

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

**Metodika určování propustnosti kolejíšť
železničních stanic s podporou počítačové simulace**

Autor: Ing. Roman Diviš

Školitel: prof. Ing. Antonín Kavička, Ph.D.

Školitel – specialista: doc. Ing. Michael Bažant, Ph.D.

Disertační práce

2020

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 31. 7. 2020

Roman Diviš

PODĚKOVÁNÍ

Chtěl bych touto cestou poděkovat svému školiteli prof. Ing. Antonínu Kavičkovi, Ph.D. a školiteli – specialistovi doc. Ing. Michaelu Bažantovi, Ph.D. za vedení během celého mého studia, rady a za věcné připomínky při tvorbě této práce.

ANOTACE

Železniční doprava představuje jeden ze základních způsobů dopravy osob a zboží. Infrastruktura železnice je restriktivnější a omezenější (než je tomu např. u infrastruktury automobilové dopravy) a případné technické problémy na zařízeních či přetížení části infrastruktury mohou negativně ovlivnit mnoho vlaků na železnici. Techniky určování kapacity na železnici slouží k analýze a vyhodnocení, jakou kvalitu provozu lze na dané části železniční infrastruktury (a s ní související realizovanou vlakovou dopravu) očekávat. Kapacitu lze zjišťovat s užitím různých technik a matematických aparátů, zahrnujících metody analytické, grafické a experimentální. Disertační práce se primárně věnuje návrhu inovativní metodiky pro určování kapacity železničních stanic s využitím techniky počítačové simulace. Navrhovaná metodika využívá techniku vnořených reflektivních simulací, která jako podpora rozhodování umožňuje hledat řešení konfliktů vznikajících při stochastických simulacích. Disertační práce se dále zaměřuje na popis techniky vnořených reflektivních simulací a jejich uplatnění jako obecné podpory rozhodování v simulaci. V poslední části práce je navržena metodika pro zjišťování kapacity železničních stanic aplikována na rozsáhlé případové studii.

KLÍČOVÁ SLOVA

Počítačová simulace, reflektivní vnořené simulace, železniční doprava, kapacita železničních stanic, metodika

TITLE

Methodology for determining the capacity of railway stations using computer simulation

ANNOTATION

Rail transport is one of the basic modes of transport of people and goods. The railway infrastructure is more restrictive and limited (than, for example, in the case of road transport infrastructure) and possible technical problems on equipment or overload of part of the infrastructure can negatively affect many trains on the railway. Techniques for determining the railway capacity are used to analyse and evaluate the quality of traffic that can be expected on a given part of the railway infrastructure (and related rail transport). Capacity can be determined using a variety of techniques and mathematical apparatus, including analytical, graphical, and experimental methods. The dissertation primarily deals with the design of an innovative methodology for determining the capacity of railway stations using computer simulation techniques. The proposed methodology uses the technique of reflective nested simulations, which as a decision-making support allows to find resolutions to conflicts arising in stochastic simulations. The dissertation further focuses on the description of the technique of reflective nested simulations and their application as a general decision-making support in simulation. In the final part of the dissertation, the proposed methodology for determining the capacity of railway stations is applied to an extensive case study.

KEYWORDS

Computer simulation, reflective nested simulations, rail transport, railway station capacity, methodology

OBSAH

Seznam obrázků	8
Seznam zkratk	10
Úvod.....	11
1 Motivace a cíle práce	12
2 Základní termíny a pojmy ze železniční dopravy.....	14
2.1 Kapacita a propustnost	14
2.2 Zpoždění.....	14
2.3 Limitní a kritické hodnoty.....	15
3 Železniční doprava a kapacita infrastruktury.....	16
3.1 Kapacita.....	17
3.2 Zjišťování kapacity železničních stanic	17
3.3 Analytické metody pro zjišťování kapacity	18
3.3.1 SŽDC SM124	18
3.3.2 UIC 406.....	19
3.4 Experimentální metody	19
3.4.1 Separátní simulace	19
4 Modelování a simulace	20
4.1 Systém.....	20
4.1.1 Model	20
4.1.2 Modelování	21
4.1.3 Verifikace a validace modelu	21
4.1.4 Simulace.....	21
4.2 Rozlišovací úrovně simulace.....	22
4.2.1 Mikroskopická simulace	22
4.2.2 Mezoskopická simulace	22
4.2.3 Makroskopická simulace	22
4.3 Simulace železniční dopravy.....	23
4.3.1 Analýza současného stavu simulátorů železniční dopravy.....	23
4.3.2 MesoRail.....	25
4.3.3 Deterministický a stochastický režim simulace železničního provozu	26
5 Podpora rozhodování v simulaci.....	28

5.1	Interaktivní režim simulace	28
5.2	Prioritní plánování.....	28
5.3	Expertní systémy	29
5.4	Metody operačního výzkumu a soft computing.....	30
5.5	Reflektivní vnořené simulace.....	30
6	Metoda reflektivních vnořených simulací	32
6.1	Technika reflektivní vnořené simulace	32
6.2	Výpočetní náročnost a paralelizace výpočtů RNS	34
6.3	Podmínka ukončení vnořených simulací	42
7	Metodika pro zjišťování kapacity železničních stanic.....	45
7.1	Základní termíny a pojmy užívané v rámci metodiky	45
7.2	Základní charakteristiky metodiky.....	47
7.2.1	Vstupy metodiky	48
7.2.2	Výstupy metodiky	50
7.3	Přehled jednotlivých fází metodiky	51
7.4	Průzkumná fáze	51
7.4.1	Vstupy průzkumné fáze	52
7.4.2	Provedení průzkumné fáze.....	53
7.4.3	Výstupy průzkumné fáze	54
7.5	Pilotní fáze	55
7.5.1	Vstupy pilotní fáze.....	55
7.5.2	Provedení pilotní fáze	55
7.5.3	Výstupy pilotní fáze.....	58
7.6	Finální fáze.....	58
7.6.1	Vstupy finální fáze.....	58
7.6.2	Provedení finální fáze	59
7.6.3	Výstupy finální fáze.....	59
7.7	Aplikace metodiky na více různých verzí systémů.....	59
8	Simulační nástroj MesoRail.....	62
8.1	Základní předpoklady.....	63
8.2	Koncepce simulačního jádra	64
8.3	Agentově orientovaný přístup	64
8.4	Základní komponenty simulátoru	64

8.5	Simulační replikace	65
8.6	Řídící logika pohybu vlaku	66
8.6.1	Zjišťování aktuální hodnoty zpoždění vlaku	68
8.6.2	Udržování rychlosti vlaku dle platných limitů a jízdního řádu	68
8.6.3	Zastavení vlaku v definované poloze na kolejišti	69
9	Případová studie.....	70
9.1	Základní charakteristika železniční infrastruktury	70
9.2	Základní informace o simulačních experimentech	74
9.3	Charakteristika podpory rozhodování v nástroji MesoRail	75
9.4	Villon.....	76
9.5	Cluster	77
9.6	Předpoklady.....	77
9.7	Průzkumná fáze	78
9.8	Pilotní fáze	81
9.9	Finální fáze.....	83
9.10	Vyhodnocení zjišťování kapacity	88
9.11	Vyhodnocení výsledků v porovnání s nástrojem Villon	90
10	Závěr	91
10.1	Perspektivy dalšího vývoje.....	91
	Bibliografie	93
	Seznam publikací autora	97
	Přílohy.....	99

Seznam obrázků

Obrázek 1 – Hierarchicky uspořádané množiny replikací ${}^0H_0(S_{0,0,r})$ a ${}^1H_3(S_{0,0,r})$	35
Obrázek 2 – Hierarchicky uspořádaná množina replikací ${}^2H_3(S_{0,0,r})$	39
Obrázek 3 – Hierarchicky uspořádaná množina replikací ${}^3H_3(S_{0,0,r})$	40
Obrázek 4 – Indikace stejných konfliktů ve vybraných replikacích z ${}^0H_3(S_{0,0,r})$, ${}^1H_3(S_{0,0,r})$, ${}^2H_3(S_{0,0,r})$, ${}^3H_3(S_{0,0,r})$	42
Obrázek 5 – Intervaly simulačních časů vybraných vnořených replikací (dle tabulky 1) při uplatnění strategií: (a) <i>hierarchicky redukované doby výhledů</i> , (b) <i>konstantní doby výhledů</i> .44	
Obrázek 6 – Obecný příklad hierarchie popsané pomocí termínů <i>verze systému, konfigurace RNS a scénář</i>	46
Obrázek 7 – Konkrétní příklad hierarchie scénářů a jejich hlavních replikací	47
Obrázek 8 – Přehled fází metodiky pro jednu verzi systému	52
Obrázek 9 – Ukázka uplatnění techniky <i>FCFS</i> při přidělování jednotlivých replikací na jednotky <i>LPU</i>	56
Obrázek 10 – Vývojový diagram uplatnění metodiky na více verzí systému	61
Obrázek 11 – Ilustrace schematického zobrazování kolejové infrastruktury v rámci simulačního nástroje <i>MesoRail</i>	62
Obrázek 12 – Vazby mezi jednotlivými skupinami komponentů nástroje <i>MesoRail</i>	65
Obrázek 13 – Zjednodušená koncepce činnosti exekutivní etapy replikace	67
Obrázek 14 – Přehledové zobrazení kolejové infrastruktury zkoumaného systému	70
Obrázek 15 – Grafikon vlakové dopravy a plán obsazení kolejí pro zkoumaný systém v rámci dopravní špičky	72
Obrázek 16 – Zaznamenaný tachogram podle ujeté vzdálenosti vlaku 750 (linky <i>RI</i>) při deterministické simulaci v nástroji <i>MesoRail</i>	73
Obrázek 17 – Špičkové využití operační paměti u replikace $ReplId = 5$ ($nestReplCount = 3$, $maxLevel = 3$).....	79

Seznam tabulek

Tabulka 1 – Hierarchicky redukované doby výhledů pro vybrané replikace vnořených simulací z obrázku 3	44
Tabulka 2 – Vybrané charakteristiky vlaků	71
Tabulka 3 – Prioritní seznamy s alternativami přidělování staničních kolejí vlakům	74
Tabulka 4 – Přehled provedených experimentů v <i>průzkumné fázi</i>	80
Tabulka 5 – Pilotní simulační experimenty aplikující 16 replikací hlavních simulací	84
Tabulka 6 – Ilustrační ukázka výsledků dopočítání „neperspektivních“ pilotních scénářů z tabulky 5	85
Tabulka 7 – Výsledky scénářů z <i>finální fáze</i> aplikujících 100 replikací hlavních simulací	86
Tabulka 8 – Ilustrační ukázka výsledků finálního dopočítání nevybraných pilotních scénářů (pouze „proveditelné“ scénáře)	87
Tabulka 9 – <i>Přírůstky zpoždění</i> jednotlivých kategorií vlaků pro scénáře z <i>finální fáze</i>	88
Tabulka 10 – <i>Limitní hodnoty</i> pro určení kvality provozu v období dopravní špičky dle hodnoty přírůstku zpoždění	88
Tabulka 11 – Přírůstky zpoždění jednotlivých vlakových linek pro scénáře z <i>finální fáze</i>	89

Seznam zkratek

2D	Dvoudimenzionální
3D	Trojdimenzionální
CPU	Central processing unit (centrální procesorová jednotka)
ČD	České dráhy
ETCS	European train control systém
FCFS	First come first served
GPU	Graphic processing unit (grafický procesor)
HDD	Hard drive disk (pevný disk)
HT	Hyper-Threading
LD	Lookahead duration (délka časového výhledu)
LPU	Logical processing unit (logická výpočetní jednotka)
RNS	Reflective nested simulation (reflektivní vnořená simulace)
RR	Round-Robin
SWDI	Sum of weighted delay increments (suma vážených přírůstků zpoždění)
SŽ	Správa železnic, s. o.
SŽDC	Správa železniční dopravní cesty, s. o. (původní název současné společnosti Správa železnic, s. o.)
UIC	International union of railways
XML	Extensible markup language

ÚVOD

Železniční doprava v dnešní době představuje jeden z důležitých způsobů, jak realizovat dopravu osob a nákladu na vnitrostátní i mezinárodní úrovni. Ročně je využívána miliony osob a jsou přepraveny miliony tun nákladu. To vše je realizováno na infrastruktuře železniční sítě, která je na rozdíl od silniční sítě výrazně menšího rozsahu. Poptávka po železniční dopravě přetrvává i vzhledem k faktům, že doprava po železnici je poměrně rychlá, umožňuje přepravu objemných a těžkých nákladů a rovněž představuje ekologičtější způsob dopravy. Železnice představuje heterogenní prostředí, kde se vyskytuje množství osobních vlaků, které typicky mají taktové jízdní řády, a vlaků nákladní dopravy, které mohou být jak pravidelné, tak nepravidelné. Správce či provozovatel železnice musí alokovat železnici a její koleje takovým způsobem, aby jednotlivé vlaky mohly bezpečně realizovat svoji cestu a navzájem se neohrožovaly. V ideálním světě by byly dodrženy veškeré jízdní řády a každý vlak by mohl dorazit načas do cílové stanice. Bohužel v realitě je možné se setkat s nehodami a poruchami, které následně dokážou narušit i nejlépe sestavené plány a jízdní řády. V důsledku omezené infrastruktury železnice může i poměrně drobná porucha nebo zpoždění vlaku mít negativní vliv na další vlaky pohybující se na železnici. Pokud by například jeden zpožděný vlak dokázal způsobit řetězovou reakci, která negativně ovlivní celý zbytek železniční sítě, pak by takový systém dopravy byl extrémně nestabilní a v praxi nepoužitelný.

