MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Bc. Jan Jeliga

Minimax v úlohách rozvrhování za nejistoty

Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: RNDr. Martin Branda, PhD.
Studijní program: Matematika
Studijní obor: Pravděpodobnost, matematická statistika a ekonometrie

Praha 2019

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V .... dne .... Podpis autora

Na tomto místě bych rád poděkoval panu RNDr. Martinu Brandovi, Ph.D. za vedení, konzultace a nesmírně cennou pomoc při psaní mé diplomové práce.

Název práce: Minimax v úlohách rozvrhování za nejistoty

Autor: Bc. Jan Jeliga

Katedra: Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: RNDr. Martin Branda, PhD., Katedra pravděpodobnosti a matematické statistiky

Abstrakt: V práci se zabýváme rozvrhovací úlohou pro úkoly s daným časem začátku a konce práce (FIS) při možnosti náhodného zpoždění konce práce. Nejprve představujeme základní deterministické úlohy FIS a možnosti jejich řešení. Dále zavádíme koncept minimaxu a představujeme dvě známé a jednu novou úlohu FIS za nejistoty, kdy jsou náhodná zpoždění úkolů uvažována z vybrané množiny rozdělení. Dále se věnujeme řešení dříve představených FIS úloh pro pět konkrétních množin pravděpodobnostních rozdělení. Uvádíme jak dříve dosažené, tak původní výsledky. Práci zakončuje shrnutí numerické studie dvou úloh. Nejprve zkoumáme možnost aplikace Lagrangeovské relaxace na první z uvedených úloh. Dále zkoumáme kvalitu aproximace umožňující řešení druhé úlohy jako LP.

Klíčová slova: optimalizace, rozvrhovací úlohy, minimax, robustifikace

Title: Minimax in scheduling problems under uncertainty

Author: Bc. Jan Jeliga

Department: Department of Probability and Mathematical Statistics

Supervisor: RNDr. Martin Branda, PhD., Department of Probability and Mathematical Statistics

Abstract: In this work, we deal with fixed interval scheduling problems with the possibility of random delay of the end of the tasks (FIS). First, we present the basic deterministic FIS problems and ways to solve them. Next, we introduce the concept of minimax and present two well-known and one new FIS problem under uncertainty, when random task delays are considered to belong to a certain uncertainty set. Next, we deal with the solution of previously presented FIS problems for five chosen uncertainty sets. We present both previously achieved and original results. The work concludes with a summary of a numerical study of two problems. First, we explore the possibility of Lagrange relaxation application to the first presented problem. Next we explore the quality of approximation allowing to solve the later of problems as LP.

Keywords: optimisation, fixed interval scheduling, minimax, robustification

Obsah

Úvod

V práci se budeme zabývat řešením optimalizačních rozvrhovacích úloh, kdy se snažíme danému počtu strojů přiřadit určitým optimálním způsobem sadu úkolů s fixními časy začátku a konce práce (FIS). V praxi však můžeme očekávat existenci možnosti zpoždění práce na jednotlivých úkolech, čímž může následně dojít k porušení rozvrhu. Další stupeň nejistoty pak spočívá v nedokonalé znalosti rozdělení onoho náhodného zpoždění, a proto uvažujeme místo jednoho konkrétního rozdělení nějakou danou množinu a snažíme se nejlepším možným způsobem pojistit proti nejhorší možnosti. Dostáváme se tak k minimaxové formulaci, čímž se úloha stává značně obtížnou. Navazujeme především na práce Branda a kol. (2016), Branda a Hájek (2017) a Branda (2018).

V první části představíme klasické deterministické FIS úlohy. Uvádíme základní formulace pro různé volby účelové funkce a jejich možné modifikace. Dále uvádíme možnosti řešení těchto úloh a příklady praktické aplikace.

V druhé části se zaměřujeme na minimaxové FIS úlohy za nejistoty. Obecně zavádíme koncept minimaxu a dáváme ho do vztahu s dále prezentovanými optimalizačními úlohami. Rozebíráme tři úlohy FIS – minimalizace střední hodnoty počtu překrytí úkolů v rozvrhu, maximalizace spolehlivosti rozvrhu a novou FIS úlohu za nejistoty, kde minimalizujeme střední hodnotu doby vzájemného překrytí práce na úkolech. Pro úlohy uvádíme jejich různé formulace a modifikace.

Ve třetí části se věnujeme konkrétním případům robustifikace FIS úloh za nejistoty, tedy řešení FIS úloh představených v druhé části pro dané množiny pravděpodobnostních rozdělení. Představíme koncepty zatěžkávacího rozdělení, *box-uncertainty*, množiny nejistoty ve tvaru elipsoidu, směsného rozdělení a definice pomocí hodnot daných momentů. K úlohám navrhujeme možnosti jejich řešení. Většina těchto konceptů zatím nebyla v nám známé literatuře v souvislosti s FIS úlohou za nejistoty nikde prezentována a v této práci jsou uvedeny poprvé.

Práci uzavírá část věnovaná numerické studii, kde se zabýváme dvěma úlohami. Nejprve pomocí metody Lagrangeovské relaxace řešíme úlohu minimalizace střední hodnoty počtu překrytí po sobě jdoucích prací při definici množiny pravděpodobnostních rozdělení pomocí směsného rozdělení. Prezentujeme řešenou úlohu, použitý algoritmus a dosažené výsledky, které srovnáváme s výsledky řešení problému formulovaného jako MIP. Dále zkoumáme kvalitu výsledků aproximační metody pro úlohu minimalizace střední hodnoty doby překrytí při znalosti středních hodnot jednotlivých zpoždění.

1. Rozvrhovací úlohy

V optimalizačních rozvrhovacích úlohách se obecně snažíme vytvořit rozvrh daných úkolů (jobs) pro stroje (machines) tak, aby byl v jistém smyslu optimální. Úloha tohoto typu je aplikovatelná na širokou škálu praktických problémů a existuje tak velké množství jejích modifikací. Může se jednat například o úlohu minimalizace množství potřebného personálu na pracovišti, řízení letového provozu na letišti nebo maximalizace zisku z výroby při pevném množství strojů v továrně. V kontextu této práce se budeme držet výše zmíněné terminologie.

V této práci se budeme věnovat typu úloh, kdy jsou pevně dány časy začátku práce na jednotlivých úkolech a jejich doby trvání. Jedná se o rozvrhovací úlohy s pevnými intervaly prací (v angličtině Fixed Interval Scheduling – FIS – problem). Tato úloha může mít více tvarů, potažmo praktických motivací, z nichž zde některé uvádíme:

- pro dané množství strojů se snažíme určit rozvrh úkolů maximalizující zisk (podrobně například v práci Kroon a kol. (1995)),
- minimalizujeme počet potřebných strojů pro vykonání všech úkolů, což odpovídá minimalizaci nákladů – tzv. taktická úloha FIS (TFIS) (podrobně například v práci Kroon a kol. (1997)),
- minimalizujeme pravděpodobnost porušení rozvrhu, nebo jiným způsobem optimalizujeme rozvrh při zahrnutí nejistoty (více například v práci Branda a kol. (2016) nebo Branda a Hájek (2017)).

V této kapitole stručně představíme první dva typy úloh, základní tvary, některé modifikace jejich zadání, metody řešení a možné aplikace. Řešení úlohy FIS při zahrnutí nejistoty se budeme věnovat v následujících kapitolách.

1.1 FIS problém s pevně daným počtem strojů

Při řešení této úlohy se snažíme s daným počtem strojů sestavit rozvrh tak, abychom, obecně řečeno, maximalizovali zisk z vykonané práce. To v praxi znamená z množiny všech úkolů co nejlepším způsobem vybrat určitou podmnožinu a úkoly z ní přiřadit dostupným strojům.

Na tomto místě zavedeme značení, kterého se budeme držet i ve zbytku práce, uvedeme některá tvrzení a matematickou formulaci úlohy, jak je můžeme najít například v práci Kroon a kol. (1995).

Předpokládejme, že máme rozvrhnout $J \in \mathbb{N}$ úkolů a pro $j = 1, \ldots, J$ je každý z nich charakterizován vektorem tří prvků (s_j, f_j, q_j), kde $s_j > 0$ a $f_j > s_j$ značí fixovaný čas začátku a ukončení práce na úkolu[1] a $q_j > 0$ značí jeho prioritu (můžeme chápat i jako zisk z vykonání daného úkolu). Pro jednoduchost předpokládáme, že úkoly jsou seřazeny podle času začátku, od nejdřívějšího k nejpozdějšímu[2]. Dále předpokládáme, že úkoly po započetí jejich vykonávání není

[1] Alternativně můžeme místo f_j uvažovat předpokládaný čas $(f_j - s_j) > 0$ trvání práce na úkolu.

[2] Při shodě zvolíme pořadí libovolně.

možné přerušit (non–preemptive jobs). Předpokládáme, že úkoly jsou rozděleny do $A \in \mathbb{N}$ disjunktních tříd a každý z nich náleží do nějaké třídy a_j. Ty můžou být pro různé úkoly identické.

K dispozici máme M strojů, které náleží do $C \in \mathbb{N}$ disjunktních tříd a počet strojů náležející do dané třídy $c = 1, \ldots, C$ značíme jako M_c. Pro každou třídu strojů c značíme množinu tříd úkolů, které je schopna vykonávat jako A_c a pro každý úkol j značíme C_j množinu tříd strojů, které jsou schopny tento úkol vykonat. Dále zavedeme množinu $S = \{s_j : j = 1, \ldots, J\}$ startovních časů všech úkolů.

Matematická formulace úlohy pak vypadá následovně:

$$\begin{array}{llrll} \max & \sum_{j=1}^{J} \sum_{c \in C_j} q_j x_{jc}, & & & \\ \text{s.t.} & \sum_{\{j: a_j \in A_c \wedge s_j \leq s < f_j\}} & x_{jc} & \leq & M_c, \quad c = 1, \ldots, C, \ s \in S, \\ & \sum_{c \in C_j} & x_{jc} & \leq & 1, \quad j = 1, \ldots, J, \\ & & x_{jc} & \in & \{0,1\}, \quad j = 1, \ldots, J, \ c = 1, \ldots, C, \end{array} \tag{1.1}$$

kde x_{jc} je binární rozhodovací proměnná určující přiřazení úkolu j třídě strojů c. První sada nerovností zaručuje, že v žádném okamžiku nebude dané třídě strojů přiřazeno více úkolů, než je počet strojů v ní. Druhá sada nerovností potom zaručuje přiřazení každého úkolu nejvýše jedné třídě strojů. Optimální řešení úlohy 1.1 nám dává možnost sestavení přípustného rozvrhu pro všechny úkoly $j \in \{j | \exists\ c : x_{jc} = 1\}$, což je přímým důsledkem následujícího *Lemma* 1 aplikovaného na jednotlivé třídy strojů.

Lemma 1. *Předpokládejme, že* $C = A = 1$. *Přípustný rozvrh pro všechny úkoly existuje právě tehdy, když*

$$\forall s \in S : \sum_{j=1}^{J} \mathbb{1}\left[s_j \leq s < f_j\right] \leq M.$$

Důkaz můžeme najít ve výše zmíněné práci Kroon a kol. (1995).

1.1.1 Řešitelnost a možnosti řešení úlohy

Pro $C = 1$ jsou známy algoritmy pro nalezení optimálního řešení pracující v polynomiálním čase. Úloha může být ekvivalentně definována například jako problém nalezení toku grafem s minimální cenou. Pro tuto formulaci definujeme orientovaný graf $G = (V, E)$ který je konstruován následovně:

- množina $V = \{v_1, \ldots, v_{J+1}\}$ obsahuje $J + 1$ vrcholů, kde $v_1, \ldots, v_J$ reprezentují všechny úkoly a v_{J+1} je stok reprezentující ukončení veškeré práce, u kterého pro úplnost uvažujeme $s_{J+1} = f_{J+1} = f_J + 1$,

- množina E obsahuje pro $j = 1, \ldots, J$ hrany (v_j, v_{j+1}) s kapacitou $c = M$ a cenou $q = 0$ a hrany (v_j, v_k) s kapacitou 1 a cenou $-q_j$, kde $k = \min\{l : f_j \leq s_l\}$, celkem tedy množina E obsahuje $2J$ hran,

- požadujeme tok velikosti M z v_1 do v_{J+1}.

Podrobnější rozbor ekvivalence obou úloh a odkaz na popis algoritmu najdeme například v práci Bouzina a Emmons (1996).

Matematická formulace úlohy pomocí lineárního programování je potom následující:

$$\begin{array}{rrcll} \min & \sum_{e \in E} q_e x_e, & & & \\ \text{s.t.} & x_e & \leq & c_e, & e \in E, \\ & 0 & \leq & x_e, & e \in E. \end{array} \tag{1.2}$$

V práci Kroon a Kolen (1991) autoři ukazují, že výše definovaný problém je, kromě některých triviálních případů, pro $C > 1$ NP–těžký. Z toho plyne, že známý algoritmus vedoucí k nalezení optimálního řešení je výpočetně značně náročný a pro větší úlohy v zásadě nepoužitelný. V praxi je ovšem nutné řešit tuto úlohu rychle za cenu dosažení ne nutně optimálního výsledku, a proto se musíme spokojit s aproximujícími algoritmy.

Jednou z možností je formulace úlohy jako nalezení toku grafem s minimální cenou s přidanými omezeními. Následně je možné využití Lagrangeovské relaxace a totální unimodularity výsledné matice omezení, která umožňuje relaxaci celočíselnosti rozhodovacích proměnných a použití algoritmů pracujících v polynomiálním čase.

Další možností je využití nějakého heuristického algoritmu. Jeden příklad můžeme najít například v práci Kroon a kol. (1995), kde je popsána konstrukce odpovídajícího grafu a algoritmu.

1.1.2 Aplikace

Na tomto místě pro představu uvedeme některé možnosti aplikace úlohy FIS s daným počtem strojů.

- Přiřazení personálu při údržbě přilétajících letadel na letišti. S omezeným počtem údržbářů se snažíme provést pokud možno všechny opravy nebo maximalizovat jejich celkovou prioritu. Kvalifikovanost jednotlivých údržbářů může být různá, což odpovídá více třídám strojů v úloze. Úloze se reálně věnuje práce Dijkstra a kol. (1991). Možnou modifikací může být například přidání nejistoty v podobě náhodného zpoždění příletu letadel nebo ukončení údržby.

- Rozvrh operačních sálů v nemocnici. Na sálech se snažíme provést co nejvíce plánovaných operací s tím, že jednotlivé případy mohou mít různou prioritu a každý operační sál je vhodný pouze pro určité typy operací.

Úlohu můžeme podle potřeby různě modifikovat přidáním dalších podmínek, jako například:

- stanovení maximálního počtu druhů úkolů, které smí stroj vykonat,

- stanovení maximální doby po kterou smí stroj pracovat,

- zavedení časových intervalů, kdy jsou jednotlivé stroje k dispozici.

K řešení modifikovaných úloh je nutné přistupovat jiným způsobem než k řešení původní úlohy, tomu se ovšem v naší práci věnovat nebudeme. Některé z těchto případů jsou popsány v dostupné literatuře.

1.2 Taktický FIS problém

Další situace, se kterou se při rozvrhování můžeme v praxi často setkat, je nutnost sestavení optimálního rozvrhu pro všechny úkoly při minimalizaci nákladů na jejich vykonání. Je-li cena vykonání úkolu na všech strojích stejná, což je nejčastější případ, a za užívaní stroje platíme fixní částku, je minimalizace nákladů ekvivalentní minimalizaci počtu potřebných strojů. V tom případě jde o tzv. taktický FIS problém (TFIS), který na tomto místě popíšeme.

Předpoklady, týkající se úkolů, které jsme uvažovali v předchozí části, budeme uvažovat i zde ale q_{jc} má v tomto kontextu význam ceny vykonání práce j na stroji třídy c a pokud nebude uvedeno jinak, předpokládáme, že je tato hodnota napříč všemi třídami strojů vždy konstantní. U předpokladů o úkolech vynecháme předpoklad týkající se celkového počtu strojů a počtů strojů v jednotlivých třídách. Dále předpokládáme, že žádná třída strojů není dominována jinou, neboli $\nexists\ c, \tilde{c} : \ A_c \subset A_{\tilde{c}}$.

Matematická formulace úlohy, jak ji můžeme najít v práci Kroon a kol. (1997), pak vypadá následovně:

$$
\begin{array}{lrcl}
\min & \displaystyle\sum_{c=1}^{C} Y_c, & & \\
\text{s.t.} & \displaystyle\sum_{\{j: a_j \in A_c \wedge s_j \le s < f_j\}} x_{jc} & \le & Y_c, \quad c = 1, \ldots, C, \ s \in S \\
& \displaystyle\sum_{c \in C_j} x_{jc} & = & 1, \quad j = 1, \ldots, J, \\
& x_{jc} & \in & \{0,1\}, \quad j = 1, \ldots, J, \, c = 1, \ldots, C, \\
& Y_c & \in & \mathbb{N},
\end{array}
\tag{1.3}
$$

kde x_{jc} má stejnou interpretaci jako u FIS s pevně daným počtem strojů a nová celočíselná rozhodovací proměnná Y_c značí, podle první nerovnosti, nejvyšší počet úkolů, který v libovolný okamžik přiřazujeme třídě strojů c, což je v důsledku optimální potřebný počet strojů třídy c při minimalizaci celkového počtu strojů. Druhá nerovnost potom zaručuje, že bude každý úkol přiřazen některé třídě strojů, která je schopná ho vykonat.

