.5204 x19_7 >= 0
z + p19_8 - 1.547 x19_8 >= 0
z + p19_9 - 1.5662 x19_9 >= 0
z + p19_10 - 1.5383 x19_10 >= 0
z + p19_11 - 1.5458 x19_11 >= 0
z + p19_12 - 1.6716 x19_12 >= 0
z + p19_13 - 1.739 x19_13 >= 0
z + p19_14 - 1.7393 x19_14 >= 0
z + p19_15 - 1.5084 x19_15 >= 0
z + p19_16 - 1.4048 x19_16 >= 0
z + p19_17 - 1.2788 x19_17 >= 0
z + p19_18 - 0.9427 x19_18 >= 0

Příloha 6 Podmínka (4.6) pro $\delta_{ij}^{c} = 0,2c_{ij}$

z + p1_2 - 0 x1_2 >= 0
z + p1_3 - 0.3428 x1_3 >= 0
z + p1_4 - 1.652 x1_4 >= 0
z + p1_5 - 1.7701 x1_5 >= 0
z + p1_6 - 2.0081 x1_6 >= 0
z + p1_7 - 2.0331 x1_7 >= 0
z + p1_8 - 2.0687 x1_8 >= 0
z + p1_9 - 2.0942 x1_9 >= 0
z + p1_10 - 2.0569 x1_10 >= 0
z + p1_11 - 2.067 x1_11 >= 0
z + p1_12 - 2.2347 x1_12 >= 0
z + p1_13 - 2.3246 x1_13 >= 0
z + p1_14 - 2.3249 x1_14 >= 0
z + p1_15 - 2.0171 x1_15 >= 0
z + p1_16 - 1.879 x1_16 >= 0
z + p1_17 - 1.7109 x1_17 >= 0
z + p1_18 - 1.2628 x1_18 >= 0
z + p1_19 - 0.0294 x1_19 >= 0

z + p2_1 - 1.3572 x2_1 >= 0
z + p2_3 - 1.0367 x2_3 >= 0
z + p2_4 - 0 x2_4 >= 0
z + p2_5 - 0.4539 x2_5 >= 0
z + p2_6 - 0.9789 x2_6 >= 0
z + p2_7 - 1.0294 x2_7 >= 0
z + p2_8 - 1.0708 x2_8 >= 0
z + p2_9 - 1.1291 x2_9 >= 0
z + p2_10 - 1.111 x2_10 >= 0
z + p2_11 - 1.1617 x2_11 >= 0
z + p2_12 - 1.2758 x2_12 >= 0
z + p2_13 - 1.3457 x2_13 >= 0
z + p2_14 - 1.3621 x2_14 >= 0
z + p2_15 - 1.1821 x2_15 >= 0
z + p2_16 - 1.044 x2_16 >= 0
z + p2_17 - 0.8759 x2_17 >= 0
z + p2_18 - 0.4943 x2_18 >= 0
z + p2_19 - 1.3495 x2_19 >= 0

z + p3_1 - 0.3477 x3_1 >= 0
z + p3_2 - 1.0442 x3_2 >= 0
z + p3_4 - 1.3384 x3_4 >= 0
z + p3_5 - 1.4564 x3_5 >= 0
z + p3_6 - 1.6945 x3_6 >= 0
z + p3_7 - 1.7195 x3_7 >= 0
z + p3_8 - 1.7551 x3_8 >= 0
z + p3_9 - 1.7806 x3_9 >= 0
z + p3_10 - 1.7433 x3_10 >= 0
z + p3_11 - 1.7534 x3_11 >= 0
z + p3_12 - 1.9211 x3_12 >= 0
z + p3_13 - 2.011 x3_13 >= 0
z + p3_14 - 2.0113 x3_14 >= 0
z + p3_15 - 1.7035 x3_15 >= 0
z + p3_16 - 1.5654 x3_16 >= 0
z + p3_17 - 1.3973 x3_17 >= 0
z + p3_18 - 0.9492 x3_18 >= 0
z + p3_19 - 0.34 x3_19 >= 0

z + p4_1 - 1.6528 x4_1 >= 0
z + p4_2 - 0.3346 x4_2 >= 0
z + p4_3 - 1.3322 x4_3 >= 0
z + p4_5 - 0 x4_5 >= 0
z + p4_6 - 0.7004 x4_6 >= 0
z + p4_7 - 0.7509 x4_7 >= 0
z + p4_8 - 0.7923 x4_8 >= 0
z + p4_9 - 0.8507 x4_9 >= 0
z + p4_10 - 0.8563 x4_10 >= 0
z + p4_11 - 0.9174 x4_11 >= 0
z + p4_12 - 0.9973 x4_12 >= 0
z + p4_13 - 1.0672 x4_13 >= 0
z + p4_14 - 1.0836 x4_14 >= 0
z + p4_15 - 0.9652 x4_15 >= 0
z + p4_16 - 0.8271 x4_16 >= 0
z + p4_17 - 0.659 x4_17 >= 0
z + p4_18 - 0.7609 x4_18 >= 0
z + p4_19 - 1.6451 x4_19 >= 0

z + p5_1 - 1.7719 x5_1 >= 0
z + p5_2 - 0.4537 x5_2 >= 0
z + p5_3 - 1.4513 x5_3 >= 0
z + p5_4 - 0.1783 x5_4 >= 0
z + p5_6 - 0.5802 x5_6 >= 0
z + p5_7 - 0.6307 x5_7 >= 0
z + p5_8 - 0.6721 x5_8 >= 0
z + p5_9 - 0.7304 x5_9 >= 0
z + p5_10 - 0.7361 x5_10 >= 0
z + p5_11 - 0.7972 x5_11 >= 0
z + p5_12 - 0.8771 x5_12 >= 0
z + p5_13 - 0.9469 x5_13 >= 0
z + p5_14 - 0.9634 x5_14 >= 0
z + p5_15 - 0.8654 x5_15 >= 0
z + p5_16 - 0.7273 x5_16 >= 0
z + p5_17 - 0.5593 x5_17 >= 0
z + p5_18 - 0.6612 x5_18 >= 0
z + p5_19 - 1.7642 x5_19 >= 0

z + p6_1 - 2.0053 x6_1 >= 0
z + p6_2 - 0.9864 x6_2 >= 0
z + p6_3 - 1.6847 x6_3 >= 0
z + p6_4 - 0.7109 x6_4 >= 0
z + p6_5 - 0.586 x6_5 >= 0
z + p6_7 - 0.0791 x6_7 >= 0
z + p6_8 - 0.1204 x6_8 >= 0
z + p6_9 - 0.1788 x6_9 >= 0
z + p6_10 - 0.1949 x6_10 >= 0
z + p6_11 - 0.2747 x6_11 >= 0
z + p6_12 - 0.3255 x6_12 >= 0
z + p6_13 - 0.3953 x6_13 >= 0
z + p6_14 - 0.4117 x6_14 >= 0
z + p6_15 - 0.3827 x6_15 >= 0

z + p6_16 - 0.4109 x6_16 >= 0
z + p6_17 - 0.3581 x6_17 >= 0
z + p6_18 - 0.7987 x6_18 >= 0
z + p6_19 - 1.9975 x6_19 >= 0

z + p7_1 - 2.0249 x7_1 >= 0
z + p7_2 - 1.0298 x7_2 >= 0
z + p7_3 - 1.7043 x7_3 >= 0
z + p7_4 - 0.7544 x7_4 >= 0
z + p7_5 - 0.6294 x7_5 >= 0
z + p7_6 - 0.0704 x7_6 >= 0
z + p7_8 - 0.0919 x7_8 >= 0
z + p7_9 - 0.1503 x7_9 >= 0
z + p7_10 - 0.1677 x7_10 >= 0
z + p7_11 - 0.2803 x7_11 >= 0
z + p7_12 - 0.2969 x7_12 >= 0
z + p7_13 - 0.3668 x7_13 >= 0
z + p7_14 - 0.3832 x7_14 >= 0
z + p7_15 - 0.3556 x7_15 >= 0
z + p7_16 - 0.4165 x7_16 >= 0
z + p7_17 - 0.3777 x7_17 >= 0
z + p7_18 - 0.8184 x7_18 >= 0
z + p7_19 - 2.0172 x7_19 >= 0

z + p8_1 - 2.0639 x8_1 >= 0
z + p8_2 - 1.086 x8_2 >= 0
z + p8_3 - 1.7434 x8_3 >= 0
z + p8_4 - 0.8106 x8_4 >= 0
z + p8_5 - 0.6856 x8_5 >= 0
z + p8_6 - 0.1266 x8_6 >= 0
z + p8_7 - 0.096 x8_7 >= 0
z + p8_9 - 0.0923 x8_9 >= 0
z + p8_10 - 0.1077 x8_10 >= 0
z + p8_11 - 0.2546 x8_11 >= 0
z + p8_12 - 0.2609 x8_12 >= 0
z + p8_13 - 0.3308 x8_13 >= 0
z + p8_14 - 0.3472 x8_14 >= 0
z + p8_15 - 0.3091 x8_15 >= 0
z + p8_16 - 0.3978 x8_16 >= 0
z + p8_17 - 0.4167 x8_17 >= 0
z + p8_18 - 0.8574 x8_18 >= 0
z + p8_19 - 2.0562 x8_19 >= 0

z + p9_1 - 2.0946 x9_1 >= 0
z + p9_2 - 1.1406 x9_2 >= 0
z + p9_3 - 1.774 x9_3 >= 0
z + p9_4 - 0.8652 x9_4 >= 0
z + p9_5 - 0.7402 x9_5 >= 0
z + p9_6 - 0.1812 x9_6 >= 0
z + p9_7 - 0.1506 x9_7 >= 0
z + p9_8 - 0.0891 x9_8 >= 0
z + p9_10 - 0.0823 x9_10 >= 0
z + p9_11 - 0.2373 x9_11 >= 0
z + p9_12 - 0.306 x9_12 >= 0
z + p9_13 - 0.3759 x9_13 >= 0
z + p9_14 - 0.3923 x9_14 >= 0
z + p9_15 - 0.2918 x9_15 >= 0

z + p9_16 - 0.4115 x9_16 >= 0
z + p9_17 - 0.4473 x9_17 >= 0
z + p9_18 - 0.888 x9_18 >= 0
z + p9_19 - 2.0868 x9_19 >= 0

z + p10_1 - 2.0617 x10_1 >= 0
z + p10_2 - 1.1331 x10_2 >= 0
z + p10_3 - 1.7411 x10_3 >= 0
z + p10_4 - 0.8708 x10_4 >= 0
z + p10_5 - 0.7459 x10_5 >= 0
z + p10_6 - 0.1961 x10_6 >= 0
z + p10_7 - 0.1683 x10_7 >= 0
z + p10_8 - 0.1032 x10_8 >= 0
z + p10_9 - 0.0864 x10_9 >= 0
z + p10_11 - 0.1942 x10_11 >= 0
z + p10_12 - 0.3051 x10_12 >= 0
z + p10_13 - 0.375 x10_13 >= 0
z + p10_14 - 0.3914 x10_14 >= 0
z + p10_15 - 0.2486 x10_15 >= 0
z + p10_16 - 0.3738 x10_16 >= 0
z + p10_17 - 0.4144 x10_17 >= 0
z + p10_18 - 0.8551 x10_18 >= 0
z + p10_19 - 2.0539 x10_19 >= 0

z + p11_1 - 2.0598 x11_1 >= 0
z + p11_2 - 1.1803 x11_2 >= 0
z + p11_3 - 1.7392 x11_3 >= 0
z + p11_4 - 0.9296 x11_4 >= 0
z + p11_5 - 0.8046 x11_5 >= 0
z + p11_6 - 0.2736 x11_6 >= 0
z + p11_7 - 0.2757 x11_7 >= 0
z + p11_8 - 0.2474 x11_8 >= 0
z + p11_9 - 0.2361 x11_9 >= 0
z + p11_10 - 0.1919 x11_10 >= 0
z + p11_12 - 0.3568 x11_12 >= 0
z + p11_13 - 0.4619 x11_13 >= 0
z + p11_14 - 0.4622 x11_14 >= 0
z + p11_15 - 0.1775 x11_15 >= 0
z + p11_16 - 0.25 x11_16 >= 0
z + p11_17 - 0.4011 x11_17 >= 0
z + p11_18 - 0.8532 x11_18 >= 0
z + p11_19 - 2.052 x11_19 >= 0

z + p12_1 - 2.2265 x12_1 >= 0
z + p12_2 - 1.2805 x12_2 >= 0
z + p12_3 - 1.9059 x12_3 >= 0
z + p12_4 - 1.0051 x12_4 >= 0
z + p12_5 - 0.8802 x12_5 >= 0
z + p12_6 - 0.3212 x12_6 >= 0
z + p12_7 - 0.2906 x12_7 >= 0
z + p12_8 - 0.2588 x12_8 >= 0
z + p12_9 - 0.3012 x12_9 >= 0
z + p12_10 - 0.3103 x12_10 >= 0
z + p12_11 - 0.356 x12_11 >= 0
z + p12_13 - 0.1754 x12_13 >= 0
z + p12_14 - 0.1757 x12_14 >= 0
z + p12_15 - 0.2636 x12_15 >= 0

z + p12_16 - 0.3962 x12_16 >= 0
z + p12_17 - 0.5766 x12_17 >= 0
z + p12_18 - 1.0199 x12_18 >= 0
z + p12_19 - 2.2187 x12_19 >= 0

z + p13_1 - 2.313 x13_1 >= 0
z + p13_2 - 1.3489 x13_2 >= 0
z + p13_3 - 1.9924 x13_3 >= 0
z + p13_4 - 1.0736 x13_4 >= 0
z + p13_5 - 0.9486 x13_5 >= 0
z + p13_6 - 0.3896 x13_6 >= 0
z + p13_7 - 0.359 x13_7 >= 0
z + p13_8 - 0.3272 x13_8 >= 0
z + p13_9 - 0.3697 x13_9 >= 0
z + p13_10 - 0.3787 x13_10 >= 0
z + p13_11 - 0.4594 x13_11 >= 0
z + p13_12 - 0.1757 x13_12 >= 0
z + p13_14 - 0.0417 x13_14 >= 0
z + p13_15 - 0.3501 x13_15 >= 0
z + p13_16 - 0.4827 x13_16 >= 0
z + p13_17 - 0.6631 x13_17 >= 0
z + p13_18 - 1.1064 x13_18 >= 0
z + p13_19 - 2.3053 x13_19 >= 0

z + p14_1 - 2.3141 x14_1 >= 0
z + p14_2 - 1.3666 x14_2 >= 0
z + p14_3 - 1.9935 x14_3 >= 0
z + p14_4 - 1.0912 x14_4 >= 0
z + p14_5 - 0.9662 x14_5 >= 0
z + p14_6 - 0.4072 x14_6 >= 0
z + p14_7 - 0.3766 x14_7 >= 0
z + p14_8 - 0.3448 x14_8 >= 0
z + p14_9 - 0.3873 x14_9 >= 0
z + p14_10 - 0.3964 x14_10 >= 0
z + p14_11 - 0.4604 x14_11 >= 0
z + p14_12 - 0.1768 x14_12 >= 0
z + p14_13 - 0.0411 x14_13 >= 0
z + p14_15 - 0.3512 x14_15 >= 0
z + p14_16 - 0.4838 x14_16 >= 0
z + p14_17 - 0.6641 x14_17 >= 0
z + p14_18 - 1.1076 x14_18 >= 0
z + p14_19 - 2.3063 x14_19 >= 0

z + p15_1 - 2.0078 x15_1 >= 0
z + p15_2 - 1.1801 x15_2 >= 0
z + p15_3 - 1.6872 x15_3 >= 0
z + p15_4 - 0.9741 x15_4 >= 0
z + p15_5 - 0.8676 x15_5 >= 0
z + p15_6 - 0.3788 x15_6 >= 0
z + p15_7 - 0.3511 x15_7 >= 0
z + p15_8 - 0.3001 x15_8 >= 0
z + p15_9 - 0.2887 x15_9 >= 0
z + p15_10 - 0.2446 x15_10 >= 0
z + p15_11 - 0.1717 x15_11 >= 0
z + p15_12 - 0.2628 x15_12 >= 0
z + p15_13 - 0.3527 x15_13 >= 0
z + p15_14 - 0.353 x15_14 >= 0
z + p15_16 - 0.1774 x15_16 >= 0
z + p15_17 - 0.3578 x15_17 >= 0
z + p15_18 - 0.8012 x15_18 >= 0
z + p15_19 - 2 x15_19 >= 0

z + p16_1 - 1.8702 x16_1 >= 0
z + p16_2 - 1.0424 x16_2 >= 0
z + p16_3 - 1.5496 x16_3 >= 0
z + p16_4 - 0.8365 x16_4 >= 0
z + p16_5 - 0.7299 x16_5 >= 0
z + p16_6 - 0.397 x16_6 >= 0
z + p16_7 - 0.3991 x16_7 >= 0
z + p16_8 - 0.3854 x16_8 >= 0
z + p16_9 - 0.3959 x16_9 >= 0
z + p16_10 - 0.3597 x16_10 >= 0
z + p16_11 - 0.2466 x16_11 >= 0
z + p16_12 - 0.3933 x16_12 >= 0
z + p16_13 - 0.4833 x16_13 >= 0
z + p16_14 - 0.4836 x16_14 >= 0
z + p16_15 - 0.1757 x16_15 >= 0
z + p16_17 - 0.2202 x16_17 >= 0
z + p16_18 - 0.6636 x16_18 >= 0
z + p16_19 - 1.8624 x16_19 >= 0

z + p17_1 - 1.7021 x17_1 >= 0
z + p17_2 - 0.8744 x17_2 >= 0
z + p17_3 - 1.3816 x17_3 >= 0
z + p17_4 - 0.6685 x17_4 >= 0
z + p17_5 - 0.5619 x17_5 >= 0
z + p17_6 - 0.3472 x17_6 >= 0
z + p17_7 - 0.3722 x17_7 >= 0
z + p17_8 - 0.4077 x17_8 >= 0
z + p17_9 - 0.4332 x17_9 >= 0
z + p17_10 - 0.3959 x17_10 >= 0
z + p17_11 - 0.3943 x17_11 >= 0
z + p17_12 - 0.5676 x17_12 >= 0
z + p17_13 - 0.6576 x17_13 >= 0
z + p17_14 - 0.6578 x17_14 >= 0
z + p17_15 - 0.35 x17_15 >= 0
z + p17_16 - 0.2119 x17_16 >= 0
z + p17_18 - 0 x17_18 >= 0
z + p17_19 - 1.6944 x17_19 >= 0

z + p18_1 - 1.2517 x18_1 >= 0
z + p18_2 - 0.4901 x18_2 >= 0
z + p18_3 - 0.9311 x18_3 >= 0
z + p18_4 - 0.7712 x18_4 >= 0
z + p18_5 - 0.6646 x18_5 >= 0
z + p18_6 - 0.7893 x18_6 >= 0
z + p18_7 - 0.8143 x18_7 >= 0
z + p18_8 - 0.8498 x18_8 >= 0
z + p18_9 - 0.8753 x18_9 >= 0
z + p18_10 - 0.8381 x18_10 >= 0
z + p18_11 - 0.8482 x18_11 >= 0
z + p18_12 - 1.0158 x18_12 >= 0
z + p18_13 - 1.1058 x18_13 >= 0
z + p18_14 - 1.1061 x18_14 >= 0

z + p18_15 - 0.7983 x18_15 >= 0
z + p18_16 - 0.6602 x18_16 >= 0
z + p18_17 - 0.4921 x18_17 >= 0
z + p18_19 - 1.2439 x18_19 >= 0

z + p19_1 - 0.0312 x19_1 >= 0
z + p19_2 - 1.3519 x19_2 >= 0
z + p19_3 - 0.3369 x19_3 >= 0
z + p19_4 - 1.6461 x19_4 >= 0
z + p19_5 - 1.7642 x19_5 >= 0
z + p19_6 - 2.0022 x19_6 >= 0
z + p19_7 - 2.0272 x19_7 >= 0
z + p19_8 - 2.0627 x19_8 >= 0
z + p19_9 - 2.0882 x19_9 >= 0
z + p19_10 - 2.051 x19_10 >= 0
z + p19_11 - 2.0611 x19_11 >= 0
z + p19_12 - 2.2288 x19_12 >= 0
z + p19_13 - 2.3187 x19_13 >= 0
z + p19_14 - 2.319 x19_14 >= 0
z + p19_15 - 2.0112 x19_15 >= 0
z + p19_16 - 1.8731 x19_16 >= 0
z + p19_17 - 1.705 x19_17 >= 0
z + p19_18 - 1.2569 x19_18 >= 0