Při plánování železniční infrastruktury i jízdních řádů pro její provoz je tak nutné přemýšlet dopředu a počítat s možnou rezervou na řešení případných problémů. Za tímto účelem se provádí zjišťování kapacity železničních tratí či stanic. Kapacitu si je možné zjednodušeně představit jako sadu charakteristik či provozních indikátorů, které popisují, jak dobře lze daný provoz na určité části železniční sítě zvládnout. Pomocí kapacitních charakteristik lze prognosticky spočítat či odhadnout, jestli bude daná část infrastruktury efektivně využívána nebo bude přetížená a její provoz bude rizikový. V minulosti byla vyvinuta řada různých metod, jak zjistit a analyzovat kapacitu, od jednoduchých analytických metod až po sofistikované experimentální metody.

V rámci této disertační práce je navržena nová inovativní metodika pro zjišťování kapacity železničních stanic, která využívá techniku počítačových simulací. V rámci metodiky je uplatněna metoda *reflektivních vnořených simulací (RNS)*, jejíž cílem je řešit konfliktní stavy, které vznikají při realizaci stochastických simulací. Metoda *RNS* užívá simulaci v simulaci, kdy jsou jednotlivé varianty (jak pokračovat při vzniklém konfliktu) simulovány po kratší dobu a následně je přijato rozhodnutí o řešení na základě zvoleného kritéria. Problém nastává, když dochází ke vzniku konfliktů v již běžící vnořené simulaci, pak je nutné proces rekurzivně opakovat nebo uplatnit jiný způsob řešení. Metoda *RNS* je v práci detailně popsána a formalizována.

Výše uvedené metoda je následně uplatněna v navržené metodice pro zjišťování kapacity železničních stanic. Navržená metodika umožňuje efektivní provádění výpočetně náročných vnořených simulací a zároveň umožňuje získání kvalitních výsledků ve stanoveném časovém limitu na provedení kapacitní studie.

1 MOTIVACE A CÍLE PRÁCE

Disertační práce se zaměřuje na řešení problematiky zjišťování kapacity železničních stanic s využitím metody počítačové simulace a techniky reflektivních vnořených simulací. Železniční doprava v dnešní době představuje jeden ze základních pilířů hromadné dopravy osob a nákladu. Ačkoliv podíl železniční dopravy na celkovém přepraveném množství osob a nákladu v průběhu posledních desetiletí výrazně poklesl, zejména na úkor osobní automobilové dopravy, stále se jedná o významný dopravní segment. V roce 2018 bylo pomocí železnice v České republice přepraveno celkem téměř 4 % cestujících a 17 % nákladu, v absolutních číslech se jedná o cca 190 milionů osob a 99 milionů tun nákladu (Ministerstvo dopravy, 2019). Segment železniční dopravy tak výrazně přispívá k celkovým dopravním kapacitám, jak v rámci vnitrostátní, ale také i mezinárodní přepravy. Na rozdíl od osobní automobilové dopravy je také železniční doprava výrazně bezpečnější a ekologičtější.

Železniční doprava operuje na omezené železniční infrastruktuře tvořené kolejemi. Železniční síť je v dnešní době menšího rozsahu, než je například síť silniční, a provoz na železnici tak musí respektovat výrazně omezenější prostředí a reflektovat jej. Významná část provozu je založena na pravidelných jízdních rádech, které definují jednotlivé vlaky a jejich časoprostorové polohy na železniční síti. V reálném světě dochází k výskytu událostí, které se dají označit za poruchy či nehody. Ty mohou vyřadit z provozu část železniční infrastruktury nebo některá železniční vozidla a v důsledku těchto událostí dochází k hledání náhradních řešení, omezení a vzniku mimořádností v dopravě. Konečným důsledkem uvedeného je vznik zpoždění vlaků a ovlivnění původně pravidelného provozu na železnici. Zpožděné vlaky mohou při pohybu po železnici dále ovlivňovat jiné vlaky a způsobit vznik následných zpoždění. To pak negativně vnímají zejména přepravovaní pasažéři v důsledku sníženého komfortu a možného vzniku dalších problémů.

Častá následná zpoždění mezi vlaky mohou být důsledkem přetížené železniční infrastruktury. Ta pak při výskytu zpožděných vlaků není schopna tyto vlaky (nebo následující vlaky) obsloužit a dochází u nich k navyšování zpoždění a přenosu zpoždění na další vlaky. Takový stav je nežádoucí a je ho možné řešit například vybudováním dalších staničních kolejí. To je časově i finančně nákladná investice, z ekonomického hlediska je rovněž třeba hledat rovnováhu mezi rozsahem infrastruktury a kvalitou provozu. Kvalitou je možné rozumět schopnost obsloužit dané vlaky a zároveň být schopen (částečně) absorbovat nepravidelnosti vzniklé zpožděnými vlaky. Pro vyjádření kvality provozu se využívají techniky zjišťující kapacitu železniční infrastruktury.

Lze definovat dva základní typy zjišťování kapacity železničních tratí a železničních stanic s ohledem na definované cíle: (a) určování rozsahu infrastruktury, (b) prověřování provozního konceptu.

V prvním případě se jedná o zjišťování, které má za cíl nalézt optimální rozsah a topologii infrastruktury. Aby výsledek byl obecně uplatnitelný, zjišťování v tomto případě by nemělo vycházet z konkrétního jízdního řádu.

Ve druhém případě, při prověřování provozního konceptu, je cílem stanovit řadu kvalitativních parametrů, jejichž hodnoty jsou závislé na konkrétním jízdním řádu a stanovené infrastruktuře. Využívá se tedy řada ukazatelů popisujících různé aspekty provozu. Na jejich základě je možné vyhodnotit, zdali je realizovaný rozsah dopravy vyhovující, rizikový nebo nevhovující. Taková informace je důležitá pro železniční dopravce a správce železniční infrastruktury. Díky ní je možné upravit plánovaný jízdni řád, tak aby poskytl ekonomický zisk pro přepravce a komfort pro cestující, anebo reagovat jinými opatřeními ke zvýšení kapacity v daném segmentu.

Za účelem zjištění kapacity se využívají různé matematické a infromatické přístupy. Různí se v informacích, které do těchto metod vstupují a které z metod vystupují. V minulosti dominantní roli hráli především analytické a grafické metody, které jsou ale vzhledem k jejich matematickému aparátu a vstupním informacím omezeny k poskytování zejména kvantitativních parametrů. V současnosti se projevuje tendence využívat aparát počítačové simulace, jako experimentální metody, k analýze a vyhodnocení chování dynamického systému železniční dopravy.

Cílem této disertační práce bylo navrhnout inovativní metodiku pro určování kapacity železničních stanic. Tato metodika využívá počítačové simulace a zároveň aplikuje techniku vnořených reflektivních simulací. Cílem je rozšíření informací o dané železniční stanici, tak aby bylo možné lépe vyhodnotit dopady zvoleného jízdniho řádu s ohledem na dnešní požadavky na kvalitu a rozsah dopravy.

2 ZÁKLADNÍ TERMÍNY A POJMY ZE ŽELEZNIČNÍ DOPRAVY

Na úvod je vhodné definovat termíny a další odborné pojmy z železniční dopravy, které se v textu práce vyskytují. V oblasti stanovování kapacity či propustnosti je řada termínů a jejich vnímání a výklad se může lišit. V textu práce se primárně vychází z terminologie, které je dána směrnici a dokumenty od Správy železnic (SŽDC, 2019).

2.1 Kapacita a propustnost

Kapacita představuje schopnost realizovat určitý výkon s určitou kvalitou. *Kapacita* je určena řadou vstupních parametrů definujících železniční infrastrukturu, rozsah a strukturu provozu, požadavky na kvalitu apod. *Ukazatel kapacity* představuje konkrétní kvantifikaci kapacity. Pro popis kapacity je typicky užíván komplexní popis, který kombinuje více různých ukazatelů kapacity.

Termínem *propustnost (propustná výkonnost)* je pak označován konkrétní ukazatel kapacity a je definován jako trvale realizovatelný rozsah dopravy v daném segmentu za daný časový úsek (typicky 24 hodin, ale může se vyjadřovat i pro kratší období). *Propustnost* tedy udává počet vlaků za čas, které je možná v požadované kvalitě na daném segmentu provozovat. Na základě propustnosti jsou dále definovány ukazatele označované jako *praktická (technická) propustnost*, *maximální propustnost* či *propustnost výhledová* a další. Tyto ukazatele vycházejí ze stejného základního předpokladu jako základní *propustnost*, ale různým způsobem zahrnují nebo manipulují s rozsahem záložních obslužných zdrojů, časem potřebným na výkon předepsaných kontrolních prohlídek, údržby zařízení a dalšími vstupními parametry.

Analytické metody pro určování kapacity obvykle používají jako základní vstupní parametr *výpočetní dobu*. Jedná se o analyzovaný časový interval provozu (za určitých výjimek se může analyzované období a výpočetní doba lišit). Mezi výstupní ukazatele kapacity pak patří *celková doba obsazení infrastruktury*, tj. součet dob obsazení daného zařízení sledovanými jízdami vlaků. Poměr celkové doby obsazení a výpočetní doby (analyzovaného období) pak udává *stupeň obsazení*. Tyto ukazatele se dají vyjádřit pro jednotlivé koleje i kolejové skupiny.

2.2 Zpoždění

Zpoždění představuje časovou odchylku vlaku od jízdního řádu, je vyjádřeno jako rozdíl mezi skutečným časem (resp. časovým okamžikem dané události) a plánovaným časem (výskytu příslušné události). Záporný rozdíl je označen jako *náskok*. Zpoždění je dále rozděleno na dva typy podle příčiny vzniku. *Primární zpoždění* je takové zpoždění, které nevzniklo v důsledku přenosu zpoždění z jiného vlaku. Jeho příčina je tak například technická závada, prodloužení pobytu ve stanici z důvodu zvýšené frekvence nástupu a výstupu cestujících a další příčiny. *Následné zpoždění* vzniká přenosem zpoždění z jiného vlaku (omezení pohybu vlaku kvůli jinému zpožděnému vlaku apod.).

V simulačních modelech je možné analyzovat zpoždění v průběhu simulačního procesu, nicméně důležité je definovat vstupní a výstupní zpoždění. *Vstupní zpoždění* je zpoždění, které

je vlaku přiděleno při jeho vstupu do simulačního modelu (např. ve výchozí stanici nebo v jiném vstupním místě), jedná se o definovatelný parametr simulace. Vstupní zpoždění umožňuje modelovat jevy, které vznikají vně simulovaného systému. V okamžiku, když je simulace ukončena nebo vlak opouští simulovaný systém (záleží který okamžik pro daný vlak nastane dříve), je možné změřit hodnotu *výstupního zpoždění* daného vlaku. Hodnota výstupního zpoždění závisí na vstupním zpoždění a na samotném simulačním procesu (může docházet k vytváření dalšího primárního zpoždění ve stanici nebo přenosem zpoždění z jiných vlaků). Rozdíl hodnot výstupního zpoždění a vstupního zpoždění je definován jako *přírůstek zpoždění* a představuje základní měřitelný výsledek účinku simulovaného systému na daný vlak. Tato hodnota je typicky analyzována v běžně užívaných simulačních nástrojích a může být použita jako jeden z ukazatelů kapacity. Přírůstek zpoždění neodpovídá hodnotě následného zpoždění a není možné tyto termíny zaměňovat.

S pojmem zpoždění je rovněž spjat obecnější termín *čekání*. U čekání je třeba rozlišit, jestli je plánováno v rámci zkonstruovaného jízdního řádu (*plánované čekání*) nebo vzniká při vlastní realizaci provozu (*čekání v provozu*). *Plánované čekání* představuje rozdíl mezi přáním dopravce a trasou realizovanou v jízdním řádu. Vzniká v důsledku konfliktů na zhlavích, předjíždění vlaků, a dalších situacích vyplývajících zejména z omezené infrastruktury.

Čekání v provozu se projevuje jako zpoždění. Hodnotu čekání je možné zjistit s využitím simulace nebo analyticky (s využitím metod teorie hromadné obsluhy). Z hodnoty čekání jsou dále odvozeny ukazatele *sumy čekání*, *koeficient čekání*, *pravděpodobnost čekání* a *koeficient pravděpodobnosti čekání*.

2.3 Limitní a kritické hodnoty

Pro jednotlivé ukazatele kapacity mohou být definovány *limitní hodnoty*. Na základě vztahu mezi aktuální hodnotou ukazatele kapacity a příslušnou limitní hodnotou je možné formulovat závěry o předpokládané kvalitě provozu nebo závěry týkající se propustnosti.

Rozlišují se dva typy limitních hodnot: (a) *optimální hodnota*, (b) *kritická hodnota*. S ohledem na kvalitu provozu je žádoucí, aby hodnota příslušného ukazatele kapacity nepřevyšovala optimální hodnotu. Pokud je aktuální hodnota vyšší než optimální hodnota, pak existuje riziko nedostatečné kvality provozu. V případě překročení kritické hodnoty lze předpokládat, že se jedná o nedostatečnou kvalitu provozu a zkoumaný segment (nebo zařízení) je přetížený.

3 ŽELEZNIČNÍ DOPRAVA A KAPACITA INFRASTRUKTURY

Provoz na železnici lze rozdělit na dvě základní kategorie dle typu přepravovaného nákladu – *osobní* a *nákladní* dopravu. Osobní vlaková doprava, sloužící k přepravě osob, je charakterizována obvykle taktovými jízdními řády s vyšší intenzitou dopravy v období ranní a odpolední špičky. Osobní dopravu je možné dále rozdělit dle vzdálenosti mezi přepravními body na *regionální* a *dálkovou*. Regionální doprava typicky spojuje města s blízkými menšími obcemi a je tvořena zejména *osobními vlaky*. Dálková doprava spojuje jednotlivá větší města nebo státy a je tvořena především *rychlíky* a *vlaky vyšší kvality*.

Nákladní doprava slouží k přepravě zboží a materiálů mezi centry průmyslu a obchodu. Částečně může být pravidelná, ale vyskytuje se zde větší množství nepravidelných nákladních vlaků (tzv. vlaků dle potřeby).