Je snadné nahlédnout, že optimální řešení úlohy splňuje lemma 1 a dává nám tak možnost sestavit přípustný rozvrh. Při sestavování samotného rozvrhu potom stačí pro jednotlivé třídy strojů vyřešit nejjednodušší případ úlohy FIS s pevně daným počtem strojů.

1.2.1 Řešitelnost a možnosti řešení úlohy

Pro $C = 1$ je úloha velice snadno řešitelná. Z lemmatu 1 plyne, že pro sestavení přípustného rozvrhu je nejnižší možný, tedy optimální, počet použitých strojů roven nejvyššímu počtu úkolů, na kterých je potřeba pracovat současně. Sestavení rozvrhu je potom triviální.

Pro $C > 1$ se úloha stává značně komplikovanou. V práci Kroon a kol. (1997) autoři ukazují, že se jedná o NP–těžkou úlohu. Autoři v textu popisují algoritmus založený na aproximaci horních a dolních mezí pro Y_c a následném použití metody *branch–and–bound*. Dále pak ukazují, že úlohu můžeme alternativně definovat jako nalezení toku orientovaným grafem s přídavnými podmínkami. Pro tuto formulaci definujeme orientovaný graf $G = (V, E)$ který je konstruován následovně:

- pro každé $c = 1, \ldots, C$ a $j \in A_c$ definujeme chronologicky uspořádanou množinu $T_c = \{t_{c,1}, \ldots, t_{c,n_c}\}$ všech s_j a f_j,
- množina vrcholů V obsahuje pro každou množinu T_c vrcholy $v_{c,1}, \ldots, v_{c,n_c}$ reprezentující všechny její prvky, vrchol v_0 reprezentující zdroj a vrchol v_s reprezentující stok,
- množina hran E obsahuje pro $c = 1, \ldots, C$ $i = 1, \ldots n_c$, hrany $(v_0, v_{c,1})$, (v_{c,n_c}, v_s) a $(v_{c,i-1}, v_{c,i})$ s neomezenou kapacitou, dále pak obsahuje hrany reprezentující úkoly, kdy pro každé $c = 1, \ldots, C$ a $j \in A_c$ spojíme hranou s kapacitou 1 vrcholy odpovídající času začátku a konce úkolu, tedy pokud $t_{c,a} = s_j$ a $t_{c,b} = f_j$, pak přidáme hranu $(v_{c,a}, v_{c,b})$.
- pro $j = 1, \ldots, J$ definujeme množinu $E_j \subseteq E$ všech hran reprezentujících úkol j,
- požadujeme minimální tok z v_0 do v_s za splnění podmínky, že pro $\forall j$ je součet toků přes hrany v E_j roven 1 a podmínky nezáporných celočíselných toků.

1.2.2 Aplikace

Na tomto místě pro představu uvádíme některé možnosti aplikace úlohy TFIS.

- Řešení úlohy minimalizace počtu potřebného personálu v provozu a sestavení rozvrhu práce. Tento koncept je široce použitelný napříč mnoha odvětvími.
- V případě, že máme v současnosti k dispozici pevně daný počet pracovníků/strojů (řešíme tedy předchozí úlohu), můžeme řešení úlohy TFIS na fiktivních scénářích požadavků použít pro získání představy o možné nutnosti změn v provozu.

Úlohu můžeme podle potřeby různě modifikovat zavedením dalších požadavků, podobně jako tomu bylo v předchozím případě.

2. FIS za nejistoty – minimaxové formulace

V této kapitole se budeme věnovat rozšíření optimalizačních úloh o zdroj nejistoty v účelové funkci při neúplné znalosti relevantního pravděpodobnostního rozdělení, konkrétně popíšeme tzv. *minimax* přístup. Obecnou teorii následně zavedeme v kontextu FIS úloh s daným počtem strojů, kde primárním zdrojem nejistoty bude možnost náhodného zpoždění konce prací na úkolech.

2.1 Zavedení a obecné výsledky

V optimalizačních úlohách se často setkáváme s nejistotou, která pramení z přirozené povahy řešeného problému, může se jednat například o náhodnost vývoje cen akcií na burze či počtu prodaných kusů zboží. Typickým a dobře prozkoumaným příkladem takové optimalizační úlohy je například optimalizace portfolia, kde minimalizujeme *CVaR* (více viz Rockafellar a kol. (2000)), nebo tzv. *Úloha prodavače novin*, kde hledáme optimální počet výtisků novin k prodeji při náhodné poptávce (více viz Gallego a Moon (1993)).

Ve většině případů při řešení takové optimalizační úlohy předpokládáme úplnou znalost rozdělení P náhodné složky a optimalizační úloha má pro nějakou známou funkci $F(x, P)$ obecně tvar:

$$\min_{x} F(x, P) \quad x \in \mathcal{X}, \tag{2.1}$$

často například $F(x, P) = \mathsf{E}_P f(x, \omega)$.

Tento předpoklad však nemusí být zcela legitimní. Špatná volba pravděpodobnostního rozdělení pomocí expertního odhadu nebo při využití empirických dat může vést ke špatným a v důsledku nákladným řešením.

Z tohoto důvodu oslabíme výše uvedený předpoklad a dále budeme předpokládat, že $P \in \mathcal{P}$, kde $\mathcal{P}$ je nějaká množina pravděpodobnostních rozdělení. To nás vede k reformulaci úlohy 2.1, jak ji můžeme vidět například v práci Dupačová (2011), úloha má potom tvar:

$$\min_{x} \max_{P} F(x, P) \quad x \in \mathcal{X},\, P \in \mathcal{P}\,. \tag{2.2}$$

Optimálním řešením 2.2 je potom situace, kdy se nejlepším možným způsobem zajistíme proti v určitém smyslu nejhoršímu případu vzhledem k dané množině $\mathcal{P}$. Může se jednat například o minimalizaci střední hodnoty ztráty nebo pravděpodobnosti, že ztráta dosáhne určité míry vzhledem k celé množině $\mathcal{P}$. Množina $\mathcal{X}$ dále v této kapitole značí obecnou množinu všech přípustných x relevantních pro danou úlohu.

Pro obecnou úlohu typu 2.2 neexistuje žádný jasný postup pro získání optimálního řešení, ten musí být pro konkrétní volbu $F(x, P)$ a $\mathcal{X}$ vždy přizpůsoben příslušné množině $\mathcal{P}$. Její volba je pro kvalitu získaného řešení zásadní a je žádoucí mít řešení robustní proti malým změnám v P, neboť v praxi se snadno můžeme setkat s případem, že skutečné $P \notin \mathcal{P}$. Proto je vždy na místě provádět analýzu citlivosti optimálního řešení vzhledem k perturbacím $\mathcal{P}$. Pokud je

$\mathcal{P}$ definována například pomocí střední hodnoty μ, je vhodné zkoumat změnu optimálního řešení při hodnotách blízkých μ.

Při konstrukci $\mathcal{P}$ je dobré využít veškeré dostupné informace, které o rozdělení P máme, je ale zároveň potřeba zavést $\mathcal{P}$ tak, aby byla úloha výpočetně únosná. Můžeme využít odhady momentů, kvalitativní vlastnosti týkající se například symetrie či blízkosti některého zvoleného pravděpodobnostního rozdělení nebo expertně zvolit konečnou množinu rozdělení. Volba $\mathcal{P}$ se odvíjí i od expertně zvoleného nosiče vybraných rozdělení, které závisí na řešeném problému (pro náhodné zpoždění nemusí být vhodné uvažovat normální rozdělení atd.).

Na tomto místě uvedeme některé konkrétní případy $\mathcal{P}$, jak je můžeme najít například v práci Dupačová (2011).

- $\mathcal{P}$ je množina rozdělení majících dané zobecněné momenty, tedy:

  $$\mathcal{P} = \{P : \ \mathsf{E}_P\, g_k(\omega) = y_k, k = 1, \ldots, K\},$$

  Kde g_k jsou nějaké zvolené funkce a y_k hodnoty zobecněných momentů (například empirické odhady). Ve většině případů se jedná o první a druhé momenty.

- $\mathcal{P}$ je jako v předchozím případě, avšak místo rovností momentů uvažujeme v některých případech nerovnosti.

- $\mathcal{P}$ je množina diskrétních rozdělení na zvolené konečné množině Ω. Rozdělení jsou tedy definována pravděpodobnostmi scénářů. Při definici $\mathcal{P}$ může být využita například apriorní informace o jejich řazení.

- $\mathcal{P}$ je definována jako okolí nějakého pravděpodobnostního rozdělení P_0, tedy:

  $$\mathcal{P} = \{P : \ d(P, P_0) \leq \epsilon\},\ \epsilon > 0,$$

  kde d je nějaká vhodně zvolená vzdálenost. Konkrétně třeba *Wassersteinova metrika*, optimalizace portfolia je pro tento případ rozebrána v práci Pflug a kol. (2012).

- $\mathcal{P}$ je tvořena konečnou množinou zvolených rozdělení, nebo jejich konvexním obalem.

2.2 Vybrané úlohy FIS s minimaxovou formulací

Nyní popíšeme FIS problém s pevným počtem identických strojů a náhodným zpožděním konce prací, jak jej můžeme vidět například v práci Branda (2018). Nejprve zavedeme potřebné definice a značení a následně popíšeme problém a možnosti jeho řešení.

Definujeme sdružené zpoždění všech prací na úkolech jako náhodnou veličinu $D(\omega)$ na pravděpodobnostním prostoru $(\Omega, \mathcal{A}, P)$. Jednotlivá marginální rozdělení zpoždění úkolů $j = 1, \ldots, J$ značíme $D_j(\omega)$ a s nimi související náhodný čas konce práce na úkolu značíme $f_j(\omega) = f_j + D_j(\omega)$.

Dále obecně nepředpokládáme vzájemnou nezávislost jednotlivých zpoždění, což vychází z praxe, neboť úkoly, které je nutné vykonávat ve stejný čas a jež jsou nějakým způsobem závislé například na počasí, mohou mít kladně korelovaná zpoždění konce práce. Tento aspekt dále rozvádíme v části 2.2.3.

V praxi však rozdělení P neznáme přesně a jeho nevhodnou volbou bychom mohli dojít k velmi nevhodným řešením, předpokládáme tedy, že P náleží do nějaké známé množiny pravděpodobnostních rozdělení $\mathcal{P}$. Z toho plyne nevhodnost aplikace klasických postupů optimalizace za nejistoty a využijeme tak výše zmíněného minimaxového přístupu.

U FIS úlohy s náhodným zpožděním konce úkolů můžeme optimalizovat například střední hodnotu doby nebo počtu případů, kdy by měl jeden stroj pracovat na více úkolech, tuto situaci nazýváme *překryv*. Dále můžeme maximalizovat pravděpodobnost, že k žádnému překryvu nedojde, tedy že rozvrh zůstane přípustným.

Každý z těchto přístupů může být vhodný v jiné situaci, záleží na konkrétních podmínkách řešeného problému. Pokud lze provádět outsourcing, může se jako vhodný jevit přístup minimalizující střední hodnotu doby trvání překryvů nebo jejich počet. Toto může nastat například v provozu nestíhajícím výrobu, který si může další výrobu zboží zaplatit jinde. V případě, že jsou stroje unikátní či pracovníci vysoce kvalifikovaní a je nutné vykonat všechny úkoly včas, bude nejspíše vhodnější přístup minimalizující pravděpodobnost výskytu překryvu.

2.2.1 Střední hodnota počtu překryvů

Na tomto místě uvedeme formulaci úlohy FIS s pevným počtem identických strojů a náhodným zpožděním konce úkolů, kdy minimalizujeme střední hodnotu počtu překryvů a následně její možnou reformulaci jako robustní úlohu barvení grafu. Tyto výsledky jsou pro pevně zvolené rozdělení zpoždění konce úkolů uvedeny v práci Branda a kol. (2016). Na tomto místě uvádíme následující minimaxovou formulaci:

$$
\begin{aligned}
\min_{x,y} \max_{P\in\mathcal{P}} \quad & \mathsf{E}_P \left[\sum_{i=1}^{M} \sum_{j=1}^{J} y_{ji}(\omega) \right], \\
\text{s.t.} \quad & \sum_{\{j: s_j \le t < f_j\}} x_{ji} \le 1, \quad i = 1,\dots,M,\ t \in S, \\
& \sum_{i=1}^{M} x_{ij} = 1, \quad j = 1,\dots,J, \\
& \sum_{\{k: f_j \le s_k < f_j(\omega)\}} x_{ki} \le y_{ji}(\omega) + J(1 - x_{ji}), \\
& \qquad i = 1,\dots,M, j = 1,\dots,J,\ \omega \in \Omega, \\
& x_{ji} \in \{0,1\}, \quad j = 1,\dots,J,\ i = 1,\dots,M, \\
& y_{ji}(\omega) \in \mathbb{N}, \quad j = 1,\dots,J,\ i = 1,\dots,M,\ \omega \in \Omega,
\end{aligned}
\tag{2.3}
$$

kde první podmínka zaručuje, že každému stroji je v jeden okamžik uložen nejvýše jeden úkol a druhá zajišťuje, že každý úkol bude vykonán, stejně jako je tomu u 1.1. Ve třetí podmínce potom zavádíme novou rozhodovací proměnnou $y_{ij}(\omega)$, která v případě, že úkol j je zadán stroji i, značí náhodný počet překryvů tohoto úkolu s dalšími úkoly zadanými tomuto stroji, v opačném případě pak bude tato proměnná rovna 0, což se dá, vzhledem k povaze optimalizační úlohy, snadno nahlédnout. Hodnota účelové funkce potom odpovídá celkově nejmenšímu očekávanému počtu překryvů vzhledem k příslušnému nejhoršímu pravděpodobnostnímu rozdělení. Úkoly, které se překrývají s úkoly jim předcházejícími, buď nebude možné vykonat, nebo bude nutné je zadat externím pracovním kapacitám. Možnou modifikací účelové funkce je přidání penalizace pro překryv úkolů j a k ($f_j \leq s_k$) představující ztrátu při nevykonání úkolu k nebo cenu za jeho outsourcing.

Úlohu je možné reformulovat jako robustní úlohu barvení grafu, která se poprvé objevuje v práci Yáñez a Ramírez (2003) a již dále uvádíme. Pro jednoduchost předpokládáme, že úkoly jsou seřazeny podle času jejich začátků. Počet barev k barvení odpovídá počtu strojů M, množina vrcholů odpovídá množině úkolů, množina E obsahuje hrany $(j, k), j < k$, pokud se úkoly určitě překrývají, tedy v případě, že $s_k < f_j$. Dále definujeme množinu $\bar{E}$ doplňkových hran, tedy hran $(j, k), j < k$, kde $f_j \leq s_k$. Tato množina odpovídá dvojicím úkolů, které se mohou překrýt pouze z důvodu vzniku náhodného zpoždění a mohou tak být přiřazeny stejnému stroji. Těmto hranám bude přiřazena penalizace závisející na rozdělení zpoždění konce prací, která se projeví v případě, že vrcholy, které spojuje, budou obarveny stejnou barvou. Penalizaci příslušející hraně (j, k) při pravděpodobnostním rozdělení zpoždění P značíme $q_{P,jk}$. Naším úkolem pak bude najít obarvení grafu (přípustný rozvrh), které celkovou penalizaci minimalizuje vzhledem k nejhorší možné realizaci $P \in \mathcal{P}$.

Předpokládáme, že M je větší nebo rovno než chromatické číslo zavedeného grafu, což je ekvivalentní existenci přípustného rozvrhu. Matematická formulace problému je potom následující:

$$
\begin{aligned}
\min_{x,y} \max_{P \in \mathcal{P}} \quad & \sum_{(j,k) \in \bar{E}} q_{P,jk} y_{jk}, \\
\text{s.t.} \quad \sum_{i=1}^{M} x_{ji} &= 1, \quad j = 1, \dots, J, \\
x_{ji} + x_{ki} &\leq 1, \quad (j,k) \in E,\ i = 1, \dots, M, \\
x_{ji} + x_{ki} &\leq 1 + y_{jk}, \quad (j,k) \in \bar{E},\ i = 1, \dots, M, \\
x_{ji} &\in \{0,1\}, \quad j = 1, \dots, J,\ i = 1, \dots, M, \\
y_{jk} &\in \{0,1\}, \quad (j,k) \in \bar{E},
\end{aligned}
\tag{2.4}
$$

kde rozhodovací proměnná $x_{ji} = 1$ indikuje obarvení vrcholu j barvou i. První podmínka potom zaručuje obarvení každého vrcholu právě jednou barvou, druhá zaručuje, že vrcholy spojené hranou budou obarveny jinou barvou a třetí zavádí další rozhodovací proměnnou y_{jk}, která, vzhledem k povaze úlohy, bude rovna jedné právě tehdy, když dva vrcholy j a $k, (j, k) \in \bar{E}$, budou obarveny stejnou barvou.

