Obsah

1 Úvod

1.1 Motivácia

TODO: úvod, motivácia, prečo?

1.2 Cieľ práce

Cieľom práce je skonštruovať cestovný poriadok zo zadaných liniek tak, aby bolo čo najviac cestujúcich spokojných pri prestupoch. T.j. optimalizovať čas na prestupoch v závislosti od počtu cestujúcich, očakávaných na danom prestupe. Súčasťou zadania práce je navrhnúť algoritmy na generovanie a optimalizáciu cestovných poriadkov na železnici.

Výsledkom práce je program PTG, ktorého názov je skratka odvodená od Periodic Timetable Generation. Vstupom pre program je jeden konkrétny cestovný poriadok, z ktorého sa načítajú potrebné údaje. Tie sa dajú modifikovať na viacerých úrovniach, čím sa do značnej dá ovplyvniť výsledok generovania. Výstupom programu sú viaceré cestovné poriadky, vygenerované rôznymi algoritmami pre porovnanie.

1.3 Štruktúra práce

V druhej kapitole je načrtnutá problematika generovania cestovných poriadkov a popísané známe algoritmy, ktoré riešia daný problém. Tretia kapitola je zameraná na formuláciu riešeného problému v tejto bakalárskej práci.

Vo štvrtej kapitole sú popísané navrhnuté algoritmy. Piata kapitola obsahuje užívateľskú dokumentáciu, ktorá objasňuje postup, ako modifikovať vstupné údaje na viacerých úrovniach a odplyniť výsledky generovania.

V šiestej kapitole sú vzájomne porovnané výsledky vygenerované navrhnutými algoritmami nad rôznymi kolekciami vstupných dát. Výsledky sú porovnané aj so skutočným cestovným poriadkom. Siedma kapitola je zameraná na implementáciu a popisuje zaujímavosti, ktoré sa vyskytli pri implementácii. V závere sú zhrnuté výsledky porovnania algoritmov a kriticky spomenuté nedostatky.

2 Problematika generovania cestovných poriadkov

V tejto kapitole a rovnako ako aj celej práci sa zameriavam na problematiku cestovných poriadkov (ďalej len pod skratkou „CP“) na železnici. Taktové CP sa používajú aj pre iné dopravné siete ako železnice, ako napríklad systém integrovanej dopravy v mestách zahrňujúci metro, električky a autobusy, ktoré chodia v kratších časových intervaloch ako vlaky. Preto sa v texte obmedzím na pojem taktové cestovné poriadky.

Špeciálne pre CP na železnici sa berie do úvahy aj infraštruktúra železničných tratí a rôzne iné obmedzenia z toho vyplývajúce, ktoré budú vysvetlené neskôr.

2.1 Taktové cestovné poriadky

Myšlienka taktových cestovných poriadkov spočíva v tom, že vlaky, prípadne vlakové spojenia, jazdia na danom úseku v pravidelných intervaloch, v tzv. taktoch. Vlakové súpravy jednej vlakovej linky sú z určitej stanice vypravované vždy v rovnaký čas v každej perióde intervalu, napríklad každú hodinu.

Taktové CP prinášajú viaceré výhody. Svoju transparentnosťou sú pre cestujúcich ľahko zapamätateľné, a to najmä pre ich pravidelnosť. Pre odchod vlaku si stačí zapamätať v ktorej minúte odchádza zo stanice. Koncept zachováva rovnaký čas potrebný na prestup počas celého dňa. Ak Vám teda uchádza prípoj len o pár minút, bude Vám pre periodicitu CP uchádzať rovnako po celý deň. Optimalizácia časov potrebných na prestupoch v závislosti od počtu cestujúcich je predmetom tejto bakalárskej práce.

Z hľadiska plánovania je výhodné, že pri generovaní stačí uvažovať iba jeden interval. Zásadnou požiadavkou je, že situácia na konci periódy musí odpovedať situácii na začiatku periódy. V prípade zhody sa základ vygenerovaného CP pre daný interval skopíruje do celého rozsahu dňa, a tým skomponuje výsledný celodenný CP. Vlaky chodia v každej perióde rovnako až na posun.