V běžném provozu na železnici se zpravidla vyskytuje doprava smíšená, která obsahuje vlaky osobní i nákladní. V ideálním světě by všechny vlaky jezdily dle předepsaných grafikonů vlakové dopravy (jízdních řádů), v realitě dochází k výskytu neočekávaných vlivů, a ty mohou negativně ovlivnit provoz vlakové dopravy (spojený se zpožděními vlaků). Takové nepravidelnosti jsou negativně vnímány především pasažéry osobní vlakové dopravy, kteří pociťují snížený komfort dopravy kvůli nutnosti čekání na vlakové spoje nebo výskytu dalších problémů, například problémům s přestupem.

Nedodržení předepsaného jízdního řádu může vzniknout vlivem výskytu náhodných událostí (mimořádností v dopravě) jako je porucha na dopravní infrastruktuře a jejích zařízeních (závada trakčního vedení nebo zabezpečovacího zařízení, poškození kolejí, ...) nebo porucha na kolejových vozidlech (např. porucha hnacího vozidla). Porucha na infrastruktuře železnice může velmi výrazně ovlivnit i další vlaky v dané oblasti. Pokud dojde k závadě na jedné koleji u dvoukolejné trati, je možné využít druhou kolej k obousměrnému provozu, ale nelze vždy očekávat, že jedna kolej dokáže v plné kvalitě obsloužit všechny původně plánované vlaky. Zpožděné vlaky mohou dále ovlivňovat další provoz na železnici a způsobit tak vznik následného zpoždění u dalších vlaků.

Aby bylo možné posoudit, zdali je dopravní infrastrukturu efektivně využívána, je nutné zjistit kapacitu dané infrastruktury a provést vyhodnocení jejího využití s ohledem na stanovené limitní hodnoty. Tak je možné charakterizovat, jak je daný jízdní řád infrastruktura schopna obsloužit. Pokud by byl rozsah dopravy na infrastruktuře naddimenzovaný, jakékoliv zpožděné vlaky by mohlo být problematické obsloužit a docházelo by k tvoření následného zpoždění u dalších vlaků. Pro optimální nebo nižší než optimální rozsah dopravy by mělo být možné do jisté míry nepravidelnosti absorbovat. Při konstrukci jízdních řádů je tedy vhodné mít informaci o tom, kolik vlaků může daná část železnice obsloužit. Tento problém řeší stanovení kapacity dané trati či železniční stanice.

3.1 Kapacita

Kapacita představuje schopnost realizovat určitý výkon s určitou kvalitou. Tato obecná definice zcela nevyjadřuje, jakou komplexnost tento termín skrývá. Kapacitu je nutné vztáhnout k určitému segmentu infrastruktury – konkrétní železniční stanici, či konkrétní trati. Není možné definovat kapacitu jako globální míru, pak by bylo velmi složité postihnout detaily jednotlivých segmentů. Vztažením kapacity na konkrétní omezenou infrastrukturu dále značí, že různá infrastruktura nabízí různou kapacitu.

Kapacita se nedá jednoduše definovat pro „obecný“ provoz. Kapacita a hodnoty jejích ukazatelů jsou obvykle závislé na definovaném provozu, na požadavcích na jeho kvalitu, ale také na očekávané míře nepravidelností a velikosti rezerv v jízdních řádech. Kapacitu je nutné vztahovat ke konkrétní situaci, která zahrnuje infrastrukturu a provozní poměry na železnici.

Kapacita jako taková není jediným číslem, slovem či obecnou hodnotou. Pro kvantifikaci kapacity slouží její ukazatele, které vyjadřují konkrétní sledované charakteristiky (vlastnosti) kapacity. Pro komplexní pochopení chování části železničního systému za vybrané provozní situace je navíc nutné sledovat více různých ukazatelů kapacity. Mezi ukazatele kapacity patří např. *propustnost, využití propustnosti, stupeň obsazení, pravděpodobnost čekání* nebo *přírůstek zpoždění*.

Samotné vyhodnocení, zdali je zkoumaná část železnice využívána nedostatečně, optimálně nebo je naopak přetížená, je následně provedeno porovnáním získaných hodnot ukazatelů kapacity se stanovenými limitními hodnotami. Tyto limitní hodnoty mohou být definovány v příslušné lokální, národní či nadnárodní metodice pro vyhodnocování kapacity infrastruktury.

Kapacitu je možné zjišťovat s využitím různých přístupů (metod), přičemž jednotlivé přístupy mohou poskytovat různé ukazatele, které mezi sebou nejsou jednoduše porovnatelné.

3.2 Zjišťování kapacity železničních stanic

Metodiky k určování kapacity železnice se vyvíjejí od samotného vzniku železniční přepravy a existuje několik základních přístupů: analytické, grafické, graficko-analytické a experimentální (Molková, 2010).

Analytické metodiky definují matematický aparát vypočítávající dílčí a souhrnné ukazatele kapacity. Podkladem těchto metodik jsou data o infrastruktuře, informace o vlacích a jejich jízdní řád. Metodika představuje postup, který zpracuje vstupní data a jejím výstupem jsou jednotlivé ukazatele kapacity. Příkladem takové metodiky může být analytická metodika definovaná ve směrnici SŽDC SM124 (SŽDC, 2019), která vychází z dříve používané směrnice D24 Českých drah (ČD, 1965).

Grafické metodiky (UIC406, 2004) využívají sestavený (kompresní) grafikon vlakové dopravy. Ten je na rozdíl od standardního grafikonu sestaven bez dodatečných mezer a je tak maximalizováno využití infrastruktury. Zkonstruovaný grafikon je následně analyzován, jsou odečteny informace o celkové době obsazení kolejíště a na základě těchto informací je vyjádřena kapacita. Nevýhodou těchto metod je nutnost konstrukce grafikonu vlakové dopravy,

kteřá je do určité míry ovlivněna osobou, kteřá danou metodiku provádí. Grafické metodiky jsou někdy rovněž nazývány jako konstrukční metodiky.

Experimentální metody využívají techniky počítačové simulace s uplatněním stochastických jevů. Na rozdíl od výše zmíněných přístupů umožňuje tato technika velmi detailní prozkoumání daného systému za různých okolností. Výsledkem procesu zkoumání mohou být různé kvantitativní i kvalitativní ukazatele kapacity.

3.3 Analytické metody pro zjišťování kapacity

Analytické metody pro určování kapacity představují konkrétní matematický aparát (obvykle využívající teorii pravděpodobnosti a statistiku), pomocí kterého je možné vyjádřit kapacitu trati nebo železniční stanice.

Vstupní údaje pro výpočet jsou následující:

- informace o infrastruktuře (topologie kolejiště),
- informace o zabezpečovacím zařízení,
- provozní intervaly, následná mezidobí,
- informace o vlakových soupravách, trasách a jízdních řádech.

Stanovením konkrétních vstupních hodnot pro danou analytickou metodiku je následně možné provést výpočty jednotlivých ukazatelů kapacity. Typické výstupní ukazatele analytických metod zahrnují:

- propustnost, využití propustnosti,
- celkovou a průměrnou dobu obsazení,
- stupeň obsazení.

Konkrétní potřebné podklady a způsob výpočtu se liší dle použité metodiky. Mezi základní zástupce analytických metodik patří například:

- metodika *SŽDC SM124 – Zjišťování kapacity dráhy*,
- metodika *UIC CODE 406 – Capacity*,
- metodika *Richtlinie 405 – Fahrwegkapazität spoločnosti DB Netz AG* (Richtlinie 405, 2008).

3.3.1 SŽDC SM124

Směrnice SŽDC SM124 (SŽDC, 2019) definuje přístupy ke zjišťování kapacity v prostředí české železnice. Směrnice definuje analytickou, konstrukční a experimentální metodu. Principy využívané analytickou a konstrukční metodou jsou postaveny na dříve využívané metodice D24.

Metodika *SŽDC (ČSD) D24* představovala základní metodu užívanou na československé železniční síti (ČD, 1965). Tato metodika vznikla již v roce 1965 a byla užívána s několika změnami až do roku 2019. Obsahuje základní zásady posuzování kapacity železniční dopravní infrastruktury ve smyslu definice ukazatelů kapacity a metod jejich stanovení.

3.3.2 UIC 406

Kodex *UIC 406 – Capacity* je dokument doporučený pro členy UIC (International union of railways) od roku 2004 (UIC406, 2004). UIC 406 využívá obdobné principy jako metodika D24, nicméně metodika se snaží reagovat na současné podmínky železničního provozu (např. definice tzv. kapacitní bilance, nebo definice standardních hodnot využití kapacity tratí).

3.4 Experimentální metody

Experimentální metody uplatňují techniky počítačové simulace, pomocí nichž je možné komplexně posoudit kapacitní charakteristiky posuzované infrastruktury. V rámci počítačové simulace je možné posoudit různé dopravní situace, které by mohly nastat v realitě – posouzení v simulátoru je ale řádově rychlejší, nemá negativní dopad na ostatní provoz a je možné posuzovat i různé varianty železniční infrastruktury, které by byly časově i finančně nákladné na vybudování.

Na rozdíl od analytických a grafických metod mohou simulační metody poskytnout celou řadu dalších ukazatelů souvisejících s analýzou průběhu simulace: například vývoj zpoždění vlaků, stabilita jízdního řádu apod. Nevýhodou tohoto přístupu je vysoká časová náročnost na vybudování simulačního modelu a jeho následnou validaci. Rozsah vstupních dat je zde řádově vyšší než u analytických metodik.

3.4.1 Separátní simulace

Metoda *separátní simulace* (realizovaná simulačním nástrojem *SepSim*) je definována jako jedna z metod ve směrnici SŽDC SM124 (SŽDC, 2019). Tato metoda pracuje na mezoskopické úrovni podrobnosti a slouží k výpočtu kapacity traťových kolejí a staničních zhlaví. Vstupními parametry metody jsou informace o vlacích, jejich prioritách a parametrech vzniku vstupního zpoždění. Dále je nutné definovat technologické doby pro všechny vzájemné dvojice vlaků, kde dochází k obsazování stejných prvků infrastruktury.

V průběhu výpočtu metody jsou zaznamenány a analyzovány informace o čekání jednotlivých vlaků a o vzniku a velikosti následného zpoždění. Tyto informace jsou následně prezentovány v podobě výstupních textových protokolů. Hlavním hodnotícím ukazatelem separátní simulace je hodnota ukazatele čekání v provozu. Ukazatel propustnosti nelze z metody přímým způsobem vypočítat, ale je možné ho zjistit alternativním způsobem – iterativním použitím metody.

4 MODELOVÁNÍ A SIMULACE

Definice a termíny užívané v následujícím textu vycházejí z terminologie specifikované v Křivý a Kindler (2003). Počítačová simulace představuje experimentální výzkumnou metodu, která umožňuje zkoumat i velmi komplexní systémy. Metoda je založena na nahrazení zkoumaného systému (originálu) příslušným simulujícím systémem (neboli simulátorem) implementovaným na počítači. Se simulátorem jsou prováděny experimenty, přičemž vyhodnocení těchto experimentů vede k doporučením aplikovatelným na originál.

Technika počítačové simulace přímo neposkytuje výpočet, který by vedl k automatickému nalezení (optimálního nebo suboptimálního) řešení daného problému dle daných kritérií. Tato experimentální metoda poskytuje prostředky k provádění sérií simulačních experimentů řídicích se odlišnými scénáři. Simulace umožňují posuzovat různé varianty zkoumaného systému (v případě železnice – různé varianty infrastruktury a provozu) bez nutnosti tyto varianty fyzicky realizovat v rámci existujícího systému.

4.1 Systém

V rámci modelování a simulace dochází ke studiu nějaké věci, resp. možné varianty nějakého objektu (obvykle) z hmotného světa. Takový objekt může existovat nebo můžeme o jeho existenci uvažovat. Dokonale popsat reálný objekt je v současné době prakticky nemožné, proto jsou zaváděny abstrakce, které zanedbávají některé aspekty objektu. Popsatelnou abstrakci, vymezenou na objektu zkoumání nazýváme *systém*.

Abstrakce může anebo nemusí zahrnovat význam času. Systém, ve kterém se od času abstrahuje, se nazývá *statický systém*. Pokud je význam času v systému zahrnut, systém je označen jako *dynamický systém*. Simulace zahrnuje pouze dynamické systémy.

Množina okamžiků, v nichž dynamický systém existuje, se nazývá (*časovou*) *existencí systému*. Dynamický systém je v každém okamžiku své existence charakterizován určitým stavem.

Definovaný systém se skládá z *prvků*. Jednotlivé prvky mají své vlastnosti, které se nazývají *atributy*. V dynamických systémech se může počet prvků v průběhu existence systému měnit. Prvky, které v průběhu existence vznikají a zanikají v systému (resp. vstupují či vystupují ze systému) se nazývají *temporární prvky*. Na rozdíl od nich prvky, které jsou v systému přítomny po celou dobu existence, se označují jako *permanentní prvky*.

Stav dynamického systému v čase t je dán prvky, které jsou v čase t v tomto systému přítomny, hodnotami jejich atributů v tomto čase a relacemi mezi těmito prvky.

4.1.1 Model

V modelování a simulaci se termín *model* používá pro analogii mezi dvěma systémy. Vztah obou systémů – *modelovaného* a *modelujícího* je dán tak, že každému prvku P modelovaného systému je přiřazen prvek Q modelujícího systému, každému atributu g prvku P je přiřazen atribut h prvku Q a pro hodnoty atributů g a h je definována nějaká relace. Charakter relace není nějak obecně omezen.