Snadno vidíme, že množina přípustných řešení úlohy barvení grafu odpovídá množině přípustných řešení 2.3, tedy množině všech přípustných rozvrhů. Pokud vhodně zvolíme penalizace $q_{P,jk}$, dá se ukázat, že formulace 2.4 je ekvivalentní formulaci 2.3. Pro důkaz ekvivalence obou úloh pak stačí ukázat, že hodnota účelové funkce bude v případě optimálního řešení stejná. Autoři v práci Branda a kol. (2016) dokazují, že úlohy jsou ekvivalentní, pokud $q_{P,jk} = P(D_j > s_k - f_j)$ (od tohoto místa dále pro jednoduchost vypouštíme (ω) při zápisu náhodných veličin a dalších s nimi spojených hodnot). Autoři úlohu uvádějí pro pevně zvolené rozdělení zpoždění P, ale výsledky jsou obecně platné i pro minimaxovou formulaci úlohy.

Autoři v práci Yáñez a Ramírez (2003) navrhují 2 možnosti řešení úlohy, které jsou aplikovatelné při pevně zvoleném rozdělení P. Úlohu navrhují řešit buď přímo pomocí známých algoritmů pro celočíselné optimalizační úlohy, nebo pomocí genetického algoritmu. Při numerických experimentech se ukázalo, že pro menší instance dávají klasické algoritmy lepší výsledky, ale jsou časově mnohem náročnější. Pro větší instance, kde z důvodu přílišných výpočetních nároků není možné použít klasické metody řešení, se genetický algoritmus ukázal jako vhodná alternativa přinášející uspokojivá řešení v přijatelném čase. Autoři v článku uvažují pouze úlohu ekvivalentní případu s identickými stroji.

Úlohu lze ekvivalentně formulovat jako úlohu nalezení toku grafem s minimální penalizací, která je uvedena v práci Branda a Hájek (2017). Tuto formulaci uvedeme a detailněji popíšeme v jedné z následujících částí naší práce.

Má-li jeden stroj vykonat více než dva úkoly, formulace 2.4 bere v úvahu i možnost, že se práce na libovolném z úkolů překryje s plánovanou dobou práce na pozdějším, než přímo následujícím úkolu. Pokud bychom chtěli brát v potaz pouze možnost překrytí dvou po sobě následujících úkolů, je nutné přistoupit k formulaci identifikující přímého následníka daného úkolu uvedené pro identické stroje a fixní P v práci Branda a kol. (2016). Minimaxová formulace úlohy je následující:

$$
\begin{aligned}
\min_{x,y}\ \max_{P\in\mathcal{P}} \quad & \sum_{(j,k)\in\bar{E}} q_{P,jk} z_{jk}, \\
\text{s.t.} \quad & \sum_{i=1}^{M} x_{ji} = 1, \quad j = 1,\dots,J, \\
& x_{ji} + x_{ki} \le 1, \quad (j,k)\in E,\ i = 1,\dots M, \\
& x_{ji} + x_{ki} \le 1 + y_{jk}, \quad (j,k)\in \bar{E},\ i = 1,\dots M, \\
& y_{jk} + \sum_{\{l:(j,l)\in\bar{E}\wedge s_l \ge f_k\}} z_{jl} \le 1, \quad (j,k)\in\bar{E}, \\
& \sum_{\{k:(j,k)\in\bar{E}\}} y_{jk} \le J \sum_{\{k:(j,k)\in\bar{E}\}} z_{jk}, \quad j = 1,\dots,J, \\
& z_{jk} \le y_{jk}, \quad (j,k)\in\bar{E}, \\
& x_{ji} \in \{0,1\}, \quad j = 1,\dots,J,\ i = 1,\dots M,
\end{aligned}
\tag{2.5}
$$

Zavádíme novou rozhodovací proměnnou z_{jk} značící přímou souslednost úkolů j a k na jednom stroji a oproti formulaci 2.4 je přidána třetí a čtvrtá uvedená nerovnost.

Třetí nerovnost zaručuje, že pokud jsou úkoly j a k zadány stejnému stroji, pak žádný z úkolů, pro který je čas začátku větší než f_k, nemůže být označen za přímého následníka. Čtvrtá nerovnost zaručuje, že pro úkol, po kterém je na stejném stroji plánován alespoň jeden další, bude označen přímý následník.

Tato sada podmínek uvedená ve zmíněné práci však ve skutečnosti nestačí k identifikaci optimálního řešení a za tímto účelem přidáváme do matematické formulace poslední nerovnost

$$z_{jk} \le y_{jk}, \quad (j,k)\in\bar{E},$$

která zaručí, že z_{jk} může být rovno 1 pouze tehdy, pokud jsou úkoly j a k zadány stejnému stroji.

2.2.2 Minimalizace střední hodnoty doby překrytí

Dále uvádíme formulaci FIS s náhodným zpožděním konce prací, kde minimalizačním kritériem je střední hodnota doby překrytí zpožděných úkolů s plánovanou dobou práce na následujících úkolech. Tuto formulaci jsme pro FIS v dostupné literatuře neobjevili.

Pro j a k takové, že $f_j \leq s_k$, zavádíme novou náhodnou proměnnou O_{jk}, závisející na D_j, jejíž hodnota odpovídá době překrytí práce na úkolech j a k v důsledku zpoždění úkolu j. Definice je následující:

$$O_{jk} = \begin{cases} 0 & D_j \leq s_k - f_j \\ D_j + f_j - s_k & s_k - f_j < D_j \leq f_k - f_j \\ f_k - s_k & f_k - f_j < D_j. \end{cases} \tag{2.6}$$

Matematická formulace problému je potom následující:

$$\begin{aligned} \min_{x,y} \max_{P \in \mathcal{P}} \quad & \mathsf{E}_P \left[\sum_{j=1}^{J} \sum_{\{k: f_j \leq s_k\}} y_{jk} O_{jk} \right], \\ \text{s.t.} \quad \sum_{\{j: s_j \leq t < f_j\}} x_{ji} & \leq 1, \quad i = 1, \ldots, M, \; t \in S, \\ \sum_{i=1}^{M} x_{ij} & = 1, \quad j = 1, \ldots, J, \\ x_{ji} + x_{ki} & \leq 1 + y_{jk}, \quad j = 1, \ldots, J, \; k = j+1, \ldots, J, \; i = 1, \ldots, M, \\ x_{ji} & \in \{0,1\}, \quad j = 1, \ldots, J, \; i = 1, \ldots, M, \\ y_{jk} & \in \{0,1\}, \quad j = 1, \ldots, J, \; k \in \{l : f_j \leq s_l\}, \end{aligned} \tag{2.7}$$

kde první dvě podmínky zaručují přípustnost rozvrhu a třetí podmínka zavádí rozhodovací proměnnou y_{jk}, která je rovna jedné, jsou-li práce j a k zadány stejnému stroji a nule jinak.

Úlohu 2.7 lze, stejně jako úlohu 2.3, reformulovat jako robustní úlohu barvení grafu 2.4, kde $q_{jk} = \mathsf{E}_P \, O_{jk}$ (rozepsáno níže). Stejně jako v předchozím případě vidíme, že množina přípustných řešení úlohy barvení grafu odpovídá množině přípustných řešení úlohy 2.7 a stačí nám ukázat ekvivalenci účelových funkcí. Ta okamžitě plyne z toho, že rozhodovací proměnná y_{jk} v obou úlohách indikuje, zda jsou úkoly j a k zadány stejnému stroji a z linearity střední hodnoty.

V hodnotě účelové funkce této formulace je zahrnuta i možnost, že se práce na úkolu z důvodu zpoždění překryje s plánovanou dobou práce na pozdějším, než přímo následujícím úkolu. Pokud bychom chtěli brát v potaz pouze možnost překrytí dvou po sobě následujících úkolů, je nutné přistoupit k reformulaci identifikující přímého následníka, podobně jako je tomu v 2.5. Další možností je reformulace úlohy jako nalezení toku grafem s minimální penalizací, přirozeně identifikující přímé následníky, která je popsána níže v 2.12 nebo 3.2. Uvedené možnosti nevyčerpávají všechny modifikace úlohy, které se v praxi budou odvíjet od charakteru úkolů, možnosti posunutí práce na nich či jejich outsorcingu a dalších faktorů.

V úloze FIS minimalizující počet překrytí stačí pro fixní $P \in \mathcal{P}$ vyčíslení hodnoty účelové funkce znalost $F_P(s_k - f_j)$ pro všechna $(j,k) \in \bar{E}$, není tedy potřeba úplná znalost P. Oproti tomu v úloze FIS minimalizující střední hodnotu doby překrytí je pro vyčíslení hodnoty účelové funkce nutná mnohem detailnější znalost P, umožňující spočítat hodnotu

$$\mathsf{E}_P O_{jk} = \int_{s_k-f_j}^{f_k-f_j} x - (s_k - f_j) dF_P(x) + (f_k - s_k)\left(1 - F_P(f_k - f_j)\right).$$

Pro jednotlivá rozdělení je tedy nutná znalost distribuční funkce na intervalu

$$\left(\min_{(j,k)\in\bar{E}} (s_k - f_j), \max_{(j,k)\in\bar{E}} (f_k - f_j)\right].$$

Pro v praxi často používanou volbu $\mathcal{P}$ jako množinu exponenciálních rozdělení s kladnou hmotou p v nule (distribuční funkce $F(x) = p + (1-p)(1-e^{-\lambda x}), p > 0$) je $\mathsf{E}_P O_{jk}$ snadno vyjádřitelná jako spojitá funkce proměnných p a λ:

$$(1-p)\left(\frac{e^{-\lambda(s_k+f_k-2f_j)}\left(e^{\lambda(s_k-f_j)}(\lambda(s_k - f_k) - 1) + e^{\lambda(f_k-f_j)}\right)}{\lambda} + e^{-\lambda(f_k-f_j)}(f_k - s_k)\right).$$

Při určité modifikaci úlohy, kterou podrobněji rozebíráme ve třetí kapitole, není při řešení minimaxové formulace potřeba detailnější znalost P a úlohu lze řešit pro $\mathcal{P}$ definovanou pouze za pomoci středních hodnot zpoždění.

2.2.3 Maximalizace spolehlivosti rozvrhu

Další možností je varianta FIS, kde se snažíme maximalizovat pravděpodobnost, že zvolený rozvrh zůstane v případě výskytu zpoždění přípustným. Tuto pravděpodobnost budeme dále označovat jako spolehlivost rozvrhu. Tato formulace se jeví jako nejvhodnější, pokud nemáme možnost outsorcingu úkolů, kdy jejich překrytí způsobí buď nemožnost vykonání následujícího úkolu, nebo zpoždění začátku práce na něm. Formulaci úlohy, její modifikace a možnosti řešení na tomto místě uvedeme.

Formulace FIS s pevně daným počtem identických strojů a náhodným zpožděním prací maximalizující spolehlivost rozvrhu je následující:

$$\begin{aligned}
\max_x \min_{P\in\mathcal{P}} \quad & \mathsf{P}\left(\sum_{\{j: s_j \le t < f_j + D_j\}} x_{ji} \le 1,\ t \in S,\ i = 1,\ldots,M\right), \\
\text{s.t.} \quad & \sum_{\{j: s_j \le t < f_j\}} x_{ji} \le 1, \quad i = 1,\ldots,M,\ t \in S, \\
& \sum_{i=1}^{M} x_{ij} = 1, \quad j = 1,\ldots,J, \\
& x_{ij} \in \{0,1\}, \quad j = 1,\ldots,J,\ i = 1,\ldots,M.
\end{aligned} \tag{2.8}$$

V účelové funkci 2.8 optimalizujeme pravděpodobnost jevu, že při realizaci náhodného zpoždění bude v každém z časů začátku práce na úkolu každému stroji přiřazen nejvýše jeden úkol, což odpovídá tomu, že nedojde k žádnému překryvu.

Obecně budeme předpokládat, že jednotlivá zpoždění D_j nemusí být navzájem nezávislá a budeme pro ně uvažovat nějaké sdružené pravděpodobnostní rozdělení. Tento předpoklad vychází z praxe, neboť pokud se úkoly mohou zpozdit například příčinou špatného počasí, je pravděpodobné, že se zpozdí více úkolů naplánovaných na stejnou dobu.

V předchozích dvou uvedených problémech FIS nebylo nutné brát sdružené rozdělení zpoždění v potaz díky vlastnostem střední hodnoty, což v tomto případě neplatí. Zároveň je úlohu 2.8 vzhledem k tvaru účelové funkce velmi obtížné řešit přímo, proto přistoupíme k zavedení určitých zobecnění a dalších předpokladů.

Vzájemnou závislost marginálních rozdělení zpoždění budeme, stejně jako v práci Branda (2018), modelovat pomocí Archimedovské kopule. Na tomto místě uvedeme některé základní definice a tvrzení, která můžeme najít v práci McNeil a kol. (2009).

Definice 1. *Řekneme, že n-rozměrná funkce $C : [0,1]^n \to [0,1]$ je kopule, pokud platí:*

(i) $C(u_1, \dots, u_n) = 0$ *pokud alespoň pro jedno* $i \in \{1, \dots, n\}$ *je* $u_i = 0$

(ii) $C(1, \dots, 1, u_i, 1, \dots, 1) = u_i \ \forall i \in \{1, \dots, n\}$

(iii) C *je n-rostoucí, tj.* $\forall S = \prod_{i=1}^{n}[a_i, b_i] \subseteq [0,1]^n$ *platí*

$$\int_S dC(u) = \sum_{z \in \times_{i=1}^{n}\{a_i, b_i\}} (-1)^{card\{i: z_i = a_i\}} C(z) \geq 0.$$

Podle definice C odpovídá sdruženému pravděpodobnostnímu rozdělení na $[0,1]^n$, kde marginální rozdělení jsou rovnoměrná na $[0,1]$. Vztah libovolného sdruženého pravděpodobnostního rozdělení a kopule popisuje následující věta.

Věta 2 (Sklarova)**.** *Nechť F je distribuční funkce n-rozměrného pravděpodobnostního rozdělení s jednorozměrnými marginálními rozděleními $F_1, \dots, F_n$. Potom existuje n-rozměrná kopule C, že $\forall x \in \mathbb{R} \cup \{\infty, -\infty\}$ platí:*

$$F(x_1, \dots, x_n) = C\left(F_1\left(x_1\right), \dots, F_n\left(x_n\right)\right).$$

Pokud jsou marginální rozdělení F spojitá, potom je C určena jednoznačně.

Vidíme, že jelikož v našem případě uvažujeme znalost všech marginálních rozdělení zpoždění (případně jejich příslušnost do množiny $\mathcal{P}$), jeví se modelování jejich sdruženého rozdělení pomocí vhodné kopule jako rozumná volba.

Důležitou skupinou kopulí jsou takzvané Archimedovské kopule.

Definice 2. *Kopuli C nazveme Archimedovskou, pokud existuje spojitá klesající funkce $\psi : [0,1] \to [0,\infty)$, $\psi(1) = 0$, $\lim_{x \to 0^+} \psi(x) = \infty$, taková, že*

$$C(u) = \psi^{-1}\left(\sum_{i=1}^{n} \psi(u_i)\right).$$

Funkci ψ nazveme (striktním) generátorem Archimedovské kopule C.

Autoři v práci McNeil a kol. (2009) uvádějí definici pro inverzní funkci k ψ, avšak vzhledem k tomu, že v další použité literatuře týkající se FIS za nejistoty je použita tato definice, rozhodli jsme se pro zde uvedenou formu.

Z věty 2 a definice 2 plyne, že pro sdružené rozdělení F s příslušnou Archimedovskou kopulí generovanou ψ platí jednoduchý vztah:

$$F(x) = \psi^{-1}\left(\sum_{i=1}^{n} \psi(F_i(x_i))\right), \tag{2.9}$$

který je však velmi užitečný při řešení úlohy 2.8.

Na tomto místě pro ilustraci uvedeme několik důležitých Archimedovských kopulí:

- Gumbelova: $\psi(u) = (-\log(u))^{\theta}$, $\theta \in [1,\infty)$,
- Claytonova: $\psi(u) = \frac{1}{\theta}(u^{-\theta} - 1)$, $\theta \in [0,\infty)$,
- Frankova: $\psi(u) = -\log\left(\frac{e^{-\theta u}-1}{e^{-\theta}-1}\right)$, $\theta \in \mathbb{R} \setminus 0$.

Speciálním případem Gumbelovy kopule je případ, kdy $\theta = 1$, $\psi(u) = -\log(u)$ a jednotlivá marginální rozdělení jsou tak nezávislá, neboť:

$$F(x) = \psi^{-1}\left(\sum_{i=1}^{n} \psi(F_i(x_i))\right) = \prod_{i=1}^{n} F_i(x_i).$$

Uvedenou teorii nyní aplikujeme na úlohu 2.8. Nechť kopule C příslušející sdruženému rozdělení zpoždění úkolů F je Archimedovská s generátorem ψ. Platnost tohoto budeme automaticky předpokládat, nebude-li uvedeno jinak.