Pre prispôsobenie taktového CP potrebám obyvateľstva a dopytu po cestovaní sa narúša periodicita. V praxi to vyzerá tak, že v prípade dopravnej špičky sa hodinový interval môže skrátiť na polhodinu, naopak mimo špičky a pri nočných spojoch natiahnuť na dvojhodinový interval.

V skutočnom CP sa pridávajú spoje aj v dôsledku zvýšenia počtu cestujúcich pred víkendom a po víkende. Počas víkendov sa interval zväčšuje. Príkladom pridaných neperiodicky sa opakujúcich spojov môžu byť školské linky a linky zabezpečujúce dopravu pre industriálne zóny.

TODO: ukážka taktového cestovného poriadku

2.2 Obmedzujúce podmienky

Obmedzujúce podmienky sa používajú pre modelovanie vzťahu medzi dvomi udalosťami. Pomocou nich sa dá sformulovať všetky možné vzťahy, ktoré majú byť obsiahnuté vo výsledne vygenerovanom CP.

V tejto sekcii popíšem ako sa dajú modelovať obmedzujúce podmienky pre taktové CP na železnici. Najskôr uvediem predpoklady a požiadavky vyplývajúce z rôznych faktorov, následne ukážem ako sa formulujú periodické podmienky na základe rôznych požiadaviek.

2.2.1 Predpoklady

Predpoklady pre model môžu byť rôzne v závislosti od infraštruktúry, vlakových spojení a iných požiadaviek na CP. Spomenuté predpoklady a požiadavky z nich vyplývajúce sú vopred zadané.

Železničná infraštruktúra

Železničnú infraštruktúru tvoria uzly a trate.

- **Uzly** reprezentujú miesta v železničnej sieti, kde sa môžu vlaky vzájomne ovplyvňovať, preto sa v nich vyžaduje koordinácia. Príkladom uzla je vlaková stanica, kríženia, výhybky a zoraďovacie koľajisko.

- **Trate** sú spojenia z uzla do najbližšie uzla, po ktorých sa presúvajú vlaky. Medzi dvojicou uzlov môže existovať aj viac položených paralelne tratí - koľají, a každému vlaku je vopred určená voľná koľaj po ktorej sa má presúvať. Trate môžu byť:

    - Jednokoľajové.
    - Dvojkoľajové, pri ktorých je každá koľaj využívaná jedným smerom.
    - Viackoľajové, (napr.: 2 koľaje pre každý smer) na ktorých môže byť riešené predbiehanie v tom zmysle, že pre koľaje v jednom smere sú pridelené vlaky podľa rýchlosti (jedna koľaj pre IC a rýchliky, druhá pre osobné a nákladné vlaky).
    - Banalizované viackoľajové, kde všetky koľaje môžu byť využívané pre oba smery podľa potreby.

- **Stanica** je uzol, určený pre zastavenie vlaku na určitý čas, pre nástup a výstup cestujúcich, ale taktiež miesto pre križovanie vlakov. Pri pohľade na stanicu ako uzol, je skrytá jej vlastná infraštruktúra. Uvažovanie a začlenenie staničnej infraštruktúry by viedlo k veľkému a komplikovanému modelu, preto v našom modeli stanice chápeme ako čierne skrinky. Pri tomto prístupe môže nastať prípad, že po vygenerovaní CP nebude realizovateľný pre obmedzenie staničnej infraštruktúry. Existujú modely a algoritmy, ktoré zahŕňajú aj staničnú infraštruktúru.

Vlaky

Jednotlivé vlaky sú uvažované vo forme vlakových liniek.

- **Vlaková linka** je priame vlakové spojenie medzi počiatočnou a konečnou stanicou po určitej, vopred definovanej trase. Každá linka jazdí v pravidelnom intervale, t.j. pravidelné sú vypravované vlaky v každej perióde intervalu. Pre každú linku je čas medzi dvomi stanicami vopred určený a fixný.

Cestujúci

Vzhľadom k počtu cestujúcich, ich zámeru odkiaľ, kam a v akom čase cestovať, je optimalizovaný celý železničný model. Podľa toho sú navrhované vlakové linky, ich periodicita, či použitie konkrétnych vlakových súprav.