Pokud je modelovaný i modelující systém statický, označujeme výsledný model jako *statický model*. V simulaci se však uplatní jen tzv. *simulační modely*, totiž modely, které splňují následující požadavky:

1. Jejich modelující i modelované systémy jsou dynamické systémy.
2. Existuje zobrazení τ existence modelovaného systému do existence modelujícího systému; je-li tedy t_1 okamžik, v němž existuje modelovaný systém M_1 , je mu přiřazen okamžik $\tau(t_1) = t_2$, v němž existuje modelující systém M_2 , a tak je pomocí zobrazení τ přiřazen i stavu $S_1(t_1) = \sigma_1$ systému M_1 stav $S_2(t_2) = \sigma_2$ systému M_2 .
3. Mezi stavy σ_1 a σ_2 jsou splněny požadavky na vztahy mezi prvky a jejich atributy, jak je uvedeno výše pro modely obecně.
4. Zobrazení τ je neklesající. Pokud nastane stav s modelovaného systému před stavem s^* téhož systému, pak stav \bar{s} (který v modelující systému odráží stav s) nastane před stavem \bar{s}^* (který odráží stav s^*), nebo mohou oba stavy nastat v modelující systému současně (totiž v případě, že modelující systém není na takové úrovni rozlišení, aby dokázal zobrazit všechny detaily v modelovaném systému), nikdy však nemůže být časové pořadí stavů v modelovaném systému a jim odpovídajících stavů v modelující systému nedodrženo (tj. je dodržena kauzalita).

Model představuje složitou strukturu, která váže dva systémy, jejich prvky a atributy a v případě simulačních modelů i existence obou systémů. Nicméně v neformálním popisu se často slovem *model* označuje modelující systém, *originál* pak představuje modelovaný systém. V případě, že jde o simulační model, označují se systémy jako *simulovaný* a *simulující*. Alternativně k termínu simulující systém se v praxi používá termín *simulační model* nebo také *simulátor*.

4.1.2 Modelování

V rámci modelování ve smyslu výzkumné techniky dochází k náhradě zkoumaného systému jeho modelující systémem („modelem“). Cílem modelování je získat pomocí pokusů s modelem informace o původním zkoumaném systému.

4.1.3 Verifikace a validace modelu

V souvislosti s počítačovou simulací se využívají pojmy verifikace a validace. *Verifikace* je proces, při kterém se prověřuje logická správnost činnosti simulujícího systému. Je tedy prověřováno, zda představa o fungování zkoumaného systému byla v simulátoru správně aplikována.

Pod pojmem *validace* simulujícího systému se rozumí proces, v jehož rámci se prověřuje, zda je chování tohoto systému dostatečně věrné při srovnání s realitou nebo s příslušným jiným již validovaným (počítačovým) simulátorem.

4.1.4 Simulace

Simulace je výzkumná technika, jejíž podstatou je náhrada zkoumaného dynamického systému jeho simulátorem s tím, že se simulátorem se experimentuje s cílem získat informace o původním zkoumaném dynamickém systému (Křivý, 2003).

4.2 Rozlišovací úrovně simulace

Simulaci lze klasifikovat dle uplatněné míry abstrakce na vymezený systém na objektu zkoumání, a to na *mikroskopickou*, *mezoskopickou* a *makroskopickou* simulaci. Simulace typicky pracuje s modelem na jedné úrovni podrobnosti, ale mohou existovat modely, které různé úrovně podrobnosti kombinují. Typickým příkladem mohou být distribuované simulace, kde jeden logický proces pracuje na mikroskopické úrovni podrobnosti a jiný logický proces na makroskopické úrovni podrobnosti. Dalším příkladem takových modelů jsou škálovatelné simulační modely (Burghout, 2004; Kettner, 2003).

4.2.1 Mikroskopická simulace

Simulátory pracující na mikroskopické úrovni podrobnosti se maximálně snaží zahrnout veškeré detaily modelovaného systému (Chung, 2009). Výsledky příslušné *mikroskopické simulace* nejpodrobněji popisují děje v originálu. U takovýchto modelů je třeba popsat jednotlivé elementární prvky reálného světa, jejich vlastnosti a jejich vzájemné vazby. Příkladem elementárních prvků z oblasti železničních systémů může být: kolej, výhybka, návěstidlo, lokomotiva, vůz, aj. V praxi to znamená, že takový simulační model je velmi komplexní a jeho vybudování může trvat netriviálně dlouhou dobu. Zároveň je vhodné poznamenat, že ani mikroskopické simulační modely obvykle nemohou pojmout veškeré detaily z reality a výsledky tedy nejsou stoprocentně shodné s výsledky, které by bylo možno naměřit provedením experimentů v realitě.

4.2.2 Mezoskopická simulace

Mezoskopické simulátory pracují s větší mírou abstrakce, než je tomu u mikroskopických simulátorů (Nuzzolo, 2017). Příslušný model neposkytuje tak přesné výsledky jako mikroskopický model, ale v určitých případech může být mezoskopická přesnost výhodnější. Zanedbáním některých detailů se zkracuje nutná doba k vybudování, verifikaci a validaci modelu. Takový přístup může být u některých druhů aplikací žádoucí. Uplatnění nalezne především tam, kde není kladen striktní důraz na přesnost výsledků, ale záleží také na rychlosti provedení simulačních studií.

4.2.3 Makroskopická simulace

Simulátory uplatňující makroskopickou úroveň podrobností abstrahují od sledování detailů (Barceló, 2010). Na rozdíl od předchozích druhů simulátorů zde obvykle nejsou sledovány konkrétní entity, ale souhrnné toky entit a jejich charakteristiky. Tento přístup se využívá pro simulace problémů, kde není potřeba sledovat konkrétní detaily, ale sleduje se charakter či trend proudů entit a jejich zpracování.

Závěrem k uvedeným úrovním podrobností simulace je možné konstatovat, že neexistuje jednoznačný předpis, jaké postupy volit pro uplatnění míry abstrakce (míry *granularity*) pro jednotlivé typy simulací. Volba vhodné úrovně podrobnosti a konkrétní realizace abstrakcí v simulačním modelu závisí na povaze zkoumání ve zpracovávané simulační studii, požadavcích na ni a očekávaných výsledcích.

4.3 Simulace železniční dopravy

Železniční doprava představuje rozsáhlý a komplexní systém, jehož simulace není triviální záležitostí. Základní statickou součástí železnice je kolejová infrastruktura, kterou je možné popsat pomocí matematického modelu typu graf, kde jednotlivé hrany odrážejí dílčí kolejové úseky. Počet hran v grafu je přímo úměrný rozsahu sledovaného území a přesnosti, se kterou je simulační model vybudován. Jednotlivé hrany (*permanentní prvky* simulujícího systému) obsahují množinu informací, které přesně popisují význam a vlastnosti příslušných kolejí (kolejový obvod, délka úseku, přítomné prvky zabezpečovacího zařízení, maximální povolené rychlosti vlaků, vlastnosti trakčního vedení, sklonové poměry a další vlastnosti).

Mezi *temporární mobilní prvky* pak patří jednak vlakové soupravy, lokomotivy a jednotlivá kolejová vozidla. U jednotlivých vlakových souprav mezi základní parametry patří identifikace vlaku, popis jednotlivých vozů soupravy a grafikon (jízdní řád). Další sledované prvky pak mohou odrážet například obslužný personál (výpravčí, technici, dispečeri apod.). V některých modelech je dále nutné sledovat cestující nebo přepravovaný náklad, záleží na typu a podrobnosti simulační studie.

Funkce celého systému je dále ovlivněna a řízena platnými dopravními předpisy a technologickými postupy s využitím staničního a traťového zabezpečovacího zařízení. Dokonalá simulace železniční dopravy by musela pojmout všechny výše zmíněné prvky včetně všech jejich vlastností. To ale v současné době není zcela možné, a tak i mikroskopické simulační nástroje zavádějí některé abstrakce, aby bylo možné simulace reálně provést. Výsledky dopravních simulací se běžně využívají jako podklad pro budoucí rekonstrukce infrastruktury nebo plánování grafikonů vlakové dopravy.

Mikroskopickou úroveň podrobnosti v rámci simulací železniční dopravy se v dnešní době zabývá již řada kvalitních komerčních softwarových produktů. Mezi významné zástupce patří nástroje *RailSys*, *OpenTrack* nebo *Villon*. Společnou charakteristikou těchto nástrojů a mikroskopického přístupu je problém, že vybudování a validace simulačního modelu na takto detailní úrovni podrobnosti trvá obvykle netriviálně dlouhou dobu.

4.3.1 Analýza současného stavu simulátorů železniční dopravy

Villon

Simulační nástroj *Villon* (Villon, 2016) vyvíjený společností *Simcon* představuje mikroskopický simulátor dopravních procesů podporující simulace železniční dopravy, silniční dopravy, závodové dopravy a manipulačních činností. Simulátor je možné uplatnit pro simulační studie osobních stanic železniční dopravy, průmyslových vleček, železničních sítí, ale i kontejnerových terminálů. *Villon* podporuje online animaci ve 2D i 3D zobrazení. Simulátor umožňuje generování post-simulačních protokolů využitelných pro statistické zpracování dat a tvorbu grafických časových protokolů.

Infrastruktura dopravní sítě v rámci simulačního modelu je připravována v měřítku dle skutečnosti a vyžaduje definování několika základních parametrů. Chování jednotlivých objektů (vlaků, vozidel, ...) v simulaci je definováno pomocí síťových grafů odrážejících

příslušné technologické postupy, které obsahují jednotlivé vykonávané operace. Nástroj *Villon* je rovněž využíván pro vytváření rozsáhlých komerčních simulačních studií.

OpenTrack

Dalším z dostupných nástrojů pro realizaci simulací v oblasti železniční dopravy je nástroj *OpenTrack* (Railway Simulation, 2017). Vznikl v polovině 90. let 20. století jako výzkumný projekt v ETH Zürich. Dnes je tento nástroj pro simulaci provozu na železnici používán železničními podniky, správci infrastruktury, dodavateli železnic, konzultanty a univerzitami v nejrůznějších zemích.

Simulátor je možné využívat pro úkoly typu: analýza kapacity tratí a stanic, konstrukce jízdního řádů, analýza vlivu zpoždění a další. Na rozdíl od nástroje *Villon* je infrastruktura modelována schematicky pomocí grafu využívajícího tzv. dvojvrcholy (obdoba *polárního grafu*), který obsahuje data o parametrech infrastruktury. Řízení vlaků je v *OpenTracku* realizováno pomocí definování časového jízdního řádu s několika parametry. Simulátor také poskytuje online animaci a sadu post-simulačních protokolů. *OpenTrack* je také často využíván pro realizaci komerčních simulací.

RailSys

Simulátor *RailSys*, který byl vytvořený společností *RMcon* (RailSys, 2017), je zaměřen na komplexní simulace železniční dopravy zahrnující konstrukci jízdních řádů, analýzu kapacity a celkovou optimalizaci plánovacích procesů. Na rozdíl od předchozích simulátorů je *RailSys* striktně vyhrazen pouze pro komerční zákazníky a není možné jej blíže představit pro nedostatek volně dostupných materiálů.

Nemo

Na opačné straně úrovně podrobnosti simulace se nachází simulační nástroj *Nemo*, který představuje makroskopický simulační nástroj železničního provozu (Kettner, 2003). Jeho hlavním cílem je podpora strategického plánování a hodnocení provozu. Umožňuje provádět vyhodnocování různých strategií pro řízení dopravních systémů, dále umožňuje vyhodnotit dopady plánovaných změn v železniční infrastruktuře nebo vliv nových dopravních služeb v systému. Dopravní síť je reprezentována jako graf. Vrcholy představují abstrakci jednotlivých stanic a křižovatek, hrany představují propojení jednotlivých vrcholů tratěmi. Jednotlivé prvky mohou mít definovanou sadu atributů blíže popisující chování a vlastnosti daných prvků.

Ostatní

Kromě výše uvedených simulátorů existuje v současnosti řada dalších převážně specializovaných simulátorů. Část simulátorů se zaměřuje na problematiku určování kapacity. Mezi tyto nástroje patří například softwarový nástroj *Luks* (LUKS – Analysis of lines and junctions, 2011), který umožňuje provádět hodnocení kapacity železničních stanic, křižovatek a tratí a provádět hodnocení kvality provozu. Nástroj je primárně využíván v rámci metodiky Richtlinie 405 společnosti DB Netz AG a nejsou o něm dostupné detailní veřejné informace.

Jako další simulátor je možné uvést specializovaný nástroj *SepSim*, určený pro provádění mezoskopických simulací, který je využíván v rámci tzv. separátní simulace definované v rámci směrnice SŽDC SM124 (SŽDC, 2019).

Dalším z dostupných nástrojů je platforma *HERMES*, která představuje komplexní platformu pro řešení železničních simulací (HERMES Rail Simulation, 2017). Platforma nabízí například možnost realizace online trenažerů nebo detailní modelování ETCS systému (podporována je úroveň ETCS 2, podpora úrovně 3 je aktuálně ve vývoji).

Na problematiku energetiky a optimalizace rozvodných soustav používaných k napájení vlaků pomocí trakčního vedení se zaměřují např. nástroje *eTraX* (Railway Traction Power Analysis, 2017) nebo *EnerGplan* (EnerGplan: Simulation Tool, 1997).

Lze tedy shrnout, že v současnosti se na trhu nachází několik majoritních zástupců mikroskopických simulátorů železničního provozu, které poskytují komplexní služby pro realizaci simulačních studií. Jednotlivé produkty se od sebe liší zvolenými přístupy při vytváření simulačních modelů (definice infrastruktury a pravidel vlakového provozu), ale jejich jádro poskytuje podobné detailní simulace. Výstupy těchto nástrojů tvoří online animace a sady post-simulačních protokolů pro následné podrobné analýzy simulovaného provozu. Všichni jmenovaní zástupci představují běžně komerčně používané simulátory.

4.3.2 MesoRail

Původní simulační nástroj *MesoRail* (Diviš, 2015) představuje vyvíjený mezoskopický simulátor železničního provozu, který se zaměřuje na zkoumání kapacity železničních uzlů. Simulátor se zaměřuje na podporu pro rychlé prototypování železniční infrastruktury a rychlé provedení specifikace provozních podmínek (zahrnující typy a složení vlakových souprav, jejich časové plány, parametrizace náhodných vstupů aj.).