Pro libovolný rozvrh určený hodnotou rozhodovací proměnné x zavádíme pro $j = 1,\ldots,J$ hodnotu $m_j(x)$ odpovídající času $s_k - f_j$, pokud jsou úkoly j,k : $(j,k) \in \bar{E}$ přiřazeny stejnému stroji a k následuje jako první po j. Dále pro jednoduchost píšeme m_j místo $m_j(x)$.

Účelovou funkci 2.8 potom můžeme zapsat jako:

$$\mathsf{P}\left(D_j \leq m_j,\, j = 1,\ldots,J\right) = F(m_1,\ldots,m_J) = \psi^{-1}\left(\sum_{j=1}^{J} \psi\left(F_j\left(m_j\right)\right)\right).$$

Vzhledem k tomu, že ψ^{-1} je striktně klesající funkce, je úloha maximalizace spolehlivosti rozvrhu ekvivalentní:

$$\min_{x\in\mathcal{X}} \max_{P\in\mathcal{P}} \sum_{j=1}^{J} \psi\left(F_j\left(m_j\right)\right). \tag{2.10}$$

Definujeme-li rozhodovací proměnnou z_{jk}, označující přímou souslednost úkolů j a k na stejném stroji (stejně jako v 2.5), můžeme 2.10 dále upravit do výrazu:

$$\min_{x\in\mathcal{X}} \max_{P\in\mathcal{P}} \sum_{j=1}^{J} \sum_{\{k: s_k \geq f_j\}} \psi\left(F_j\left(s_k - f_j\right)\right) z_{jk}, \tag{2.11}$$

z čehož okamžitě plyne[1], že úlohu 2.8 lze ekvivalentně formulovat jako úlohu 2.5 (ať už pro jednu či více tříd strojů), kde penalizace hran $q_{P,jk} = \psi\left(F_j\left(s_k - f_j\right)\right)$, což je ukázáno také v práci Branda a kol. (2016).

[1]Množina indexů (j,k), kterou určuje kombinace sum v 2.11 je ekvivalentní množině $\bar{E}$ z 2.5.

V práci Branda a Hájek (2017) je uvedena další možná formulace úlohy 2.8 založená na principu nalezení toku grafem s minimální penalizací. Pro tuto formulaci budeme vycházet z grafu definovaného pro úlohu 2.4. K množině vrcholů (značíme pouze pomocí indexů) přidáme vrcholy 0 – zdroj a $(J+1)$ – stok. Jako množinu hran uvažujeme $\bar{E}$ rozšířenou o hrany $(0, J+1)$, $(0, j)$ a $(j, J+1)$ $\forall j = 1, \ldots, J$, s kapacitou 1 a příslušnými penalizacemi rovnými nule.

V případě, že uvažujeme pouze jednu třídu strojů, je jedna z možných formulací úlohy následující:

$$
\begin{aligned}
\min_{y} \max_{P \in \mathcal{P}} \quad & \sum_{(j,k) \in \bar{E}} q_{P,jk} y_{jk}, \\
\text{s.t.} \quad & \sum_{j=1}^{J+1} y_{0j} = M, \\
& \sum_{\{k: f_k \leq s_j\}} y_{kj} = \sum_{\{k: f_j \leq s_k\}} y_{jk}, \quad j = 1, \ldots, J, \\
& \sum_{\{k: f_j \leq s_k\}} y_{jk} = 1, \quad j = 1, \ldots, J, \\
& \sum_{j=0}^{J} y_{j(J+1)} = M, \\
& y_{jk} \in \{0,1\}, \quad j, k = 1, \ldots, J,
\end{aligned}
\tag{2.12}
$$

kde $y_{jk} = 1$ pokud je nějakým strojem úkol k vykonáván přímo po úkolu j, jinak $y_{jk} = 0$. Intuitivní představa úlohy je taková, že chceme nalézt kombinaci M toků ze zdroje do stoku, která minimalizuje celkovou penalizaci za průtok hranami spojujícími jednotlivé vrcholy tak, že jednotlivé toky nemají kromě zdroje a stoku žádný společný vrchol a zároveň jsou v nich dohromady zahrnuty všechny vrcholy.

První rovnost zajišťuje počátek všech M toků ve zdroji. Druhá rovnost zaručuje, rovnováhu přítoku a odtoku v každém vrcholu mezi zdrojem a stokem. Třetí rovnost zaručuje, že každým vrcholem prochází právě jeden tok a poslední rovnost zaručuje, že všechny toky skončí ve stoku. Ekvivalence 2.12 a 2.8, ve smyslu nalezení stejného optimálního rozvrhu, je zjevná.

Pokud uvažujeme více tříd strojů, je nejprve nutné formulaci 2.12 upravit tak, aby její každé přípustné řešení identifikovalo stroje odpovídající jednotlivým tokům. Z toho důvodu zavádíme místo y_{jk} proměnnou y_{jki} značící průchod toku i hranou spojující vrcholy j a k, což odpovídá přiřazení těchto úkolů stroji i. Analogicky zavádíme penalizaci $q_{P,jki}$ za průchod toku i hranou spojující vrcholy j a k. Ve formulaci úlohy je nutné rozšířit původní sadu rovností při využití množin C_j všech strojů, na kterých je možné vykonat úkol j. Výsledná formulace je potom následující:

$$
\begin{aligned}
\min_{y} \max_{P \in \mathcal{P}} \quad & \sum_{i=1}^{M} \sum_{(j,k) \in \bar{E}} q_{P,jki} y_{jki}, \\
\text{s.t.} \quad & \sum_{j=1}^{J+1} y_{0ji} &=& \ 1, \quad i = 1, \dots, M, \\
& \sum_{\{k : i \in C_k \wedge f_k \leq s_j\}} y_{kji} &=& \sum_{\{k : i \in C_k \wedge f_j \leq s_k\}} y_{jki}, \quad j = 1, \dots, J,\ i \in C_j, \\
& \sum_{i \in C_j} \sum_{\{k : i \in C_k \wedge f_j \leq s_k\}} y_{jki} &=& \ 1, \quad j = 1, \dots, J, \\
& \sum_{j=0}^{J} y_{j(J+1)i} &=& \ 1, \quad i = 1, \dots, M, \\
& y_{jki} &\in& \ \{0{,}1\}, \quad (j,k) \in \bar{E},\ i \in C_j \cap C_k,
\end{aligned}
\tag{2.13}
$$

Pro pevně zvolené P je k řešení možné využít například metody zmíněné výše v 1.1.1.

3. Robustifikace

V této kapitole se budeme věnovat reformulaci a řešení výše uvedených úloh pro vybrané případy množin $\mathcal{P}$. Obecně o takových úlohách hovoříme jako o robustních úlohách rozvrhování za nejistoty, neboť jejich řešení je robustní vzhledem k nejistotě dané množinou $\mathcal{P}$. Uvádíme zde výsledky dosažené jinými autory, případnou možnost rozšíření jejich aplikace i výsledky původní. V celé kapitole řešíme FIS úlohy pouze pro případ, kdy jsou všechny stroje identické.

3.1 Maximalizace spolehlivosti rozvrhu při fixním počtu zatěžkávacích rozdělení

V práci Branda (2018) jsou uvedeny výsledky řešení úlohy maximalizace spolehlivosti rozvrhu 2.12 pro speciální případ množiny $\mathcal{P}$, které zde uvedeme. V tomto případě jsou pro marginální rozdělení uvažovány pouze 2 možnosti a $\mathcal{P}$ je množina všech jejich různých kombinací.

Pro popis množiny $\mathcal{P}$ budeme vycházet z bodu, kdy uvažujeme všechna rozdělení D_j identická s distribuční funkcí $F_j(x) = F(x), j = 1, \dots, J$. Penalizace $q_{jk} = \psi\left(F\left(s_k - f_j\right)\right)$. Dále uvažujeme nějaké zatěžkávací rozdělení $\tilde{F}(x) : \tilde{F}(x) < F(x)$, $x \geq 0$. Příslušné pravděpodobnosti a penalizace značíme $\tilde{P}$ a $\tilde{q}$. Pro pevně zvolené $\Gamma \in \{1, \dots, J - M\}$ potom definujeme množinu:

$$\begin{aligned}\mathcal{P}_\Gamma = \{P : \Gamma \text{ marginálních rozdělení } F_j \text{ je změněno na } \tilde{F}, \\ \text{zbylých } (J - \Gamma)\ F_j \text{ zůstane beze změny.}\}.\end{aligned}$$

Tento předpoklad je z praktického hlediska odůvodnitelný, pokud sledujeme, že jistá část úkolů mívá za určitých okolností zpoždění výrazně vyšší než zbytek.

Autoři v práci Branda (2018) uvádějí jednoduchý příklad, kdy volíme $F(x) = p + (1 - p)\left(1 - e^{-\lambda x}\right)$, $x \geq 0$ a $F(x) = 0$, $x < 0$. Jedná se o případ, kdy s pravděpodobností p nedojde k žádnému zpoždění a s pravděpodobností $(1 - p)$ má zpoždění exponenciální rozdělení s parametrem λ. p a λ dále uvažujeme jako bodové odhady skutečných hodnot parametrů za předpokladu platnosti modelu. Nechť dále existují konfidenční intervaly pro odhad p a λ s dolními mezemi $\tilde{p}$ a $\tilde{\lambda}$. Potom definujeme $\tilde{F}(x) = \tilde{p} + (1 - \tilde{p})\left(1 - e^{-\tilde{\lambda} x}\right)$ $x \geq 0$.

Autoři docházejí při předpokladu identických strojů k následujícímu výsledku:

Tvrzení 3. *Za platnosti dříve uvedených předpokladů lze pro $\mathcal{P} = \mathcal{P}_\Gamma$ reformulovat účelovou funkci 2.12 jako:*

$$\sum_{(j,k) \in \bar{E}} q_{jk} y_{jk} \ + \ \Gamma \mathrm{CVaR}_\alpha\left(Z\left(y\right)\right),$$

kde náhodná veličina $Z(y)$ představující ztrátu je rovnoměrně rozdělená na $\{\Delta_{jk} y_{jk}\}$, $\Delta_{jk} = \tilde{q}_{jk} - q_{jk} > 0$ a $\alpha = 1 - \Gamma / |\bar{E}|$. CVaR *je obecně známá koherentní míra rizika* Conditional Value at Risk.

Důkaz. Viz Branda (2018).

$\square$

Autoři také navrhují způsob řešení takto reformulované úlohy. Ukazují, že úlohu je možné řešit jako dvoustupňovou optimalizační úlohu.

Pro navrhovaný způsob řešení je však nutné upravit graf zavedený pro formulaci úlohy 2.12, aby bylo možné využít totální unimodularitu matice omezení. Množinu vrcholů V rozšíříme o vrcholy j' $j = 1, \ldots, J$, do množiny hran $\bar{E}$ přidáme hrany (j, j') s kapacitou 1 a penalizací rovnou 0, hrany $(j, k) \in \bar{E}$ nahradíme hranami (j', k) s identickou kapacitou a penalizací a hrany $(j, J+1)$ nahradíme hranami $(j', J+1)$ s kapacitou 1 a penalizací 0. Výsledné množiny hran a vrcholů značíme $\tilde{E}$ a $\tilde{V}$. Dále pro vrcholy zavádíme poptávku po toku $d_0 = -M$, $d_{J+1} = M$, $d_j = 1$, $d_{j'} = -1$, $j = 1, \ldots, J$.

Ve výsledné dvoustupňové úloze potom v prvním stupni řešíme:

$$\min_{\theta} \Gamma\theta + \varphi(\theta) \text{ s.t. } \theta \in \left[0, \max_{(j,k)\in\hat{E}} \Delta_{jk}\right] \tag{3.1}$$

a ve druhém stupni řešíme:

$$\begin{aligned} \varphi(\theta) = \min_{y} \quad & \sum_{(j,k)\in\bar{E}} q_{jk} y_{jk} + \sum_{(j,k)\in\bar{E}} \max\{\Delta_{jk} y_{jk} - \theta, 0\}, \\ \text{s.t.} \quad & \sum_{(j,k)\in\tilde{E}} y_{jk} - \sum_{(k,j)\in\tilde{E}} y_{kj} \;=\; d_j, \quad j \in \tilde{V}, \\ & \qquad\qquad\qquad\qquad y_{jk} \;\in\; \{0,1\}, \quad (j,k) \in \tilde{E}. \end{aligned} \tag{3.2}$$

Vzhledem k tomu, že hodnota θ je při řešení 3.2 fixována, platí:

$$\max\{\Delta_{jk} y_{jk} - \theta, 0\} = y_{jk} \max\{\Delta_{jk} - \theta, 0\}$$

a problém 3.2 tak můžeme řešit jako úlohu nalezení toku grafem s minimální penalizací, kde jednotlivé penalizace hran mají tvar $q_{jk} + \max\{\Delta_{jk} - \theta, 0\}$. Jelikož je matice omezení 3.2 totálně unimodulární, je možné relaxovat celočíselnost y_{jk} a nahradit poslední omezení 3.2 omezením $0 \leq y_{jk} \leq 1$. Úloha se tak stává LP a lze ji efektivně řešit v polynomiálním čase

Pro nalezení optimálního řešení 3.1 je nutné opakovaně vyřešit 3.2 pro různé hodnoty θ. Autoři v článku navrhují pro nalezení optimální hodnoty θ použití algoritmu *golden–section–search*, kdy počet nutných iterací algoritmu $\tilde{n}$ lze shora odhadnout pomocí nerovnice:

$$0.618^{\tilde{n}-1} \leq \frac{\epsilon}{\max \Delta_{jk}},$$

kde ϵ je stanovená přesnost optimálního řešení 3.1.

Je zjevné, že úlohy 2.4 a 2.7, v jejichž formulaci uvažujeme možnost překryvu pouze u navazujících úkolů, lze pro $\mathcal{P} = \mathcal{P}_\Gamma$ reformulovat a řešit naprosto stejným způsobem s jediným rozdílem ve významu penalizace q_{jk}.

V dalších řešených úlohách budeme automaticky předpokládat, že úloha nalezení optimálního rozvrhu je reformulována jako úloha nalezení toku grafem, jehož konstrukci jsme popsali výše, s minimální penalizací.

3.2 *Box uncertainty*

V práci Zhu a Fukushima (2009) autoři řeší robustní úlohu optimalizace portfolia při minimalizaci *CVaR*, tzv. *worst–case CVaR* pro různé případy množiny pravděpodobnostních rozdělení ztráty. Jedním z nich je tzv. *box uncerntainty* (BU), který zde krátce představíme a následně podobný přístup aplikujeme při řešení vybraných FIS úloh. Tento přístup dosud nebyl v souvislosti s úlohou FIS za nejistoty v nám známé literatuře uveden, jedná se tedy o původní výsledek.

V případě BU v úloze přirozeně vystupuje diskrétní rozdělení, kde atomy odpovídají jednotlivým scénářům výnosů akcií. Běžně se při optimalizaci portfolia vzhledem k *CVaR* považují všechny scénáře za stejně pravděpodobné, což nemusí odpovídat realitě a může vést ke špatným výsledkům. Autoři proto navrhují uvažovat všechna možná diskrétní rozdělení π v nějakém okolí zvoleného diskrétního rozdělení π^0, což může být například zmiňované rovnoměrné rozdělení.

Množina uvažovaných pravděpodobnostních rozdělení π – v tomto případě ztotožňujeme rozdělení s vektorem pravděpodobností – má potom tvar:

$$\mathcal{P} = \{\pi : \pi = \pi^0 + \eta, \mathbb{1}'\eta = 0, \underline{\eta} \leq \eta \leq \overline{\eta}\}, \tag{3.3}$$

kde nezápornost π je obsažena v omezení $\underline{\eta} \leq \eta \leq \overline{\eta}$ a podmínka $\mathbb{1}'\eta = 0$ zaručuje, že půjde o pravděpodobnostní rozdělení.

Autoři v práci ukazují, že při této volbě množiny $\mathcal{P}$ lze úlohu řešit jako konvexní nebo lineární program.

3.2.1 Minimalizace střední hodnoty počtu překryvů a doby překrytí

Jak už bylo řečeno dříve, pro vyčíslení hodnoty účelové funkce úlohy 2.4 stačí znát hodnoty distribuční funkce F v bodech $(s_k - f_j) : (j,k) \in \bar{E}$ a tak je pro tuto úlohu přirozená možnost diskretizace rozdělení zpoždění. Pro verzi úlohy 2.4 uvažující pouze možnost překrytí dvou po sobě jdoucích úkolů formulovanou jako nalezení toku sítí definovanou pro 3.2 s minimální cenou tedy předvedeme možnost jejího řešení při BU.