Špeciálne požiadavky

CP musia spĺňať viacero požiadaviek, ktoré sú kladené na bezpečnostnú reguláciu, na dosiahnutie určitého štandardu poskytovaných služieb a v neposlednom rade musia byť realizovateľné samotné CP.

- **Pobyt vlaku na stanici** - je časový interval počas ktoré vlak zastaví na stanici. Spodná hranica predstavuje minimálny potrebný čas na nástup a výstup pasažierov. Na druhej strane horná hranica obmedzuje dobu státia a po uplynutí ktorej už vlak nevyťažuje kapacitu stanice. Započítava sa do celkovej doby cestovania.

- **Prípojové vlaky** - dva vlaky môžeme označiť ako prípojové, ak je medzi nimi plánovaný prestup. Dôležitým faktorom je príchod prvého nasledovaný odchodom druhého vlaku zo stanice. Taká situácia nastane ak medzi dvomi stanicami neexistuje priame vlakové spojenie, čiže nutnosť prestupu.

- **Spájanie vlakov** - spájanie môže vzniknúť medzi dvomi vlakmi, ktoré majú v určitom úseku spoločnú trasu. Takáto kombinácia dvoch vlakov si vyžaduje prítomnosť oboch v stanici, v ktorej sú spájané do jednej vlakovej súpravy. Optimalizuje sa tým veľkosť posádky (na spoločnej trase postačí jedna) a šetrí sa kapacita tratí (dva vlaky by museli mať medzi sebou bezpečnostné časové rozstupy).

- **Synchronizácia vlakov** býva výhodná, ak dve vlakové linky majú spoločnú časť na ich trasách. Tak potom čas odchodu z prvého spoločného uzla je synchronizovaný. Napríklad pri rovnakej perióde majú dve vlakové linky frekvenciu jedna, po synchronizácii je poskytovanie prepravy osôb v spoločnej časti trasy s frekvenciou dva.

- **Otáčanie vlakovej súpravy** v konečnej stanici býva v dôsledku jej využitia pre vlakovú linku v opačnom smere. Čas strávený v tomto uzle je naplánovaný tak, že započítava dobu prepojovania súpravy, prípadné overenie technického stavu a taktiež aj čas slúžiaci na zmenšovanie alebo absorbovanie meškaní.

- **Fixované príchody a odchody** sa vyskytujú napríklad pri vlakových linkách, ktoré sú zároveň medzinárodnými linkami. Čas príchodu na hranice štátu je vymedzený vzájomnou dohodou susediacich železničných spoločností a nie ako výsledok plánovacieho procesu.

- **Bezpečnostná regulácia** spočíva v tom, že dva vlaky využívajúce rovnakú trať v určitom smere sú od seba oddelené bezpečnostnými časovými rozstupmi. Časový rozdiel medzi nimi musí byť dodržaný rovnako v počiatočnom aj v koncovom uzle

trate. Bezpečnostné opatrenia taktiež nepovoľujú stretávanie a predbiehanie vlakov na jednej koľaji.

2.2.2 Formulácia obmedzujúcich podmienok

V tejto sekcii si ukážeme ako sa dajú predchádzajúce požiadavky pretransformovať do obmedzujúcich periodických podmienok. Väčšina známych algoritmov na generovanie CP je založená na splňovaní obmedzujúcich podmienok.

Periodická obmedzujúca podmienka

Cestovný poriadok pozostáva z časov príchodu a odchodu pre všetky linky v každej stanici, ktorou prechádzajú. Pri modelovaní sa používajú rozhodovacie premenné pre príchody a odchody nasledovne:

$$a_n^t \in \{0,..,T-1\} \text{ čas príchodu vlaku } t \text{ do uzla } n$$
$$d_n^t \in \{0,..,T-1\} \text{ čas odchodu vlaku } t \text{ z uzla } n$$

Kde parameter T je interval cestovné poriadku v minútach. Rozhodovanie premenné a_n^t a d_n^t nadobúdajú celočíselné hodnoty z domény $\{0,..,T-1\}$. Vlak t podľa CP príde do stanice s v čase :25, pobudne v stanice jednu minútu a odíde v čase :26, zapíšeme pomocou obmedzujúcej podmienky takto:

$$d_s^t - a_s^t = 1$$

Inými slovami udalosť a_s^t nastane jednu minútu pred udalosťou d_s^t, a naopak udalosť d_s^t nastane jednu minútu po a_s^t.