Příloha 7 Výstup z Gurobi pro $\delta_{ij}^{c} = 10\%$ $\Gamma = 18$

```
Academic license - for non-commercial use only
Set parameter timelimit to value 50

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/10/spravny/10_18.lp
Reading time = 0.01 seconds
: 1086 rows, 705 columns, 3749 nonzeros
Optimize a model with 1086 rows, 705 columns and 3749 nonzeros
Variable types: 363 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [1e-02, 2e+01]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Presolve removed 382 rows and 2 columns
Presolve time: 0.03s
Presolved: 704 rows, 703 columns, 5333 nonzeros
Variable types: 361 continuous, 342 integer (342 binary)

Root relaxation: objective 2.008097e+01, 98 iterations, 0.02 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0   20.08097    0   33          -   20.08097      -     -    0s
H    0     0                       65.8786000   20.08097  69.5%     -    0s
H    0     0                       61.2454000   20.08097  67.2%     -    0s
H    0     0                       46.4069000   20.08097  56.7%     -    0s
     0     0   21.97540    0   40   46.40690   21.97540  52.6%     -    0s
     0     0   22.38796    0   48   46.40690   22.38796  51.8%     -    0s
     0     0   22.41170    0   46   46.40690   22.41170  51.7%     -    0s
     0     0   22.44715    0   46   46.40690   22.44715  51.6%     -    0s
     0     0   22.44722    0   46   46.40690   22.44722  51.6%     -    0s
     0     0   22.48607    0   48   46.40690   22.48607  51.5%     -    0s
     0     0   22.48609    0   48   46.40690   22.48609  51.5%     -    0s
     0     0   22.50298    0   49   46.40690   22.50298  51.5%     -    0s
     0     0   22.50479    0   49   46.40690   22.50479  51.5%     -    0s
     0     0   22.52328    0   49   46.40690   22.52328  51.5%     -    0s
     0     0   22.52430    0   48   46.40690   22.52430  51.5%     -    0s
     0     0   22.55729    0   48   46.40690   22.55729  51.4%     -    0s
     0     0   22.55736    0   48   46.40690   22.55736  51.4%     -    0s
     0     0   22.56828    0   48   46.40690   22.56828  51.4%     -    0s
     0     0   22.56828    0   48   46.40690   22.56828  51.4%     -    0s
     0     0   22.58962    0   49   46.40690   22.58962  51.3%     -    0s
     0     0   22.59395    0   48   46.40690   22.59395  51.3%     -    0s
     0     0   22.59402    0   48   46.40690   22.59402  51.3%     -    0s
     0     0   22.59898    0   48   46.40690   22.59898  51.3%     -    0s
     0     0   22.59924    0   49   46.40690   22.59924  51.3%     -    0s
     0     0   22.63539    0   51   46.40690   22.63539  51.2%     -    0s
     0     0   22.63588    0   51   46.40690   22.63588  51.2%     -    0s
     0     0   22.64385    0   51   46.40690   22.64385  51.2%     -    0s
     0     0   22.64425    0   50   46.40690   22.64425  51.2%     -    0s
```

```
   0    0  22.64459   0  51  46.40690  22.64459  51.2%    -   0s
   0    0  22.64528   0  50  46.40690  22.64528  51.2%    -   0s
   0    0  22.64605   0  50  46.40690  22.64605  51.2%    -   0s
   0    0  22.64605   0  50  46.40690  22.64605  51.2%    -   0s
   0    2  22.64605   0  50  46.40690  22.64605  51.2%    -   0s
H  183   178              44.7819000  22.88462  48.9%  11.8   0s
*  230   214         111    43.3563000  22.98361  47.0%  12.2   0s
H  443   362              43.2584000  23.09593  46.6%  13.1   0s
H  595   509              41.8416000  23.19167  44.6%  12.9   0s
H  880   676              40.3632000  24.78713  38.6%  17.4   2s
* 1106   766          61    36.6384000  24.78713  32.3%  18.8   2s
H 1556   900              36.6296000  24.80454  32.3%  19.9   3s
* 2145   960          41    33.8895000  25.09677  25.9%  19.7   3s
* 2305   945          56    33.5122000  25.10346  25.1%  19.7   3s
* 2306   945          56    33.5113000  25.10346  25.1%  19.7   3s
 5098  2801   27.02475   44   15   33.51130   25.36797  24.3%  17.1    5s
* 6826  3877          35     33.5023000   25.46374  24.0%  16.2    6s
* 6828  3875          35     33.4904000   25.46374  24.0%  16.2    6s
 10573  6111   30.50990   40   37   33.49040   25.56472  23.7%  16.0   10s
 19649  8604   26.41780   51   12   33.49040   25.61103  23.5%  15.4   15s
H19653  8300              33.4904000   25.61103  23.5%  15.4   15s
 21024  8663   27.26769   56   69   33.49040   25.63265  23.5%  15.4   20s
 21065  8690   30.22410   63   81   33.49040   25.63265  23.5%  15.3   25s
 21111  8721   26.02958   49   65   33.49040   25.63265  23.5%  15.3   30s
 21148  8746   31.30707   63   67   33.49040   25.63265  23.5%  15.3   35s
 21200  8783   25.63265   42   51   33.49040   25.63265  23.5%  15.7   40s
 24374 10156   28.28729   76   21   33.49040   25.63265  23.5%  16.5   45s
 30319 12693   28.68016   81   34   33.49040   25.68280  23.3%  16.9   50s

Cutting planes:
  Learned: 2
  Gomory: 22
  Cover: 3
  Implied bound: 10
  Clique: 1
  MIR: 160
  StrongCG: 1
  Flow cover: 401
  Inf proof: 2
  Zero half: 1

Explored 30661 nodes (521846 simplex iterations) in 50.01 seconds
Thread count was 4 (of 4 available processors)

Solution count 10: 33.4904 33.4904 33.5023 ... 41.8416

Time limit reached
Best objective 3.349039997251e+01, best bound 2.568280397199e+01, gap 23.3129%

Wrote result file '/Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/10/spravny/1018.sol'
```

Příloha 8 Výstup z Gurobi pro $\delta_{ij}^{c} = 10\%\ \Gamma = 3$

```
Academic license - for non-commercial use only
Set parameter timelimit to value 50

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/10/spravny/10_18.lp
Reading time = 0.01 seconds
: 1086 rows, 705 columns, 3749 nonzeros
Optimize a model with 1086 rows, 705 columns and 3749 nonzeros
Variable types: 363 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [1e-02, 4e+01]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Presolve removed 382 rows and 2 columns
Presolve time: 0.04s
Presolved: 704 rows, 703 columns, 5333 nonzeros
Variable types: 361 continuous, 342 integer (342 binary)

Root relaxation: objective 2.033062e+01, 131 iterations, 0.02 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0   20.33062    0   44          -   20.33062      -     -    0s
H    0     0                      67.1528000   20.33062  69.7%     -    0s
H    0     0                      50.1578000   20.33062  59.5%     -    0s
     0     0   22.58469    0   46   50.15780   22.58469  55.0%     -    0s
     0     0   22.58469    0   46   50.15780   22.58469  55.0%     -    0s
     0     0   22.94259    0   57   50.15780   22.94259  54.3%     -    0s
     0     0   23.00404    0   54   50.15780   23.00404  54.1%     -    0s
     0     0   23.00404    0   54   50.15780   23.00404  54.1%     -    0s
     0     0   23.00843    0   54   50.15780   23.00843  54.1%     -    0s
     0     0   23.00863    0   54   50.15780   23.00863  54.1%     -    0s
     0     0   23.01324    0   53   50.15780   23.01324  54.1%     -    0s
     0     0   23.01324    0   53   50.15780   23.01324  54.1%     -    0s
     0     0   23.01497    0   52   50.15780   23.01497  54.1%     -    0s
     0     0   23.01497    0   52   50.15780   23.01497  54.1%     -    0s
     0     2   23.01497    0   52   50.15780   23.01497  54.1%     -    0s
H   88    85                      44.0594000   23.31950  47.1%  20.1    0s
H  111   103                      40.3887000   23.31950  42.3%  19.1    0s
*  119   103              57      39.8299000   23.31950  41.5%  18.6    0s
*  128   101              61      33.6171000   23.32140  30.6%  18.8    0s
H  882   619                      33.5896000   24.02389  28.5%  22.5    2s
H 1478   836                      33.5060000   25.11749  25.0%  22.8    2s
 5671  3404   28.86307   40   15   33.50600   25.63532  23.5%  18.4    5s
 18369 10239   30.74118   43    8   33.50600   25.94520  22.6%  15.3   10s
 20711 11490   26.35282   30   52   33.50600   25.97869  22.5%  15.1   16s
 20754 11519   27.04685   31   76   33.50600   25.97869  22.5%  15.0   20s
 21516 11817   26.75355   39   26   33.50600   25.97869  22.5%  15.4   25s
 30347 15191   26.41254   39    6   33.50600   25.97869  22.5%  14.9   30s
 41645 18776   26.83337   37   13   33.50600   25.97869  22.5%  14.5   35s
```

```
H48566 20280                33.5059999   25.97869  22.5%  14.4   38s
 52661 21411   26.87660   55   25   33.50600   26.00676  22.4%  14.3   40s
 62915 27365   29.24033   49    9   33.50600   26.07986  22.2%  14.3   45s
 73297 33342   28.78665   51   12   33.50600   26.14092  22.0%  14.1   50s

Cutting planes:
  Learned: 5
  Gomory: 27
  Implied bound: 9
  Clique: 2
  MIR: 22
  Flow cover: 221
  Inf proof: 16
  Zero half: 8

Explored 73718 nodes (1041974 simplex iterations) in 50.01 seconds
Thread count was 4 (of 4 available processors)

Solution count 9: 33.506 33.506 33.5896 ... 67.1528

Time limit reached
Best objective 3.350599990275e+01, best bound 2.614302665259e+01, gap 21.9751%
```

Příloha 9 Výstup z Gurobi pro $\delta_{ij}^{c} = 10\%$ Γ=56

```
Academic license - for non-commercial use only
Set parameter timelimit to value 50

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/10/spravny/10_18.lp
Reading time = 0.02 seconds
: 1086 rows, 705 columns, 3749 nonzeros
Optimize a model with 1086 rows, 705 columns and 3749 nonzeros
Variable types: 363 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [1e-02, 5e+01]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Presolve removed 382 rows and 2 columns
Presolve time: 0.05s
Presolved: 704 rows, 703 columns, 5333 nonzeros
Variable types: 361 continuous, 342 integer (342 binary)

Root relaxation: objective 2.035560e+01, 123 iterations, 0.02 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

    0     0   20.35560    0   30          -   20.35560      -     -    0s
H   0     0                       70.6310000   20.35560  71.2%     -    0s
H   0     0                       51.2314000   20.35560  60.3%     -    0s
    0     0   22.65815    0   24   51.23140   22.65815  55.8%     -    0s
    0     0   22.66689    0   24   51.23140   22.66689  55.8%     -    0s
```

```
   0    0  22.96542   0  30   51.23140   22.96542  55.2%    -   0s
H   0    0             48.0403000   22.96542  52.2%    -   0s
   0    0  23.12235   0  22   48.04030   23.12235  51.9%    -   0s
   0    0  23.12429   0  27   48.04030   23.12429  51.9%    -   0s
   0    0  23.12530   0  22   48.04030   23.12530  51.9%    -   0s
   0    0  23.12530   0  22   48.04030   23.12530  51.9%    -   0s
   0    0  23.12695   0  27   48.04030   23.12695  51.9%    -   0s
   0    0  23.12695   0  27   48.04030   23.12695  51.9%    -   0s
   0    0  23.12695   0  27   48.04030   23.12695  51.9%    -   0s
   0    0  23.12695   0  27   48.04030   23.12695  51.9%    -   0s
   0    2  23.12695   0  27   48.04030   23.12695  51.9%    -   0s
H  69   68              45.5994000   23.34859  48.8%  23.5   0s
H 345  218              33.6171000   23.75940  29.3%  17.6   0s
H 802  478              33.5896000   25.33876  24.6%  17.6   2s
* 1727  819          58    33.5511000   25.51497  24.0%  17.3   3s
* 1728  795          57    33.5401000   25.51497  23.9%  17.3   3s
H 2219 1066             33.5400999   25.53760  23.9%  16.0   3s
 5017 3024  32.93107  40    6   33.54010   25.66931  23.5%  14.0   5s
H12330 7947              33.5400999   25.82396  23.0%  12.6   7s
H16176 10461              33.5400997   25.88461  22.8%  12.4   9s
 17903 11480  33.46379   41   15   33.54010   25.90355  22.8%  12.4  10s
H21021 13490              33.5269000   25.94426  22.6%  12.3  16s
 21054 13512  26.16853   33   75   33.52690   25.94426  22.6%  12.3  20s
 26574 15791  26.21127   38   25   33.52690   25.94426  22.6%  12.5  25s
 38122 19814    cutoff   62        33.52690   25.94426  22.6%  12.7  30s
 50027 22956 infeasible   60        33.52690   26.02190  22.4%  12.9  35s
 62498 26102   28.04938   45    9   33.52690   26.12216  22.1%  13.1  40s
 75646 33415   30.68677   59   12   33.52690   26.19916  21.9%  13.1  45s
 89874 41161 infeasible   46        33.52690   26.27715  21.6%  13.2  50s

Cutting planes:
  Learned: 6
  Gomory: 14
  Cover: 25
  Implied bound: 10
  Clique: 4
  MIR: 12
  Flow cover: 243
  Inf proof: 8
  Zero half: 6

Explored 90219 nodes (1190105 simplex iterations) in 50.00 seconds
Thread count was 4 (of 4 available processors)

Solution count 10: 33.5269 33.5401 33.5401 ... 48.0403

Time limit reached
Best objective 3.352690000000e+01, best bound 2.627845033581e+01, gap 21.6198%
```