Cílem simulátoru je umožnit provádění simulačních studií v kratší době, než je tomu u standardních mikroskopických simulačních nástrojů. Z tohoto důvodu jsou v *MesoRailu* zavedeny některé abstrakce (uvedené níže), které simulátor stavějí do úrovně mezoskopické simulace. V kontrastu s těmito abstrakcemi je fakt, že některé aspekty simulátoru odpovídají mikroskopické úrovni podrobnosti, ale kvůli výše zmíněným abstrakcím, bude v dalším výkladu *MesoRail* označován jako mezoskopický simulační nástroj. Hlavním problémem je nalezení vhodného kompromisu a míry abstrakce pro jednotlivé části simulátoru tak, aby bylo dosaženo požadovaného výsledku, resp. požadované přesnosti. Pro stanovení vhodných abstrakcí byly využity konzultace s železničními experty.

Ve vyvíjeném simulátoru jsou uplatněny následující abstrakce – (i) schematické zobrazení železniční infrastruktury, (ii) zjednodušení výpočtů dynamiky jízdy vlaku (Diviš, 2014), (iii) zanedbání mobilních zdrojů obsluhy (zaměstnanci), (iv) zanedbání některých obslužných technologických postupů (např. posunů), (v) zanedbání entit cestujících. *MesoRail* nalezne uplatnění například pro zjišťování kapacity železničních uzlů, proto je nutné splnit následující požadavky – (i) rychlé prototypování infrastruktury, (ii) reálné jízdní vlastnosti vlaků, (iii) respektování funkčnosti zabezpečovacích zařízení, (iv) uplatnění staničních a traťových intervalů, (v) simulování deterministických a stochastických proudů vlaků, (vi) animační

výstupy v průběhu simulace, (vii) post simulační statistiky a grafické časové protokoly. Uvedený seznam je pouze výčtem základních předpokladů, které se od činnosti simulátoru očekávají. Na jejich základě je nutné stanovit úroveň abstrakce pro jednotlivé komponenty.

Simulátor je schopný provádět deterministické simulace provozu železničního uzlu a také stochastické simulace s využitím metody *vnořených simulací* nebo pomocí metody *prioritního plánování* jako podpory rozhodování pro řešení konfliktních situací. Tyto metody jsou blíže představeny v kapitole 5.

4.3.3 Deterministický a stochastický režim simulace železničního provozu

Deterministická simulace představuje simulaci, která nezahrnuje výskyty náhodných událostí, resp. náhodné vstupy. Taková simulace je opakovatelná a výsledky je možné potenciálně spočítat manuálně (v případě mikroskopických simulací je takový postup velmi pracný a prakticky nerealizovatelný).

Konkrétním příkladem může být deterministická simulace provozu železniční stanice, která představuje případ, kdy všechny vlaky jezdí dle jízdního řádu, nedochází k žádným náhodným výlukám, zpožděním nebo jiným nestandardním událostem. Taková simulace je pak předvídatelná a její průběh je vždy stejný. Tento přístup je uplatnitelný pro prvotní posouzení proveditelnosti konstruovaného grafikonu za ideálních podmínek (bezkonfliktní grafikon).

Stochastické simulace přinášejí do výpočtu prvek neurčitosti. Minimálně jedním ze vstupů simulace je náhodná veličina. V dokonalém případě by pak takové simulace nebyly reprodukovatelné a každé provedení simulace by mohlo dojít do jiného koncového stavu. V počítači by bylo potřebné náhodné veličiny reprezentovat pomocí generátorů náhodných čísel. Avšak, dokonalý generátor náhodných čísel, který by poskytoval ideální instance náhodné proměnné, by však byl technicky jen velmi obtížně realizovatelný. Místo generátorů náhodných čísel se běžně používají generátory pseudonáhodných čísel, které jsou parametrizovatelné tzv. hodnotou počáteční násady (*seed*). Ta poté ovlivňuje generovanou posloupnost pseudonáhodných čísel. Zvolením konkrétní hodnoty násady generátoru pseudonáhodných čísel je určena tzv. *replikace simulace*. Replikace jsou reprodukovatelné, a aby stochastická simulace poskytovala kvalitní výsledky, je obvykle nutné provést větší množství různých replikací a jejich výsledky pak statisticky vyhodnotit.

V rámci simulace provozu železniční stanice při uvažování stochastických vstupů může docházet například k (i) náhodnému zpoždění jednotlivých vlaků na příjezdu, (ii) poruchám vlaků nebo technologických zařízení, (iii) příjezdu neplánovaného vlaku a k dalším jevům. V rámci stochastické simulace může dojít k tomu, že v časovém okamžiku simulačního času t ve stanici na koleji k stojí vlak (plánovaný) a jiný vlak v důsledku velkého zpoždění přijíždí v okamžiku t ke stanici a měl by zastavit rovněž na koleji k , která je obsazená. Kolej k tak představuje z pohledu simulačního experimentu zdroj s omezenou kapacitou (typicky jeden vlak či lokomotiva), který je v důsledku stochastických jevů již alokovan a není pro druhý vlak dostupný. Toto je jeden příklad *konfliktu*, který může nastat (na rozdíl od bezkonfliktní deterministické simulace). Podobná situace může nastat např. při současném odjezdu dvou vlaků, které požadují stejnou traťovou kolej, nebo pokud by měly dva vlaky vjet na stejnou

traťovou kolej v opačných směrech. Takovéto provozní konflikty je možné řešit pomocí dvou přístupů: (i) v rámci *interaktivního režimu simulace* (řešení s využitím vstupu od uživatele) nebo (ii) pomocí některé techniky *automatické podpory rozhodování*.

5 PODPORA ROZHODOVÁNÍ V SIMULACI

Simulátory železničního provozu mohou řešit vznikající provozní konflikty v průběhu simulačních experimentů pomocí různých přístupů, resp. metod. Příslušná řešení mohou být získávána přímo od uživatelů simulátoru (v rámci interaktivního režimu simulace) anebo mohou být uplatňovány automatizované výpočty založené na exaktních nebo heuristických metodách *operačního výzkumu* (např. *celočíselné lineární programování*, *vícekriteriální hodnocení variant*), metodách *soft computingu* (např. využívajících *umělé neuronové sítě*, *fuzzy logiku* apod.), expertním statickým *prioritním plánování* atd. Další možností je využití tzv. *reflektivních vnořených simulací*, které pro daný časový výhled (vzhledem k okamžiku výskytu danému konfliktu) prediktivně vyhodnotí odlišné varianty řešení.

5.1 Interaktivní režim simulace

Je-li v případě výskytu konfliktu automatický simulační výpočet pozastaven a přepnut do interaktivního režimu, rozhodne o příslušném řešení uživatel. Toto řešení je dále uplatněno pro pokračující automatický výpočet, který je opět pozastaven při výskytu dalšího konfliktu. Tento přístup může být u některých experimentů dostačující a dále může být uplatňován u vybraných typů trenažerů, které se nezaměřují na výcvik v reálném čase. Tyto trenažéry umožňují školit a trénovat uživatele v rozličných dispečerských oblastech, přičemž využívají běžící simulace, které uživatelé ovlivňují svými vstupy/řešeními. Jako příklad komerčního simulátoru železničního provozu, který může být využit jako zmíněný typ trenažeru, lze uvést produkt *Villon* (Adamko, 2008).

Pokud je však potřebné provádět časově náročnější simulace s velkým množstvím replikací, tak dochází ke značnému navýšení počtu konfliktních stavů. Potom by provádění simulací s interaktivním rozhodováním bylo pomalé a náročné pro uživatele. Navíc, rozhodování uživatelů obvykle nebývá v daných situacích optimální, což může způsobit zhoršení kvality výsledků simulace. Pro tyto případy je tedy vhodnější využívat softwarové automatizované podpory rozhodování.

5.2 Prioritní plánování

V rámci železniční dopravy jsou provozní konflikty typicky spojeny se soupeřením vlaků, resp. souprav železničních vozů o zdroje (tj. části kolejové infrastruktury, obslužný personál, technická zařízení, posunovací lokomotivy apod.). Konflikt je v daném časovém okamžiku typicky reprezentován požadavkem na přidělení zdroje, který není v tomto čase disponibilní. U železniční dopravy se předpokládá existence příslušných (statických) plánů provozu - např. jízdních řádů, plánů obsazení kolejí v železničních stanicích, pracovních rozvrhů obslužného personálu apod., podle nichž je železniční provoz organizován. Konflikty potom vznikají v situacích, kdy není možné v rámci aktuálního provozu dodržet některý z uvedených plánů a je potřebné rozhodnout o příslušných alternativních řešeních.

Statické prioritní plánování představuje velmi jednoduchý způsob automatického rozhodování ohledně konfliktů, přičemž je využíváno vybranými komerčními softwarovými nástroji (*OpenTrack* (Nash, 2004), *Villon* (Adamko, 2008)), které se specializují na simulace železniční

dopravy. Před vlastním zahájením simulací je pro každý typ konfliktu vytvořena prioritní fronta možných alternativních řešení (ohledně přidělení zdrojů). Řešení s nejvyšší prioritou obvykle odpovídá přidělení zdroje v souladu s příslušným provozním plánem. Při výskytu konfliktního stavu v simulaci je automaticky zvoleno použitelné řešení s nejvyšší prioritou. V případě, že žádné z řešení z fronty nelze realizovat, žadatel o zdroj obvykle čeká do té doby, než je některé z alternativních řešení proveditelné.

Jako příklady prioritních front lze uvést: (i) seznam alternativních nástupištních kolejí pro pobyt osobních vlaků přijíždějících do železniční stanice ze směru D_i a odjíždějících ze stanice do směru D_j , (ii) seznam alternativních jízdních cest pro tranzitní nákladní vlaky přijíždějící do stanice ze směru D_k a odjíždějící ze stanice do směru D_l apod. Je doporučeno, aby prioritní fronty byly vytvářeny odborníky na železniční dopravu, kteří jsou schopni kvalifikovaně posoudit, která alternativní řešení jsou aplikovatelná. Z tohoto hlediska není formálně uplatňováno přesně specifikované optimalizační kritérium, ale pouze zkušenosti odborníka, který odhaduje míru narušení plánovaného provozu pro jednotlivá alternativní řešení. Je nutné konstatovat, že příprava zmíněných front je pro rozsáhlejší modelované systémy poměrně pracná.

Prioritní plánování založené na expertně dobře sestavených prioritních frontách nicméně neumožňuje automatizovaně detailně vyhodnocovat, která z řešení jsou objektivně lepší než jiná, respektive jaké dopady mohou mít uplatněná řešení pro budoucí provoz.

5.3 Expertní systémy

Expertní systémy představují skupinu počítačových programů, jejichž cílem je poskytovat expertní rady korespondující se znalostmi odborníků v dané aplikační doméně. Expertní systémy se dělí podle typu použití na několik základních druhů. Typickými představiteli jsou softwarové systémy zaměřené na provádění různých druhů diagnostiky nebo plánování. Diagnostické expertní systémy slouží k nalezení diagnózy (řešení) daného problému. Uplatnění tohoto typu systému nemusí být jen v oblasti medicíny pro určení diagnózy choroby pacienta, ale také může být například využitelný pro řešení dopravních, logistických a jiných problémů (Tan, 2016).

Na rozdíl od konvenčních programů jsou znalosti expertních systémů uloženy odděleně od dat a inferenční mechanismus, který řídí expertní systém, využívá dat i znalostí k nalezení řešení. Vlastní inferenční mechanismus může být založen na různých principech (např. vyhodnocování logických pravidel, fuzzy logika, umělé neuronové sítě apod.).

Expertní systémy jsou rovněž využívány v oblasti železniční dopravy, kde poskytují prostředek pro operativní řízení provozu a plánování jízd vlaků a jejich vzájemných interakcí. Autoři Fay a Schnieder (Alexander, 1999) představují základní problematiku řízení a plánování železničního provozu s užitím expertních systémů a poskytují přehled již existujících expertních systémů využívaných ve zmíněné oblasti. Dále je představen vlastní návrh expertního systému s využitím znalostí a fuzzy logiky. Další vylepšení tohoto systému je představeno v rámci (Fay, 2000), kde je navržen a popsán podpůrný dispečerský systém založený na fuzzy Petriho sítích.

Jiný příklad expertního systému založeného na pravidlech využitého pro plánování železniční dopravy je ESTRAC-III (Komaya, 1990). Jeho výstupem je plán změn, který je uplatněn na existující grafikon vlakové dopravy. Systém pro rozhodování využívá i počítačové simulace, pomocí kterých zjišťuje dopady zpoždění vlaků a navrhuje plánovaná opatření. Simulace je založena na technice „partial simulation“, která seskupuje související události spojené s jednotlivými stanicemi, a tímto způsobem rozděluje simulovaný systém železniční dopravy na dílčí podsystémy (Komaya, 1991).

Obdobný problém je řešen i realtime expertním systémem Sepia (Larroche, 1994). Na rozdíl od předchozího případu je rozhodovací mechanismus založen na prohledávání a úpravách struktury stavového grafu.

Příklad expertního systému použitého pro operativní řízení na železnici v aktuální době může být systém založený na fuzzy logice použitý pro plánování nákladní dopravy na Daqin Line v Číně (DENG, 2018). Pro řešení výlukových stavů byl navržen expertní systém založený na fuzzy Petriho sítích, který byl testován na případové studii železnice v Taiwanu (Cheng, 2009).

Přestože výše uvedené expertní systémy obvykle cílí na využití v reálném provozu, je možné je rovněž využívat i jako prostředek podpory rozhodování v simulátorech (Standridge, 2000). Navzdory této skutečnosti nejsou uvedené expertní systémy doposud reálně dostupné pro použití v obecných simulátorech železničního provozu.

5.4 Metody operačního výzkumu a soft computing

Operační výzkum představuje rozsáhlou disciplínu zahrnující poměrně širokou škálu exaktních a heuristických optimalizačních metod. Některé z těchto metod mohou být potenciálně využitelné pro automatizovanou podporu rozhodování v rámci simulátorů železničního provozu. Jako příklady lze uvést metody smíšeného celočíselného lineárního programování (Bai, 2014; Jánošíková, 2014), metody vícekriteriálního hodnocení variant (Bažant, 2019), metody z teorie grafů (De Luca Cardillo, 1998) apod.