Periodicita obmedzujúcich podmienok sa dosiahne počítaním v modulo T (pri perióde T). Pre zovšeobecnenie myšlienky uvažujme a_n^t ako príchod vlaku t do uzla n, $d_m^{t'}$ ako odchod iného vlaku t' z uzla m, a ich vzťah vytvoríme pomocou časového okna $[l,u]$ namiesto fixovanej hodnoty. Časové okno predstavuje interval s dolnou a hornou hranicou pre rozdiel dvoch rozhodujúcich premenných.

$$a_n^t - d_m^{t'} + Tp \in [l,u] \qquad p \in \mathbb{Z}$$

Ekvivalentne sa predchádzajúci zápis pre prehľadnosť skracuje na:

$$a_n^t - d_m^{t'} \in [l,u]_T$$

Pre jednotlivé špeciálne požiadavky spomenuté v 2.2.1 sa formulujú obmedzujúce podmienky nasledovne (všetky konkrétne čísla použité v obmedzujúcich podmienkach sú uvedené ako príklad):

- **Pobyt vlaku *t* na stanici *s***, ktorý má byť minimálne 2 minúty a maximálne 10 minút sa dá vyjadriť
$$d_s^t - a_s^t \in [2,10]_{60}$$
- **Prípojové vlaky** - plánovaný vzájomný prestup medzi linkami t_1 a t_2 v stanici s, sa definuje dvojicou obmedzujúcich podmienok. Požadovaný čas na prestup od 2 do 10 minúť medzi oboma navzájom.
$$d_s^{t_2} - a_s^{t_1} \in [2,10]_{60}$$
$$d_s^{t_1} - a_s^{t_2} \in [2,10]_{60}$$

- **Spájanie dvoch vlakov** t_1 a t_2, respektíve spájanie ich vlakových súprav do jednej (napr. t_1) na spoločnej trase ohraničenej stanicami s_1 a s_2 si vyžaduje, aby boli v oboch hraničných staniciach obidva dva vlaky prítomné. Presnejšie v stanici s_1, kde sa vlaky spájajú do t_1, príchod vlaku t_2 musí predchádzať odchodu vlaku t_1. Druhá obmedzujúca podmienka vyplýva analogicky pre stanicu, kde sa rozpájajú:
$$d_{s_1}^{t_1} - a_{s_1}^{t_2} \in [5,10]_{60}$$
$$d_{s_2}^{t_2} - a_{s_2}^{t_1} \in [5,10]_{60}$$

- **Otáčanie vlakov** v konečnej stanici pre použitie vlakovej súpravy na linke v opačnom smere vyjadruje obmedzujúca podmienka
$$d_s^{t_2} - a_s^{t_1} \in [20,50]_{60}$$
kde je požadovaný minimálny čas 20 minút predtým ako vlaková súprava opustí konečnú stanicu a 50 minút je maximálny čas, ktorý v nej môže pobudnúť.