Příloha 10 Výstup z Gurobi $\delta_{ij}^{c} = 15\%$ Γz = 18

```
Academic license - for non-commercial use only
Set parameter timelimit to value 40

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/15/10_18.lp
Reading time = 0.01 seconds
: 1086 rows, 705 columns, 3748 nonzeros
Optimize a model with 1086 rows, 705 columns and 3748 nonzeros
Variable types: 363 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [2e-02, 2e+01]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Presolve removed 383 rows and 3 columns
Presolve time: 0.01s
Presolved: 703 rows, 702 columns, 5330 nonzeros
Variable types: 360 continuous, 342 integer (342 binary)

Root relaxation: objective 2.085546e+01, 100 iterations, 0.00 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0   20.85546    0   33          -   20.85546      -     -    0s
H    0     0                       68.8663000   20.85546  69.7%     -    0s
H    0     0                       64.0220000   20.85546  67.4%     -    0s
H    0     0                       53.0224000   20.85546  60.7%     -    0s
     0     0   22.53465    0   43   53.02240   22.53465  57.5%     -    0s
H    0     0                       49.6135000   22.53465  54.6%     -    0s
     0     0   22.53800    0   40   49.61350   22.53800  54.6%     -    0s
     0     0   22.89314    0   45   49.61350   22.89314  53.9%     -    0s
     0     0   22.90890    0   43   49.61350   22.90890  53.8%     -    0s
     0     0   22.96181    0   43   49.61350   22.96181  53.7%     -    0s
H    0     0                       41.6663000   22.96181  44.9%     -    0s
     0     0   22.96471    0   40   41.66630   22.96471  44.9%     -    0s
     0     0   22.99937    0   44   41.66630   22.99937  44.8%     -    0s
     0     0   23.05990    0   44   41.66630   23.05990  44.7%     -    0s
     0     0   23.05990    0   44   41.66630   23.05990  44.7%     -    0s
     0     0   23.09556    0   44   41.66630   23.09556  44.6%     -    0s
     0     0   23.09556    0   44   41.66630   23.09556  44.6%     -    0s
     0     0   23.12684    0   44   41.66630   23.12684  44.5%     -    0s
     0     0   23.17263    0   42   41.66630   23.17263  44.4%     -    0s
     0     0   23.17263    0   42   41.66630   23.17263  44.4%     -    0s
     0     0   23.17263    0   42   41.66630   23.17263  44.4%     -    0s
     0     0   23.17263    0   42   41.66630   23.17263  44.4%     -    0s
     0     2   23.17263    0   42   41.66630   23.17263  44.4%     -    0s
H  211   181                       41.5946259   23.69600  43.0%  14.3    0s
H  220   187                       41.5792000   23.69600  43.0%  14.2    0s
*  517   416              58      39.7289000   23.86975  39.9%  14.1    0s
*  519   416              57      39.7275000   23.86975  39.9%  14.1    0s
H  747   569                       38.9409000   24.01974  38.3%  14.3    0s
* 1085   721              49      37.2412000   24.01974  35.5%  17.7    2s
```

```
H 1402   796               37.1320000   24.15192  35.0%  18.3    2s
H 1447   794               37.0286000   24.15192  34.8%  18.5    2s
* 4017  2161          38     35.7981000   25.83043  27.8%  17.6    4s
* 4018  2161          38     35.7967000   25.83043  27.8%  17.6    4s
H 4446  2299               35.1874000   25.90542  26.4%  17.5    4s
 5011  2705   27.50979   41    9   35.18740   26.04586  26.0%  17.3    5s
H 5049  2664               35.0411000   26.04586  25.7%  17.3    5s
 15415  8703   26.91629   25   18   35.04110   26.55258  24.2%  14.6   10s
 20771 11646   29.63551   32   42   35.04110   26.64537  24.0%  14.2   18s
 20803 11667   28.09187   34   72   35.04110   26.64537  24.0%  14.2   20s
 20854 11701   27.53405   27   77   35.04110   26.64537  24.0%  14.1   25s
 20898 11731   27.80601   35   80   35.04110   26.64537  24.0%  14.1   30s
H20914 11154               35.0275000   26.64537  23.9%  14.5   33s
 20920 11160   26.64537   26   54   35.02750   26.64537  23.9%  14.5   35s
*21249 10757          67     34.7632000   26.64537  23.4%  15.0   36s
*21250 10224          68     34.7515000   26.64537  23.3%  15.0   36s
H21582  9821               34.7445000   26.64537  23.3%  15.2   36s
*22790  9768          76     34.7444000   26.64537  23.3%  15.5   38s
 23439 10006 infeasible   67       34.74440   26.64537  23.3%  15.7   40s

Cutting planes:
  Learned: 1
  Gomory: 11
  Implied bound: 10
  Clique: 1
  MIR: 106
  Flow cover: 197
  Inf proof: 3
  Zero half: 5

Explored 23470 nodes (369741 simplex iterations) in 40.01 seconds
Thread count was 4 (of 4 available processors)

Solution count 10: 34.7444 34.7445 34.7515 ... 37.0286

Time limit reached
Best objective 3.474440000000e+01, best bound 2.664536983686e+01, gap 23.3103%
```

Příloha 11 Výstup z Gurobi pro $\delta_{ij}^{c} = 15\%,\ \Gamma = 36$

```
Academic license - for non-commercial use only
Set parameter timelimit to value 40

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust
new/15/10_18.lp
Reading time = 0.00 seconds
: 1086 rows, 705 columns, 3748 nonzeros
Optimize a model with 1086 rows, 705 columns and 3748 nonzeros
Variable types: 363 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [2e-02, 4e+01]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+01]
Presolve removed 383 rows and 3 columns
Presolve time: 0.01s
Presolved: 703 rows, 702 columns, 5330 nonzeros
Variable types: 360 continuous, 342 integer (342 binary)

Root relaxation: objective 2.122518e+01, 139 iterations, 0.00 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

    0     0   21.22518    0   46          -   21.22518      -     -    0s
H   0     0                       70.2050000   21.22518  69.8%     -    0s
H   0     0                       52.3661000   21.22518  59.5%     -    0s
    0     0   23.38362    0   52   52.36610   23.38362  55.3%     -    0s
    0     0   23.40059    0   46   52.36610   23.40059  55.3%     -    0s
    0     0   23.73216    0   49   52.36610   23.73216  54.7%     -    0s
H   0     0                       46.5425000   23.73216  49.0%     -    0s
    0     0   23.73216    0   49   46.54250   23.73216  49.0%     -    0s
    0     0   23.74616    0   46   46.54250   23.74616  49.0%     -    0s
    0     0   23.74823    0   44   46.54250   23.74823  49.0%     -    0s
H   0     0                       45.9663000   23.74823  48.3%     -    0s
    0     0   23.74823    0   44   45.96630   23.74823  48.3%     -    0s
    0     0   23.75885    0   44   45.96630   23.75885  48.3%     -    0s
    0     0   23.75885    0   44   45.96630   23.75885  48.3%     -    0s
    0     0   23.77050    0   46   45.96630   23.77050  48.3%     -    0s
    0     0   23.77050    0   46   45.96630   23.77050  48.3%     -    0s
    0     0   23.77311    0   44   45.96630   23.77311  48.3%     -    0s
    0     0   23.77370    0   46   45.96630   23.77370  48.3%     -    0s
    0     0   23.77494    0   46   45.96630   23.77494  48.3%     -    0s
    0     0   23.77534    0   44   45.96630   23.77534  48.3%     -    0s
    0     0   23.78182    0   44   45.96630   23.78182  48.3%     -    0s
    0     0   23.78182    0   44   45.96630   23.78182  48.3%     -    0s
    0     0   23.82524    0   48   45.96630   23.82524  48.2%     -    0s
    0     0   23.84132    0   53   45.96630   23.84132  48.1%     -    0s
    0     0   23.84172    0   51   45.96630   23.84172  48.1%     -    0s
    0     0   23.84285    0   51   45.96630   23.84285  48.1%     -    0s
    0     0   23.84299    0   52   45.96630   23.84299  48.1%     -    0s
    0     0   24.13049    0   55   45.96630   24.13049  47.5%     -    0s
    0     0   24.13759    0   60   45.96630   24.13759  47.5%     -    0s
```

```
   0    0  24.13759   0  60   45.96630   24.13759  47.5%    -   0s
   0    0  24.14368   0  59   45.96630   24.14368  47.5%    -   0s
   0    0  24.14430   0  58   45.96630   24.14430  47.5%    -   0s
   0    0  24.14657   0  57   45.96630   24.14657  47.5%    -   0s
   0    0  24.14657   0  57   45.96630   24.14657  47.5%    -   0s
   0    0  24.15057   0  56   45.96630   24.15057  47.5%    -   0s
   0    0  24.15057   0  56   45.96630   24.15057  47.5%    -   0s
   0    2  24.15057   0  56   45.96630   24.15057  47.5%    -   0s
H   8    9              44.8322000   25.10328  44.0%  70.6   0s
H   9   11              43.5730000   25.10328  42.4%  73.6   0s
H  100   94              40.6971000   25.10328  38.3%  29.8   0s
H  136   109             40.5313000   25.18279  37.9%  25.8   0s
H  481   398             40.4060000   25.70076  36.4%  24.9   0s
*  532   428         48    40.3236000   25.70076  36.3%  24.3   0s
*  537   427         48    40.3120000   25.70076  36.2%  24.1   0s
H  743   524             37.0518000   26.06599  29.6%  22.5   1s
 5195  3024   34.28749  45   23   37.05180   26.58551  28.2%  16.9   5s
H 5197  3025              37.0345000   26.58551  28.2%  16.9   5s
* 8562  5075          45    37.0255000   26.71738  27.8%  15.6   7s
* 8565  5073          45    37.0138000   26.71738  27.8%  15.6   7s
H12383  7072             35.4267000   26.80710  24.3%  14.8   8s
 15766  8855    cutoff   26      35.42670   26.86987  24.2%  14.3   10s
H19184 10356              34.8606000   26.92031  22.8%  14.1   11s
H19413 10435              34.7734000   26.92120  22.6%  14.0   11s
H20747 11115              34.7422000   26.94126  22.5%  13.9   18s
 20767 11129   30.56840   30   84   34.74220   26.94126  22.5%  13.9   20s
 20811 11158   32.88698   44   84   34.74220   26.94126  22.5%  13.9   25s
H20872 10638              34.6573000   26.94126  22.3%  14.2   29s
 20876 10640   26.94126   28   17   34.65730   26.94126  22.3%  14.2   30s
 29047 13199   27.47393   45   19   34.65730   26.94126  22.3%  14.7   35s
H33935 14202              34.6572999   26.94126  22.3%  14.9   37s
 39439 15957   28.96013   48    4   34.65730   26.94126  22.3%  15.0   40s

Cutting planes:
  Learned: 4
  Gomory: 27
  Implied bound: 14
  Clique: 1
  MIR: 52
  Flow cover: 186
  Inf proof: 10
  Zero half: 9

Explored 39819 nodes (595813 simplex iterations) in 40.01 seconds
Thread count was 4 (of 4 available processors)

Solution count 10: 34.6573 34.6573 34.7422 ... 37.0518

Time limit reached
Best objective 3.465729992476e+01, best bound 2.694126274510e+01, gap 22.2638%

Wrote result file '/Users/nehy/Documents/škola/DP/prakticka/TSP_robust/TSP_Robust new/15/1536.sol'
```

Příloha 12 Matematický model TSP s časovými okny

Minimize

7.468 x1_2 + 1.886 x1_3 + 9.086 x1_4 + 9.735 x1_5 + 11.045 x1_6 + 11.182 x1_7 + 11.378 x1_8 + 11.518 x1_9 + 11.313 x1_10 + 11.369 x1_11 + 12.291 x1_12 + 12.785 x1_13 + 12.787 x1_14 + 11.094 x1_15 + 10.335 x1_16 + 9.41 x1_17 + 6.946 x1_18 + 0.162 x1_19 + 7.465 x2_1 + 5.702 x2_3 + 1.847 x2_4 + 2.497 x2_5 + 5.384 x2_6 + 5.662 x2_7 + 5.889 x2_8 + 6.21 x2_9 + 6.111 x2_10 + 6.389 x2_11 + 7.017 x2_12 + 7.401 x2_13 + 7.491 x2_14 + 6.502 x2_15 + 5.742 x2_16 + 4.818 x2_17 + 2.719 x2_18 + 7.422 x2_19 + 5.743 x3_2 + 1.912 x3_1 + 7.361 x3_4 + 8.01 x3_5 + 9.32 x3_6 + 9.457 x3_7 + 9.653 x3_8 + 9.793 x3_9 + 9.588 x3_10 + 9.644 x3_11 + 10.566 x3_12 + 11.061 x3_13 + 11.062 x3_14 + 9.369 x3_15 + 8.61 x3_16 + 7.685 x3_17 + 5.221 x3_18 + 1.87 x3_19 + 1.84 x4_2 + 7.327 x4_3 + 9.09 x4_1 + 0.965 x4_5 + 3.852 x4_6 + 4.13 x4_7 + 4.358 x4_8 + 4.679 x4_9 + 4.71 x4_10 + 5.046 x4_11 + 5.485 x4_12 + 5.87 x4_13 + 5.96 x4_14 + 5.308 x4_15 + 4.549 x4_16 + 3.625 x4_17 + 4.185 x4_18 + 9.048 x4_19 + 2.495 x5_2 + 7.982 x5_3 + 0.981 x5_4 + 9.745 x5_1 + 3.191 x5_6 + 3.469 x5_7 + 3.696 x5_8 + 4.017 x5_9 + 4.048 x5_10 + 4.384 x5_11 + 4.824 x5_12 + 5.208 x5_13 + 5.299 x5_14 + 4.76 x5_15 + 4 x5_16 + 3.076 x5_17 + 3.636 x5_18 + 9.703 x5_19 + 5.425 x6_2 + 9.266 x6_3 + 3.91 x6_4 + 3.223 x6_5 + 11.029 x6_1 + 0.435 x6_7 + 0.662 x6_8 + 0.983 x6_9 + 1.072 x6_10 + 1.511 x6_11 + 1.79 x6_12 + 2.174 x6_13 + 2.264 x6_14 + 2.105 x6_15 + 2.26 x6_16 + 1.969 x6_17 + 4.393 x6_18 + 10.986 x6_19 + 5.664 x7_2 + 9.374 x7_3 + 4.149 x7_4 + 3.462 x7_5 + 0.387 x7_6 + 11.137 x7_1 + 0.505 x7_8 + 0.827 x7_9 + 0.922 x7_10 + 1.542 x7_11 + 1.633 x7_12 + 2.018 x7_13 + 2.108 x7_14 + 1.956 x7_15 + 2.291 x7_16 + 2.077 x7_17 + 4.501 x7_18 + 11.094 x7_19 + 5.973 x8_2 + 9.589 x8_3 + 4.458 x8_4 + 3.771 x8_5 + 0.696 x8_6 + 0.528 x8_7 + 11.352 x8_1 + 0.508 x8_9 + 0.592 x8_10 + 1.4 x8_11 + 1.435 x8_12 + 1.819 x8_13 + 1.91 x8_14 + 1.7 x8_15 + 2.188 x8_16 + 2.292 x8_17 + 4.716 x8_18 + 11.309 x8_19 + 6.273 x9_2 + 9.757 x9_3 + 4.758 x9_4 + 4.071 x9_5 + 0.996 x9_6 + 0.828 x9_7 + 0.49 x9_8 + 11.52 x9_1 + 0.453 x9_10 + 1.305 x9_11 + 1.683 x9_12 + 2.067 x9_13 + 2.158 x9_14 + 1.605 x9_15 + 2.263 x9_16 + 2.46 x9_17 + 4.884 x9_18 + 11.478 x9_19 + 6.232 x10_2 + 9.576 x10_3 + 4.79 x10_4 + 4.102 x10_5 + 1.079 x10_6 + 0.926 x10_7 + 0.568 x10_8 + 0.475 x10_9 + 11.339 x10_1 + 1.068 x10_11 + 1.678 x10_12 + 2.063 x10_13 + 2.153 x10_14 + 1.367 x10_15 + 2.056 x10_16 + 2.279 x10_17 + 4.703 x10_18 + 11.297 x10_19 + 6.492 x11_2 + 9.565 x11_3 + 5.113 x11_4 + 4.425 x11_5 + 1.505 x11_6 + 1.516 x11_7 + 1.361 x11_8 + 1.299 x11_9 + 1.056 x11_10 + 11.329 x11_1 + 1.962 x11_12 + 2.541 x11_13 + 2.542 x11_14 + 0.976 x11_15 + 1.375 x11_16 + 2.206 x11_17 + 4.693 x11_18 + 11.286 x11_19 + 7.043 x12_2 + 10.482 x12_3 + 5.528 x12_4 + 4.841 x12_5 + 1.766 x12_6 + 1.598 x12_7 + 1.423 x12_8 + 1.657 x12_9 + 1.707 x12_10 + 1.958 x12_11 + 12.246 x12_1 + 0.965 x12_13 + 0.966 x12_14 + 1.45 x12_15 + 2.179 x12_16 + 3.171 x12_17 + 5.61 x12_18 + 12.203 x12_19 + 7.419 x13_2 + 10.958 x13_3 + 5.905 x13_4 + 5.217 x13_5 + 2.143 x13_6 + 1.975 x13_7 + 1.8 x13_8 + 2.033 x13_9 + 2.083 x13_10 + 2.527 x13_11 + 0.966 x13_12 + 12.722 x13_1 + 0.229 x13_14 + 1.926 x13_15 + 2.655 x13_16 + 3.647 x13_17 + 6.085 x13_18 + 12.679 x13_19 + 7.516 x14_2 + 10.964 x14_3 + 6.001 x14_4 + 5.314 x14_5 + 2.24 x14_6 + 2.071 x14_7 + 1.897 x14_8 + 2.13 x14_9 + 2.18 x14_10 + 2.532 x14_11 + 0.973 x14_12 + 0.226 x14_13 + 12.728 x14_1 + 1.931 x14_15 + 2.661 x14_16 + 3.653 x14_17 + 6.092 x14_18 + 12.685 x14_19 + 6.49 x15_2 + 9.279 x15_3 + 5.358 x15_4 + 4.772 x15_5 + 2.084 x15_6 + 1.931 x15_7 + 1.65 x15_8 + 1.588 x15_9 + 1.345 x15_10 + 0.944 x15_11 + 1.445 x15_12 + 1.94 x15_13 + 1.942 x15_14 + 11.043 x15_1 + 0.976 x15_16 + 1.968 x15_17 + 4.407 x15_18 + 11 x15_19 + 5.733 x16_2 + 8.523 x16_3 + 4.601 x16_4 + 4.015 x16_5 + 2.184 x16_6 + 2.195 x16_7 + 2.12 x16_8 + 2.177 x16_9 + 1.978 x16_10 + 1.356 x16_11 + 2.163 x16_12 + 2.658 x16_13 + 2.66 x16_14 + 0.966 x16_15 + 10.286 x16_1 + 1.211 x16_17 + 3.65 x16_18 + 10.243 x16_19 + 4.809 x17_2 + 7.599 x17_3 + 3.677 x17_4 + 3.09 x17_5 + 1.909 x17_6 + 2.047 x17_7 + 2.242 x17_8 + 2.383 x17_9 + 2.178 x17_10 + 2.169 x17_11 + 3.122 x17_12 + 3.617 x17_13 + 3.618 x17_14 + 1.925 x17_15 + 1.165 x17_16 + 9.362 x17_1 + 2.726 x17_18 + 9.319 x17_19 + 2.695 x18_2 + 5.121 x18_3 + 4.241 x18_4 + 3.655 x18_5 + 4.341 x18_6 + 4.479 x18_7 + 4.674 x18_8 + 4.814 x18_9 + 4.609 x18_10 + 4.665 x18_11 + 5.587 x18_12 + 6.082 x18_13 + 6.083 x18_14 + 4.391 x18_15 + 3.631 x18_16 + 2.707 x18_17 + 6.884 x18_1 + 6.842 x18_19 + 7.435 x19_2 + 1.853 x19_3 + 9.053 x19_4 + 9.703 x19_5 + 11.012 x19_6 + 11.15 x19_7 + 11.345 x19_8 + 11.485 x19_9 + 11.281 x19_10 + 11.336 x19_11 + 12.258 x19_12 + 12.753 x19_13 + 12.755 x19_14 + 11.061 x19_15 + 10.302 x19_16 + 9.378 x19_17 + 6.913 x19_18 + 0.171 x19_1 + W1 + W2 + W3 + W4 + W5 + W6 + W7 + W8 + W9 + W10 + W11 + W12 + W13 + W14 + W15 + W16 + W17 + W18 + W19