Nejprve zavedeme způsob diskretizace rozdělení P. Nechť $T = \{t_1, \ldots, t_N\}$ je množina všech různých hodnot $(s_k - f_j) : (j,k) \in \bar{E}$ seřazená podle velikosti. Pravděpodobnostní rozdělení π na množině $T \cup \{t_0 = t_1 - 1\}$[1] definujeme následujícím způsobem:

$$\pi_i = \begin{cases} F(t_1) & i = 0 \\ F(t_{i+1}) - F(t_i) & i = 1, \ldots, N-1 \\ 1 - F(t_N) & i = N \end{cases} \tag{3.4}$$

Verze úlohy 2.4, uvažující možnost překrytí pouze přímo po sobě následujících úkolů, formulovaná jako úloha nalezení toku grafem s minimální penalizací má

[1] t_0 může mít libovolnou hodnotu menší než t_1, na řešení úlohy toto nemá vliv.

potom následující tvar:

$$\begin{array}{rrcl}\min\limits_{y} \max\limits_{\pi\in\{\pi:\pi=\pi^0+\eta,\, \mathbb{1}'\eta=0,\, \underline{\eta}\le\eta\le\overline{\eta}\}} & \sum\limits_{(j,k)\in\tilde{E}} q_{jk}(\pi)y_{jk}, & & \\ \text{s.t.} & \sum\limits_{(j,k)\in\tilde{E}} y_{jk} - \sum\limits_{(k,j)\in\tilde{E}} y_{kj} & = & d_j, \quad j\in\tilde{V}, \\ & y_{jk} & \in & \{0{,}1\}, \quad (j,k)\in\tilde{E}, \end{array} \tag{3.5}$$

Dále budeme rozhodovací proměnné y v účelové funkci značit pouze pomocí indexů $i = 1,\dots,\bar{N}$, $\bar{N} = |\bar{E}|$, pro zjednodušení zavádíme množinu Y všech přípustných hodnot y a zavádíme hodnotu r_i značící pořadí délky času mezi pracemi spojenými hranou, které odpovídá y_i, v množině T. Tedy pokud y_i odpovídá hraně (j,k) a $(s_k - f_j) = t_m$, pak $r_i = m$. Pro penalizace platí $q_{jk}(\pi) = \sum_{j=r_i}^{N} \pi_j = 1 - F(t_i)$. Formulace úlohy vypadá následovně:

$$\begin{array}{rrcl}\min\limits_{y\in Y}\max\limits_{\pi} & \sum\limits_{i=1}^{\bar{N}} y_i \sum\limits_{j=r_i}^{N} \pi_j, & & \\ \text{s.t.} & \pi & = & \pi^0+\eta, \\ & \mathbb{1}'\eta & = & 0, \\ & \underline{\eta} & \le & \eta, \\ & \eta & \le & \overline{\eta}. \end{array} \tag{3.6}$$

Účelovou funkci můžeme také zapsat jako:

$$\pi'\mathbf{L}y, \text{ kde } \mathbf{L} = \begin{pmatrix} 0 & \dots & 0 \\ 1 & \dots & \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{pmatrix}. \tag{3.7}$$

Pro i-tý sloupcový vektor matice $\mathbf{L}$ platí, že jeho prvních r_i prvků je rovno nule a zbytek prvků je roven jedné. V případě, kdy $N = \bar{N}$ je $\mathbf{L}$ bez prvního nulového řádku dolní trojúhelníková matice mající všechny prvky pod diagonálou rovny jedné.

Tvar úlohy je potom následující:

$$\begin{array}{rrcl}\min\limits_{y\in Y} \pi^{0'}\mathbf{L}y + \max\limits_{\eta} & \eta'\mathbf{L}y, & & \\ \text{s.t.} & \mathbb{1}'\eta & = & 0, \\ & \underline{\eta} & \le & \eta, \\ & \eta & \le & \overline{\eta}. \end{array} \tag{3.8}$$

Podobně jako v práci Zhu a Fukushima (2009) převedeme maximalizační část úlohy 3.6 na její duální tvar. Výsledná úloha má tvar:

$$\begin{array}{rrcl}(P_1)\ \min\limits_{y\in Y}\min\limits_{\xi,\omega,z} & \pi^{0'}\mathbf{L}y - \underline{\eta}'\omega + \overline{\eta}'\xi, & & \\ \text{s.t.} & \xi - \omega + \mathbb{1}z & = & \mathbf{L}y, \\ & \xi & \ge & 0, \\ & \omega & \ge & 0, \\ & z & \in & \mathbb{R}. \end{array} \tag{3.9}$$

Z duality LP vyplývá že pro optimální řešení $(y^*, \xi^*, \omega^*, z^*)$ úlohy 3.9 platí, že existuje η^* takové, že (y^*, η^*) je optimální řešení 3.8[2] a $\eta^{*\prime}\mathbf{L}y^* = \overline{\eta}'\xi^* - \underline{\eta}'\omega^*$. Existence optimálního řešení 3.8, potažmo 3.9, je zaručena, neboť v maximalizační podúloze 3.8 maximalizujeme lineární funkci na kompaktní množině a minimalizujeme přes konečnou množinu přípustných toků (rozvrhů).

Ve výše zmíněné práci Zhu a Fukushima (2009) se v tomto bodě úloha stala klasickou lineární optimalizační úlohou s konvexními či lineárními omezeními a šla proto dobře řešit. V našem případě je však y binární proměnná a kvůli prvnímu omezení 3.9 nelze použít celočíselnou relaxaci jako např. v 3.2. Dále tedy postupujeme za použití Lagrangeovské relaxace (LR), pro jejíž účel definujeme duální funkci

$$\begin{aligned} \theta_1(v) = \inf_{y,\xi,\omega,z} \quad & \pi^{0\prime}\mathbf{L}y - \underline{\eta}'\omega + \overline{\eta}'\xi + v'(\xi - \omega + \mathbb{1}z - \mathbf{L}y), \\ \text{s.t.} \quad & y \in Y, \xi \geq 0, \omega \geq 0, z \ \in \mathbb{R}. \end{aligned} \tag{3.10}$$

Část výrazu $\theta_1(v)$ obsahující y je pro libovolné fixované $v \in \mathbb{R}^{N+1}$ omezená a lze ji interpretovat jako hodnotu toku grafem s vektorem penalizací $(\pi^0 - v)'\mathbf{L}$. Můžeme tedy relaxovat celočíselnost y – v definici Y nahradíme podmínku $y \in \{0, 1\}$ podmínkou $0 \leq y \leq 1$ – a relaxovanou množinu přípustných hodnot y značíme $\hat{Y}$. Dále v definici $\theta_1(v)$ uvažujeme $\hat{Y}$ místo Y a zavádíme množinu $X = \{(y, \xi, \omega, z) : y \in \hat{Y}, \xi \geq 0, \omega \geq 0, z \in \mathbb{R}\}$. Vidíme, že pro nalezení hodnoty $\theta(v)$ pro libovolnou hodnotu v stačí vyřešit úlohu lineárního programování.

Duální problém má potom tvar:

$$(D_1) \ \sup_{v \in \mathbb{R}^{N+1}} \inf_{X} \pi^{0\prime}\mathbf{L}y - \underline{\eta}'\omega + \overline{\eta}'\xi + v'(\xi - \omega + \mathbb{1}z - \mathbf{L}y). \tag{3.11}$$

Platí, že $(P_1) \geq (D_1)$, a rovnost nastává v případě, že jsou splněny podmínky silné věty o dualitě (podrobnosti můžeme nalézt například v práci Boyd a Vandenberghe (2009)). První podmínkou je konvexita množiny X, která je zjevná. Druhou podmínkou je konvexita minimalizovaného výrazu $\theta_1(v)$, která plyne z toho, že $\pi^{0\prime}\mathbf{L}y - \underline{\eta}'\omega + \overline{\eta}'\xi$ je konvexní funkce a $h(x) = \xi - \omega + \mathbb{1}z - \mathbf{L}y$ je afinní ($x \in X$). Zbývá ověřit *Slaterovu* podmínku: $0 \in \text{int}h(X)$. Nechť $\tilde{y} \in Y$ a $\tilde{z} < 0$, potom $\mathbb{1}\tilde{z} - \mathbf{L}\tilde{y} < 0$ a $0 \in \text{int}\,\{\xi + \mathbb{1}\tilde{z} - \mathbf{L}\tilde{y} : \xi \geq 0\} \subset \text{int}\,(h(X))$. Podmínky věty jsou splněny a v tomto případě tedy platí $(P_1) = (D_1)$. Při hledání optimálního řešení 3.11 můžeme využít následujícího pozorování:

$$\theta_1(v) \neq -\infty \Leftrightarrow -\overline{\eta} \leq v \ \leq -\underline{\eta} \wedge \mathbb{1}'v = 0.$$

Popsané výsledky stručně shrnuje následující tvrzení.

Tvrzení 4. *Úlohu 3.6 můžeme ekvivalentně řešit jako úlohu*

$$\max_{v \in \mathbb{R}^{N+1}} \min_{y \in \hat{Y}} (\pi^0 + v)'\mathbf{L}y, \quad \underline{\eta} \leq v \leq \overline{\eta}, \mathbb{1}'v = 0,$$

kde $\mathbf{L}$ *je jako v 3.7 a* $\hat{Y}$ *je výše popsaná množina přípustných toků s relaxovanou celočíselností.*

[2] Nebo jinak, že y^* je optimální řešení minimalizační úlohy 3.8.

Důkaz. Tvrzení plyne z výše popsaného postupu a ze substituce $v = -v$. □

Podobný způsob můžeme aplikovat i při řešení úlohy 2.7 minimalizace střední hodnoty doby překrytí po sobě jdoucích prací. V tom případě bychom museli pracovat s aproximací střední hodnoty O_{jk}. Rozdíl by tak byl ve tvaru matice $\mathbf{L}$. Nechť y_i odpovídá hraně spojující úkol j s úkolem k, pro které $s_k - f_j = t_u$ a $f_k - f_j = t_v$. Hodnoty sloupce i matice $\mathbf{L}$, značené $\mathbf{L}_{li}$, lze potom zavést například následujícím způsobem:

$$\mathbf{L}_{li} = \begin{cases} 0 & l \leq u \\ t_l - t_u & u < l \leq v \\ t_v - t_u & v < l. \end{cases} \tag{3.12}$$

Při malém počtu různých $t \in T$ by mohlo být vhodné zvolit jemnější dělení pro definici π, tedy uměle rozšířit množinu T, což má vliv i na tvar matice $\mathbf{L}$.

3.2.2 Maximalizace spolehlivosti rozvrhu

Přístup k definici $\mathcal{P}$ pomocí BU je možné aplikovat i na úlohu 2.12. Pro diskrétní rozdělení π je v tomto případě přirozená definice 3.4 a penalizace jsou ve tvaru $q_i(\pi) = \psi(\sum_{j=0}^{r_i-1} \pi_j)$. Problém potom můžeme zapsat jako:

$$\begin{array}{rrcl} \min\limits_{y \in Y} \max\limits_{\eta} & q'y, & & \\ \text{s.t.} & \mathbb{1}'\eta & = & 0, \\ & \underline{\eta} & \leq & \eta, \\ & \eta & \leq & \overline{\eta}. \\ & \tilde{\pi} & = & \tilde{\mathbf{L}}(\pi^0 + \eta), \end{array} \tag{3.13}$$

kde $\tilde{\mathbf{L}} = (\mathbb{1}_{N+1}\mathbb{1}'_{\bar{N}} - \mathbf{L})'$ a $q = (\psi_1, \ldots, \psi_{\bar{N}})$, $\psi_i = \psi(\tilde{\pi}_i)$.

Vzhledem k nelinearitě způsobené tvarem penalizací se celá úloha stává obtížněji řešitelnou, neboť v tomto případě není možné snadno využít duality LP (nebo jiné) a převést tak celou úlohu na minimalizační problém. Je proto nutné přistoupit k hledání optimálního řešení jiným způsobem.

Možností je aplikace vybraného algoritmu pro prohledávání diskrétní množiny Y. Tím může být genetický algoritmus, jak je popsán například v práci Deep a kol. (2009), modifikace *tabu-search* algoritmu popsaného v práci (Branda a kol., 2016), nebo některý další vhodný algoritmus. Tento přístup je však výpočetně náročnější, neboť pro ohodnocení každého uvažovaného řešení y je potřeba spočítat konvexní NLP úlohu – maximalizační podúlohu 3.13 s parametrem y.

3.3 Eliptická množina nejistoty

Další možností volby množiny $\mathcal{P}$ diskrétních pravděpodobnostních rozdělení představené v práci Zhu a Fukushima (2009) je tzv. *Ellipsoidal Uncertainty* (EU). Tento přístup dosud nebyl v souvislosti s úlohou FIS za nejistoty v nám známé literatuře uveden, jedná se tedy o původní výsledek.

Předpokládáme, že rozdělení zpoždění P je diskretizováno stejně jako v 3.4 a uvažujeme π z množiny ve tvaru elipsoidu, kterou definujeme následujícím způsobem:

$$\mathcal{P} = \{\pi : \pi = \pi^0 + \mathbf{A}\eta, \mathbb{1}'\mathbf{A}\eta = 0, \pi^0 + \mathbf{A}\eta \geq 0, \|\eta\|_2 = 1\}, \tag{3.14}$$

kde $\mathbf{A}$ je čtvercová škálovací matice mající plnou hodnost. Aplikujeme-li tento přístup na úlohu 2.5 minimalizace střední hodnoty počtu překryvů po sobě jdoucích prací formulovanou jako úlohu nalezení toku grafem s minimální penalizací, dostáváme úlohu:

$$\begin{array}{lrcl} \min\limits_{y \in Y} \pi^{0'}\mathbf{L}y + \max\limits_{\eta} & \eta'\mathbf{A}'\mathbf{L}y, & & \\ \text{s.t.} & \mathbb{1}'\mathbf{A}\eta & = & 0, \\ & \pi^0 + \mathbf{A}\eta & \geq & 0, \\ & \|\eta\|_2 & \leq & 1. \end{array} \tag{3.15}$$

Maximalizační podúlohu 3.15 převedeme na její duální úlohu, podobně jako autoři v práci Zhu a Fukushima (2009) při optimalizaci portfolia pomocí minimalizace *CVaR*. V tomto případě se nejedná o klasickou LP dualitu, ale o SOCP dualitu. Dostáváme úlohu ve tvaru:

$$\begin{array}{llrcl} (P_3) & \min\limits_{y \in Y, \gamma, \xi, \omega, z} & \pi^{0'}\mathbf{L}y + \gamma + \pi^{0'}\omega, & & \\ & \text{s.t.} & -\xi - \mathbf{A}'\omega + \mathbf{A}'\mathbb{1}z & = & \mathbf{A}'\mathbf{L}y, \\ & & \|\xi\|_2 & \leq & \gamma. \\ & & \omega & \geq & 0. \end{array} \tag{3.16}$$

Opět platí silná věta o (SOCP) dualitě, jak ji můžeme nalézt například v Andersen a kol. (2002). Stačí ověřit, že v obou úlohách existuje přípustné řešení, pro které v omezeních platí ostré nerovnosti, což se dá snadno nahlédnout. Z toho můžeme vyvodit, že je-li y^* optimálním řešení 3.15, pak je také optimálním řešení 3.16 a obráceně.

Minimalizační úloha 3.16 je však obtížně řešitelná, protože matice omezení y není totálně unimodulární a nelze tak relaxovat celočíselnost y. Opět tedy můžeme využít metodu LR a s ní spojenou celočíselnou relaxaci. Definujeme duální funkci:

$$\begin{array}{rl} \theta_3(v) = \inf\limits_{y, \gamma, \xi, \omega, z} & \pi^{0'}\mathbf{L}y + \gamma + \pi^{0'}\omega + v'(-\xi - \mathbf{A}'\omega + \mathbf{A}'\mathbb{1}z - \mathbf{A}'\mathbf{L}y), \\ \text{s.t.} & y \in \hat{Y}, \|\xi\|_2 \leq \gamma, \omega \geq 0, \end{array} \tag{3.17}$$

jejíž hodnotu lze pro libovolné v nalézt v polynomiálním čase. Dále definujeme duální úlohu:

$$(D_3) \sup_{v \in \mathbb{R}^{N+1}} \theta_3(v).$$

V tomto případě opět platí silná věta o Lagrangeovské dualitě – splnění předpokladů lze snadno nahlédnout – z čehož plyne $(P_3) = (D_3)$.

Stejným způsobem můžeme řešit i úlohu minimalizace střední hodnoty času překrytí po sobě jdoucích prací. V tom případě bychom museli pracovat s aproximací střední hodnoty. Možností je využít například tu popsanou v 3.12.

3.4 Smíšené rozdělení

Třetí možnost volby $\mathcal{P}$ prezentovaná v práci Zhu a Fukushima (2009) je takzvaná *mixture-distribution*, neboli směsné rozdělení, kdy nejprve uvažujeme konečnou množinu $\{P_1, P_2, \ldots, P_K\}$ daných jednorozměrných rozdělení, která mohou být určena například jako maximálně věrohodné odhady vybraných rozdělení. Tento přístup dosud nebyl v souvislosti s úlohou FIS za nejistoty v nám známé literatuře uveden, jedná se tedy o původní výsledek.