- **Fixované príchody a odchody** sa vyskytujú pri medzinárodných linkách. Vlak vstupuje na územie štátu o :25 a opúšťa územie o :34. Vstup odpovedá odchodu medzinárodného vlaku od uzla na hranici, ktorý je podľa dohody napríklad v rozpätí :23 a :27. Opúšťanie odpovedá príchodu vlaku do uzla hranice v rozpätí :32 a :36.
$$d_b^t = 25$$
$$a_b^t = 34$$
$$d_b^t \in [23,27]$$
$$a_b^t \in [32,36]$$
Obmedzujúce podmienky nie sú ale periodické. Vyjadrujú presne hodnotu v minútach. Pre zapísanie do obecného tvaru potrebujeme pridať pomocnú premennú
$$\beta \in \{0, \dots, T-1\}$$
Dostávame upravené periodické obmedzujúce podmienky:
$$d_b^t - \beta = [25]_{60}$$
$$a_b^t - \beta = [34]_{60}$$
$$d_b^t - \beta \in [23,27]_{60}$$
$$a_b^t - \beta \in [32,36]_{60}$$
Každá obmedzujúca podmienka dve rozhodujúce premenné. Ak nejaký cestovný poriadok spĺňa všetky obmedzujúce podmienky, tak pridaním m minút pre každý čas príchodu a odchodu dostaneme nový cestovný poriadok, rovnako spĺňajúci všetky podmienky. Oba CP sú v podstate rovnaké, až o posunutie o m minút. Ak pri zvolenom posunutí sú všetky pomocné premenné $\beta = 0$, tak všetky obmedzujúce podmienky pre fixné odchody a príchody sú splniteľné.

- **Synchronizáciu dvoch vlakov** t_1 a t_2, ktoré majú značnú časť trasy spoločnú a frekvenciu jedna v perióde cestovného poriadku, skonštruujeme nasledovne: chceme posunúť odchod vlaku t_2 voči odchodu vlaku t_1 o 30 minút s chybou 2 minúty. Synchronizácia sa vzťahuje na celú spoločnú trasu so stanicami $s_i, i \in \{1, \dots, n\}$.
$$d_{s_i}^{t_2} - d_{s_i}^{t_1} \in [28,32]_{60} \qquad i \in \{1, \dots, n\}$$
Po synchronizácii vlakov o 30 minút poskytuje osobnú dopravu pre na spoločnej trase s frekvenciou dva. Podobným spôsobom sa dajú synchronizovať viac ako dva vlaky.

- **Bezpečnostná regulácia** rozstupov (napr. 3 minúty) dvoch po sebe idúcich vlakov v jenom smere po tej istej trati znamená, že ak vlak t_1 opustí stanicu s v určitom čase, tak vlak t_2 nesmie opustiť stanicu do 3 minút pred ani po odchode vlaku t_1. Keďže bezpečnostný rozstup sa týka vlakov po celej dĺžke trate medzi stanicami s_1 a s_2, tak

dostávame dve podmienky:

$$d_{s_1}^{t_2} - d_{s_1}^{t_1} \in [3{,}57]_{60}$$

2.3 Periodic Event Scheduling Problem

Táto sekcia opisuje Periodic Event Scheduling Problem (PESP), ktorý sformulovali Serafini a Ukovich (1). PESP poskytuje akýsi framework pre podmienky typu popísané v sekcii **Error! Reference source not found.**.

2.3.1 Definícia PESP

Definícia: Nech je daná N množina udalostí, množina $A \subseteq N \times N$, časová perióda T, a časové okná $[l_{ij}, u_{ij}]$ pre všetky $(i,j) \in A$. PESP má nájsť periodický rozvrh $v_i \in [0,T)$, $i \in N$, ktorý spĺňa

$$(v_j - v_i)\ mod\ T \in [l_{ij}\, u_{ij}]pre\ všetky\ (i,j) \in A$$

alebo rozhodne, že vstupné podmienky sú nesplniteľné a rozvrh neexistuje.

Spomínaný problém splniteľnosti zadaných obmedzujúcich podmienok je NP - úplný, dokázaný autormi v (1).

2.4 Známe algoritmy

V tejto sekcii je prehľad algoritmov, ktoré riešia problém generovania a optimalizácie taktových cestovných poriadkov.

2.4.1 Odijk

2.4.2 Voorhoeve

2.4.3 Liebchen

2.4.4 Peeters

2.5 Prípadová štúdia

Nasledujúca sekcia stručne popisuje situáciu ako bol použitý matematický optimalizačný model v praxi, t.j. pri generovaní CP pre sieť liniek metra. (2)(3)

2.5.1 Berlínske metro

Sieť berlínskeho metra

Sieť berlínskeho metra má dĺžku všetkých tratí 144 km TODO: (Pražské metro ?? km) s počtom 170 staníc, kde 19 z nich je prestupných. Priemerná doba cestovania je 6 km alebo 8 staníc. Počas jedného dňa je prepraví okolo 1,3 milióna cestujúcich. Z úvodných informácií sa dá usúdiť, že ide o rozľahnú sieť, ktorá je len časťou siete integrovanej dopravy v Berlíne.