Subject to

x1_2 + x1_3 + x1_4 + x1_5 + x1_6 + x1_7 + x1_8 + x1_9 + x1_10 + x1_11 + x1_12 + x1_13 + x1_14 + x1_15 + x1_16 + x1_17 + x1_18 + x1_19 = 1
x2_1 + x2_3 + x2_4 + x2_5 + x2_6 + x2_7 + x2_8 + x2_9 + x2_10 + x2_11 + x2_12 + x2_13 + x2_14 + x2_15 + x2_16 + x2_17 + x2_18 + x2_19 = 1
x3_2 + x3_1 + x3_4 + x3_5 + x3_6 + x3_7 + x3_8 + x3_9 + x3_10 + x3_11 + x3_12 + x3_13 + x3_14 + x3_15 + x3_16 + x3_17 + x3_18 + x3_19 = 1
x4_2 + x4_3 + x4_1 + x4_5 + x4_6 + x4_7 + x4_8 + x4_9 + x4_10 + x4_11 + x4_12 + x4_13 + x4_14 + x4_15 + x4_16 + x4_17 + x4_18 + x4_19 = 1
x5_2 + x5_3 + x5_4 + x5_1 + x5_6 + x5_7 + x5_8 + x5_9 + x5_10 + x5_11 + x5_12 + x5_13 + x5_14 + x5_15 + x5_16 + x5_17 + x5_18 + x5_19 = 1
x6_2 + x6_3 + x6_4 + x6_5 + x6_1 + x6_7 + x6_8 + x6_9 + x6_10 + x6_11 + x6_12 + x6_13 + x6_14 + x6_15 + x6_16 + x6_17 + x6_18 + x6_19 = 1
x7_2 + x7_3 + x7_4 + x7_5 + x7_6 + x7_1 + x7_8 + x7_9 + x7_10 + x7_11 + x7_12 + x7_13 + x7_14 + x7_15 + x7_16 + x7_17 + x7_18 + x7_19 = 1
x8_2 + x8_3 + x8_4 + x8_5 + x8_6 + x8_7 + x8_1 + x8_9 + x8_10 + x8_11 + x8_12 + x8_13 + x8_14 + x8_15 + x8_16 + x8_17 + x8_18 + x8_19 = 1
x9_2 + x9_3 + x9_4 + x9_5 + x9_6 + x9_7 + x9_8 + x9_1 + x9_10 + x9_11 + x9_12 + x9_13 + x9_14 + x9_15 + x9_16 + x9_17 + x9_18 + x9_19 = 1
x10_1 + x10_2 + x10_3 + x10_4 + x10_5 + x10_6 + x10_7 + x10_8 + x10_9 + x10_11 + x10_12 + x10_13 + x10_14 + x10_15 + x10_16 + x10_17 + x10_18 + x10_19 = 1
x11_1 + x11_2 + x11_3 + x11_4 + x11_5 + x11_6 + x11_7 + x11_8 + x11_9 + x11_10 + x11_12 + x11_13 + x11_14 + x11_15 + x11_16 + x11_17 + x11_18 + x11_19 = 1
x12_1 + x12_2 + x12_3 + x12_4 + x12_5 + x12_6 + x12_7 + x12_8 + x12_9 + x12_11 + x12_10 + x12_13 + x12_14 + x12_15 + x12_16 + x12_17 + x12_18 + x12_19 = 1
x13_1 + x13_2 + x13_3 + x13_4 + x13_5 + x13_6 + x13_7 + x13_8 + x13_9 + x13_11 + x13_12 + x13_10 + x13_14 + x13_15 + x13_16 + x13_17 + x13_18 + x13_19 = 1
x14_1 + x14_2 + x14_3 + x14_4 + x14_5 + x14_6 + x14_7 + x14_8 + x14_9 + x14_11 + x14_12 + x14_13 + x14_10 + x14_15 + x14_16 + x14_17 + x14_18 + x14_19 = 1
x15_1 + x15_2 + x15_3 + x15_4 + x15_5 + x15_6 + x15_7 + x15_8 + x15_9 + x15_11 + x15_12 + x15_13 + x15_14 + x15_10 + x15_16 + x15_17 + x15_18 + x15_19 = 1
x16_1 + x16_2 + x16_3 + x16_4 + x16_5 + x16_6 + x16_7 + x16_8 + x16_9 + x16_11 + x16_12 + x16_13 + x16_14 + x16_15 + x16_10 + x16_17 + x16_18 + x16_19 = 1
x17_1 + x17_2 + x17_3 + x17_4 + x17_5 + x17_6 + x17_7 + x17_8 + x17_9 + x17_11 + x17_12 + x17_13 + x17_14 + x17_15 + x17_16 + x17_10 + x17_18 + x17_19 = 1
x18_1 + x18_2 + x18_3 + x18_4 + x18_5 + x18_6 + x18_7 + x18_8 + x18_9 + x18_11 + x18_12 + x18_13 + x18_14 + x18_15 + x18_16 + x18_17 + x18_10 + x18_19 = 1
x19_1 + x19_2 + x19_3 + x19_4 + x19_5 + x19_6 + x19_7 + x19_8 + x19_9 + x19_11 + x19_12 + x19_13 + x19_14 + x19_15 + x19_16 + x19_17 + x19_18 + x19_10 = 1

x2_1 + x3_1 + x4_1 + x5_1 + x6_1 + x7_1 + x8_1 + x9_1 + x10_1 + x11_1 + x12_1 + x13_1 + x14_1 + x15_1 + x16_1 + x17_1 + x18_1 + x19_1 = 1
x1_2 + x3_2 + x4_2 + x5_2 + x6_2 + x7_2 + x8_2 + x9_2 + x10_2 + x11_2 + x12_2 + x13_2 + x14_2 + x15_2 + x16_2 + x17_2 + x18_2 + x19_2 = 1
x1_3 + x2_3 + x4_3 + x5_3 + x6_3 + x7_3 + x8_3 + x9_3 + x10_3 + x11_3 + x12_3 + x13_3 + x14_3 + x15_3 + x16_3 + x17_3 + x18_3 + x19_3 = 1
x1_4 + x2_4 + x3_4 + x5_4 + x6_4 + x7_4 + x8_4 + x9_4 + x10_4 + x11_4 + x12_4 + x13_4 + x14_4 + x15_4 + x16_4 + x17_4 + x18_4 + x19_4 = 1
x1_5 + x2_5 + x3_5 + x4_5 + x6_5 + x7_5 + x8_5 + x9_5 + x10_5 + x11_5 + x12_5 + x13_5 + x14_5 + x15_5 + x16_5 + x17_5 + x18_5 + x19_5 = 1
x1_6 + x2_6 + x3_6 + x4_6 + x5_6 + x7_6 + x8_6 + x9_6 + x10_6 + x11_6 + x12_6 + x13_6 + x14_6 + x15_6 + x16_6 + x17_6 + x18_6 + x19_6 = 1
x1_7 + x2_7 + x3_7 + x4_7 + x5_7 + x6_7 + x8_7 + x9_7 + x10_7 + x11_7 + x12_7 + x13_7 + x14_7 + x15_7 + x16_7 + x17_7 + x18_7 + x19_7 = 1
x1_8 + x2_8 + x3_8 + x4_8 + x5_8 + x6_8 + x7_8 + x9_8 + x10_8 + x11_8 + x12_8 + x13_8 + x14_8 + x15_8 + x16_8 + x17_8 + x18_8 + x19_8 = 1

x1_9 + x2_9 + x3_9 + x4_9 + x5_9 + x6_9 + x7_9 + x8_9 + x10_9 + x11_9 + x12_9 + x13_9 + x14_9 + x15_9 + x16_9 + x17_9 + x18_9 + x19_9 = 1
x1_10 + x2_10 + x3_10 + x4_10 + x5_10 + x6_10 + x7_10 + x8_10 + x9_10 + x11_10 + x12_10 + x13_10 + x14_10 + x15_10 + x16_10 + x17_10 + x18_10 + x19_10 = 1
x1_11 + x2_11 + x3_11 + x4_11 + x5_11 + x6_11 + x7_11 + x8_11 + x9_11 + x10_11 + x12_11 + x13_11 + x14_11 + x15_11 + x16_11 + x17_11 + x18_11 + x19_11 = 1
x1_12 + x2_12 + x3_12 + x4_12 + x5_12 + x6_12 + x7_12 + x8_12 + x9_12 + x10_12 + x11_12 + x13_12 + x14_12 + x15_12 + x16_12 + x17_12 + x18_12 + x19_12 = 1
x1_13 + x2_13 + x3_13 + x4_13 + x5_13 + x6_13 + x7_13 + x8_13 + x9_13 + x10_13 + x11_13 + x12_13 + x14_13 + x15_13 + x16_13 + x17_13 + x18_13 + x19_13 = 1
x1_14 + x2_14 + x3_14 + x4_14 + x5_14 + x6_14 + x7_14 + x8_14 + x9_14 + x10_14 + x11_14 + x12_14 + x13_14 + x15_14 + x16_14 + x17_14 + x18_14 + x19_14 = 1
x1_15 + x2_15 + x3_15 + x4_15 + x5_15 + x6_15 + x7_15 + x8_15 + x9_15 + x10_15 + x11_15 + x12_15 + x13_15 + x14_15 + x16_15 + x17_15 + x18_15 + x19_15 = 1
x1_16 + x2_16 + x3_16 + x4_16 + x5_16 + x6_16 + x7_16 + x8_16 + x9_16 + x10_16 + x11_16 + x12_16 + x13_16 + x14_16 + x15_16 + x17_16 + x18_16 + x19_16 = 1
x1_17 + x2_17 + x3_17 + x4_17 + x5_17 + x6_17 + x7_17 + x8_17 + x9_17 + x10_17 + x11_17 + x12_17 + x13_17 + x14_17 + x15_17 + x16_17 + x18_17 + x19_17 = 1
x1_18 + x2_18 + x3_18 + x4_18 + x5_18 + x6_18 + x7_18 + x8_18 + x9_18 + x10_18 + x11_18 + x12_18 + x13_18 + x14_18 + x15_18 + x16_18 + x17_18 + x19_18 = 1
x1_19 + x2_19 + x3_19 + x4_19 + x5_19 + x6_19 + x7_19 + x8_19 + x9_19 + x10_19 + x11_19 + x12_19 + x13_19 + x14_19 + x15_19 + x16_19 + x17_19 + x18_19 = 1

u1 - u2 + 19 x1_2 <= 18
u1 - u3 + 19 x1_3 <= 18
u1 - u4 + 19 x1_4 <= 18
u1 - u5 + 19 x1_5 <= 18
u1 - u6 + 19 x1_6 <= 18
u1 - u7 + 19 x1_7 <= 18
u1 - u8 + 19 x1_8 <= 18
u1 - u9 + 19 x1_9 <= 18
u1 - u10 + 19 x1_10 <= 18
u1 - u11 + 19 x1_11 <= 18
u1 - u12 + 19 x1_12 <= 18
u1 - u13 + 19 x1_13 <= 18
u1 - u14 + 19 x1_14 <= 18
u1 - u15 + 19 x1_15 <= 18
u1 - u16 + 19 x1_16 <= 18
u1 - u17 + 19 x1_17 <= 18
u1 - u18 + 19 x1_18 <= 18
u1 - u19 + 19 x1_19 <= 18

u2 - u3 + 19 x2_3 <= 18
u2 - u4 + 19 x2_4 <= 18
u2 - u5 + 19 x2_5 <= 18
u2 - u6 + 19 x2_6 <= 18
u2 - u7 + 19 x2_7 <= 18
u2 - u8 + 19 x2_8 <= 18
u2 - u9 + 19 x2_9 <= 18
u2 - u10 + 19 x2_10 <= 18
u2 - u11 + 19 x2_11 <= 18
u2 - u12 + 19 x2_12 <= 18
u2 - u13 + 19 x2_13 <= 18
u2 - u14 + 19 x2_14 <= 18
u2 - u15 + 19 x2_15 <= 18
u2 - u16 + 19 x2_16 <= 18
u2 - u17 + 19 x2_17 <= 18
u2 - u18 + 19 x2_18 <= 18
u2 - u19 + 19 x2_19 <= 18

u3 - u2 + 19 x3_2 <= 18
u3 - u4 + 19 x3_4 <= 18
u3 - u5 + 19 x3_5 <= 18
u3 - u6 + 19 x3_6 <= 18
u3 - u7 + 19 x3_7 <= 18
u3 - u8 + 19 x3_8 <= 18
u3 - u9 + 19 x3_9 <= 18
u3 - u10 + 19 x3_10 <= 18
u3 - u11 + 19 x3_11 <= 18
u3 - u12 + 19 x3_12 <= 18
u3 - u13 + 19 x3_13 <= 18
u3 - u14 + 19 x3_14 <= 18
u3 - u15 + 19 x3_15 <= 18
u3 - u16 + 19 x3_16 <= 18
u3 - u17 + 19 x3_17 <= 18
u3 - u18 + 19 x3_18 <= 18
u3 - u19 + 19 x3_19 <= 18

u4 - u2 + 19 x4_2 <= 18
u4 - u3 + 19 x4_3 <= 18
u4 - u5 + 19 x4_5 <= 18
u4 - u6 + 19 x4_6 <= 18
u4 - u7 + 19 x4_7 <= 18
u4 - u8 + 19 x4_8 <= 18
u4 - u9 + 19 x4_9 <= 18
u4 - u10 + 19 x4_10 <= 18
u4 - u11 + 19 x4_11 <= 18
u4 - u12 + 19 x4_12 <= 18
u4 - u13 + 19 x4_13 <= 18
u4 - u14 + 19 x4_14 <= 18
u4 - u15 + 19 x4_15 <= 18
u4 - u16 + 19 x4_16 <= 18

u4 - u17 + 19 x4_17 <= 18
u4 - u18 + 19 x4_18 <= 18
u4 - u19 + 19 x4_19 <= 18

u5 - u2 + 19 x5_2 <= 18
u5 - u3 + 19 x5_3 <= 18
u5 - u4 + 19 x5_4 <= 18
u5 - u6 + 19 x5_6 <= 18
u5 - u7 + 19 x5_7 <= 18
u5 - u8 + 19 x5_8 <= 18
u5 - u9 + 19 x5_9 <= 18
u5 - u10 + 19 x5_10 <= 18
u5 - u11 + 19 x5_11 <= 18
u5 - u12 + 19 x5_12 <= 18
u5 - u13 + 19 x5_13 <= 18
u5 - u14 + 19 x5_14 <= 18
u5 - u15 + 19 x5_15 <= 18
u5 - u16 + 19 x5_16 <= 18
u5 - u17 + 19 x5_17 <= 18
u5 - u18 + 19 x5_18 <= 18
u5 - u19 + 19 x5_19 <= 18

u6 - u2 + 19 x6_2 <= 18
u6 - u3 + 19 x6_3 <= 18
u6 - u4 + 19 x6_4 <= 18
u6 - u5 + 19 x6_5 <= 18
u6 - u7 + 19 x6_7 <= 18
u6 - u8 + 19 x6_8 <= 18
u6 - u9 + 19 x6_9 <= 18
u6 - u10 + 19 x6_10 <= 18
u6 - u11 + 19 x6_11 <= 18
u6 - u12 + 19 x6_12 <= 18
u6 - u13 + 19 x6_13 <= 18
u6 - u14 + 19 x6_14 <= 18
u6 - u15 + 19 x6_15 <= 18
u6 - u16 + 19 x6_16 <= 18
u6 - u17 + 19 x6_17 <= 18
u6 - u18 + 19 x6_18 <= 18
u6 - u19 + 19 x6_19 <= 18

u7 - u2 + 19 x7_2 <= 18
u7 - u3 + 19 x7_3 <= 18
u7 - u4 + 19 x7_4 <= 18
u7 - u5 + 19 x7_5 <= 18
u7 - u6 + 19 x7_6 <= 18
u7 - u8 + 19 x7_8 <= 18
u7 - u9 + 19 x7_9 <= 18
u7 - u10 + 19 x7_10 <= 18
u7 - u11 + 19 x7_11 <= 18
u7 - u12 + 19 x7_12 <= 18
u7 - u13 + 19 x7_13 <= 18
u7 - u14 + 19 x7_14 <= 18
u7 - u15 + 19 x7_15 <= 18
u7 - u16 + 19 x7_16 <= 18
u7 - u17 + 19 x7_17 <= 18
u7 - u18 + 19 x7_18 <= 18
u7 - u19 + 19 x7_19 <= 18

u8 - u2 + 19 x8_2 <= 18
u8 - u3 + 19 x8_3 <= 18
u8 - u4 + 19 x8_4 <= 18
u8 - u5 + 19 x8_5 <= 18
u8 - u6 + 19 x8_6 <= 18
u8 - u7 + 19 x8_7 <= 18
u8 - u9 + 19 x8_9 <= 18
u8 - u10 + 19 x8_10 <= 18
u8 - u11 + 19 x8_11 <= 18
u8 - u12 + 19 x8_12 <= 18
u8 - u13 + 19 x8_13 <= 18
u8 - u14 + 19 x8_14 <= 18
u8 - u15 + 19 x8_15 <= 18
u8 - u16 + 19 x8_16 <= 18
u8 - u17 + 19 x8_17 <= 18
u8 - u18 + 19 x8_18 <= 18
u8 - u19 + 19 x8_19 <= 18

u9 - u2 + 19 x9_2 <= 18
u9 - u3 + 19 x9_3 <= 18
u9 - u4 + 19 x9_4 <= 18
u9 - u5 + 19 x9_5 <= 18
u9 - u6 + 19 x9_6 <= 18
u9 - u7 + 19 x9_7 <= 18
u9 - u8 + 19 x9_8 <= 18
u9 - u10 + 19 x9_10 <= 18
u9 - u11 + 19 x9_11 <= 18
u9 - u12 + 19 x9_12 <= 18
u9 - u13 + 19 x9_13 <= 18
u9 - u14 + 19 x9_14 <= 18
u9 - u15 + 19 x9_15 <= 18
u9 - u16 + 19 x9_16 <= 18
u9 - u17 + 19 x9_17 <= 18
u9 - u18 + 19 x9_18 <= 18
u9 - u19 + 19 x9_19 <= 18