Z oblasti výpočetní inteligence (*soft computing*) lze rovněž potenciálně využívat některé metody, resp. přístupy pro získávání řešení ohledně konfliktních stavů v dopravních simulacích. V příslušných rozhodovacích komponentech simulátorů je možné aplikovat umělé neuronové sítě (Bažant, 2009), fuzzy Petriho sítě (Cheng, 2009) apod.

Výše uvedené metody, resp. přístupy jsou obvykle testovány v rámci jednotlivých případových studií, avšak nejsou standardně začleňovány do běžně dostupných komerčních simulátorů železničního provozu. Toto je způsobeno zejména tím, že je v rámci těchto simulátorů poměrně složité uživatelsky přátelským způsobem adaptovat, resp. parametrizovat tyto metody (zejména z důvodu specifikace požadovaného optimalizačního kritéria) pro řešení daných typů konfliktů.

5.5 Reflektivní vnořené simulace

Reflektivní vnořené simulace (dále pro ně bude používán akronym *RNS – Reflective Nested Simulations*), někdy též označované jako *rekurzivní simulace*, představují další techniku uplatnitelnou pro podporu rozhodování v simulátorech. Princip této techniky spočívá

v pozastavení vlastní hlavní simulace za podmínky, že došlo k výskytu konfliktu. Následně je pozastavená hlavní simulace naklonována do několika variant. Jednotlivé klony (podpůrné *RNS*) jsou odlišně parametrizovány tak, aby byly prověřeny různé možnosti řešení konfliktu v pozastavené *původní* simulaci. Tyto rekurzivní simulace (v zásadě se jedná o odlišné výhledy do budoucnosti) jsou spuštěny, po určité omezené době ukončeny a následně vyhodnoceny. Varianta vykazující nejlepší predikované výsledky (podle definovaného kritéria) je následně použita pro pokračování pozastavené hlavní simulace.

Výzkumem *RNS* se zabývala řada autorů, přičemž využití této techniky je uplatnitelné v odlišných aplikačních doménách. Větší množství publikací týkajících se zmíněné odborné oblasti uskutečnil E. Kindler, který se zaměřil jednak na teoretický popis vnořených simulací (Kindler, 2010), jejich kategorizaci, příslušnou odbornou terminologii (Krivý, 2006), a také na praktické uplatnění v různých odvětvích (Kindler, 2006).

Autoři Gilmer a Sullivan se v několika pracích zabývají efektivitou velkého množství replikací proti tzv. *vícetrajektorové simulaci* (Gilmer, 1999). Tento typ simulace představuje alternativní způsob prozkoumání stavového prostoru. Jediná (výchozí) simulace je v jednotlivých konfliktech (rozhodovacích stavech) rozvětvena a zkoumání pokračuje potenciálně ve všech možných variantách. Vzniká tak stromová hierarchie simulací. Oblastí zájmu Gilmera a Sullivana je zejména vojenský simulátor Eaglet, který simuluje pohyb vojenských jednotek dvou armád a jejich vzájemnou interakci. Autoři se věnovali i problematice využití *RNS* pro podporu rozhodování (Gilmer, 2000).

Dále je možné najít využití vnořených simulací pro vybudování škálovatelné simulace, technika vnořených simulací je zde využita pro řešení konfliktů vzniklých v subsystému pracujícím na makroskopické úrovni podrobnosti. Jednotlivé vnořené simulace jsou realizovány uvnitř submodelu pracujícím na mikroskopické úrovni podrobnosti, výsledky z vnořených simulací jsou po vyřešení konfliktu promítnuty zpět do makroskopického submodelu (Bonté, 2009). Samostatná skupina publikací prezentuje využití vnořených simulací (dvouúrovňových simulací) pro potřeby finančního a rizikového managementu (Gordy, 2010).

Navzdory skutečnosti, že se diskutované oblasti věnuje řada odborných publikací, není k dispozici ucelená metodika pojednávající o obecném využití *RNS* pro podporu rozhodování ohledně řešení konfliktů v průběhu simulačních experimentů.

Základní princip implementace *RNS* je poměrně jednoduchý a pro jejich spouštění se předpokládá možnost využívání výpočetních jader ze stávajících simulátorů. Avšak pro fyzickou realizaci *RNS* je nutné dále vyřešit několik technických problémů. Bližší popis problematiky metodiky a techniky *RNS* a jejího uplatnění jako podpory rozhodování obsahují následující kapitoly.

6 METODA REFLEKTIVNÍCH VNOŘENÝCH SIMULACÍ

Jak již bylo zmíněno v předchozí kapitole, *reflektivní vnořené simulace* představují jednu z možných podpor pro řešení konfliktů (například v simulátorech železničního provozu). V dalším výkladu bude věnována pozornost jednak původnímu technickému řešení *RNS* a jednak původní metodice jejich využívání v rámci simulačních studií. Uvedené technické řešení *RNS* realizuje obecnou podporu rozhodování, která není vázána na konkrétní aplikační doménu či specifický simulátor.

6.1 Technika reflektivní vnořené simulace

Implementace původní softwarové podpory (aplikující *RNS*) pro automatické řešení konfliktů v průběhu simulačních experimentů může využívat standardní výpočetní jádro (implementované v rámci příslušného simulátoru) jak pro realizaci hlavní simulace, tak pro realizace vnořených simulací. Nicméně pro integrovaný simulační výpočet zahrnující vnořené simulace je uplatňován původní řídicí postup (Diviš, 2016).

Před uvedením koncepce řízení simulačních výpočtů je potřebné uvést některá používaná označení a dále parametry, funkce a softwarové komponenty, jejichž nastavení, resp. implementace potom ovlivňuje průběh simulačních experimentů:

mainReplCount parametr udávající počet replikací hlavní simulace.

nestReplCount parametr udávající počet replikací prováděných pro jednotlivé varianty řešení konfliktů.

stopCond parametr, který definuje ukončovací podmínku běhu replikace vnořené simulace. Podmínka může být určena buď konkrétním okamžikem simulačního času (tj. délkou časového výhledu) nebo výskytem určitého stavu, resp. události). Problematika délky časového výhledu v rámci rekurzivních simulací je dále popsána v podkapitole 6.3.

maxVarnts parametr určující maximální povolený počet prověřovaných variant řešení pro každý z konfliktů (avšak neurčuje aktuální počet prověřovaných variant pro daný konkrétní konflikt).

maxLevel parametr určující maximální povolený počet úrovní vnoření pro *RNS* vzhledem k hlavní simulaci. Pokud dojde k výskytu konfliktu v replikaci vnořené simulace je možné jej buď řešit opět pomocí vnořených simulací anebo využít pro řešení konfliktu jiný přístup. Nastavení parametru *maxLevel* tedy definuje, do jaké maximální hloubky je povoleno rekurzivní simulace provádět.

VarntsGen(i) softwarový generátor, který poskytne pro *i*-tý konflikt varianty, které budou následně prověřovány prostřednictvím vnořených simulací.

CrOptim(j) funkce, která vyhodnotí podle daného optimalizačního kritéria příslušnou *j*-tou variantu řešení příslušného konfliktu.

- $S_{i,j,k}$ označení replikace simulačního experimentu. Pro vnořenou simulaci se jedná o k -tou replikaci příslušející j -té variantě řešení i -tého konfliktu, kde $k \in \langle 1 \dots nestReplCount \rangle$, $j \in \langle 1 \dots n \rangle$, $n \leq maxVarnts$ a $i \in N^+$ (množina kladných celých čísel). V případě, že se jedná o k -tou replikaci hlavní (tj. „nevnořené“) simulace, pak $k \in \langle 1 \dots mainReplCount \rangle$, $j = 0$, $i = 0$.
- ${}^dH_b(S_{0,0,r})$ označení hierarchicky uspořádané množiny replikací. Na hierarchicky nejvyšší (nulté) úrovni je r -tá replikace hlavní simulace $S_{0,0,r}$ (prvek-kořen), na nižších úrovních jsou příslušné replikace vnořených/rekurzivních simulací (vzhledem k $S_{0,0,r}$). Hodnoty d (*depth*) a b (*batch*) odpovídají příslušným nastavením parametrů $maxLevel$ a $nestReplCount$. Jestliže r -tá replikace hlavní simulace $S_{0,0,r}$ nevyužívá vnořené simulace, potom je příslušná množina označena ${}^0H_0(S_{0,0,r})$ a ${}^0H_0(S_{0,0,r}) = \{S_{0,0,r}\}$, $r \in \langle 1 \dots mainReplCount \rangle$.
- ${}^dC_i(V_j)$ označení i -tého konfliktu v rámci libovolné replikace $X \in {}^dH_b(S_{0,0,r})$ pro dané r . Po výskytu konfliktu pokračuje výpočet replikace X jeho j -tou variantou řešení (V_j).
- $Confl(S_{i,j,k})$ označení lineárně uspořádané množiny konfliktů, které se vyskytly v rámci replikace $S_{i,j,k}$. Uspořádání zohledňuje časová razítka výskytů příslušných konfliktů.

Hlavní část řídicího algoritmu simulačního výpočtu pro r -tou replikaci hlavní simulace využívající RNS (tj. $maxLevel > 0$) je následující:

ALGORITMUS 1: Řízení výpočtu replikace hlavní simulace

- (1) Inicializace čítače konfliktů: $i \leftarrow 1$
- (2) Zahájení simulačního výpočtu r -té replikace hlavní simulace ($S_{0,0,r}$).
- (3) Provádění výpočtu $S_{0,0,r}$ až do výskytu nejbližšího (i -tého) konfliktu v okamžiku t_i simulačního času. Simulace je v čase t_i pozastavena.
- (4) S využitím $VarntsGen(i)$ je navrženo n alternativních variant pro řešení i -tého konfliktu, kde $n \leq MaxVarnts$.
- (5) Stavový prostor $S_{0,0,r}$ je naklonován a pro každou variantu řešení daného aktuálního (i -tého) konfliktu jsou pro každý klon inicializovány replikace vnořených simulací: $S_{i,j,k}$, $j = 1 \dots n$, $k = 1 \dots nestReplCount$.
- (6) Jsou kompletně spočítány replikace vnořených simulací: $S_{i,j,k}$, $j = 1 \dots n$, $k = 1 \dots nestReplCount$. Výpočet každé z replikací je ukončen, jestliže je pro něj splněna příslušná ukončovací podmínka $stopCond$.
- (7) Jsou vyhodnoceny výsledky pro n odlišných variant řešení i -tého konfliktu. Vyhodnocení každé j -té varianty ($j = 1 \dots n$) je založené na statistickém vyhodnocení výsledků jejích k replikací ($k = 1 \dots nestReplCount$). Následně je pro pokračování $S_{0,0,r}$ vybrána ta varianta řešení i -tého konfliktu, která vykázala nejlepší výsledek zjištěný prostřednictvím funkce $CrOptim(j)$.

(8) Aktualizace čítače konfliktů: $i \leftarrow i + 1$

(9) Návrat na krok (3).

Popsaný algoritmus je ilustrován na obrázku 1b, kde je stylizovaně zobrazena hierarchicky uspořádaná množina ${}^1H_3(S_{0,0,r})$, která sestává z 22 replikací:

$${}^1H_3(S_{0,0,r}) = \{S_{0,0,r}\} \cup \{S_{i,j,k} \mid i=1; j=1\dots 3; k=1\dots 3\} \cup \{S_{i,j,k} \mid i=5,7; j=1\dots 2; k=1\dots 3\}.$$

Na nejvyšší (nulté) hierarchické úrovni je r -tá replikace hlavní simulace ($S_{0,0,r}$) a na první úrovni jsou příslušné vnořené simulace (parametr $maxLevel = 1$).

Konflikty ${}^1C_1, {}^1C_5, {}^1C_7 \in Confl(S_{0,0,r})$, které se vyskytly v průběhu hlavní simulace, jsou řešeny pomocí *RNS*. V uvedené hierarchii jsou pro každou variantu řešení konfliktu spuštěny tři replikace (parametr $nestReplCount = 3$). Pro konflikty z vnořených simulací: ${}^1C_2, {}^1C_3 \in Confl(S_{1,1,1})$, ${}^1C_4 \in Confl(S_{1,3,3})$, ${}^1C_6 \in Confl(S_{5,1,3})$ a ${}^1C_8 \in Confl(S_{7,2,3})$ již nejsou uplatňovány další vnořené simulace, ale tyto jsou řešeny pomocí jiného přístupu (například s využitím prioritních seznamů).

Obrázek 1a demonstruje specifický případ hierarchie ${}^0H_0(S_{0,0,r})$, která sestává pouze z jedné replikace hlavní simulace ($maxLevel = 0$) s konflikty: ${}^0C_1, {}^0C_2 \in Confl(S_{0,0,r})$. Pro řešení těchto konfliktů se *RNS* nevyužívají.

Uvedený algoritmus je možné dále podrobněji rozpracovat v tom smyslu, že i vnořené simulace mohou své konflikty řešit rekurzivně pomocí dalších podpůrných vnořených simulací, jak bude dále popsáno.