Počas dopravnej špičky a v pracovných dňoch sú intervaly na linkách aspoň 5 minút. Intervaly vo večerných a nočných hodinách a počas víkendov sú 10 minút. Predmetom ich štúdie bola prevádzka berlínskeho metra mimo dopravnej špičky pri intervale T=10.

Požiadavky

Cieľom ich štúdie bolo vylepšiť aktuálny CP metra, redukovať dobu čakania cestujúcich. Kritéria, ktoré sa brali do úvahy, sa týkali počtu použitých vlakových súprav, priemernej doby potrebnej na prestup, priemernú rýchlosť vlakov, a počtu plánovaných prestupov.

Ďalšími kritériami boli vyrovnanosť medzi všetkými prestupmi (neexistujú príliš zlé časy na prestupoch), a stabilita CP, schopnosť absorbovať v určitej miere odchýlenie sa od plánovaného rozvrhu.

Optimalizácia

Pri modelovaní použili PESP model s určitými doplneniami. Výslednú úlohu lineárneho programovania riešili CPLEX MIP-Solverom. Skonštruovali CP, ktorý podstatne vylepšil ten predchádzajúci vo všetkých zadaných kritériách.

Výsledok

Prvý výsledný CP zredukoval vážený priemer čakacej doby cestujúcich z 34% doby intervalu na menej ako 17%. Pre 24 najdôležitejších prestupov sa priemerná doba čakania znížila z 30% na 4% doby intervalu. Zvýšil počet potrebných vlakových súprav zo 71 na 78. Spoločnosť BVG, prevádzkujúca berlínske metro, ho zamietla.

Nasledujúca spolupráca medzi vývojovým tímom CP a BVG viedla k vzniku viacerých CP. Pri každom z nich sa vyskytli situácie, ktoré viedli k zmene pôvodných podmienok alebo sa pridávali dodatočné nové požiadavky. Konečný akceptovaný CP bol v porovnaní s pôvodným CP z pohľadu vylepšenia doby čakania len o niečo lepší, ale výrazne však zlepšoval vyrovnanosť a stabilitu.

Prípad optimalizácie ukázal, že výsledný CP je významne lepší ako manuálne skonštruovaný v splniteľnosti kritérií. Prípad demonštroval úspešné uvedenie prvého, matematicky optimalizovaného, taktového CP v praxi.

3 Formulácia riešeného problému

Táto kapitola obsahuje formuláciu riešeného problému, ktorý vychádza už zo zadania ročníkového projektu a jeho špecifikácie. Definuje vstup a požadovaný výstup, popisuje minimalizačnú funkciu v závislosti od ktorej prebieha optimalizácia problému.

3.1 Vstup

Vstupom je existujúci cestovný poriadok získaný z IDOSu, ktorý slúži pre potreby načítania vstupných dát. Vstupnými dátami sú rozplánované vlakové linky, ktoré majú definovanú pevnú trasu, číslo linky, časový interval a obsahujú zoznam zastávok. Zastávka obsahuje názov stanice, čas príchodu a odchodu do stanice, a na akom kilometri od počiatku trasy sa nachádza. Časovanie vlakových linie je tak kompletne zadané.

Ďalším parametrom vstupu je zoznam staníc. Každú stanicu špecifikuje názov stanice, mesto v ktorom sa nachádza a počet obyvateľov aglomerácie spadajúcich pod danú stanicu. Položka mesto nie je pre stanicu povinná, slúži pre potreby optimalizácie. Nie sú vyhľadávané vlakové spojenia medzi dvomi stanicami jedného mesta, inak by nezohľadňovali realitu. Obyvatelia mesta preferujú MHD, ktorá poskytuje frekventovanejšiu osobnú dopravu.