u10 - u2 + 19 x10_2 <= 18
u10 - u3 + 19 x10_3 <= 18
u10 - u4 + 19 x10_4 <= 18
u10 - u5 + 19 x10_5 <= 18
u10 - u6 + 19 x10_6 <= 18
u10 - u7 + 19 x10_7 <= 18
u10 - u8 + 19 x10_8 <= 18
u10 - u9 + 19 x10_9 <= 18
u10 - u11 + 19 x10_11 <= 18
u10 - u12 + 19 x10_12 <= 18
u10 - u13 + 19 x10_13 <= 18
u10 - u14 + 19 x10_14 <= 18
u10 - u15 + 19 x10_15 <= 18
u10 - u16 + 19 x10_16 <= 18
u10 - u17 + 19 x10_17 <= 18
u10 - u18 + 19 x10_18 <= 18
u10 - u19 + 19 x10_19 <= 18

u11 - u2 + 19 x11_2 <= 18
u11 - u3 + 19 x11_3 <= 18

u11 - u4 + 19 x11_4 <= 18
u11 - u5 + 19 x11_5 <= 18
u11 - u6 + 19 x11_6 <= 18
u11 - u7 + 19 x11_7 <= 18
u11 - u8 + 19 x11_8 <= 18
u11 - u9 + 19 x11_9 <= 18
u11 - u10 + 19 x11_10 <= 18
u11 - u12 + 19 x11_12 <= 18
u11 - u13 + 19 x11_13 <= 18
u11 - u14 + 19 x11_14 <= 18
u11 - u15 + 19 x11_15 <= 18
u11 - u16 + 19 x11_16 <= 18
u11 - u17 + 19 x11_17 <= 18
u11 - u18 + 19 x11_18 <= 18
u11 - u19 + 19 x11_19 <= 18

u12 - u2 + 19 x12_2 <= 18
u12 - u3 + 19 x12_3 <= 18
u12 - u4 + 19 x12_4 <= 18
u12 - u5 + 19 x12_5 <= 18
u12 - u6 + 19 x12_6 <= 18
u12 - u7 + 19 x12_7 <= 18
u12 - u8 + 19 x12_8 <= 18
u12 - u9 + 19 x12_9 <= 18
u12 - u10 + 19 x12_10 <= 18
u12 - u11 + 19 x12_11 <= 18
u12 - u13 + 19 x12_13 <= 18
u12 - u14 + 19 x12_14 <= 18
u12 - u15 + 19 x12_15 <= 18
u12 - u16 + 19 x12_16 <= 18
u12 - u17 + 19 x12_17 <= 18
u12 - u18 + 19 x12_18 <= 18
u12 - u19 + 19 x12_19 <= 18

u13 - u2 + 19 x13_2 <= 18
u13 - u3 + 19 x13_3 <= 18
u13 - u4 + 19 x13_4 <= 18
u13 - u5 + 19 x13_5 <= 18
u13 - u6 + 19 x13_6 <= 18
u13 - u7 + 19 x13_7 <= 18
u13 - u8 + 19 x13_8 <= 18
u13 - u9 + 19 x13_9 <= 18
u13 - u10 + 19 x13_10 <= 18
u13 - u11 + 19 x13_11 <= 18
u13 - u12 + 19 x13_12 <= 18
u13 - u14 + 19 x13_14 <= 18
u13 - u15 + 19 x13_15 <= 18
u13 - u16 + 19 x13_16 <= 18
u13 - u17 + 19 x13_17 <= 18
u13 - u18 + 19 x13_18 <= 18
u13 - u19 + 19 x13_19 <= 18

u14 - u2 + 19 x14_2 <= 18
u14 - u3 + 19 x14_3 <= 18
u14 - u4 + 19 x14_4 <= 18
u14 - u5 + 19 x14_5 <= 18
u14 - u6 + 19 x14_6 <= 18
u14 - u7 + 19 x14_7 <= 18
u14 - u8 + 19 x14_8 <= 18
u14 - u9 + 19 x14_9 <= 18
u14 - u10 + 19 x14_10 <= 18
u14 - u11 + 19 x14_11 <= 18
u14 - u12 + 19 x14_12 <= 18
u14 - u13 + 19 x14_13 <= 18
u14 - u15 + 19 x14_15 <= 18
u14 - u16 + 19 x14_16 <= 18
u14 - u17 + 19 x14_17 <= 18
u14 - u18 + 19 x14_18 <= 18
u14 - u19 + 19 x14_19 <= 18

u15 - u2 + 19 x15_2 <= 18
u15 - u3 + 19 x15_3 <= 18
u15 - u4 + 19 x15_4 <= 18
u15 - u5 + 19 x15_5 <= 18
u15 - u6 + 19 x15_6 <= 18
u15 - u7 + 19 x15_7 <= 18
u15 - u8 + 19 x15_8 <= 18
u15 - u9 + 19 x15_9 <= 18
u15 - u10 + 19 x15_10 <= 18
u15 - u11 + 19 x15_11 <= 18
u15 - u12 + 19 x15_12 <= 18
u15 - u13 + 19 x15_13 <= 18
u15 - u14 + 19 x15_14 <= 18
u15 - u16 + 19 x15_16 <= 18
u15 - u17 + 19 x15_17 <= 18
u15 - u18 + 19 x15_18 <= 18
u15 - u19 + 19 x15_19 <= 18

u16 - u2 + 19 x16_2 <= 18
u16 - u3 + 19 x16_3 <= 18
u16 - u4 + 19 x16_4 <= 18
u16 - u5 + 19 x16_5 <= 18
u16 - u6 + 19 x16_6 <= 18
u16 - u7 + 19 x16_7 <= 18
u16 - u8 + 19 x16_8 <= 18
u16 - u9 + 19 x16_9 <= 18
u16 - u10 + 19 x16_10 <= 18
u16 - u11 + 19 x16_11 <= 18
u16 - u12 + 19 x16_12 <= 18
u16 - u13 + 19 x16_13 <= 18
u16 - u14 + 19 x16_14 <= 18
u16 - u15 + 19 x16_15 <= 18
u16 - u17 + 19 x16_17 <= 18
u16 - u18 + 19 x16_18 <= 18
u16 - u19 + 19 x16_19 <= 18

u17 - u2 + 19 x17_2 <= 18
u17 - u3 + 19 x17_3 <= 18
u17 - u4 + 19 x17_4 <= 18
u17 - u5 + 19 x17_5 <= 18
u17 - u6 + 19 x17_6 <= 18
u17 - u7 + 19 x17_7 <= 18
u17 - u8 + 19 x17_8 <= 18
u17 - u9 + 19 x17_9 <= 18

u17 - u10 + 19 x17_10 <= 18
u17 - u11 + 19 x17_11 <= 18
u17 - u12 + 19 x17_12 <= 18
u17 - u13 + 19 x17_13 <= 18
u17 - u14 + 19 x17_14 <= 18
u17 - u15 + 19 x17_15 <= 18
u17 - u16 + 19 x17_16 <= 18
u17 - u18 + 19 x17_18 <= 18
u17 - u19 + 19 x17_19 <= 18

u18 - u2 + 19 x18_2 <= 18
u18 - u3 + 19 x18_3 <= 18
u18 - u4 + 19 x18_4 <= 18
u18 - u5 + 19 x18_5 <= 18
u18 - u6 + 19 x18_6 <= 18
u18 - u7 + 19 x18_7 <= 18
u18 - u8 + 19 x18_8 <= 18
u18 - u9 + 19 x18_9 <= 18
u18 - u10 + 19 x18_10 <= 18
u18 - u11 + 19 x18_11 <= 18
u18 - u12 + 19 x18_12 <= 18
u18 - u13 + 19 x18_13 <= 18
u18 - u14 + 19 x18_14 <= 18
u18 - u15 + 19 x18_15 <= 18
u18 - u16 + 19 x18_16 <= 18
u18 - u17 + 19 x18_17 <= 18
u18 - u19 + 19 x18_19 <= 18

u19 - u2 + 19 x19_2 <= 18
u19 - u3 + 19 x19_3 <= 18
u19 - u4 + 19 x19_4 <= 18
u19 - u5 + 19 x19_5 <= 18
u19 - u6 + 19 x19_6 <= 18
u19 - u7 + 19 x19_7 <= 18
u19 - u8 + 19 x19_8 <= 18
u19 - u9 + 19 x19_9 <= 18
u19 - u10 + 19 x19_10 <= 18
u19 - u11 + 19 x19_11 <= 18
u19 - u12 + 19 x19_12 <= 18
u19 - u13 + 19 x19_13 <= 18
u19 - u14 + 19 x19_14 <= 18
u19 - u15 + 19 x19_15 <= 18
u19 - u16 + 19 x19_16 <= 18
u19 - u17 + 19 x19_17 <= 18
u19 - u18 + 19 x19_18 <= 18

u1 >= 1
u2 >= 1
u3 >= 1
u4 >= 1
u5 >= 1
u6 >= 1
u7 >= 1
u8 >= 1
u9 >= 1
u10 >= 1
u11 >= 1
u12 >= 1
u13 >= 1
u14 >= 1
u15 >= 1
u16 >= 1
u17 >= 1
u18 >= 1
u19 >= 1

u1 <= 19
u2 <= 19
u3 <= 19
u4 <= 19
u5 <= 19
u6 <= 19
u7 <= 19
u8 <= 19
u9 <= 19
u10 <= 19
u11 <= 19
u12 <= 19
u13 <= 19
u14 <= 19
u15 <= 19
u16 <= 19
u17 <= 19
u18 <= 19
u19 <= 19

a1 = 0
a2 >= 8
a2 <= 24
a4 >= 24
a4 <= 28
a5 >= 24
a5 <= 32
a6 >= 48
a6 <= 56
a7 >= 48
a7 <= 56
a8 >= 48
a8 <= 56
a9 >= 48
a9 <= 56
a10 >= 48
a10 <= 56
a11 >= 72
a11 <= 80
a12 >= 72
a12 <= 80
a13 >= 72
a13 <= 80
a14 >= 72
a14 <= 80
a15 >= 72
a15 <= 80
a16 >= 72

a16 <= 80
a17 >= 96
a17 <= 100

a18 >= 100
a18 <= 104
a19 >= 120

a1 - a3 + 1000000 x1_3 + W3 + v1_3 = 999998.114
a1 - a4 + 1000000 x1_4 + W4 + v1_4 = 999990.164
a1 - a5 + 1000000 x1_5 + W5 + v1_5 = 999989.515
a1 - a6 + 1000000 x1_6 + W6 + v1_6 = 999988.205
a1 - a7 + 1000000 x1_7 + W7 + v1_7 = 999988.068
a1 - a8 + 1000000 x1_8 + W8 + v1_8 = 999987.872
a1 - a9 + 1000000 x1_9 + W9 + v1_9 = 999987.732
a1 - a10 + 1000000 x1_10 + W10 + v1_10 = 999987.937
a1 - a11 + 1000000 x1_11 + W11 + v1_11 = 999987.882
a1 - a12 + 1000000 x1_12 + W12 + v1_12 = 999986.959
a1 - a13 + 1000000 x1_13 + W13 + v1_13 = 999986.465
a1 - a14 + 1000000 x1_14 + W14 + v1_14 = 999986.463
a1 - a15 + 1000000 x1_15 + W15 + v1_15 = 999988.156
a1 - a16 + 1000000 x1_16 + W16 + v1_16 = 999988.916
a1 - a17 + 1000000 x1_17 + W17 + v1_17 = 999989.84
a1 - a18 + 1000000 x1_18 + W18 + v1_18 = 999992.304
a1 - a19 + 1000000 x1_19 + W19 + v1_19 = 999999.088

a2 - a3 + 1000000 x2_3 + W3 + v2_3 = 999994.298
a2 - a4 + 1000000 x2_4 + W4 + v2_4 = 999997.403
a2 - a5 + 1000000 x2_5 + W5 + v2_5 = 999996.753
a2 - a6 + 1000000 x2_6 + W6 + v2_6 = 999993.866
a2 - a7 + 1000000 x2_7 + W7 + v2_7 = 999993.588
a2 - a8 + 1000000 x2_8 + W8 + v2_8 = 999993.361
a2 - a9 + 1000000 x2_9 + W9 + v2_9 = 999993.04
a2 - a10 + 1000000 x2_10 + W10 + v2_10 = 999993.14
a2 - a11 + 1000000 x2_11 + W11 + v2_11 = 999992.861
a2 - a12 + 1000000 x2_12 + W12 + v2_12 = 999992.233
a2 - a13 + 1000000 x2_13 + W13 + v2_13 = 999991.849
a2 - a14 + 1000000 x2_14 + W14 + v2_14 = 999991.759
a2 - a15 + 1000000 x2_15 + W15 + v2_15 = 999992.748
a2 - a16 + 1000000 x2_16 + W16 + v2_16 = 999993.508
a2 - a17 + 1000000 x2_17 + W17 + v2_17 = 999994.432
a2 - a18 + 1000000 x2_18 + W18 + v2_18 = 999996.531
a2 - a19 + 1000000 x2_19 + W19 + v2_19 = 999991.828

a3 - a2 + 1000000 x3_2 + W2 + v3_2 = 999985.257
a3 - a4 + 1000000 x3_4 + W4 + v3_4 = 999991.889
a3 - a5 + 1000000 x3_5 + W5 + v3_5 = 999991.24
a3 - a6 + 1000000 x3_6 + W6 + v3_6 = 999989.93
a3 - a7 + 1000000 x3_7 + W7 + v3_7 = 999989.793
a3 - a8 + 1000000 x3_8 + W8 + v3_8 = 999989.597
a3 - a9 + 1000000 x3_9 + W9 + v3_9 = 999989.457
a3 - a10 + 1000000 x3_10 + W10 + v3_10 = 999989.662
a3 - a11 + 1000000 x3_11 + W11 + v3_11 = 999989.606
a3 - a12 + 1000000 x3_12 + W12 + v3_12 = 999988.684
a3 - a13 + 1000000 x3_13 + W13 + v3_13 = 999988.19
a3 - a14 + 1000000 x3_14 + W14 + v3_14 = 999988.188
a3 - a15 + 1000000 x3_15 + W15 + v3_15 = 999989.881
a3 - a16 + 1000000 x3_16 + W16 + v3_16 = 999990.64
a3 - a17 + 1000000 x3_17 + W17 + v3_17 = 999991.565

a3 - a18 + 1000000 x3_18 + W18 + v3_18 = 999994.029
a3 - a19 + 1000000 x3_19 + W19 + v3_19 = 999997.38

a4 - a2 + 1000000 x4_2 + W2 + v4_2 = 999989.16
a4 - a3 + 1000000 x4_3 + W3 + v4_3 = 999992.673
a4 - a5 + 1000000 x4_5 + W5 + v4_5 = 999998.285
a4 - a6 + 1000000 x4_6 + W6 + v4_6 = 999995.398
a4 - a7 + 1000000 x4_7 + W7 + v4_7 = 999995.12
a4 - a8 + 1000000 x4_8 + W8 + v4_8 = 999994.892
a4 - a9 + 1000000 x4_9 + W9 + v4_9 = 999994.571
a4 - a10 + 1000000 x4_10 + W10 + v4_10 = 999994.54
a4 - a11 + 1000000 x4_11 + W11 + v4_11 = 999994.204
a4 - a12 + 1000000 x4_12 + W12 + v4_12 = 999993.765
a4 - a13 + 1000000 x4_13 + W13 + v4_13 = 999993.38
a4 - a14 + 1000000 x4_14 + W14 + v4_14 = 999993.29
a4 - a15 + 1000000 x4_15 + W15 + v4_15 = 999993.942
a4 - a16 + 1000000 x4_16 + W16 + v4_16 = 999994.701
a4 - a17 + 1000000 x4_17 + W17 + v4_17 = 999995.626
a4 - a18 + 1000000 x4_18 + W18 + v4_18 = 999995.065
a4 - a19 + 1000000 x4_19 + W19 + v4_19 = 999990.202

a5 - a2 + 1000000 x5_2 + W2 + v5_2 = 999988.505
a5 - a3 + 1000000 x5_3 + W3 + v5_3 = 999992.018
a5 - a4 + 1000000 x5_4 + W4 + v5_4 = 999998.269
a5 - a6 + 1000000 x5_6 + W6 + v5_6 = 999996.059
a5 - a7 + 1000000 x5_7 + W7 + v5_7 = 999995.781
a5 - a8 + 1000000 x5_8 + W8 + v5_8 = 999995.554
a5 - a9 + 1000000 x5_9 + W9 + v5_9 = 999995.233
a5 - a10 + 1000000 x5_10 + W10 + v5_10 = 999995.202
a5 - a11 + 1000000 x5_11 + W11 + v5_11 = 999994.866
a5 - a12 + 1000000 x5_12 + W12 + v5_12 = 999994.426
a5 - a13 + 1000000 x5_13 + W13 + v5_13 = 999994.042
a5 - a14 + 1000000 x5_14 + W14 + v5_14 = 999993.951
a5 - a15 + 1000000 x5_15 + W15 + v5_15 = 999994.49
a5 - a16 + 1000000 x5_16 + W16 + v5_16 = 999995.25
a5 - a17 + 1000000 x5_17 + W17 + v5_17 = 999996.174
a5 - a18 + 1000000 x5_18 + W18 + v5_18 = 999995.614
a5 - a19 + 1000000 x5_19 + W19 + v5_19 = 999989.547

a6 - a2 + 1000000 x6_2 + W2 + v6_2 = 999985.575
a6 - a3 + 1000000 x6_3 + W3 + v6_3 = 999990.734
a6 - a4 + 1000000 x6_4 + W4 + v6_4 = 999995.34
a6 - a5 + 1000000 x6_5 + W5 + v6_5 = 999996.027
a6 - a7 + 1000000 x6_7 + W7 + v6_7 = 999998.815
a6 - a8 + 1000000 x6_8 + W8 + v6_8 = 999998.588
a6 - a9 + 1000000 x6_9 + W9 + v6_9 = 999998.267
a6 - a10 + 1000000 x6_10 + W10 + v6_10 = 999998.178
a6 - a11 + 1000000 x6_11 + W11 + v6_11 = 999997.739
a6 - a12 + 1000000 x6_12 + W12 + v6_12 = 999997.46
a6 - a13 + 1000000 x6_13 + W13 + v6_13 = 999997.076
a6 - a14 + 1000000 x6_14 + W14 + v6_14 = 999996.986
a6 - a15 + 1000000 x6_15 + W15 + v6_15 = 999997.145
a6 - a16 + 1000000 x6_16 + W16 + v6_16 = 999996.99
a6 - a17 + 1000000 x6_17 + W17 + v6_17 = 999997.281

a6 - a18 + 1000000 x6_18 + W18 + v6_18 = 999994.857
a6 - a19 + 1000000 x6_19 + W19 + v6_19 = 999988.264