Marginální rozdělení všech zpoždění uvažujeme v této sekci identická a $\mathcal{P}$ je v tomto případě množina všech uvažovaných jednorozměrných rozdělení. $\mathcal{P}$ definujeme jako konvexní obal množiny $\{P_1, \ldots, P_K\}$. Pro každé $P \in \mathcal{P}$ a $x \geq 0$ potom platí:

$$F_P(x) = \sum_{i=1}^{K} \lambda_i F_{P_i}(x),\ \sum_{i=1}^{K} \lambda_i = 1, \lambda_i \geq 0, i = 1, \ldots, K. \tag{3.18}$$

Nejprve tento přístup aplikujeme na úlohu minimalizace střední hodnoty počtu překryvů po sobě jdoucích prací 2.5 formulovanou jako úlohu nalezení toku grafem s minimální penalizací. Pro proměnnou y_i odpovídající hraně spojující práce j a k uvažujeme časy $t_i = s_k - f_j$ odpovídající délce intervalu mezi koncem jednoho úkolu a začátkem následujícího. Množinu všech $t_i, i = 1, \ldots, \bar{N}$ značíme $\bar{T}$. Formulace úlohy vypadá následovně:

$$\begin{array}{rrcl} \min\limits_{y \in Y} \max\limits_{\lambda} & \lambda' \mathbf{F} y, & & \\ \text{s.t.} & \mathbb{1}'\lambda & = & 1, \\ & \lambda & \geq & 0, \end{array} \tag{3.19}$$

kde

$$\mathbf{F} = \begin{pmatrix} 1 - F_{P_1}(t_1) & \ldots & 1 - F_{P_1}(t_{\bar{N}}) \\ \vdots & \vdots & \vdots \\ 1 - F_{P_K}(t_1) & \ldots & 1 - F_{P_K}(t_{\bar{N}}) \end{pmatrix}. \tag{3.20}$$

Formulaci a způsob řešení úlohy můžeme v tomto případě dále zjednodušit. Nechť y je pro řešení maximalizační podúlohy 3.19 pevně dané a $I \subset \{1, \ldots, \bar{N}\}$ je množina indexů taková, že $y_i = 1 \Leftrightarrow i \in I$. Potom $y_F = \mathbf{F}y$ je konstantní vektor, kde prvek na místě k je roven $\sum_{i \in I}(1 - F_{P_k}(t_i))$. Nechť maximální prvek vektoru y_F leží na místě m, lze snadno nahlédnout, že optimálním řešením maximalizační podúlohy 3.19 je pak vektor délky K, jehož prvek na místě m je roven jedné a ostatní jsou rovny nule. Úlohu 3.19 potom můžeme přeformulovat jako $\min\limits_{y \in Y} \max\limits_{k=1,\ldots,K} \mathbf{F}'_{\cdot k} y$, ale jelikož v tomto případě není možné prohodit min a max, zůstává úloha obtížně řešitelnou a je nutné využít některý z výše zmiňovaných heuristických algoritmů. Další možností je využití Lagrangeovské duality, jako tomu bylo v předchozích případech. Po převedení maximalizační podúlohy 3.19 na její duální tvar dostáváme následující úlohu

$$\begin{array}{rrcl} \min\limits_{y \in Y, \xi \in \mathbb{R}} & \xi, & & \\ \text{s.t.} & \mathbf{F}y & \leq & \mathbb{1}\xi. \end{array} \tag{3.21}$$

Analogicky jako v 3.10 zavádíme duální funkci $\theta_4(u)$ a aplikujeme celočíselnou relaxaci, Lagrangeovská duální úloha má potom tvar:

$$(D_4) \sup_{u \geq 0,\, u \in \mathbb{R}^k} \inf_{y \in \hat{Y},\, \xi \in \mathbb{R}} \quad \xi + u'(\mathbf{F}y - \mathbb{1}\xi). \tag{3.22}$$

Vidíme, že platí:

$$\theta_4(u) \neq -\infty \Leftrightarrow \mathbb{1}'u = 1.$$

Popsané výsledky stručně shrnuje následující tvrzení.

Tvrzení 5. *Úlohu 3.22 můžeme ekvivalentně řešit jako úlohu*

$$\max_{u \geq 0,\, u \in \mathbb{R}^k} \min_{y \in \hat{Y}} u'\mathbf{F}y, \quad \mathbb{1}'u = 1,$$

kde $\mathbf{F}$ *je jako v 3.20 a* $\hat{Y}$ *je výše popsaná množina přípustných toků s relaxovanou celočíselností.*

Důkaz. Tvrzení plyne z výše popsaného postupu, potažmo z tvrzení 6.

□

Tuto úlohu řešíme v rámci numerické studie v poslední části práce.

Analogický přístup můžeme volit i pro úlohu minimalizace střední hodnoty doby překrytí po sobě jdoucích prací. Nechť F_P je jako v 3.18 a nechť O_{jk} odpovídá době překryvu prací spojených hranou i s příslušející binární proměnnou y_i (značíme jako O_i), z linearity střední hodnoty plyne: $\mathsf{E}_P O_i = \sum_{k=1}^{K} \lambda_k \mathsf{E}_{P_k} O_i$. Formulace úlohy tedy bude stejná s tím rozdílem, že místo matice $\mathbf{F}$ použijeme matici $\mathbf{O} = (\mathbf{O}_{ki} = \mathsf{E}_{P_k} O_i)\,, k = 1, \ldots, K, i = 1, \ldots, \bar{N}$.

Tento přístup lze aplikovat i na úlohu maximalizace spolehlivosti rozvrhu navzdory nelinearitě ψ. Pro pevné y řešíme úlohu

$$\max_{\lambda} \sum_{i=1}^{\bar{N}} \psi\left(\sum_{k=1}^{K} \lambda_k F_{P_k}(t_i)\right) y_i, \quad \text{s.t. } \mathbb{1}'\lambda = 1, \lambda \geq 0. \tag{3.23}$$

Z konvexity ψ a *Jensenovy* nerovnosti plyne:

$$\begin{aligned} \sum_{i=1}^{\bar{N}} \psi\left(\sum_{k=1}^{K} \lambda_k F_{P_k}(t_i)\right) y_i &\leq \sum_{i=1}^{\bar{N}} \sum_{k=1}^{K} \lambda_k \psi\left(F_{P_k}(t_i)\right) y_i = \\ &= \sum_{k=1}^{K} \lambda_k \sum_{i=1}^{\bar{N}} \psi\left(F_{P_k}(t_i)\right) y_i \leq \max_k \{\sum_{i=1}^{\bar{N}} \psi\left(F_{P_k}(t_i)\right) y_i\}. \end{aligned} \tag{3.24}$$

z čehož okamžitě plyne, že maximum v 3.23 nastává, pokud F_P odpovídá jednomu z F_{P_k}. Úlohu potom můžeme formulovat a řešit stejně jako 3.19 s tím rozdílem, že místo matice $\mathbf{F}$ uvažujeme matici $\mathbf{P}$, kde $\mathbf{P}_{ki} = \psi\left(F_{P_k}(t_i)\right)$.

Výsledky přístupu použitého pro řešení úloh v této sekci a v sekci 3.2 shrnuje následující tvrzení.

Tvrzení 6. *Nechť* $\mathbf{T}_{p \times n}$ *je totálně unimodulární matice,* $c \in \mathbb{Z}^p$, $Y = \{y \in \mathbb{Z}^n : \mathbf{T}y = c\}$ *a* $X = \{x \in \mathbb{R}^m : \mathbf{B}x = b, l \leq x \leq u\}$. *Dále nechť pro úlohu:*

$$(P) \min_{y \in Y} \max_{x \in X} x'\mathbf{A}y$$

existuje optimální řešení (x^*, y^*).

Potom platí:

$$(P) = (D) \max_{x \in X} \min_{y \in \hat{Y}} x'\mathbf{A}y,$$

kde $\hat{Y} = \{y \in \mathbb{R}^n : \mathbf{T}y = c\}$ *a optimální hodnoty* (D) *se nabývá v bodě* (x^*, y^*).

Důkaz. Maximalizační podúlohu (P) převedeme na její duální tvar, čímž získáme úlohu ve tvaru:

$$\begin{array}{rrcl} (P_2) \min\limits_{y \in Y} \min\limits_{\alpha\,\beta\,\gamma} & \alpha' b + \beta' u + \gamma' l, & & \\ \text{s.t.} & \mathbf{B}'\alpha + \beta + \gamma & = & \mathbf{A}y, \\ & \alpha & \in & \mathbb{R}^k, \\ & \beta, \gamma & \in & \mathbb{R}^m, \\ & \beta & \geq & 0, \\ & \gamma & \leq & 0. \end{array}$$

Z existence optimálního řešení (P) a duality LP plyne existence optimálního řešení (P_2) a $(P_2) = (P)$. Dále pro $v \in \mathbb{R}^m$ zavedeme duální funkci:

$$\begin{array}{rl} \theta(v) = \inf\limits_{y\,\alpha\,\beta\,\gamma} & \alpha' b + \beta' u + \gamma' l + v'(\mathbf{A}y - \mathbf{B}'\alpha - \beta - \gamma), \\ \text{s.t.} & y \in Y, \alpha \in \mathbb{R}^k, \beta, \gamma \in \mathbb{R}^m, \beta \geq 0, \gamma \leq 0. \end{array}$$

Z totální unimodularity $\mathbf{T}$, celočíselnosti vektoru c a linearity funkce minimalizované v $\theta(v)$ v y plyne, že můžeme v definici $\theta(v)$ relaxovat celočíselnost y a platí:

$$\begin{array}{rl} \theta(v) = \inf\limits_{y\,\alpha\,\beta\,\gamma} & \alpha' b + \beta' u + \gamma' l + v'(\mathbf{A}y - \mathbf{B}'\alpha - \beta - \gamma), \\ \text{s.t.} & y \in \hat{Y}, \alpha \in \mathbb{R}^k, \beta, \gamma \in \mathbb{R}^m, \beta \geq 0, \gamma \leq 0. \end{array}$$

Formulujeme duální Lagrangeovskou úlohu:

$$(D) \sup_{v \in \mathbb{R}^m} \theta(v).$$

Obecně platí $(D) \leq (P)$. Ověříme předpoklady silné věty o Lagrangeovské dualitě. Splnění podmínky konvexity (P_2) je zjevné. Dále stačí ověřit *Slaterovu* podmínku:

$$0 \in \operatorname{int}\{\mathbf{A}y - \mathbf{B}'\alpha - \beta - \gamma |\, y \in \hat{Y}, \alpha \in \mathbb{R}^k, \beta, \gamma \in \mathbb{R}^m, \beta \geq 0, \gamma \leq 0\},$$

jejíž splnění se dá snadno nahlédnout.

Z toho plyne, že $(D) = (P_2) = (P) \in \mathbb{R}$, což dále implikuje $\mathbf{B}v = b$ a $l \leq v \leq u$, neboť v jiném případě $\theta(v) = -\infty$. Duální úlohu tedy můžeme ekvivalentně zapsat jako:

$$(D) \sup_{x \in X} \inf_{y \in \hat{Y}} x'\mathbf{A}y.$$

Jelikož (x^*, y^*) je přípustné řešení (D) a přípustné optimální řešení (P), je to i optimální řešení (D). Ve formulaci (D) tedy můžeme místo sup a inf psát max a min, čímž je tvrzení dokázáno.

□

V případě, že je X v tvrzení definována pomocí nerovnosti, můžeme postupovat analogicky nebo za pomoci zavedení skluzových proměnných v lineárních omezeních pro x, které nijak nepokazí možnost relaxace celočíselnosti y.

3.5 Znalost hodnot prvních momentů zpoždění

Jednou z nejpřirozenějších voleb charakterizace neznámého rozdělení při řešení robustních optimalizačních úloh zahrnujících náhodný prvek je jeho definice pomocí hodnot daných (zobecněných) momentů. Tento přístup byl aplikován například na úlohu prodavače novin v práci Scarf (1957) nebo při optimalizaci portfolia vzhledem k parciálním momentům, *VaR* a *CVaR* ztráty v práci Chen a kol. (2011). My budeme definovat množinu $\mathcal{P}$ pomocí hodnot prvních momentů pravděpodobnostního rozdělení. V souvislosti s úlohou FIS za nejistoty nebyl dosud tento přístup v nám známé literatuře uveden, jedná se tedy o původní výsledek.

3.5.1 Minimalizace střední hodnoty doby překrytí

Podobný přístup budeme nyní aplikovat na vybrané verze úlohy 2.7 minimalizace střední hodnoty doby překrytí po sobě jdoucích úkolů.

Definujeme množinu $\mathcal{P}_\mu = \{P : \mathsf{E}_P\, D = \mu\}$ pomocí vektoru středních hodnot zpoždění prací. Nepředpokládáme, že všechna zpoždění mají identická rozdělení. Řešenou úlohu můžeme zapsat ve tvaru:

$$\min_{y \in Y} \max_{P \in \mathcal{P}_\mu} \mathsf{E}_P \sum_{i=1}^{\bar{N}} y_i \min\left(\max(0, D_i - t_i), \hat{t}_i\right), \tag{3.25}$$

kde $t_i \in \bar{T}$ a $\hat{t}_i$ je délka pozdějšího z úkolů spojených hranou, které odpovídá y_i, koresponduje-li y_i s hranou $(j,k) \in \bar{E}$, potom $\hat{t}_i = f_k - s_k$. $\min\left(\max(0, D_i - t_i), \hat{t}_i\right)$ je nelineární, nekonvexní, nekonkávní a po částech lineární funkce náhodné veličiny D_i závisející na P, což dělá problém složitějším a znemožňuje přímé využití některých dosažených výsledků. Proto přistoupíme k určitým modifikacím úlohy, kde budeme uvažovat konvexní funkce.

Restrikce rozdělení délky překryvu

První spočívá v modifikaci uvažovaného (skutečného) rozdělení D pomocí restrikce jednotlivých D_i na interval $\left[0, t_i + \hat{t}_i\right]$ s tím, že předpokládáme znalost, nebo možnost odhadu střední hodnoty $\tilde{\mu}_i$ při této restrikci. Restrikci D značíme $\tilde{D}$. Nechť F_i a f_i jsou distribuční funkce a hustota D_i, hustotu $\tilde{f}_i$ zpoždění $\tilde{D}_i$ můžeme definovat následovně:

$$\tilde{f}_i(x) = \begin{cases} \frac{f_i(x)}{F_i(t_i + \hat{t}_i)} & x \in \left[0, t_i + \hat{t}_i\right] \\ 0 & \text{jinak.} \end{cases} \tag{3.26}$$

Další možností je ponechat hustotu $f_i(x)$ na intervalu $\left[0, t_i + \hat{t}_i\right]$ beze změny, jinde ji definovat jako 0 a do bodu $t_i + \hat{t}_i$ přidat masu o velikosti $1 - F_i(t_i + \hat{t}_i)$. V prvním případě by se jednalo o odhad pomocí useknutých (truncated) dat, v druhém případě by šlo o použití dat cenzorovaných.

Vektor středních hodnot $\tilde{D}$ značíme $\tilde{\mu}$. Definujeme množinu $\tilde{\mathcal{P}}_{\tilde{\mu}}$ všech rozdělení zpoždění, jejichž marginální rozdělení mají obor hodnot $\left[0, t_i + \hat{t}_i\right]$ a střední hodnotu $\tilde{\mu}_i, i = 1, \ldots, \bar{N}$.

V tomto případě platí, že $\min\left(\max(0,\tilde{D}_i - t_i),\hat{t}_i\right) = \left(\tilde{D}_i - t_i\right)_+$ [3], což je konvexní funkce náhodné veličiny $\tilde{D}_i$.

Dále můžeme využít výsledku odvozeného v *Example 2.2* v práci Dupačová (2011). Nechť $f(x,z) = \sum_{i=1}^{\bar{N}} f_i(x,z_i)$, kde pro fixní x jsou $\forall i\, f_i(x,z_i)$ konvexní funkce proměnné z. Dále definujeme $\mathcal{P}_\mu$ jako množinu všech rozdělení náhodného vektoru $Z = (Z_1,\ldots,Z_{\bar{N}})$, pro který platí, že náhodné veličiny Z_i nabývají hodnot z intervalu $[a_i,b_i]\,, a_i < b_i$ a $\mathsf{E}\ Z_i = \mu_i$. Potom platí:

$$\max_{P\in\mathcal{P}_\mu} \mathsf{E}_P\, f(x,Z) = \sum_{i=1}^{\bar{N}} \lambda_i f_i(x,a_i) + \sum_{i=1}^{\bar{N}} (1-\lambda_i) f_i(x,b_i),\ \lambda_i = \frac{b_i - \mu_i}{b_i - a_i}. \tag{3.27}$$

V našem případě nahradíme vektor x vektorem všech t a y, náhodnou veličinu Z_i nahradíme zpožděním $\tilde{D}_i$ se střední hodnotu $\tilde{\mu}_i$, $a_i = 0, b_i = t_i + \hat{t}_i$ a pro všechna i definujeme funkci f_i identicky jako $f_i(t_i,y_i,\tilde{D}_i) = y_i(\tilde{D}_i - t_i)_+$, což je, jak bylo řečeno dříve, konvexní funkce náhodné veličiny $\tilde{D}_i$. Dosadíme-li do 3.27, platí:

$$f_i(t_i,y_i,0) = 0 \text{ a } \tilde{w}_i(y_i) = (1-\lambda_i) f_i(t_i,y_i,b_i) = y_i \frac{\tilde{\mu}_i \hat{t}_i}{t_i + \hat{t}_i}, i = 1,\ldots,\bar{N}. \tag{3.28}$$

Výše uvedené stručně shrnuje následující tvrzení.