Parametrami vstupu aj zoznam navzájom prepojených liniek. Vzájomne prepojené linky tvoria triedu ekvivalencie. Ak posuniem odchod linky o m minút, posunú sa rovnako o m minút odchody všetkých liniek, ktoré sú s ňou prepojené.

3.2 Požadovaný výstup

Výstupom sú skonštruované taktové cestovné poriadky. Sú vygenerované rôznymi navrhnutými algoritmami optimalizované vzhľadom k minimalizačnej funkcii. Pre štatistické účely porovnania sú súčasťou výstupu pre jednotlivé CP aj: výsledná hodnota minimalizačnej funkcie a počet progresívnych zmien, ktoré vylepšili aktuálnu hodnotu minimalizačnej funkcie.

3.3 Minimalizačná funkcia

V navrhnuté algoritmy počítajú s minimalizačnou funkciou, ktorá je navrhnutá ako súčet cez všetky prestupy z množiny P súčinu váhy a času maximálnej doby prestupu. Formálne zapísané ako:

$$\sum^{P} w_{ij} \left(d_s^{t_j} - a_s^{t_i} \right)$$

P je množina prestupov v CP
$p \in P$ je prestup medzi dvoma linkami v stanici s, z linky t_i na linku t_j
$a_s^{t_i}$ je čas príchodu linky t_i do prestupnej stanice s
$d_s^{t_j}$ je čas odchodu linky t_j z prestupnej stanice s
w_s^{ij} je počet očakávaných cestujúcich na prestupe z linky t_i na linku t_j
N je počet vlakových liniek
$i, j \in \{1, \dots, N\}$
S je množina všetkých staníc
$s \in S$ je prestupná stanica pre linky t_i a t_j

3.4 Viacúrovňové zadávanie údajov

Procesu optimalizácie predchádza možnosť viacúrovňového zadávania, respektíve modifikácie vstupných údajov. Možnosť modifikácie je na úrovni vlakových liniek, ich zastávok a staníc. Po vygenerovaní vlakových spojení medzi všetkými stanicami, sa dajú modifikovať údaje o vlakových spojeniach medzi jednotlivými stanicami.

Takéto viac úrovňové zadávanie údajov ovplyvňuje výsledné vygenerované taktové cestovné poriadky.

3.5 Genereovanie a optimalizácia

Predmetom generovania a konštrukcie CP je nájsť vzájomné posunutie zadaných vlakových liniek v čase tak, aby výsledný CP zohľadňoval problém optimalizácie.

Problém optimalizácie je minimalizovať časy na prestupoch tak, aby bolo čo najviac cestujúcich spokojných. Minimalizačná funkcia je definovaná v sekcii 3.3.

4 Navrhnuté algoritmy

V použitých algoritmoch sa pre zjednodušenie obmedzím na iba na niektoré obmedzujúce podmienky zo sekcie 2.2.2.

4.1 Náhodné generovanie s vylepšovaním

Prvý z navrhnutých a implementovaných algoritmov je založený na vygenerovaní náhodného CP a jeho modifikáciou sa ho postupe snaží vylepšovať.

Pseudokód algoritmu

```
– definitions
AllLines : is a set of all used lines
AvailableLines : is a set of lines
StableLines : is a set of lines
selectedLine : is a line

- initialization
T := createRandomizedTimetable(AllLines)
AvailableLines := AllLines
StableLines := 0

- main loop
while AvailableLines ≠ 0 do
    selectedLine := chooseRandomlyFrom(AvailableLines)

    for all possible shifts {0,..T − 1} in period of selected line do
        calculate line related transfers` rating value of current shift
        newRatingValue := remember the minimum of line`s rating value
    od

    if newRatingValue < oldRatingValue then
        StableLines := 0
        AvailableLines := AllLines
    fi

    StableLines := StableLines ∪ selectedLine
    AvailableLines := AvailableLines \ selectedLine
Od
```

Popis algoritmu

Algoritmus je navrhnutý tak, že si v inicializačnej časti vygeneruje náhodný CP, a potom sa ho snaží vylepšovať. Súčasťou inicializácie je aj nastavenie množiny stabilných liniek na prázdnu a do množiny dostupných liniek, s ktorými ešte môžeme posúvať, sa pridajú všetky zadané linky.