a7 - a2 + 1000000 x7_2 + W2 + v7_2 = 999985.336
a7 - a3 + 1000000 x7_3 + W3 + v7_3 = 999990.626
a7 - a4 + 1000000 x7_4 + W4 + v7_4 = 999995.101
a7 - a5 + 1000000 x7_5 + W5 + v7_5 = 999995.788
a7 - a6 + 1000000 x7_6 + W6 + v7_6 = 999998.863
a7 - a8 + 1000000 x7_8 + W8 + v7_8 = 999998.745
a7 - a9 + 1000000 x7_9 + W9 + v7_9 = 999998.423
a7 - a10 + 1000000 x7_10 + W10 + v7_10 = 999998.328
a7 - a11 + 1000000 x7_11 + W11 + v7_11 = 999997.708
a7 - a12 + 1000000 x7_12 + W12 + v7_12 = 999997.617
a7 - a13 + 1000000 x7_13 + W13 + v7_13 = 999997.232
a7 - a14 + 1000000 x7_14 + W14 + v7_14 = 999997.142
a7 - a15 + 1000000 x7_15 + W15 + v7_15 = 999997.294
a7 - a16 + 1000000 x7_16 + W16 + v7_16 = 999996.959
a7 - a17 + 1000000 x7_17 + W17 + v7_17 = 999997.173
a7 - a18 + 1000000 x7_18 + W18 + v7_18 = 999994.749
a7 - a19 + 1000000 x7_19 + W19 + v7_19 = 999988.156

a8 - a2 + 1000000 x8_2 + W2 + v8_2 = 999985.027
a8 - a3 + 1000000 x8_3 + W3 + v8_3 = 999990.411
a8 - a4 + 1000000 x8_4 + W4 + v8_4 = 999994.792
a8 - a5 + 1000000 x8_5 + W5 + v8_5 = 999995.479
a8 - a6 + 1000000 x8_6 + W6 + v8_6 = 999998.554
a8 - a7 + 1000000 x8_7 + W7 + v8_7 = 999998.722
a8 - a9 + 1000000 x8_9 + W9 + v8_9 = 999998.742
a8 - a10 + 1000000 x8_10 + W10 + v8_10 = 999998.658
a8 - a11 + 1000000 x8_11 + W11 + v8_11 = 999997.85
a8 - a12 + 1000000 x8_12 + W12 + v8_12 = 999997.815
a8 - a13 + 1000000 x8_13 + W13 + v8_13 = 999997.431
a8 - a14 + 1000000 x8_14 + W14 + v8_14 = 999997.34
a8 - a15 + 1000000 x8_15 + W15 + v8_15 = 999997.55
a8 - a16 + 1000000 x8_16 + W16 + v8_16 = 999997.062
a8 - a17 + 1000000 x8_17 + W17 + v8_17 = 999996.958
a8 - a18 + 1000000 x8_18 + W18 + v8_18 = 999994.534
a8 - a19 + 1000000 x8_19 + W19 + v8_19 = 999987.941

a9 - a2 + 1000000 x9_2 + W2 + v9_2 = 999984.727
a9 - a3 + 1000000 x9_3 + W3 + v9_3 = 999990.243
a9 - a4 + 1000000 x9_4 + W4 + v9_4 = 999994.492
a9 - a5 + 1000000 x9_5 + W5 + v9_5 = 999995.179
a9 - a6 + 1000000 x9_6 + W6 + v9_6 = 999998.254
a9 - a7 + 1000000 x9_7 + W7 + v9_7 = 999998.422
a9 - a8 + 1000000 x9_8 + W8 + v9_8 = 999998.76
a9 - a10 + 1000000 x9_10 + W10 + v9_10 = 999998.797
a9 - a11 + 1000000 x9_11 + W11 + v9_11 = 999997.945
a9 - a12 + 1000000 x9_12 + W12 + v9_12 = 999997.567
a9 - a13 + 1000000 x9_13 + W13 + v9_13 = 999997.183
a9 - a14 + 1000000 x9_14 + W14 + v9_14 = 999997.092
a9 - a15 + 1000000 x9_15 + W15 + v9_15 = 999997.645
a9 - a16 + 1000000 x9_16 + W16 + v9_16 = 999996.987
a9 - a17 + 1000000 x9_17 + W17 + v9_17 = 999996.79

a9 - a18 + 1000000 x9_18 + W18 + v9_18 = 999994.366
a9 - a19 + 1000000 x9_19 + W19 + v9_19 = 999987.772

a10 - a2 + 1000000 x10_2 + W2 + v10_2 = 999984.768
a10 - a3 + 1000000 x10_3 + W3 + v10_3 = 999990.424
a10 - a4 + 1000000 x10_4 + W4 + v10_4 = 999994.46
a10 - a5 + 1000000 x10_5 + W5 + v10_5 = 999995.148
a10 - a6 + 1000000 x10_6 + W6 + v10_6 = 999998.171
a10 - a7 + 1000000 x10_7 + W7 + v10_7 = 999998.324
a10 - a8 + 1000000 x10_8 + W8 + v10_8 = 999998.682
a10 - a9 + 1000000 x10_9 + W9 + v10_9 = 999998.775
a10 - a11 + 1000000 x10_11 + W11 + v10_11 = 999998.182
a10 - a12 + 1000000 x10_12 + W12 + v10_12 = 999997.572
a10 - a13 + 1000000 x10_13 + W13 + v10_13 = 999997.188
a10 - a14 + 1000000 x10_14 + W14 + v10_14 = 999997.097
a10 - a15 + 1000000 x10_15 + W15 + v10_15 = 999997.883
a10 - a16 + 1000000 x10_16 + W16 + v10_16 = 999997.194
a10 - a17 + 1000000 x10_17 + W17 + v10_17 = 999996.971
a10 - a18 + 1000000 x10_18 + W18 + v10_18 = 999994.547
a10 - a19 + 1000000 x10_19 + W19 + v10_19 = 999987.953

a11 - a2 + 1000000 x11_2 + W2 + v11_2 = 999984.508
a11 - a3 + 1000000 x11_3 + W3 + v11_3 = 999990.435
a11 - a4 + 1000000 x11_4 + W4 + v11_4 = 999994.137
a11 - a5 + 1000000 x11_5 + W5 + v11_5 = 999994.825
a11 - a6 + 1000000 x11_6 + W6 + v11_6 = 999997.745
a11 - a7 + 1000000 x11_7 + W7 + v11_7 = 999997.734
a11 - a8 + 1000000 x11_8 + W8 + v11_8 = 999997.889
a11 - a9 + 1000000 x11_9 + W9 + v11_9 = 999997.951
a11 - a10 + 1000000 x11_10 + W10 + v11_10 = 999998.194
a11 - a12 + 1000000 x11_12 + W12 + v11_12 = 999997.288
a11 - a13 + 1000000 x11_13 + W13 + v11_13 = 999996.709
a11 - a14 + 1000000 x11_14 + W14 + v11_14 = 999996.708
a11 - a15 + 1000000 x11_15 + W15 + v11_15 = 999998.274
a11 - a16 + 1000000 x11_16 + W16 + v11_16 = 999997.875
a11 - a17 + 1000000 x11_17 + W17 + v11_17 = 999997.044
a11 - a18 + 1000000 x11_18 + W18 + v11_18 = 999994.557
a11 - a19 + 1000000 x11_19 + W19 + v11_19 = 999987.964

a12 - a2 + 1000000 x12_2 + W2 + v12_2 = 999983.957
a12 - a3 + 1000000 x12_3 + W3 + v12_3 = 999989.518
a12 - a4 + 1000000 x12_4 + W4 + v12_4 = 999993.722
a12 - a5 + 1000000 x12_5 + W5 + v12_5 = 999994.409
a12 - a6 + 1000000 x12_6 + W6 + v12_6 = 999997.484
a12 - a7 + 1000000 x12_7 + W7 + v12_7 = 999997.652
a12 - a8 + 1000000 x12_8 + W8 + v12_8 = 999997.827
a12 - a9 + 1000000 x12_9 + W9 + v12_9 = 999997.593
a12 - a10 + 1000000 x12_10 + W10 + v12_10 = 999997.543
a12 - a11 + 1000000 x12_11 + W11 + v12_11 = 999997.292
a12 - a13 + 1000000 x12_13 + W13 + v12_13 = 999998.285
a12 - a14 + 1000000 x12_14 + W14 + v12_14 = 999998.284
a12 - a15 + 1000000 x12_15 + W15 + v12_15 = 999997.8
a12 - a16 + 1000000 x12_16 + W16 + v12_16 = 999997.071
a12 - a17 + 1000000 x12_17 + W17 + v12_17 = 999996.079

a12 - a18 + 1000000 x12_18 + W18 + v12_18 = 999993.64
a12 - a19 + 1000000 x12_19 + W19 + v12_19 = 999987.047

a13 - a2 + 1000000 x13_2 + W2 + v13_2 = 999983.581
a13 - a3 + 1000000 x13_3 + W3 + v13_3 = 999989.042
a13 - a4 + 1000000 x13_4 + W4 + v13_4 = 999993.345
a13 - a5 + 1000000 x13_5 + W5 + v13_5 = 999994.033
a13 - a6 + 1000000 x13_6 + W6 + v13_6 = 999997.107
a13 - a7 + 1000000 x13_7 + W7 + v13_7 = 999997.276
a13 - a8 + 1000000 x13_8 + W8 + v13_8 = 999997.45
a13 - a9 + 1000000 x13_9 + W9 + v13_9 = 999997.217
a13 - a10 + 1000000 x13_10 + W10 + v13_10 = 999997.167
a13 - a11 + 1000000 x13_11 + W11 + v13_11 = 999996.723
a13 - a12 + 1000000 x13_12 + W12 + v13_12 = 999998.284
a13 - a14 + 1000000 x13_14 + W14 + v13_14 = 999999.021
a13 - a15 + 1000000 x13_15 + W15 + v13_15 = 999997.324
a13 - a16 + 1000000 x13_16 + W16 + v13_16 = 999996.595
a13 - a17 + 1000000 x13_17 + W17 + v13_17 = 999995.603
a13 - a18 + 1000000 x13_18 + W18 + v13_18 = 999993.165
a13 - a19 + 1000000 x13_19 + W19 + v13_19 = 999986.571

a14 - a2 + 1000000 x14_2 + W2 + v14_2 = 999983.484
a14 - a3 + 1000000 x14_3 + W3 + v14_3 = 999989.036
a14 - a4 + 1000000 x14_4 + W4 + v14_4 = 999993.249
a14 - a5 + 1000000 x14_5 + W5 + v14_5 = 999993.936
a14 - a6 + 1000000 x14_6 + W6 + v14_6 = 999997.01
a14 - a7 + 1000000 x14_7 + W7 + v14_7 = 999997.179
a14 - a8 + 1000000 x14_8 + W8 + v14_8 = 999997.353
a14 - a9 + 1000000 x14_9 + W9 + v14_9 = 999997.12
a14 - a10 + 1000000 x14_10 + W10 + v14_10 = 999997.07
a14 - a11 + 1000000 x14_11 + W11 + v14_11 = 999996.718
a14 - a12 + 1000000 x14_12 + W12 + v14_12 = 999998.277
a14 - a13 + 1000000 x14_13 + W13 + v14_13 = 999999.024
a14 - a15 + 1000000 x14_15 + W15 + v14_15 = 999997.319
a14 - a16 + 1000000 x14_16 + W16 + v14_16 = 999996.589
a14 - a17 + 1000000 x14_17 + W17 + v14_17 = 999995.597
a14 - a18 + 1000000 x14_18 + W18 + v14_18 = 999993.158
a14 - a19 + 1000000 x14_19 + W19 + v14_19 = 999986.565

a15 - a2 + 1000000 x15_2 + W2 + v15_2 = 999984.51
a15 - a3 + 1000000 x15_3 + W3 + v15_3 = 999990.721
a15 - a4 + 1000000 x15_4 + W4 + v15_4 = 999993.892
a15 - a5 + 1000000 x15_5 + W5 + v15_5 = 999994.478
a15 - a6 + 1000000 x15_6 + W6 + v15_6 = 999997.166
a15 - a7 + 1000000 x15_7 + W7 + v15_7 = 999997.319
a15 - a8 + 1000000 x15_8 + W8 + v15_8 = 999997.6
a15 - a9 + 1000000 x15_9 + W9 + v15_9 = 999997.662
a15 - a10 + 1000000 x15_10 + W10 + v15_10 = 999997.905
a15 - a11 + 1000000 x15_11 + W11 + v15_11 = 999998.306
a15 - a12 + 1000000 x15_12 + W12 + v15_12 = 999997.805
a15 - a13 + 1000000 x15_13 + W13 + v15_13 = 999997.31
a15 - a14 + 1000000 x15_14 + W14 + v15_14 = 999997.309
a15 - a16 + 1000000 x15_16 + W16 + v15_16 = 999998.274
a15 - a17 + 1000000 x15_17 + W17 + v15_17 = 999997.282

a15 - a18 + 1000000 x15_18 + W18 + v15_18 = 999994.843
a15 - a19 + 1000000 x15_19 + W19 + v15_19 = 999988.25

a16 - a2 + 1000000 x16_2 + W2 + v16_2 = 999985.267
a16 - a3 + 1000000 x16_3 + W3 + v16_3 = 999991.477
a16 - a4 + 1000000 x16_4 + W4 + v16_4 = 999994.649
a16 - a5 + 1000000 x16_5 + W5 + v16_5 = 999995.235
a16 - a6 + 1000000 x16_6 + W6 + v16_6 = 999997.067
a16 - a7 + 1000000 x16_7 + W7 + v16_7 = 999997.055
a16 - a8 + 1000000 x16_8 + W8 + v16_8 = 999997.13
a16 - a9 + 1000000 x16_9 + W9 + v16_9 = 999997.073
a16 - a10 + 1000000 x16_10 + W10 + v16_10 = 999997.272
a16 - a11 + 1000000 x16_11 + W11 + v16_11 = 999997.894
a16 - a12 + 1000000 x16_12 + W12 + v16_12 = 999997.087
a16 - a13 + 1000000 x16_13 + W13 + v16_13 = 999996.592
a16 - a14 + 1000000 x16_14 + W14 + v16_14 = 999996.59
a16 - a15 + 1000000 x16_15 + W15 + v16_15 = 999998.284
a16 - a17 + 1000000 x16_17 + W17 + v16_17 = 999998.039
a16 - a18 + 1000000 x16_18 + W18 + v16_18 = 999995.6
a16 - a19 + 1000000 x16_19 + W19 + v16_19 = 999989.007

a17 - a2 + 1000000 x17_2 + W2 + v17_2 = 999986.191
a17 - a3 + 1000000 x17_3 + W3 + v17_3 = 999992.401
a17 - a4 + 1000000 x17_4 + W4 + v17_4 = 999995.573
a17 - a5 + 1000000 x17_5 + W5 + v17_5 = 999996.16
a17 - a6 + 1000000 x17_6 + W6 + v17_6 = 999997.341
a17 - a7 + 1000000 x17_7 + W7 + v17_7 = 999997.203
a17 - a8 + 1000000 x17_8 + W8 + v17_8 = 999997.008
a17 - a9 + 1000000 x17_9 + W9 + v17_9 = 999996.867
a17 - a10 + 1000000 x17_10 + W10 + v17_10 = 999997.072
a17 - a11 + 1000000 x17_11 + W11 + v17_11 = 999997.081
a17 - a12 + 1000000 x17_12 + W12 + v17_12 = 999996.128
a17 - a13 + 1000000 x17_13 + W13 + v17_13 = 999995.633
a17 - a14 + 1000000 x17_14 + W14 + v17_14 = 999995.632
a17 - a15 + 1000000 x17_15 + W15 + v17_15 = 999997.325
a17 - a16 + 1000000 x17_16 + W16 + v17_16 = 999998.085
a17 - a18 + 1000000 x17_18 + W18 + v17_18 = 999996.524
a17 - a19 + 1000000 x17_19 + W19 + v17_19 = 999989.931

a18 - a2 + 1000000 x18_2 + W2 + v18_2 = 999988.305
a18 - a3 + 1000000 x18_3 + W3 + v18_3 = 999994.879
a18 - a4 + 1000000 x18_4 + W4 + v18_4 = 999995.009
a18 - a5 + 1000000 x18_5 + W5 + v18_5 = 999995.595
a18 - a6 + 1000000 x18_6 + W6 + v18_6 = 999994.909
a18 - a7 + 1000000 x18_7 + W7 + v18_7 = 999994.771
a18 - a8 + 1000000 x18_8 + W8 + v18_8 = 999994.576
a18 - a9 + 1000000 x18_9 + W9 + v18_9 = 999994.436
a18 - a10 + 1000000 x18_10 + W10 + v18_10 = 999994.641
a18 - a11 + 1000000 x18_11 + W11 + v18_11 = 999994.585
a18 - a12 + 1000000 x18_12 + W12 + v18_12 = 999993.663
a18 - a13 + 1000000 x18_13 + W13 + v18_13 = 999993.168
a18 - a14 + 1000000 x18_14 + W14 + v18_14 = 999993.167
a18 - a15 + 1000000 x18_15 + W15 + v18_15 = 999994.859
a18 - a16 + 1000000 x18_16 + W16 + v18_16 = 999995.619

a18 - a17 + 1000000 x18_17 + W17 + v18_17 = 999996.543
a18 - a19 + 1000000 x18_19 + W19 + v18_19 = 999992.408

a19 - a2 + 1000000 x19_2 + W2 + v19_2 = 999983.565
a19 - a3 + 1000000 x19_3 + W3 + v19_3 = 999998.147
a19 - a4 + 1000000 x19_4 + W4 + v19_4 = 999990.197
a19 - a5 + 1000000 x19_5 + W5 + v19_5 = 999989.547
a19 - a6 + 1000000 x19_6 + W6 + v19_6 = 999988.238
a19 - a7 + 1000000 x19_7 + W7 + v19_7 = 999988.1
a19 - a8 + 1000000 x19_8 + W8 + v19_8 = 999987.905
a19 - a9 + 1000000 x19_9 + W9 + v19_9 = 999987.765
a19 - a10 + 1000000 x19_10 + W10 + v19_10 = 999987.97
a19 - a11 + 1000000 x19_11 + W11 + v19_11 = 999987.914
a19 - a12 + 1000000 x19_12 + W12 + v19_12 = 999986.992
a19 - a13 + 1000000 x19_13 + W13 + v19_13 = 999986.497
a19 - a14 + 1000000 x19_14 + W14 + v19_14 = 999986.496
a19 - a15 + 1000000 x19_15 + W15 + v19_15 = 999988.189
a19 - a16 + 1000000 x19_16 + W16 + v19_16 = 999988.948
a19 - a17 + 1000000 x19_17 + W17 + v19_17 = 999989.873
a19 - a18 + 1000000 x19_18 + W18 + v19_18 = 999992.337