Tvrzení 7. *Úlohu*

$$\min_{y\in Y} \max_{P\in\tilde{\mathcal{P}}_{\tilde{\mu}}} \mathsf{E}_P \sum_{i=1}^{\bar{N}} y_i \min\left(\max(0,\tilde{D}_i - t_i),\hat{t}_i\right),$$

můžeme reformulovat jako:

$$\min_{y\in\hat{Y}} \sum_{i=1}^{\bar{N}} \tilde{w}_i(y_i),$$

kde $\tilde{w}_i$ je jako v 3.28.

Důkaz. Tvrzení plyne z výše popsaného postupu a z možnosti relaxovat celočíselnost y, plynoucí z toho, že se jedná o úlohu s lineární účelovou funkcí, totálně unimodulární maticí omezení a celočíselnou pravou stranou omezení.

□

Neomezená délka překryvu

Druhou možností je uvažovat jako překryv dobu, po kterou se na prvním úkolu pracuje po začátku následujícího úkolu, prakticky tedy připouštíme délku překryvu delší než délku úkolu následujícího po úkolu zpožděném.

Definujeme nekonečnou posloupnost restrikcí $\tilde{D}_{i,n}\ n \in \mathbb{N}$ definovaných na intervalech $[0, t_i + \Delta_n]$ se střední hodnotou $\tilde{\mu}_{i,n}$. Předpokládáme, že $\Delta_n \geq 0$ a

[3] Jedná se o rovnost dvou funkcí téže náhodné veličiny, potažmo o rovnost dvou náhodných veličin.

$\Delta_n \nearrow \infty$ pro $n \to \infty$, přirozeně potom $\tilde{\mu}_{i,n} \underset{n\to\infty}{\longrightarrow} \mu_i$. Příslušnou množinu pravděpodobnostních rozdělení značíme $\mathcal{P}_n$ a úloha je formulována následovně:

$$\min_{y\in Y} \max_{P\in\mathcal{P}_n} \mathsf{E}_P \sum_{i=1}^{\bar{N}} y_i \max\left(\tilde{D}_{i,n} - t_i\right)_+. \tag{3.29}$$

S využitím 3.27 pro maximalizační podúlohu 3.29 potom platí:

$$\max_{P\in\mathcal{P}_n} \mathsf{E}_P \sum_{i=1}^{\bar{N}} y_i \left(\tilde{D}_{i,n} - t_i\right)_+ = \sum_{i=1}^{\bar{N}} y_i \frac{\tilde{\mu}_{i,n}\Delta_n}{t_i + \Delta_n} \underset{n\to\infty}{\longrightarrow} \sum_{i=1}^{\bar{N}} y_i\mu_i = \sup_{P\in\mathcal{P}_\mu} \mathsf{E}_P \sum_{i=1}^{\bar{N}} y_i \left(D_i - t_i\right)_+. \tag{3.30}$$

Výše uvedený přístup shrnuje následující tvrzení.

Tvrzení 8. *Nechť je doba překryvu úkolů spojených hranou, které odpovídá proměnná y_i, definována jako $(D_i - t_i)_+$. Úlohu 3.29 minimalizace střední hodnoty doby překrytí po sobě jdoucích úkolů při znalosti středních hodnot zpoždění potom můžeme reformulovat jako úlohu:*

$$\min_{y\in\hat{Y}} \sum_{i=1}^{\bar{N}} y_i\mu_i.$$

Důkaz. Tvrzení plyne z výše uvedených výsledků a z možnosti relaxovat celočíselnost y, plynoucí z toho, že se jedná o úlohu s lineární účelovou funkcí, totálně unimodulární maticí omezení a celočíselnou pravou stranou omezení.

□

Stejně jako v předchozím případě se jedná o úlohu LP, která je snadno řešitelná v polynomiálním čase. Kvalitu této aproximace zkoumáme na simulovaných datech v numerické části práce.

3.5.2 Minimalizace střední hodnoty počtu překryvů

Na tomto místě řešíme úlohu minimalizace střední hodnoty počtu překryvů při znalosti středních hodnot jednotlivých zpoždění. Jako množinu všech možných rozdělení tedy uvažujeme dříve definovanou $\mathcal{P}_\mu$. Formulace úlohy je potom:

$$\min_{y\in Y} \sup_{P\in\mathcal{P}_\mu} \sum_{i=1}^{\bar{N}} y_i \mathsf{P}(D_i > t_i). \tag{3.31}$$

Vzhledem k tomu, že pro jednotlivá zpoždění neuvažujeme obecně stejná rozdělení, lze pro pevné y reformulovat maximalizační podúlohu 3.31 jako:

$$\sum_{i=1}^{\bar{N}} y_i \sup_{\{D_i : D_i \geq 0,\, \mathsf{E}\, D_i = \mu_i\}} \mathsf{P}(D_i > t_i). \tag{3.32}$$

Z *Věty 3.3* uvedené v práci Bertsimas a Popescu (2005) plyne, že

$$\sup_{\{D_i : D_i \geq 0,\, \mathsf{E}\, D_i = \mu_i\}} \mathsf{P}(D_i > t_i) = \begin{cases} 1 & t_i \leq \mu_i \\ \frac{\mu_i}{t_i} & t_i > \mu_i. \end{cases} \tag{3.33}$$

Dosažené výsledky shrnuje následující tvrzení.

Tvrzení 9. *Úlohu 3.31 lze ekvivalentně reformulovat jako úlohu:*

$$\min_{y \in \hat{Y}} \sum_{i=1}^{\bar{N}} \left(\frac{\mu_i}{t_i} + \frac{t_i - \mu_i}{t_i} \mathbb{1}\left[t_i \leq \mu_i\right] \right) y_i$$

Důkaz. Tvrzení plyne z výše uvedeného postupu a z toho, že se jedná o úlohu s lineární účelovou funkcí, totálně unimodulární maticí omezení a celočíselnou pravou stranou omezení.

□

Při definici $\mathcal{P}$ pomocí střední hodnoty a rozptylu není pro $D_i \geq 0$ k dispozici uzavřený tvar $\sup_{\mathcal{P}} \mathsf{P}(D_i > t_i)$ jako v 3.33 a úloha vedoucí k jeho nalezení je *NP–těžká*. Při definici $\mathcal{P}$ pomocí prvních tří momentů je pro $D \geq 0$ uzavřený tvar suprema k dispozici a úloha je potom opět snadno řešitelná. Podrobnosti můžeme nalézt v práci Bertsimas a Popescu (2005).

4. Numerická studie

V této závěrečné části práce představíme výsledky numerické studie, ve které řešíme dvě výše představené úlohy. Jako první řešíme úlohu 3.19 minimalizace střední hodnoty počtu překryvů při definici $\mathcal{P}$ jako množiny všech konvexních kombinací několika daných pravděpodobnostních rozdělení. Dále se zabýváme kvalitou řešení získaných pomocí aproximace úlohy minimalizace střední hodnoty doby překrytí při znalosti středních hodnot zpoždění popsané v tvrzení 8.

Ke všem výpočtům byl použit software *AMPL 3.5.0.201802211250* a počítač s operačním systémem *Windows 10 Pro*, procesorem *Intel Core i5 M560* 2.67GHz a 4GB RAM. Pro řešení úloh LP byl použit solver *Cplex 12.9.0.0* a pro řešení MIP úloh byl použit solver *Gurobi 8.1.0*. Data pro všechny řešené úlohy byla generována pomocí programu *R* (R Core Team, 2014). Kódy použité k výpočtům a generování dat jsou součástí elektronické přílohy k práci.

4.1 Minimalizace střední hodnoty počtu překryvů při smíšeném rozdělení

Ke generování sad úloh přistupujeme velmi podobně jako autoři v práci Branda (2018). Aby byla zaručena existence přípustného rozvrhu, generujeme pro každý z daného počtu strojů $\tilde{M}$ určitý počet úkolů j, tedy celkový počet úkolů $J = \tilde{M} \cdot j$, a následně při řešení úlohy předpokládáme, že počet strojů k dispozici je $M > \tilde{M}$. Tím zároveň vzniká prostor pro optimalizaci, protože potom existuje více než jedno přípustné řešení. Délky úkolů jsou generovány z exponenciálního rozdělení se střední hodnotou $\mu^u = 3$ a mezery mezi pracemi jsou generovány z exponenciálního rozdělení se střední hodnotou $\mu^m = 4$. Takto uvažujeme tři možnosti, potažmo velikosti jednotlivých úloh:

1. $\tilde{M} = 30$, $j = 20$, $J = 600$ a $M = 45$,
2. $\tilde{M} = 40$, $j = 20$, $J = 800$ a $M = 60$,
3. $\tilde{M} = 50$, $j = 25$, $J = 1250$ a $M = 60$.

Pro každou z možností generujeme po 10 úlohách. Počet nerovností ve formulaci se při řešení dané úlohy odvíjí přímo od počtu úkolů J a je roven $2 + 2J$. Oproti tomu počet proměnných, který je roven $1 + 3J + |\bar{E}|$, závisí především na počtu hran s nenulovou penalizací $|\bar{E}|$, jež má výsledný graf, který je pro každý scénář jiný. Údaje o velikosti jednotlivých úloh můžeme najít ve shrnující tabulce 4.1.

Pro definici $\mathcal{P}$ uvažujeme tři různá rozdělení ($K = 3$), a to:

- *Gompertzovo* s přidanou masou $p = 0{,}9$ v 0 a parametry $a = 10^{-4}$ a b =0,22,
- *Gompertzovo* s přidanou masou $p = 0{,}9$ v 0 a parametry $a = 1{,}4 \times 10^{-4}$ a b =,21,
- *Gompertzovo* s přidanou masou $p = 0{,}9$ v 0 a parametry $a = 1{,}8 \times 10^{-4}$ a b =0,20.

Předpokládáme parametrizaci, kdy distribuční funkce F_{pab} má tvar

$$F_{pab}(x) = p + (1-p)\left(1 - exp\left(-\frac{b}{a}\left(exp\left(ax\right) - 1\right)\right)\right) \quad x \geq 0.$$

K řešení úlohy přistupujeme pomocí metody Lagrangeovy relaxace (LR) a při úpravě postupujeme analogicky jako v důkazu tvrzení 6. Pro připomenutí znovu uvádíme formulaci úlohy 3.22:

$$\sup_{u \geq 0,\, u \in \mathbb{R}^K} \inf_{y \in \hat{Y},\, \xi \in \mathbb{R}} \xi + u'(\mathbf{F}y - \mathbb{1}_K \xi). \tag{4.1}$$

Z tvrzení 5 plyne, že numericky řešená úloha má tvar:

$$\max_{u \in \mathbb{R}^K} \min_{y \in \hat{Y}} u'\mathbf{F}y, \quad u \geq 0, u'\mathbb{1}_K = 1. \tag{4.2}$$

Minimalizační podúloha 4.2 je pro pevné u snadno řešitelná v polynomiálním čase a problém tedy zůstává v nalezení jeho optimální hodnoty skrze sekvenci iterací. Úprava multiplikátorů probíhá v jednotlivých iteracích podobně, jako je popsáno v práci Fisher (2004) pro obecnou lineární MIP úlohu, a to pomocí upravené metody subgradientů (v našem případě je subgradient v iteraci k roven $\mathbf{F}y_k$). Rozdílem je, že hodnoty multiplikátorů vždy dělíme jejich součtem, protože musí být dodržena podmínka $u'\mathbb{1}_K = 1$. Pro tuto metodu není ani v její původní formě zaručena konvergence ke globálnímu maximu, v praxi však metoda přináší dobré výsledky. Výsledné řešení značíme y^*. Zápis algoritmu LR je následující:

$\tilde{y} := \arg\min_{y \in \hat{Y}} \mathbf{F}_{h\cdot} y$
$UB := \max_i (\mathbf{F}_{i\cdot}\tilde{y})$
$u_1 := 1/K \mathbb{1}_K$
$\lambda := 1$
$LB := 0$
$s := 0$
for $k := 1, \ldots, Z$ **do**
  $y_k := \arg\min_{y \in \hat{Y}} u_k'\mathbf{F}y$
  if $u_k'\mathbf{F}y_k > LB + \epsilon$ **then**
    $LB := u_k'\mathbf{F}y_k$
  else
    $s := s + 1$
  end if
  if $s = 2$ **then**
    $\lambda := \lambda/2$
    $s := 0$
  end if
  $\tilde{u}_{k+1} := u_k + \lambda_k \frac{UB - u_k'\mathbf{F}y_k}{\|\mathbf{F}y_k\|_2}\mathbf{F}y_k$
  $u_{k+1} := \frac{\tilde{u}_{k+1}}{\mathbb{1}_K'\tilde{u}_{k+1}}$
end for
if $\max_i(\mathbf{F}_{i\cdot}y_Z) \leq UB$ **then**
  $y^* := y_Z$
else

$y^* := \tilde{y}$

end if

Z principu LR plyne, že v každé iteraci k bude hodnota duální funkce $u_k'\mathbf{F}y_k$ vždy menší než UB, což však nemusí platit pro hodnotu účelové funkce v aktuálním řešení. Proto je v algoritmu poslední podmínka pro volbu výsledného řešení. Další možností je zpětně zkoumat řešení v jednotlivých iteracích a vybrat nejlepší z nich. Při řešení úloh vedoucích k hodnotám v tabulce 4.1 nebyla tato podmínka zahrnuta pro možnost přesnější prezentace výsledků metody LR. Optimální řešení získané pomocí metody LR dále značíme y^*

K uvedenému algoritmu lze přidat ukončovací podmínku týkající se například změny LB, nebo změny hodnot $\lambda_k \frac{UB - u_k'\mathbf{F}y_k}{\|\mathbf{F}y_k\|_2}$, kdy algoritmus zastaví, pokud je změna příliš malá. V praxi je však používána i uvedená forma s pevným počtem iterací. V našem případě jsme volili $Z = 10$.

Index h můžeme volit libovolně z množiny $\{1, \ldots, Z\}$, v našem případě $h = 2$. Hodnotu ϵ jsme během výpočtů volili jako 10^{-6}. Jeho volba záleží především na řádu hodnot ve kterých se pohybuje účelová funkce a velikost rozdílů jejích hodnot pro různá přípustná řešení.

Všech 30 úloh počítáme také jako MIP úlohu 3.21 s $5 + 2J$ omezeními[1] a $1 + 2J + |\bar{E}|$ proměnnými. Získané výsledky potom porovnáváme s výsledky získanými pomocí přístupu používajícím metodu LR.

V tabulce 4.1 jsou uvedeny popisné statistiky shrnující výsledky výpočtů a porovnání formulací pomocí LR a MIP. Každý sloupec shrnuje výsledky výpočtu 10 úloh pro dané hodnoty M a J.

V prvním řádku jsou průměrné počty hran s penalizací v jednotlivých úlohách, což přibližně odpovídá počtu proměnných.

Další čtyři řádky uvádějí průměrnou, průměrnou relativní (vzhledem k hodnotám dosaženým pomocí LR), maximální a minimální hodnotu $UB - \max_i(\mathbf{F}_{i\cdot}y^*)$, tedy rozdílu hodnoty účelové funkce v bodě $\tilde{y}$ a y^*. Až na jeden případ bylo ve všech dosaženo nižší hodnoty účelové funkce, než v počátečním řešení.

Následující čtyři řádky uvádějí průměrnou, průměrnou relativní (vzhledem k hodnotám dosaženým pomocí MIP formulace), maximální a minimální hodnotu rozdílu $\max_i(\mathbf{F}_{i\cdot}y^*) - m^*$, kde m^* značí hodnotou účelové funkce v řešení získaném výpočtem MIP formulace úlohy. Hodnota $\max_i(\mathbf{F}_{i\cdot}y^*)$ byla ve všech případech větší, rozdíl však není příliš velký a námi navrhovaná metoda je mnohem rychlejší, jak dokládají poslední dva řádky.

Další čtyři řádky porovnávají MIP formulaci s její LP relaxací. Stejně jako v předchozích případech uvádíme průměrnou, průměrnou relativní (vzhledem k hodnotám dosaženým pomocí LP relaxace), maximální a minimální hodnotu rozdílu hodnot účelových funkcí v optimálním, potažmo výsledném řešení.

Poslední dva řádky porovnávají výpočetní/časovou náročnost formulace pomocí LR a MIP. První z nich ukazuje průměrný CPU čas potřebný k vyřešení jedné iterace LR. Údaj uvádíme v tomto formátu, neboť vhodný počet iterací se bude pro jiná data/rozdělení nebo úlohu lišit. Pro přímé srovnání délky výpočtu v našem případě stačí uvedené hodnoty vynásobit 10. Druhý z nich ukazuje průměrný čas pro vyřešení MIP formulace. Vzhledem k nízké variabilitě hodnot neuvádíme minima a maxima.