Vylepšovanie prebieha tak, že algoritmus si náhodne vyberie linku, ktorá ešte môže byť vylepšená, respektíve vylepšenie jej prírastku do minimalizačnej funkcie. Vybranou linkou algoritmus posúva v tom zmysle, že nastavuje čas odchod z počiatočnej stanice prechádzaním celej množiny periódy $\{0, \dots, T-1\}$. Množina reprezentuje minúty, o ktoré môže byť

posunutý aktuálny CP linky. Pre každé posunutie si spočíta minimalizačnú funkciu pre prestupy obsahujúce vybranú linku. Prejdením celej množiny nájde minimum možného prírastku k výslednej hodnote minimalizačnej pre minútu posunutia, a ten porovná s predchádzajúcim, doteraz aktuálnym prírastkom.

Ak je nový prírastok lepší ako doteraz aktuálny, čiže menší z pohľadu minimalizácie, tak na vybranej linke sa nastaví odchod z počiatočnej stanice na minútu posunutia, pre ktorú bol vygenerovaný nový prírastok. Rovnako sa zmení aj aktuálny prírastok pre linku. Súčasne sa množina stabilných liniek nastaví na prázdnu, a množina dostupných liniek sa rozšíri opäť na všetky linky, podobne ako v inicializačnej časti na algoritmu.

V oboch prípadoch, či už je nový prírastok lepší alebo nie, sa vybraná linka pridá do množiny stabilných liniek a odoberie z množiny dostupných liniek.

Konečnosť algoritmu

4.2 Generovanie s použitím discrete set

.

Pseudokód algoritmu

```
--Propagation part
- definitions
- transfers : defined as from line to line on specified station
- constraints : derived from transfer, contains discrete set
- I : set of {1,…,N} where N is number of train lines used in constraints
- matrix : matrix of discrete set

- initialization
constraints := createConstraints(transfer)
constraints := createSetAndMinimizationFactor(constraints)
constraints := mergeEquivalentConstraints(constraints)

- matrix initialization
for i,j ∈ I do
    if i = j then
        matrix[i,j] := {0}
    else
        matrix[i,j] := {0,..T − 1}
    fi
od
foreach constraint on i,j ∈ I do
    matrix[i,j] := constraint.set
    matrix[j,i] := {0} – matrix[i,j]
od

- main part
propagate(matrix)
```

```
--Search part (Backtracking)
- paramaters
matrix : matrix of discrete set

- main loop
while (true) do
    propagate(matrix)

    if matrix is not valid then
        return
    fi

    bestRecord := choose best record from Set derived from constraints only
    if bestRecord was not found then
        solution found
        end of algoritm
    fi

    fixOnePotentialOfSet(matrix, bestRecord)

    search(matrix)

    removedFixedPotentialOfSet(matrix, bestRecord)
od
```

```
--Propagate method
- parameters
matrix : matrix of discrete set

- definitions
changed : boolean
I : set of {1,…,N} where N is number of train lines used in constraints

- initialization
changed := true

- main loop
while changed do
    changed := false

    for i,j,k ∈ I and i ≤ j and k≠i,j do
        if matrix[i,j] ⊈ matrix[i,k] + matrix[k,j] then
            matrix[i,j] := matrix[i,j] ∩ (matrix[i,k] + matrix[k,j])
            matrix[j,i] := {0} − matrix[i,j]
            changed := true
        fi
    od

od
```

```
--Construction of Timetables
- parameters
matrix : matrix of discrete set

- solution
matrix[0] represents solution
```

Popis algoritmu

Konečnosť algoritmu

4.2.1 Varianty algoritmu

Druhý navrhnutý algoritmus má viacero varianty, ktoré vznikli až vo fáze implementácie. Boli navrhnuté rôzne postupy, ktoré pozmenili hlavné metódy pre generáciu `propagate` a `search`.

Variant A

Variant B

Variant C

Variant D

4.3 Možné využitie a obmedzenia