v1_2 >= 0
v1_3 >= 0
v1_4 >= 0
v1_5 >= 0
v1_6 >= 0
v1_7 >= 0
v1_8 >= 0
v1_9 >= 0
v1_10 >= 0
v1_11 >= 0
v1_12 >= 0
v1_13 >= 0
v1_14 >= 0
v1_15 >= 0
v1_16 >= 0
v1_17 >= 0
v1_18 >= 0
v1_19 >= 0

v2_3 >= 0
v2_4 >= 0
v2_5 >= 0
v2_6 >= 0
v2_7 >= 0
v2_8 >= 0
v2_9 >= 0
v2_10 >= 0
v2_11 >= 0
v2_12 >= 0
v2_13 >= 0
v2_14 >= 0
v2_15 >= 0
v2_16 >= 0
v2_17 >= 0
v2_18 >= 0
v2_19 >= 0

v3_2 >= 0
v3_4 >= 0
v3_5 >= 0
v3_6 >= 0
v3_7 >= 0
v3_8 >= 0
v3_9 >= 0
v3_10 >= 0
v3_11 >= 0
v3_12 >= 0
v3_13 >= 0
v3_14 >= 0
v3_15 >= 0
v3_16 >= 0
v3_17 >= 0
v3_18 >= 0
v3_19 >= 0

v4_2 >= 0
v4_3 >= 0
v4_5 >= 0
v4_6 >= 0
v4_7 >= 0
v4_8 >= 0
v4_9 >= 0
v4_10 >= 0
v4_11 >= 0
v4_12 >= 0
v4_13 >= 0
v4_14 >= 0
v4_15 >= 0
v4_16 >= 0
v4_17 >= 0
v4_18 >= 0

v4_19 >= 0

v5_2 >= 0
v5_3 >= 0
v5_4 >= 0
v5_6 >= 0
v5_7 >= 0
v5_8 >= 0
v5_9 >= 0
v5_10 >= 0
v5_11 >= 0
v5_12 >= 0
v5_13 >= 0
v5_14 >= 0
v5_15 >= 0
v5_16 >= 0
v5_17 >= 0
v5_18 >= 0
v5_19 >= 0

v6_2 >= 0
v6_3 >= 0
v6_4 >= 0
v6_5 >= 0
v6_7 >= 0
v6_8 >= 0
v6_9 >= 0
v6_10 >= 0
v6_11 >= 0
v6_12 >= 0
v6_13 >= 0
v6_14 >= 0
v6_15 >= 0
v6_16 >= 0
v6_17 >= 0
v6_18 >= 0
v6_19 >= 0

v7_2 >= 0
v7_3 >= 0
v7_4 >= 0
v7_5 >= 0
v7_6 >= 0
v7_8 >= 0
v7_9 >= 0
v7_10 >= 0
v7_11 >= 0
v7_12 >= 0
v7_13 >= 0
v7_14 >= 0
v7_15 >= 0
v7_16 >= 0
v7_17 >= 0
v7_18 >= 0
v7_19 >= 0

v8_2 >= 0
v8_3 >= 0
v8_4 >= 0
v8_5 >= 0
v8_6 >= 0
v8_7 >= 0
v8_9 >= 0
v8_10 >= 0
v8_11 >= 0
v8_12 >= 0
v8_13 >= 0
v8_14 >= 0
v8_15 >= 0
v8_16 >= 0
v8_17 >= 0
v8_18 >= 0
v8_19 >= 0

v9_2 >= 0
v9_3 >= 0
v9_4 >= 0
v9_5 >= 0
v9_6 >= 0
v9_7 >= 0
v9_8 >= 0
v9_10 >= 0
v9_11 >= 0
v9_12 >= 0
v9_13 >= 0
v9_14 >= 0
v9_15 >= 0
v9_16 >= 0
v9_17 >= 0
v9_18 >= 0
v9_19 >= 0

v10_2 >= 0
v10_3 >= 0
v10_4 >= 0
v10_5 >= 0
v10_6 >= 0
v10_7 >= 0
v10_8 >= 0
v10_9 >= 0
v10_11 >= 0
v10_12 >= 0
v10_13 >= 0
v10_14 >= 0
v10_15 >= 0
v10_16 >= 0
v10_17 >= 0
v10_18 >= 0
v10_19 >= 0

v11_2 >= 0
v11_3 >= 0
v11_4 >= 0
v11_5 >= 0

v11_6 >= 0
v11_7 >= 0
v11_8 >= 0
v11_9 >= 0
v11_10 >= 0
v11_12 >= 0
v11_13 >= 0
v11_14 >= 0
v11_15 >= 0
v11_16 >= 0
v11_17 >= 0
v11_18 >= 0
v11_19 >= 0

v12_2 >= 0
v12_3 >= 0
v12_4 >= 0
v12_5 >= 0
v12_6 >= 0
v12_7 >= 0
v12_8 >= 0
v12_9 >= 0
v12_10 >= 0
v12_11 >= 0
v12_13 >= 0
v12_14 >= 0
v12_15 >= 0
v12_16 >= 0
v12_17 >= 0
v12_18 >= 0
v12_19 >= 0

v13_2 >= 0
v13_3 >= 0
v13_4 >= 0
v13_5 >= 0
v13_6 >= 0
v13_7 >= 0
v13_8 >= 0
v13_9 >= 0
v13_10 >= 0
v13_11 >= 0
v13_12 >= 0
v13_14 >= 0
v13_15 >= 0
v13_16 >= 0
v13_17 >= 0
v13_18 >= 0
v13_19 >= 0

v14_2 >= 0
v14_3 >= 0
v14_4 >= 0
v14_5 >= 0
v14_6 >= 0
v14_7 >= 0
v14_8 >= 0
v14_9 >= 0
v14_10 >= 0
v14_11 >= 0
v14_12 >= 0
v14_13 >= 0
v14_15 >= 0
v14_16 >= 0
v14_17 >= 0
v14_18 >= 0
v14_19 >= 0

v15_2 >= 0
v15_3 >= 0
v15_4 >= 0
v15_5 >= 0
v15_6 >= 0
v15_7 >= 0
v15_8 >= 0
v15_9 >= 0
v15_10 >= 0
v15_11 >= 0
v15_12 >= 0
v15_13 >= 0
v15_14 >= 0
v15_16 >= 0
v15_17 >= 0
v15_18 >= 0
v15_19 >= 0

v16_2 >= 0
v16_3 >= 0
v16_4 >= 0
v16_5 >= 0
v16_6 >= 0
v16_7 >= 0
v16_8 >= 0
v16_9 >= 0
v16_10 >= 0
v16_11 >= 0
v16_12 >= 0
v16_13 >= 0
v16_14 >= 0
v16_15 >= 0
v16_17 >= 0
v16_18 >= 0
v16_19 >= 0

v17_2 >= 0
v17_3 >= 0
v17_4 >= 0
v17_5 >= 0
v17_6 >= 0
v17_7 >= 0
v17_8 >= 0
v17_9 >= 0
v17_10 >= 0
v17_11 >= 0

v17_12 >= 0
v17_13 >= 0
v17_14 >= 0
v17_15 >= 0
v17_16 >= 0
v17_18 >= 0
v17_19 >= 0

v18_2 >= 0
v18_3 >= 0
v18_4 >= 0
v18_5 >= 0
v18_6 >= 0
v18_7 >= 0
v18_8 >= 0
v18_9 >= 0
v18_10 >= 0
v18_11 >= 0
v18_12 >= 0
v18_13 >= 0
v18_14 >= 0
v18_15 >= 0
v18_16 >= 0
v18_17 >= 0
v18_19 >= 0

v19_2 >= 0
v19_3 >= 0
v19_4 >= 0
v19_5 >= 0
v19_6 >= 0
v19_7 >= 0
v19_8 >= 0
v19_9 >= 0
v19_10 >= 0
v19_11 >= 0
v19_12 >= 0
v19_13 >= 0
v19_14 >= 0
v19_15 >= 0
v19_16 >= 0
v19_17 >= 0
v19_18 >= 0

v1_2 + 2000000 x1_2 <= 2000000
v1_3 + 2000000 x1_3 <= 2000000
v1_4 + 2000000 x1_4 <= 2000000
v1_5 + 2000000 x1_5 <= 2000000
v1_6 + 2000000 x1_6 <= 2000000
v1_7 + 2000000 x1_7 <= 2000000
v1_8 + 2000000 x1_8 <= 2000000
v1_9 + 2000000 x1_9 <= 2000000
v1_10 + 2000000 x1_10 <= 2000000
v1_11 + 2000000 x1_11 <= 2000000
v1_12 + 2000000 x1_12 <= 2000000
v1_13 + 2000000 x1_13 <= 2000000
v1_14 + 2000000 x1_14 <= 2000000
v1_15 + 2000000 x1_15 <= 2000000
v1_16 + 2000000 x1_16 <= 2000000
v1_17 + 2000000 x1_17 <= 2000000
v1_18 + 2000000 x1_18 <= 2000000
v1_19 + 2000000 x1_19 <= 2000000

v2_3 + 2000000 x2_3 <= 2000000
v2_4 + 2000000 x2_4 <= 2000000
v2_5 + 2000000 x2_5 <= 2000000
v2_6 + 2000000 x2_6 <= 2000000
v2_7 + 2000000 x2_7 <= 2000000
v2_8 + 2000000 x2_8 <= 2000000
v2_9 + 2000000 x2_9 <= 2000000
v2_10 + 2000000 x2_10 <= 2000000
v2_11 + 2000000 x2_11 <= 2000000
v2_12 + 2000000 x2_12 <= 2000000
v2_13 + 2000000 x2_13 <= 2000000
v2_14 + 2000000 x2_14 <= 2000000
v2_15 + 2000000 x2_15 <= 2000000
v2_16 + 2000000 x2_16 <= 2000000
v2_17 + 2000000 x2_17 <= 2000000
v2_18 + 2000000 x2_18 <= 2000000
v2_19 + 2000000 x2_19 <= 2000000

v3_2 + 2000000 x3_2 <= 2000000
v3_4 + 2000000 x3_4 <= 2000000
v3_5 + 2000000 x3_5 <= 2000000
v3_6 + 2000000 x3_6 <= 2000000
v3_7 + 2000000 x3_7 <= 2000000
v3_8 + 2000000 x3_8 <= 2000000
v3_9 + 2000000 x3_9 <= 2000000
v3_10 + 2000000 x3_10 <= 2000000
v3_11 + 2000000 x3_11 <= 2000000
v3_12 + 2000000 x3_12 <= 2000000
v3_13 + 2000000 x3_13 <= 2000000
v3_14 + 2000000 x3_14 <= 2000000
v3_15 + 2000000 x3_15 <= 2000000
v3_16 + 2000000 x3_16 <= 2000000
v3_17 + 2000000 x3_17 <= 2000000
v3_18 + 2000000 x3_18 <= 2000000
v3_19 + 2000000 x3_19 <= 2000000

v4_2 + 2000000 x4_2 <= 2000000
v4_3 + 2000000 x4_3 <= 2000000
v4_5 + 2000000 x4_5 <= 2000000
v4_6 + 2000000 x4_6 <= 2000000
v4_7 + 2000000 x4_7 <= 2000000
v4_8 + 2000000 x4_8 <= 2000000
v4_9 + 2000000 x4_9 <= 2000000
v4_10 + 2000000 x4_10 <= 2000000
v4_11 + 2000000 x4_11 <= 2000000
v4_12 + 2000000 x4_12 <= 2000000
v4_13 + 2000000 x4_13 <= 2000000
v4_14 + 2000000 x4_14 <= 2000000
v4_15 + 2000000 x4_15 <= 2000000
v4_16 + 2000000 x4_16 <= 2000000
v4_17 + 2000000 x4_17 <= 2000000

v4_18 + 2000000 x4_18 <= 2000000
v4_19 + 2000000 x4_19 <= 2000000

v5_2 + 2000000 x5_2 <= 2000000
v5_3 + 2000000 x5_3 <= 2000000
v5_4 + 2000000 x5_4 <= 2000000
v5_6 + 2000000 x5_6 <= 2000000
v5_7 + 2000000 x5_7 <= 2000000
v5_8 + 2000000 x5_8 <= 2000000
v5_9 + 2000000 x5_9 <= 2000000
v5_10 + 2000000 x5_10 <= 2000000
v5_11 + 2000000 x5_11 <= 2000000
v5_12 + 2000000 x5_12 <= 2000000
v5_13 + 2000000 x5_13 <= 2000000
v5_14 + 2000000 x5_14 <= 2000000
v5_15 + 2000000 x5_15 <= 2000000
v5_16 + 2000000 x5_16 <= 2000000
v5_17 + 2000000 x5_17 <= 2000000
v5_18 + 2000000 x5_18 <= 2000000
v5_19 + 2000000 x5_19 <= 2000000

v6_2 + 2000000 x6_2 <= 2000000
v6_3 + 2000000 x6_3 <= 2000000
v6_4 + 2000000 x6_4 <= 2000000
v6_5 + 2000000 x6_5 <= 2000000
v6_7 + 2000000 x6_7 <= 2000000
v6_8 + 2000000 x6_8 <= 2000000
v6_9 + 2000000 x6_9 <= 2000000
v6_10 + 2000000 x6_10 <= 2000000
v6_11 + 2000000 x6_11 <= 2000000
v6_12 + 2000000 x6_12 <= 2000000
v6_13 + 2000000 x6_13 <= 2000000
v6_14 + 2000000 x6_14 <= 2000000
v6_15 + 2000000 x6_15 <= 2000000
v6_16 + 2000000 x6_16 <= 2000000
v6_17 + 2000000 x6_17 <= 2000000
v6_18 + 2000000 x6_18 <= 2000000
v6_19 + 2000000 x6_19 <= 2000000

v7_2 + 2000000 x7_2 <= 2000000
v7_3 + 2000000 x7_3 <= 2000000
v7_4 + 2000000 x7_4 <= 2000000
v7_5 + 2000000 x7_5 <= 2000000
v7_6 + 2000000 x7_6 <= 2000000
v7_8 + 2000000 x7_8 <= 2000000
v7_9 + 2000000 x7_9 <= 2000000
v7_10 + 2000000 x7_10 <= 2000000
v7_11 + 2000000 x7_11 <= 2000000
v7_12 + 2000000 x7_12 <= 2000000
v7_13 + 2000000 x7_13 <= 2000000
v7_14 + 2000000 x7_14 <= 2000000
v7_15 + 2000000 x7_15 <= 2000000
v7_16 + 2000000 x7_16 <= 2000000
v7_17 + 2000000 x7_17 <= 2000000
v7_18 + 2000000 x7_18 <= 2000000
v7_19 + 2000000 x7_19 <= 2000000

v8_2 + 2000000 x8_2 <= 2000000
v8_3 + 2000000 x8_3 <= 2000000
v8_4 + 2000000 x8_4 <= 2000000
v8_5 + 2000000 x8_5 <= 2000000
v8_6 + 2000000 x8_6 <= 2000000
v8_7 + 2000000 x8_7 <= 2000000
v8_9 + 2000000 x8_9 <= 2000000
v8_10 + 2000000 x8_10 <= 2000000
v8_11 + 2000000 x8_11 <= 2000000
v8_12 + 2000000 x8_12 <= 2000000
v8_13 + 2000000 x8_13 <= 2000000
v8_14 + 2000000 x8_14 <= 2000000
v8_15 + 2000000 x8_15 <= 2000000
v8_16 + 2000000 x8_16 <= 2000000
v8_17 + 2000000 x8_17 <= 2000000
v8_18 + 2000000 x8_18 <= 2000000
v8_19 + 2000000 x8_19 <= 2000000

v9_2 + 2000000 x9_2 <= 2000000
v9_3 + 2000000 x9_3 <= 2000000
v9_4 + 2000000 x9_4 <= 2000000
v9_5 + 2000000 x9_5 <= 2000000
v9_6 + 2000000 x9_6 <= 2000000
v9_7 + 2000000 x9_7 <= 2000000
v9_8 + 2000000 x9_8 <= 2000000
v9_10 + 2000000 x9_10 <= 2000000
v9_11 + 2000000 x9_11 <= 2000000
v9_12 + 2000000 x9_12 <= 2000000
v9_13 + 2000000 x9_13 <= 2000000
v9_14 + 2000000 x9_14 <= 2000000
v9_15 + 2000000 x9_15 <= 2000000
v9_16 + 2000000 x9_16 <= 2000000
v9_17 + 2000000 x9_17 <= 2000000
v9_18 + 2000000 x9_18 <= 2000000
v9_19 + 2000000 x9_19 <= 2000000

v10_2 + 2000000 x10_2 <= 2000000
v10_3 + 2000000 x10_3 <= 2000000
v10_4 + 2000000 x10_4 <= 2000000
v10_5 + 2000000 x10_5 <= 2000000
v10_6 + 2000000 x10_6 <= 2000000
v10_7 + 2000000 x10_7 <= 2000000
v10_8 + 2000000 x10_8 <= 2000000
v10_9 + 2000000 x10_9 <= 2000000
v10_11 + 2000000 x10_11 <= 2000000
v10_12 + 2000000 x10_12 <= 2000000
v10_13 + 2000000 x10_13 <= 2000000
v10_14 + 2000000 x10_14 <= 2000000
v10_15 + 2000000 x10_15 <= 2000000
v10_16 + 2000000 x10_16 <= 2000000
v10_17 + 2000000 x10_17 <= 2000000
v10_18 + 2000000 x10_18 <= 2000000
v10_19 + 2000000 x10_19 <= 2000000

v11_2 + 2000000 x11_2 <= 2000000
v11_3 + 2000000 x11_3 <= 2000000
v11_4 + 2000000 x11_4 <= 2000000

v11_5 + 2000000 x11_5 <= 2000000
v11_6 + 2000000 x11_6 <= 2000000
v11_7 + 2000000 x11_7 <= 2000000
v11_8 + 2000000 x11_8 <= 2000000
v11_9 + 2000000 x11_9 <= 2000000
v11_10 + 2000000 x11_10 <= 2000000
v11_12 + 2000000 x11_12 <= 2000000
v11_13 + 2000000 x11_13 <= 2000000
v11_14 + 2000000 x11_14 <= 2000000
v11_15 + 2000000 x11_15 <= 2000000
v11_16 + 2000000 x11_16 <= 2000000
v11_17 + 2000000 x11_17 <= 2000000
v11_18 + 2000000 x11_18 <= 2000000
v11_19 + 2000000 x11_19 <= 2000000