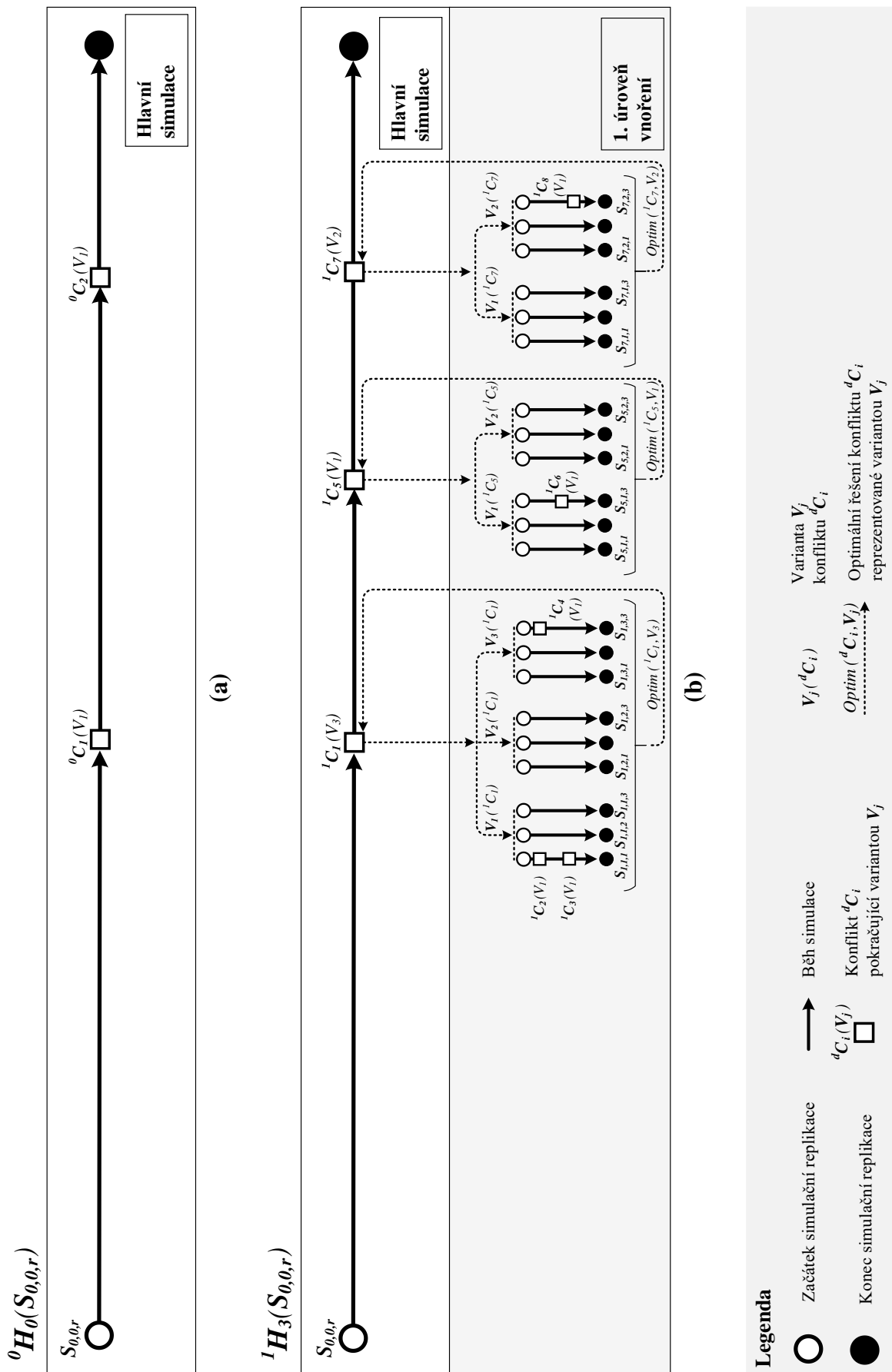
Samostatný problém představuje stanovení variant, které mají být pro každý konflikt prověřovány. Řešení tohoto problému (implementované v rámci generátoru *VarntsGen*) nelze příliš zobecnit, neboť je vždy závislé na typu konfliktní situace a na použité metodě pro určování variant, která může být ve výsledku spojena s řešením netriviální úlohy.

Použití vnořených simulací v rámci simulátorů železniční dopravy je spojeno pouze s technicky a technologicky přípustnými variantami řešení konfliktů. Přípustné varianty zohledňují: (i) konkrétní prověřovanou topologii kolejové infrastruktury, (ii) příslušný typ použitého zabezpečovacího zařízení a (iii) provozní předpisy platné pro zkoumaný provoz.

6.2 Výpočetní náročnost a paralelizace výpočtů RNS

Samostatný a rozsáhlý problém je u vnořených simulací spojen s technikou jejich výpočtů, které představují náročnou výpočetní úlohu. Složitost uvedené úlohy ovlivňují zejména následující faktory:

- počet konfliktů (v rámci hlavní simulace i vnořených simulací), pro jejichž řešení se využívá prověřování variant prostřednictvím prediktivních vnořených simulací,
- počet variant prověřovaných při řešení konfliktů,
- počet replikací vnořených simulací prováděných pro každou prověřovanou variantu,



Obrázek 1 – Hierarchicky uspořádané množiny replikací ${}^0H_0(S_{0,0,r})$ a ${}^1H_3(S_{0,0,r})$, zdroj: vlastní

- *délka časového výhledu* (resp. ukončovací podmínka) uplatňovaná ve vnořených simulacích,
- *počet replikací hlavní simulace*.

Výskyty konfliktů v rámci vnořených simulací mohou být spojeny s rekurzivním prováděním dalších vnořených simulací, což způsobuje exponenciální nárůst počtu spouštěných instancí simulačních běhů. Ohraničení rekurzivního šíření simulačních výpočtů lze zabezpečit prostřednictvím výše zmíněného parametru *maxLevel*. Řešení konfliktů ve vnořených simulacích pracujících na nejnižší povolené úrovni vnoření je potom založeno na uplatnění „nesimulačního“ přístupu (např. s využitím prioritních seznamů apod.).

Při technické realizaci výpočtů vnořených simulací lze využít skutečnost, že se jednotlivé replikace vnořených simulací (pracující na stejné úrovni vnoření) vzájemně neovlivňují a běží autonomně. Technicky je tedy možné vhodnou implementací podpořit provádění tzv. *multisimulací*, tj. paralelního zpracování nezávislých simulací. K tomuto účelu je možné využívat vhodné hardwarové výpočetní prostředky.

V rámci jednoho počítačového uzlu je možné *multisimulace* provádět s využitím více procesů nebo více vláken, čímž lze využít jednotlivá fyzická i virtuální jádra CPU pro efektivnější paralelní zpracování jednotlivých simulací. Další možnost realizace simulačních výpočtů představuje distribuce těchto výpočtů na více počítačových uzlů, které jsou typicky propojeny pomocí počítačové sítě nebo prostřednictvím internetu.

Pro provádění *multisimulací* je nutné, aby simulátor podporoval:

- uložení aktuálního stavu replikace simulačního výpočtu,
- modifikaci uloženého stavu pro přípravu spuštění příslušné varianty řešení konfliktní situace,
- serializaci uloženého stavu do datového souboru.

Vytvořené datové soubory je nutné vhodným způsobem předat jednotlivým procesům nebo výpočetním uzlům. Dle použitého způsobu realizace výpočtů se může například využívat:

- sdílená paměť pro výpočty v rámci jednoho výpočetního uzlu,
- sdílené souborové úložiště nebo komunikace prostřednictvím síťových soketů u distribuovaných výpočtů.

V průběhu výpočtů samotných vnořených simulací nevzniká potřeba jejich vzájemné komunikace nebo jejich synchronizace. Po dokončení každé replikace vnořené simulace je nutné nadřazené replikaci předat informace o příslušných výsledcích.

Z hlediska paralelizace výpočtů jednotlivých replikací lze výše specifikovaný *Algoritmus 1* dále rozpracovat. Níže je uveden příslušný pseudokód řídicího programu paralelních výpočtů, který zčásti využívá označení uvedená v podkapitole 6.1 a dále zavádí:

MainSimulation() hlavní program spouštějící paralelní výpočet replikací hlavní simulace.

<i>RunRepl(i,j,k,L)</i>	rekurzivní podprogram spouštějící replikaci $S_{i,j,k}$ pracující na L -té úrovni vnoření.
<i>Initialize(i,j,k,L)</i>	podprogram provádějící inicializaci replikace $S_{i,j,k}$ pracující na L -té úrovni vnoření.
<i>SaveResult(i,j,k,L)</i>	podprogram pro uložení výsledků replikace $S_{i,j,k}$ pracující na L -té úrovni vnoření. Výsledky zahrnují vyhodnocení replikace podle příslušného optimalizačního kritéria.
<i>MultirunVariant(i,j,L)</i>	podprogram zaměřený na paralelní výpočet replikací $S_{i,j,k}$ ($k = 1 \dots nestReplCount$) příslušejících j -té variantě řešení i -tého konfliktu.
<i>Simulate()</i>	základní rutina výpočetního jádra realizující životní cyklus běhu replikace.
<i>SelectOptim(i)</i>	funkce provádějící výběr optimální ze simulovaných variant řešení i -tého konfliktu (jednotlivé varianty jsou vyhodnoceny pomocí funkce <i>CrOptim</i>).
<i>EstimOptim(i)</i>	funkce provádějící („nesimulační“) odhad optimálního řešení i -tého konfliktu.
<i>FinishPoint(),</i> <i>ConflictPoint()</i>	booleovské funkce vyhodnocující, zda v probíhající replikaci došlo k výskytu konfliktu, resp. ke splnění podmínky jejího ukončení (dle parametru <i>stopCond</i>).
<i>ContinueVariant(i,j)</i>	podprogram nastavující parametry pokračování běhu replikace (po výskytu i -tého konfliktu) pro podmínky j -té varianty řešení.
<i>conflictID</i>	globální proměnná (čítač konfliktů).
<i>currConflict,</i> <i>bestVarnt, k, r, v</i>	pomocné proměnné.

V rámci pseudokódu jsou využívána klíčová slova „parbegin“ a „parend“. Výpočty podprogramů, jejichž volání jsou zapouzdřena uvedenými klíčovými slovy, jsou prováděny paralelně (pokud jsou k dispozici příslušné výpočetní prostředky).

ALGORITMUS 2: Řízení paralelních výpočtů replikací

```

program MainSimulation()
  conflictID ← 1
  for  $r = 1$  to mainReplCount do
    parbegin
      Initialize(0,0, $r$ ,0)
      RunRepl(0,0, $r$ ,0)
    parend
  end
end