[1]Tento počet vyplývá z formátu dat užitého pro řešení úlohy.

	$J = 600$ $M = 30$	$J = 800$ $M = 60$	$J = 1250$ $M = 60$
pr. $\lvert \bar{E} \rvert$	173 004,9	307 437,8	756 888,1
pr. LR gap	0,00525	0,00523	0,00917
pr. LR rel. gap	0,012 %	0,009 %	0,009 %
max LR gap	0,01293	0,01322	0,01764
min LR gap	0,00004	-0,00000	0,00006
pr. MIPxLR gap	0,00423	0,00575	0,00586
pr. MIPxLR rel. gap	0,009 %	0,010 %	0,006 %
max MIPxLR gap	0,00815	0,01083	0,00911
min MIPxLR gap	0,00012	0,00112	0,00045
pr. MIP gap	0,00002	0,00002	0,00003
pr. MIP rel. gap	4×10^{-5} %	3×10^{-5} %	3×10^{-5} %
max MIP gap	0,00002	0,00005	0,00005
min MIP gap	0,00001	0,00001	0,00003
pr. CPU čas 1 iter. LR	0,61 s	1,17 s	3,40 s
pr. CPU čas MIP	10,00 s	32,65 s	115,04 s

Tabulka 4.1: Shrnutí výsledků numerické studie.

```
k         step         Lagrangian    obj. f. val.
1      1.3891377e-05    45.0928810    45.1765317
2      1.3891395e-05    45.0928809    45.1765246
3      1.3890748e-05    45.0928848    45.1765196
4      1.3891329e-05    45.0928813    45.1765212
5      6.9455236e-06    45.0928830    45.1765134
6      6.9460240e-06    45.0928769    45.1765040
7      6.9452174e-06    45.0928867    45.1765424
8      6.9454618e-06    45.0928837    45.1765284
9      3.4728534e-06    45.0928807    45.1765249
10     3.4728025e-06    45.0928820    45.1765324
11     1.7364763e-06    45.0928783    45.1765153
12     1.7363870e-06    45.0928827    45.1765128
13     8.6822800e-07    45.0928793    45.1765172
14     8.6818030e-07    45.0928840    45.1765296
15     4.3408489e-07    45.0928850    45.1765143
```

Tabulka 4.2: Příklad průběhu 15 iterací algoritmu pro $(J, M) = (600, 45)$.

V tabulce 4.2 je k dispozici přehled výsledků v průběhu 15 iterací algoritmu. Ve sloupci *step* jsou hodnoty $\lambda_k \frac{UB - u_k' \mathbf{F} y_k}{\|\mathbf{F} y_k\|_2}$, tedy určité délky kroku. V druhém sloupci jsou hodnoty duální funkce v jednotlivých iteracích a v posledním je hodnota účelové funkce původní úlohy v aktuálním řešení y_k.

Uvedená metoda LR se obecně jeví jako vhodná možnost pro výše zmíněné minimaxové úlohy, jejichž MIP formulace již nebude, vzhledem ke svojí velikosti, spočitatelná v rozumném čase[2] a kdy pomocí LR snadno získáme relativně dobré přípustné řešení. To může případně sloužit jako startovní bod pro nějaký heuristický algoritmus nebo pro MIP formulaci daného problému. Stejný přístup by šel aplikovat na libovolný případ úlohy, na kterou lze aplikovat tvrzení 6 o řešení speciálního druhu minimaxové úlohy.

Jak je vidět z výsledků numerické studie, nedosahujeme pomocí metody LR optimálního řešení, jak by tomu teoreticky podle tvrzení 6 mohlo být. Problematickou částí zůstává způsob úpravy multiplikátorů v jednotlivých iteracích, konkrétně krok škálování zaručující $\mathbb{1}_K' u_k = 1$. Testovali jsem i možnost projekce $\tilde{u}_{k+1}$ do roviny bodů se součtem prvků rovným jedné, ale s horšími výsledky. Pokud by se podařilo nedostatek metody v tomto směru vyřešit, stala by se LR kvalitní a rychlou alternativou k MIP pro všechny úlohy na které lze aplikovat tvrzení 6. Toto ponecháváme jako předmět k dalšímu zkoumání.

4.2 Aproximace střední hodnoty doby překrytí při znalosti středních hodnot zpoždění

V této sekci se budeme věnovat kvalitě aproximace úlohy minimalizace střední hodnoty doby překrytí při znalosti středních hodnot zpoždění, jak je uvedena v tvrzení 8. Aproximace spočívá v redefinici překryvu jakožto shora neomezené doby zpoždění přesahující začátek následující práce. Řešená úloha má tvar:

$$\min_{y \in \hat{Y}} \sum_{i=1}^{\bar{N}} y_i \mu_i,$$

kde $\hat{Y}$ je, stejně jako výše, množina všech přípustných toků (rozvrhů) při relaxaci celočíselnosti proměnné y.

Při generování náhodných scénářů jsme postupovali podobně jako v předchozím případě. Rozdíl je v tom, že pro každou ze tří velikostí úlohy, charakterizovanou hodnotami M, J a $\tilde{M}$, jsme generovali po 50 úlohách pro $(\mu^m, \mu^u) = (1, 2)$ a pro $(\mu^m, \mu^u) = (3, 4)$. Každé z těchto 6 možností odpovídá jeden sloupec shrnující tabulky 4.3. Oproti předchozímu případu volíme v nejmenších úlohách $\tilde{M} = 20$ a v největších $M = 75$.

Při řešení úlohy předpokládáme, že skutečné rozdělení všech zpoždění je exponenciální s přidanou hmotou 0.9 v 0, pro které lze snadno vyjádřit a spočítat střední hodnotu doby překrytí. Střední hodnoty jednotlivých zpoždění (v případě, že nastane) byly vygenerovány z rovnoměrného rozdělení na intervalu $(3, 6)$.

Při posuzování kvality aproximace postupujeme tím způsobem, že nejprve spočítáme optimální řešení při úplné znalosti rozdělení, jedná se tak o spodní

[2] V testovací úloze s $|\bar{E}| \sim 1{,}9 \times 10^6$ byl čas potřebný pro vyřešení jedné iterace LR přibližně 11 s a čas potřebný pro vyřešení MIP formulace přibližně 497 s.

	$J = 400$		$J = 800$		$J = 1250$	
	$M = 30$		$M = 60$		$M = 75$	
(μ^m, μ^u)	$(1,2)$	$(3,4)$	$(1,2)$	$(3,4)$	$(1,2)$	$(3,4)$
pr. $(a^* - l^*)$	1,130	0,770	3,089	2,216	5,602	4,853
min $(a^* - l^{*)}$	0,263	0,088	1,764	0,983	2,329	3,329
max $(a^* - l^*)$	1,981	1,812	4,365	3,251	7,687	6,852
s.o. $(a^* - l^*)$	0,419	0,343	0,624	0,574	1,077	0,822
pr. rel. $(a^* - l^{*)}$ (%)	40,271	53,755	47,744	60,919	49,708	71,354
min rel. $(a^* - l^{*)}$ (%)	9,377	4,604	23,439	23,936	18,223	43,344
max rel. $(a^* - l^*)$ (%)	76,037	129,406	75,253	89,651	78,807	102,847
s.o. rel. $(a^* - l^*)$ (%)	16,586	26,653	11,845	18,120	10,856	15,290
pr. $(u^* - a^*)$	4,990	4,908	11,201	11,904	17,139	18,201
min $(u^* - a^*)$	4,299	4,306	9,787	11,055	15,181	16,211
max $(u^* - a^*)$	6,033	5,497	12,384	13,169	19,009	20,041
s.o. $(u^* - a^*)$	0,379	0,293	0,520	0,493	0,992	0,798
pr. rel $(u^* - a^*)$ (%)	127,203	221,423	116,627	203,335	101,678	155,918
min rel. $(u^* - a^{*)}$ (%)	89,866	134,074	89,306	161,029	81,952	119,422
max rel. $(u^* - a^*)$ (%)	199,480	316,229	153,156	259,684	125,823	181.563
s.o. rel. $(u^* - a^*)$ (%)	23,045	37,142	11,162	24,218	11,242	13,928
pr. $\|\bar{E}\|$	75248,1	75826,2	300941,3	303449,4	743564,6	748689,2
pr. CPU čas (s)	0,42	0,42	2,06	2,00	6,45	5,88

Tabulka 4.3: Porovnání výsledků dosažených pomocí aproximace s jejich horní a dolní hranicí při exponenciálním rozdělení.

hranici, které lze při řešení pomocí aproximace dosáhnout; hodnotu značíme l^*. Dále nalezneme optimální řešení aproximace a spočteme jeho hodnotu při úplné znalosti rozdělení; značíme a^*. Obě hodnoty porovnáváme a shrnutí můžeme najít v prvních 8 řádcích tabulky 4.3, kde uvádíme průměrnou, maximální a minimální hodnotu rozdílu l^* a a^* a jeho směrodatnou odchylku v 50 úlohách pro dané hodnoty M, J, μ^u a μ^m. Stejné údaje uvádíme i pro relativní rozdíl l^* a a^* vzhledem k hodnotě l^*.

Provádíme i porovnání a^* s horní hranicí, kterou je nejhorší řešení, kterého lze vzhledem ke skutečnému rozdělení dosáhnout; hodnotu značíme u^*. Shrnutí je uvedeno na posledních 8 řádcích tabulky 4.3, kde pro rozdíl u^* a a^* uvádíme stejné údaje jako v první půlce tabulky. Relativní rozdíl je počítán vzhledem k hodnotě a^*.

Vidíme, že hodnoty a^* jsou napříč všemi velikostmi úloh mnohem blíže k l^*, než k u^*. a^* je v průměru přibližně 1,5× větší než l^* a u^* má v průměru 2-3× vyšší hodnotu než a^*. V této modelové situaci se tedy uvedená aproximace jeví jako poměrně kvalitní, neboť její výsledky nejsou příliš daleko od jejich spodní hranice a lze je velmi snadno získat jako řešení LP. Tato aproximace se však může zdát příliš zjednodušující a v případě, že lze získat odhady středních hodnot pro restrikce zpoždění, může se jako vhodnější jevit aproximace popsaná v tvrzení 7.

Poslední 2 řádky uvádějí doplňující informace o průměrném počtu hran s penalizací $\bar{E}$ a průměrném CPU času potřebném pro vyřešení úlohy.

Závěr

Tématem práce jsou rozvrhovací úlohy s fixními začátky a konci úkolů (FIS) za nejistoty způsobené možností zpoždění konce práce. Pozornost byla věnována především úlohám s minimaxovou formulací, kde neuvažujeme konkrétní pravděpodobnostní rozdělení, ale nějakou danou množinu, a snažíme se nejlepším možným způsobem zajistit proti nejhoršímu možnému případu rozdělení náhodného zpoždění.

Nejprve popisujeme základní deterministické úlohy FIS, možnosti jejich formulace, modifikace a řešení. Dále uvádíme příklady praktické aplikace těchto úloh.

V druhé části práce zavádíme výše zmíněný princip minimaxu pro FIS úlohy za nejistoty. Zkoumáme dvě známé úlohy FIS za nejistoty, konkrétně úlohu minimalizace střední hodnoty počtu překrytí a úlohu maximalizace spolehlivosti rozvrhu při modelování sdruženého rozdělení zpoždění pomocí *Archimedovské* kopule. Dále uvádíme úlohu minimalizace střední hodnoty doby překrytí práce na po sobě jdoucích úkolech, kterou jsme v dostupné literatuře v souvislosti s FIS nezaznamenali. Pro úlohy uvádíme jejich různé reformulace, například jako robustní úlohu barvení grafu nebo jako úlohu nalezení toku sítí s minimální penalizací a dále jejich možné modifikace.

Ve třetí, stěžejní části práce se věnujeme řešení úloh představených v druhé části formulovaných jako nalezení toku grafem s minimální penalizací pro vybrané množiny pravděpodobnostních rozdělení. Pro ně představujeme známý koncept *stressing distribution* (představeno pro úlohu FIS v práci Branda (2018)) a dále koncepty *box-uncertainty*, množiny nejistoty ve tvaru elipsoidu, směsného rozdělení a definice pomocí hodnot daných momentů, které se v souvislosti s robustifikací úlohy FIS za nejistoty objevují prvně. Pro uvedené úlohy uvádíme jejich formulace a navrhujeme možnosti řešení.

Práci zakončuje numerická studie dvou vybraných úloh uvedených ve třetí části práce. Prezentujeme implementaci řešení Lagrangeovské relaxace úlohy minimalizace střední hodnoty počtu překryvů pro množinu nejistoty definovanou pomocí směsného rozdělení. Uvádíme popis řešené úlohy a použitého algoritmu, prezentujeme výsledky a srovnáváme je s řešením úlohy formulované jako MIP. Upozorňujeme na výhody i nedostatky použitého přístupu. Dále prezentujeme výsledky zkoumání kvality aproximace úlohy minimalizace střední hodnoty doby překrytí při znalosti středních hodnot zpoždění, která se jeví jako uspokojivá.

Seznam použité literatury

ANDERSEN, E. D., ROOS, C. a TERLAKY, T. (2002). Notes on duality in second order and p-order cone optimization. *Optimization: A Journal of Mathematical Programming and Operations Research*, **51**(4), 627–643.

BERTSIMAS, D. a POPESCU, I. (2005). Optimal inequalities in probability theory: A convex optimization approach. *SIAM Journal on Optimization*, **15**(3), 780–804.

BOUZINA, K. I. a EMMONS, H. (1996). Interval scheduling on identical machines. *Journal of Global Optimization*, **9**(3-4), 379–393.

BOYD, S. a VANDENBERGHE, L. (2009). *Convex optimization*. Seventh printing with corrections. Cambridge university press, Cambridge. ISBN 978-0-521-83378-3.

BRANDA, M. (2018). Distributionally robust fixed interval scheduling on parallel identical machines under uncertain finishing times. *Computers & Operations Research*, **98**, 231–239.

BRANDA, M. a HÁJEK, Š. (2017). Flow-based formulations for operational fixed interval scheduling problems with random delays. *Computational Management Science*, **14**(1), 161–177.

BRANDA, M., NOVOTNỲ, J. a OLSTAD, A. (2016). Fixed interval scheduling under uncertainty–a tabu search algorithm for an extended robust coloring formulation. *Computers & Industrial Engineering*, **93**, 45–54.

CHEN, L., HE, S. a ZHANG, S. (2011). Tight bounds for some risk measures, with applications to robust portfolio selection. *Operations Research*, **59**(4), 847–865.

DEEP, K., SINGH, K. P., KANSAL, M. L. a MOHAN, C. (2009). A real coded genetic algorithm for solving integer and mixed integer optimization problems. *Applied Mathematics and Computation*, **212**(2), 505–518.

DIJKSTRA, M. C., KROON, L. G., VAN NUNEN, J. A. a SALOMON, M. (1991). A dss for capacity planning of aircraft maintenance personnel. *International Journal of Production Economics*, **23**(1-3), 69–78.

DUPAČOVÁ, J. (2011). Uncertainties in minimax stochastic programs. *Optimization*, **60**(10-11), 1235–1250.

FISHER, M. L. (2004). The lagrangian relaxation method for solving integer programming problems. *Management science*, **50**(12_supplement), 1861–1871.

GALLEGO, G. a MOON, I. (1993). The distribution free newsboy problem: review and extensions. *Journal of the Operational Research Society*, **44**(8), 825–834.

KROON, L. a KOLEN, A. (1991). On the computational complexity of (maximum) class scheduling. *European Journal of Operational Research*, pages 23–38.

Kroon, L. G., Salomon, M. a Van Wassenhove, L. N. (1995). Exact and approximation algorithms for the operational fixed interval scheduling problem. *European Journal of Operational Research*, **82**(1), 190–205.

Kroon, L. G., Salomon, M. a Van Wassenhove, L. N. (1997). Exact and approximation algorithms for the tactical fixed interval scheduling problem. *Operations Research*, **45**(4), 624–638.

McNeil, A. J., Nešlehová, J. a kol. (2009). Multivariate archimedean copulas, d-monotone functions and 1-norm symmetric distributions. *The Annals of Statistics*, **37**(5B), 3059–3097.

Pflug, G. C., Pichler, A. a Wozabal, D. (2012). The 1/n investment strategy is optimal under high model ambiguity. *Journal of Banking & Finance*, **36**(2), 410–417.

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Rockafellar, R. T., Uryasev, S. a kol. (2000). Optimization of conditional value-at-risk. *Journal of risk*, **2**, 21–42.

Scarf, H. E. (1957). A min-max solution of an inventory problem. Technical report, RAND CORP SANTA MONICA CALIF.

Yáñez, J. a Ramírez, J. (2003). The robust coloring problem. *European Journal of Operational Research*, **148**(3), 546–558.

Zhu, S. a Fukushima, M. (2009). Worst-case conditional value-at-risk with application to robust portfolio management. *Operations research*, **57**(5), 1155–1168.