v12_2 + 2000000 x12_2 <= 2000000
v12_3 + 2000000 x12_3 <= 2000000
v12_4 + 2000000 x12_4 <= 2000000
v12_5 + 2000000 x12_5 <= 2000000
v12_6 + 2000000 x12_6 <= 2000000
v12_7 + 2000000 x12_7 <= 2000000
v12_8 + 2000000 x12_8 <= 2000000
v12_9 + 2000000 x12_9 <= 2000000
v12_10 + 2000000 x12_10 <= 2000000
v12_11 + 2000000 x12_11 <= 2000000
v12_13 + 2000000 x12_13 <= 2000000
v12_14 + 2000000 x12_14 <= 2000000
v12_15 + 2000000 x12_15 <= 2000000
v12_16 + 2000000 x12_16 <= 2000000
v12_17 + 2000000 x12_17 <= 2000000
v12_18 + 2000000 x12_18 <= 2000000
v12_19 + 2000000 x12_19 <= 2000000

v12_2 + 2000000 x12_2 <= 2000000
v12_3 + 2000000 x12_3 <= 2000000
v12_4 + 2000000 x12_4 <= 2000000
v12_5 + 2000000 x12_5 <= 2000000
v12_6 + 2000000 x12_6 <= 2000000
v12_7 + 2000000 x12_7 <= 2000000
v12_8 + 2000000 x12_8 <= 2000000
v12_9 + 2000000 x12_9 <= 2000000
v12_10 + 2000000 x12_10 <= 2000000
v12_11 + 2000000 x12_11 <= 2000000
v12_13 + 2000000 x12_13 <= 2000000
v12_14 + 2000000 x12_14 <= 2000000
v12_15 + 2000000 x12_15 <= 2000000
v12_16 + 2000000 x12_16 <= 2000000
v12_17 + 2000000 x12_17 <= 2000000
v12_18 + 2000000 x12_18 <= 2000000
v12_19 + 2000000 x12_19 <= 2000000

v13_2 + 2000000 x13_2 <= 2000000
v13_3 + 2000000 x13_3 <= 2000000
v13_4 + 2000000 x13_4 <= 2000000
v13_5 + 2000000 x13_5 <= 2000000
v13_6 + 2000000 x13_6 <= 2000000
v13_7 + 2000000 x13_7 <= 2000000
v13_8 + 2000000 x13_8 <= 2000000
v13_9 + 2000000 x13_9 <= 2000000
v13_10 + 2000000 x13_10 <= 2000000
v13_11 + 2000000 x13_11 <= 2000000
v13_12 + 2000000 x13_12 <= 2000000
v13_14 + 2000000 x13_14 <= 2000000
v13_15 + 2000000 x13_15 <= 2000000
v13_16 + 2000000 x13_16 <= 2000000
v13_17 + 2000000 x13_17 <= 2000000
v13_18 + 2000000 x13_18 <= 2000000
v13_19 + 2000000 x13_19 <= 2000000

v14_2 + 2000000 x14_2 <= 2000000
v14_3 + 2000000 x14_3 <= 2000000
v14_4 + 2000000 x14_4 <= 2000000
v14_5 + 2000000 x14_5 <= 2000000
v14_6 + 2000000 x14_6 <= 2000000
v14_7 + 2000000 x14_7 <= 2000000
v14_8 + 2000000 x14_8 <= 2000000
v14_9 + 2000000 x14_9 <= 2000000
v14_10 + 2000000 x14_10 <= 2000000
v14_11 + 2000000 x14_11 <= 2000000
v14_12 + 2000000 x14_12 <= 2000000
v14_13 + 2000000 x14_13 <= 2000000
v14_15 + 2000000 x14_15 <= 2000000
v14_16 + 2000000 x14_16 <= 2000000
v14_17 + 2000000 x14_17 <= 2000000
v14_18 + 2000000 x14_18 <= 2000000
v14_19 + 2000000 x14_19 <= 2000000

v15_2 + 2000000 x15_2 <= 2000000
v15_3 + 2000000 x15_3 <= 2000000
v15_4 + 2000000 x15_4 <= 2000000
v15_5 + 2000000 x15_5 <= 2000000
v15_6 + 2000000 x15_6 <= 2000000
v15_7 + 2000000 x15_7 <= 2000000
v15_8 + 2000000 x15_8 <= 2000000
v15_9 + 2000000 x15_9 <= 2000000
v15_10 + 2000000 x15_10 <= 2000000
v15_11 + 2000000 x15_11 <= 2000000
v15_12 + 2000000 x15_12 <= 2000000
v15_13 + 2000000 x15_13 <= 2000000
v15_14 + 2000000 x15_14 <= 2000000
v15_16 + 2000000 x15_16 <= 2000000
v15_17 + 2000000 x15_17 <= 2000000
v15_18 + 2000000 x15_18 <= 2000000
v15_19 + 2000000 x15_19 <= 2000000

v16_2 + 2000000 x16_2 <= 2000000
v16_3 + 2000000 x16_3 <= 2000000
v16_4 + 2000000 x16_4 <= 2000000
v16_5 + 2000000 x16_5 <= 2000000
v16_6 + 2000000 x16_6 <= 2000000
v16_7 + 2000000 x16_7 <= 2000000
v16_8 + 2000000 x16_8 <= 2000000
v16_9 + 2000000 x16_9 <= 2000000
v16_10 + 2000000 x16_10 <= 2000000

v16_11 + 2000000 x16_11 <= 2000000
v16_12 + 2000000 x16_12 <= 2000000
v16_13 + 2000000 x16_13 <= 2000000
v16_14 + 2000000 x16_14 <= 2000000
v16_15 + 2000000 x16_15 <= 2000000
v16_17 + 2000000 x16_17 <= 2000000
v16_18 + 2000000 x16_18 <= 2000000
v16_19 + 2000000 x16_19 <= 2000000

v17_2 + 2000000 x17_2 <= 2000000
v17_3 + 2000000 x17_3 <= 2000000
v17_4 + 2000000 x17_4 <= 2000000
v17_5 + 2000000 x17_5 <= 2000000
v17_6 + 2000000 x17_6 <= 2000000
v17_7 + 2000000 x17_7 <= 2000000
v17_8 + 2000000 x17_8 <= 2000000
v17_9 + 2000000 x17_9 <= 2000000
v17_10 + 2000000 x17_10 <= 2000000
v17_11 + 2000000 x17_11 <= 2000000
v17_12 + 2000000 x17_12 <= 2000000
v17_13 + 2000000 x17_13 <= 2000000
v17_14 + 2000000 x17_14 <= 2000000
v17_15 + 2000000 x17_15 <= 2000000
v17_16 + 2000000 x17_16 <= 2000000
v17_18 + 2000000 x17_18 <= 2000000
v17_19 + 2000000 x17_19 <= 2000000

v18_2 + 2000000 x18_2 <= 2000000
v18_3 + 2000000 x18_3 <= 2000000
v18_4 + 2000000 x18_4 <= 2000000
v18_5 + 2000000 x18_5 <= 2000000
v18_6 + 2000000 x18_6 <= 2000000
v18_7 + 2000000 x18_7 <= 2000000
v18_8 + 2000000 x18_8 <= 2000000
v18_9 + 2000000 x18_9 <= 2000000
v18_10 + 2000000 x18_10 <= 2000000
v18_11 + 2000000 x18_11 <= 2000000
v18_12 + 2000000 x18_12 <= 2000000
v18_13 + 2000000 x18_13 <= 2000000
v18_14 + 2000000 x18_14 <= 2000000
v18_15 + 2000000 x18_15 <= 2000000
v18_16 + 2000000 x18_16 <= 2000000
v18_17 + 2000000 x18_17 <= 2000000
v18_19 + 2000000 x18_19 <= 2000000

v19_2 + 2000000 x19_2 <= 2000000
v19_3 + 2000000 x19_3 <= 2000000
v19_4 + 2000000 x19_4 <= 2000000
v19_5 + 2000000 x19_5 <= 2000000
v19_6 + 2000000 x19_6 <= 2000000
v19_7 + 2000000 x19_7 <= 2000000
v19_8 + 2000000 x19_8 <= 2000000
v19_9 + 2000000 x19_9 <= 2000000
v19_10 + 2000000 x19_10 <= 2000000
v19_11 + 2000000 x19_11 <= 2000000
v19_12 + 2000000 x19_12 <= 2000000
v19_13 + 2000000 x19_13 <= 2000000
v19_14 + 2000000 x19_14 <= 2000000
v19_15 + 2000000 x19_15 <= 2000000
v19_16 + 2000000 x19_16 <= 2000000
v19_17 + 2000000 x19_17 <= 2000000
v19_18 + 2000000 x19_18 <= 2000000

Binaries

x1_2 x1_3 x1_4 x1_5 x1_6 x1_7 x1_8 x1_9 x1_10 x1_11 x1_12 x1_13 x1_14 x1_15 x1_16 x1_17 x1_18 x1_19
x2_1 x2_3 x2_4 x2_5 x2_6 x2_7 x2_8 x2_9 x2_10 x2_11 x2_12 x2_13 x2_14 x2_15 x2_16 x2_17 x2_18 x2_19
x3_2 x3_1 x3_4 x3_5 x3_6 x3_7 x3_8 x3_9 x3_10 x3_11 x3_12 x3_13 x3_14 x3_15 x3_16 x3_17 x3_18 x3_19
x4_2 x4_3 x4_1 x4_5 x4_6 x4_7 x4_8 x4_9 x4_10 x4_11 x4_12 x4_13 x4_14 x4_15 x4_16 x4_17 x4_18 x4_19
x5_2 x5_3 x5_4 x5_1 x5_6 x5_7 x5_8 x5_9 x5_10 x5_11 x5_12 x5_13 x5_14 x5_15 x5_16 x5_17 x5_18 x5_19
x6_2 x6_3 x6_4 x6_5 x6_1 x6_7 x6_8 x6_9 x6_10 x6_11 x6_12 x6_13 x6_14 x6_15 x6_16 x6_17 x6_18 x6_19
x7_2 x7_3 x7_4 x7_5 x7_6 x7_1 x7_8 x7_9 x7_10 x7_11 x7_12 x7_13 x7_14 x7_15 x7_16 x7_17 x7_18 x7_19
x8_2 x8_3 x8_4 x8_5 x8_6 x8_7 x8_1 x8_9 x8_10 x8_11 x8_12 x8_13 x8_14 x8_15 x8_16 x8_17 x8_18 x8_19
x9_2 x9_3 x9_4 x9_5 x9_6 x9_7 x9_8 x9_1 x9_10 x9_11 x9_12 x9_13 x9_14 x9_15 x9_16 x9_17 x9_18 x9_19
x10_2 x10_3 x10_4 x10_5 x10_6 x10_7 x10_8 x10_9 x10_1 x10_11 x10_12 x10_13 x10_14 x10_15 x10_16
x10_17 x10_18 x10_19
x11_2 x11_3 x11_4 x11_5 x11_6 x11_7 x11_8 x11_9 x11_10 x11_1 x11_12 x11_13 x11_14 x11_15 x11_16
x11_17 x11_18 x11_19
x12_2 x12_3 x12_4 x12_5 x12_6 x12_7 x12_8 x12_9 x12_10 x12_11 x12_1 x12_13 x12_14 x12_15 x12_16
x12_17 x12_18 x12_19
x13_2 x13_3 x13_4 x13_5 x13_6 x13_7 x13_8 x13_9 x13_10 x13_11 x13_12 x13_1 x13_14 x13_15 x13_16
x13_17 x13_18 x13_19
x14_2 x14_3 x14_4 x14_5 x14_6 x14_7 x14_8 x14_9 x14_10 x14_11 x14_12 x14_13 x14_1 x14_15 x14_16
x14_17 x14_18 x14_19
x15_2 x15_3 x15_4 x15_5 x15_6 x15_7 x15_8 x15_9 x15_10 x15_11 x15_12 x15_13 x15_14 x15_1 x15_16
x15_17 x15_18 x15_19

x16_2 x16_3 x16_4 x16_5 x16_6 x16_7 x16_8 x16_9 x16_10 x16_11 x16_12 x16_13 x16_14 x16_15 x16_1 x16_17 x16_18 x16_19
x17_2 x17_3 x17_4 x17_5 x17_6 x17_7 x17_8 x17_9 x17_10 x17_11 x17_12 x17_13 x17_14 x17_15 x17_16 x17_1 x17_18 x17_19
x18_2 x18_3 x18_4 x18_5 x18_6 x18_7 x18_8 x18_9 x18_10 x18_11 x18_12 x18_13 x18_14 x18_15 x18_16 x18_17 x18_1 x18_19
x19_2 x19_3 x19_4 x19_5 x19_6 x19_7 x19_8 x19_9 x19_10 x19_11 x19_12 x19_13 x19_14 x19_15 x19_16 x19_17 x19_18 x19_1

End

Příloha 13 Výstup z programu gurobi pro TSP s časovými okny

```
Academic license - for non-commercial use only
Set parameter timelimit to value 40

Gurobi Optimizer version 8.1.0 build v8.1.0rc1 (mac64)
Copyright (c) 2018, Gurobi Optimization, LLC

Read LP format model from file /Users/nehy/Documents/škola/DP/prakticka/TSPTW/okynka.lp
Reading time = 0.00 seconds
: 1423 rows, 723 columns, 4350 nonzeros
Optimize a model with 1423 rows, 723 columns and 4350 nonzeros
Variable types: 381 continuous, 342 integer (342 binary)
Coefficient statistics:
  Matrix range     [1e+00, 2e+06]
  Objective range  [2e-01, 1e+01]
  Bounds range     [1e+00, 1e+00]
  RHS range        [1e+00, 2e+06]
Presolve removed 1006 rows and 438 columns
Presolve time: 0.02s
Presolved: 417 rows, 285 columns, 3036 nonzeros
Variable types: 173 continuous, 112 integer (112 binary)

Root relaxation: objective 3.197650e+01, 76 iterations, 0.00 seconds

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0   31.97650    0   24          -   31.97650      -     -    0s
H    0     0                       95.2370000   31.97650  66.4%     -    0s
H    0     0                       94.2590000   31.97650  66.1%     -    0s
     0     0   38.13331    0   30   94.25900   38.13331  59.5%     -    0s
     0     0   43.87630    0   32   94.25900   43.87630  53.5%     -    0s
     0     0   46.63404    0   33   94.25900   46.63404  50.5%     -    0s
     0     0   46.88698    0   33   94.25900   46.88698  50.3%     -    0s
     0     0   46.89339    0   27   94.25900   46.89339  50.3%     -    0s
     0     0   48.12843    0   28   94.25900   48.12843  48.9%     -    0s
     0     0   48.65402    0   38   94.25900   48.65402  48.4%     -    0s
     0     0   48.65854    0   29   94.25900   48.65854  48.4%     -    0s
     0     0   48.65854    0   29   94.25900   48.65854  48.4%     -    0s
     0     0   50.50346    0   29   94.25900   50.50346  46.4%     -    0s
     0     0   50.60557    0   28   94.25900   50.60557  46.3%     -    0s
     0     0   62.20554    0   21   94.25900   62.20554  34.0%     -    0s
     0     0   69.72443    0   23   94.25900   69.72443  26.0%     -    0s
     0     0   69.78799    0   23   94.25900   69.78799  26.0%     -    0s
     0     0   69.79024    0   24   94.25900   69.79024  26.0%     -    0s
     0     0   70.45150    0   20   94.25900   70.45150  25.3%     -    0s
```

```
   0   0  70.45177   0  20  94.25900  70.45177  25.3%   -   0s
   0   0  70.65036   0  20  94.25900  70.65036  25.0%   -   0s
   0   0  70.68147   0  23  94.25900  70.68147  25.0%   -   0s
   0   0  70.68753   0  25  94.25900  70.68753  25.0%   -   0s
   0   0  70.92430   0  22  94.25900  70.92430  24.8%   -   0s
   0   0  70.92506   0  21  94.25900  70.92506  24.8%   -   0s
   0   0  71.08225   0  16  94.25900  71.08225  24.6%   -   0s
   0   0  71.08235   0  16  94.25900  71.08235  24.6%   -   0s
   0   0  75.40370   0  24  94.25900  75.40370  20.0%   -   0s
   0   0  75.52978   0  24  94.25900  75.52978  19.9%   -   0s
   0   0  75.68890   0  26  94.25900  75.68890  19.7%   -   0s
   0   0  75.68890   0  26  94.25900  75.68890  19.7%   -   0s
   0   0  83.50569   0  30  94.25900  83.50569  11.4%   -   0s
   0   0  83.82650   0  27  94.25900  83.82650  11.1%   -   0s
   0   0  83.82650   0  27  94.25900  83.82650  11.1%   -   0s
   0   0  83.83071   0  27  94.25900  83.83071  11.1%   -   0s
   0   0  83.83071   0  27  94.25900  83.83071  11.1%   -   0s
   0   0  83.83221   0  27  94.25900  83.83221  11.1%   -   0s
   0   0  83.83334   0  27  94.25900  83.83334  11.1%   -   0s
   0   0  83.85741   0  29  94.25900  83.85741  11.0%   -   0s
   0   0  83.85741   0  29  94.25900  83.85741  11.0%   -   0s
   0   2  83.85741   0  29  94.25900  83.85741  11.0%   -   0s
*  21   21          7    91.7760025  84.18456  8.27%  17.1   0s
* 103   52          12    91.6960025  85.09608  7.20%  14.4   0s
 5864   493  91.29789  31   6  91.69600  89.31290  2.60%  11.8   5s

Cutting planes:
  Gomory: 3
  Cover: 1
  Implied bound: 36
  Clique: 4
  MIR: 33
  Flow cover: 86
  Inf proof: 3
  Zero half: 2

Explored 12037 nodes (142441 simplex iterations) in 8.45 seconds
Thread count was 4 (of 4 available processors)

Solution count 4: 91.696 91.776 94.259 95.237
No other solutions better than 91.696

Optimal solution found (tolerance 1.00e-04)
Best objective 9.169600249514e+01, best bound 9.169600249514e+01, gap 0.0000%

Wrote result file '/Users/nehy/Documents/škola/DP/prakticka/TSPTW/okynka1.sol'
```