```

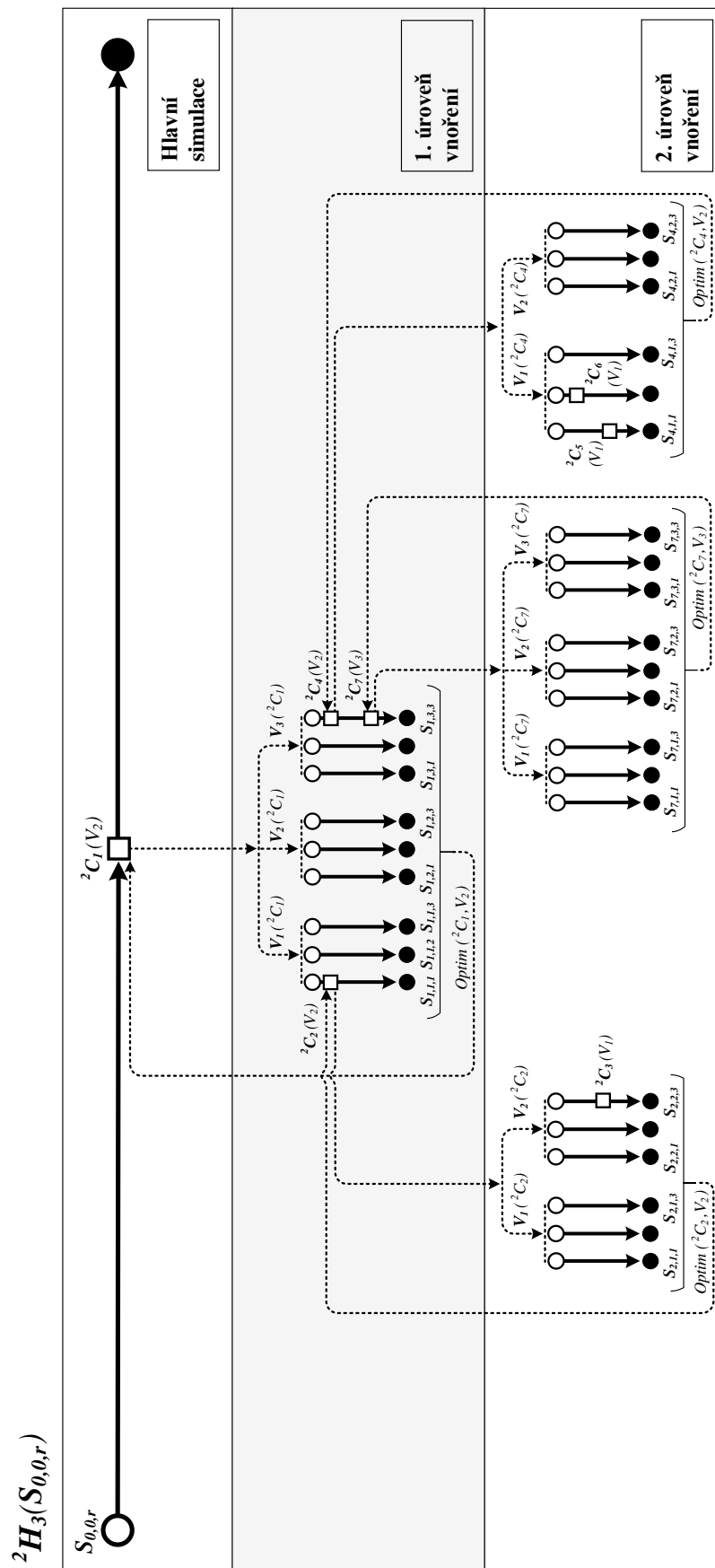
```

function RunRepl(i,j,k,L)
  while not FinishPoint() do
    repeat
      Simulate()
    until FinishPoint() or ConflictPoint()
    if FinishPoint() then
      SaveResult(i,j,k,L)
      exit
    else
      currConflict ← conflictID
      conflictID ← conflictID + 1
      if L < maxLevel then
        for v = 1 to VarntsGen(currConflict) do
          parbegin
            MultirunVariant(currConflict,v,L+1)
          parend
        end
        bestVarnt ← SelectOptim(currConflict)
      else
        bestVarnt ← EstimOptim(currConflict)
      end
      ContinueVariant(i,bestVarnt)
    end
  end
end

function MultirunVariant(i,j,L)
  for k = 1 to nestReplCount do
    parbegin
      Initialize(i,j,k,L)
      RunRepl(i,j,k,L)
    parend
  end
end

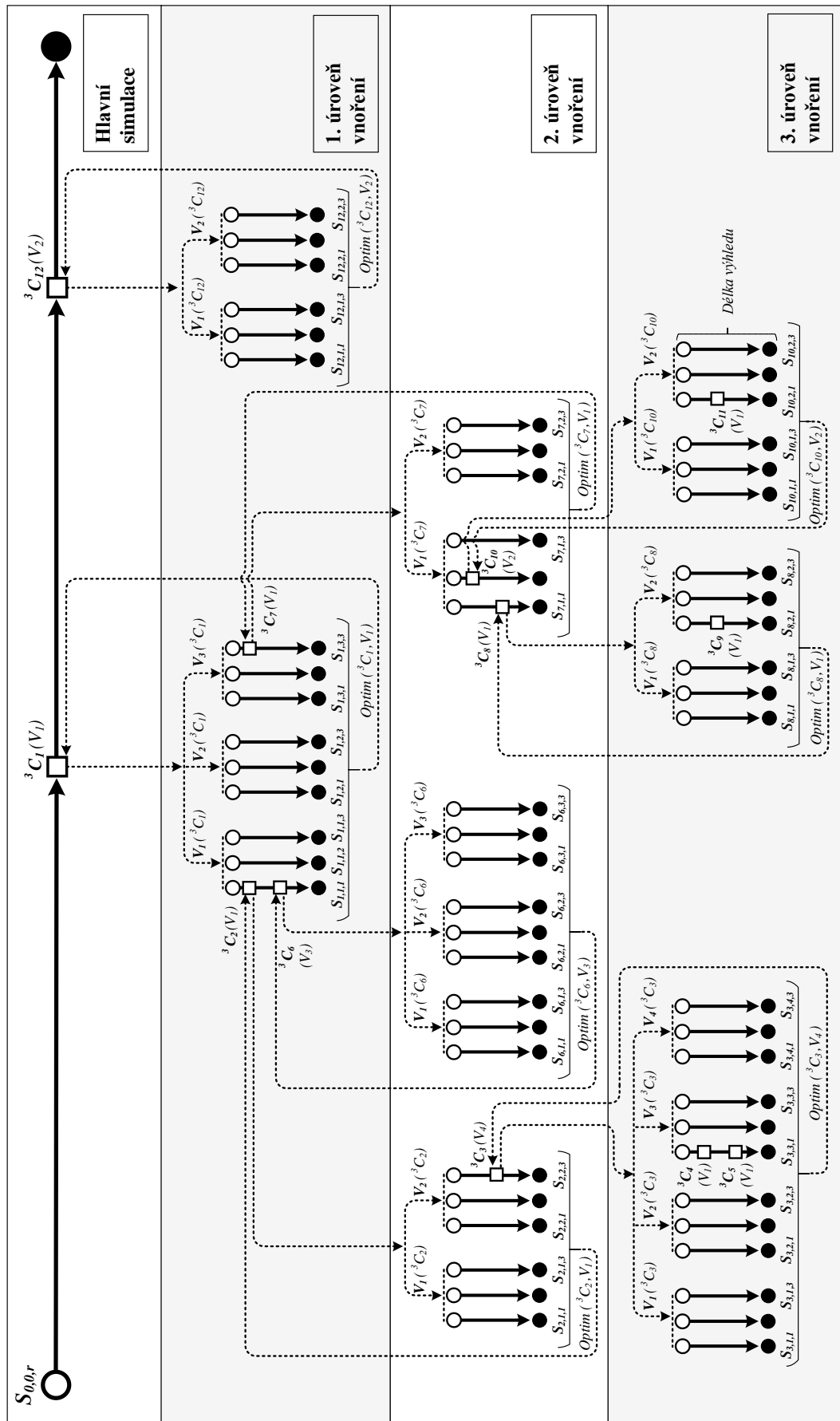
```

Příklady běhu představeného algoritmu s uplatněním odlišných nastavení parametru *maxLevel* jsou zjednodušeně graficky ilustrovány na obrázcích 2 a 3. Použitá značení jsou stejná jako na obrázku 1.



Obrázek 2 – Hierarchicky uspořádaná množina replikací ${}^2H_3(S_{0,0,r})$, zdroj: vlastní

${}^3H_3(S_{0,0,r})$



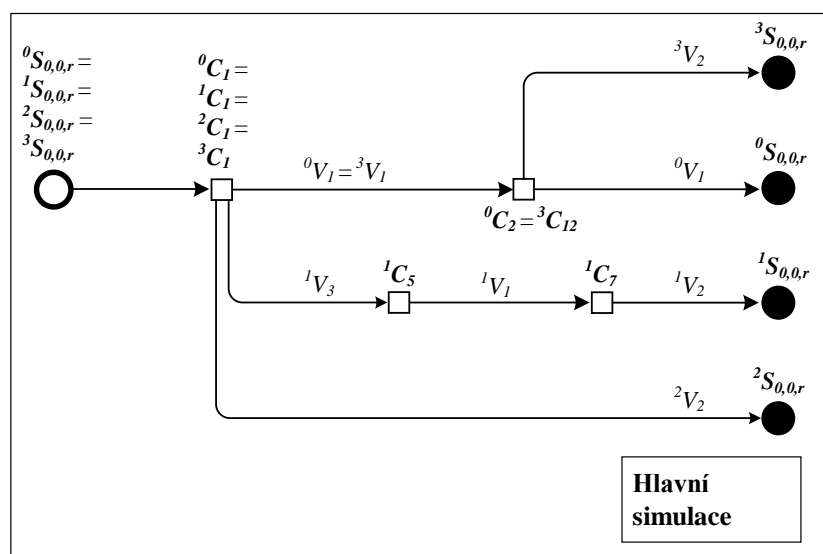
Obrázek 3 – Hierarchicky uspořádaná množina replikací ${}^3H_3(S_{0,0,r})$, zdroj: vlastní

V případě, že bude pro všechny uvedené demonstrační množiny ${}^0H_0(S_{0,0,r})$, ${}^1H_3(S_{0,0,r})$, ${}^2H_3(S_{0,0,r})$ a ${}^3H_3(S_{0,0,r})$ zvolena stejná strategie nastavování násad generátorů pseudonáhodných čísel v rámci jednotlivých replikací, potom lze indikovat stejné konflikty v rámci replikací z odlišných množin. Pro r -té replikace hlavních simulací, které se vyskytují ve zmíněných odlišných množinách, jsou stejné konflikty ilustrovány na obrázku 4. Analogicky je možné indikovat stejné konflikty i u vzájemně odpovídajících replikací vnořených simulací.

V rámci testování různých technik implementace vnořených simulací byl prověřován postup (na železničním systému popsaném v 9. kapitole), v jehož rámci byly prováděny simulační experimenty s postupně se zvyšující hodnotou maximálního povoleného počtu úrovní vnoření (*maxLevel*). Motivací pro uplatnění tohoto systematického přístupu bylo otestovat, zda by se vyplatilo průběžně dočasně uchovávat stavy paměťového prostoru simulátoru, které jsou spojené s výskyty všech konfliktů, pro jejichž řešení se aplikuje *RNS*. Pokud by při výpočtu určité replikace bylo identifikováno, že by část tohoto výpočtu byla totožná s již dříve realizovaným výpočtem jiné replikace (toto je ilustrováno na obrázku 4), nebylo by potřebné zmíněný částečný výpočet opakovat. Simulátor by byl potom převeden přímo do příslušného stavu, který byl uchován v paměti. Pro experimenty s odlišným nastavením parametru *maxLevel* byla vždy uplatněna stejná koncepce nastavování násad generátorů pseudonáhodných čísel pro jednotlivé replikace. Po zpracování dat z provedených testovacích simulačních experimentů bylo vyhodnoceno (Diviš, 2019), že uvedený postup nepřinesl významnější zefektivnění simulačních výpočtů využívajících *RNS*, a proto nebyl v rámci níže uvedené případové studie aplikován. Za hlavní důvody nedoporučení diskutovaného postupu lze označit:

- výraznou paměťovou náročnost (byť dočasně) uchovávání poměrně velkých počtů stavů simulátoru,
- nemalou časovou režii způsobenou častým načítáním a obnovováním stavů simulátoru a
- relativně malé úspory výpočetního času oproti případu, kdy není zmíněný postup využíván.

V rámci výše uvedeného článku (Diviš, 2019) byly dále navrženy heuristické metody pro snížení výpočetní náročnosti *RNS*. Jejich princip spočívá v prořezávání hierarchie vnořených replikací, tak aby nebylo nutné provádět veškeré simulace. Tyto metody v rámci prvotních experimentů vykazovaly úsporu výpočetního času v důsledku snížení potřebného počtu vnořených simulací o několik desítek procent. Tyto experimenty nicméně nevyhodnocovaly, jakým způsobem je ovlivněna kvalita rozhodování a jakých výsledků časové náročnosti lze reálně dosáhnout. Aplikace heuristických (i jiných) opatření za účelem snížení výpočetní náročnosti *RNS* je tak primárně záležitostí dalšího budoucího zkoumání a není dále detailně v práci popisována.



Legenda

${}^d S_{0,0,r}$ r-tá replikace hlavní simulace,
 ${}^d S_{0,0,r} \in {}^d H_3(S_{0,0,r})$

${}^d V_j$ Varianta řešení V_j
konfliktu ${}^d C_i$

Obrázek 4 – Indikace stejných konfliktů ve vybraných replikacích z ${}^0 H_3(S_{0,0,r})$, ${}^1 H_3(S_{0,0,r})$, ${}^2 H_3(S_{0,0,r})$, ${}^3 H_3(S_{0,0,r})$, zdroj: vlastní

6.3 Podmínka ukončení vnořených simulací

Parametr *stopCond* obecně definuje podmínku, která slouží k ukončení výpočtu replikace vnořené simulace. Parametr typicky může definovat podmínku, která je založena na výskytu specifického stavu v simulaci (tzv. *ukončovací* stav) nebo může definovat časový okamžik, kdy dojde k ukončení simulace (související s tzv. *délkou časového výhledu*). *Délka časového výhledu* (*LD – Lookahead Duration*) tedy udává interval simulačního času, který má replikace vnořené simulace k dispozici.

Pokud je parametr *stopCond* spojen s *délkou časového výhledu* a zároveň hodnota parametru *maxLevel* > 1, pak je nutné stanovit, jakým způsobem je *délka časového výhledu* uplatňována na nižších úrovních vnoření. Obecně je možné stanovit tři základní strategie, které lze uplatnit:

1. *konstantní doby výhledů*,
2. *hierarchicky redukováné doby výhledů*,
3. kombinace výše uvedených strategií.

Základní dvě strategie jsou blíže dále popsány. Dle potřeby řešeného problému je možné je kombinovat. V určitých případech je možné uvažovat i kombinaci použití *délky časového výhledu* a specifického *ukončovacího stavu*, kde například na první úrovni vnoření je uplatněna jedna strategie a na nižších úrovních vnoření je uplatněna alternativní strategie.

Konstantní doby výhledů

Strategie *konstantní doby výhledů* (*constant lookahead duration*) představuje jednoduchý přístup, kdy pro každou replikaci vnořené simulace, bez ohledu na její úroveň vnoření nebo další parametry, je přiřazen vždy stejný (výchozí) časový interval *délky časového výhledu*. Pokud při použití této strategie dojde k výskytu konfliktu ve vnořené simulaci, pak na nižší hierarchické úrovni vnoření je vždy uplatněna stejná délka výhledu. Praktický důsledek je takový, že na nižší úrovni vnoření dochází k simulaci budoucích stavů, časy jejich výskytů přesahují horní limit simulačního času u původní (hierarchicky nadřazené) vnořené simulace (dle simulačního času výskytu konfliktu a stanovené *doby výhledu*). V důsledku neomezování délky výhledu může docházet k výskytu dalších konfliktů a s ohledem na parametr *maxLevel* i k několikanásobnému prodloužení původní hodnoty *délky časového výhledu* vzhledem k základnímu časovému okamžiku hlavní simulace, kdy došlo k jejímu primárnímu přerušení.

Hierarchicky redukováné doby výhledů

Tato strategie uplatňuje následující postup, pokud se v průběhu výpočtu replikace vnořené simulace vyskytne další konflikt, který je řešen rovněž za pomoci *RNS*, je hodnota doby výhledu pro simulace na nižší hierarchické úrovni zkrácena o poměrnou část, která již byla „spotřebována“ na předešlých hierarchických úrovních. Tuto strategii lze nazvat jako *hierarchicky redukováné doby výhledů* (*hierarchically reduced lookahead durations*) a její použití lze demonstrovat na příkladu množiny replikací znázorněné na obrázku 3. Doba výhledu určená pro replikace na první úrovni vnoření se označuje jako *primární doba výhledu*. V rámci replikace $S_{0,0,r}$ hlavní simulace (na nulté hierarchické úrovni) došlo ke konfliktu 3C_1 v simulačním čase 09:00 hod. Následně jsou spuštěny vnořené simulace v první úrovni vnoření (prověřující varianty řešení V_1 , V_2 a V_3), pro něž je *primární doba výhledu* nastavena na 30 min. V tabulce 1 jsou demonstrovány přidělované doby časových výhledů pro vybrané replikace vnořených simulací souvisejících s primárním prověřováním řešení konfliktu 3C_1 . Z uvedené tabulky je patrné, že pro replikace na druhé úrovni vnoření se uplatní sekundární doba výhledu 25 min. (při řešení konfliktu 3C_2), resp. 5 min. (při řešení konfliktu 3C_6). Ve třetí úrovni vnoření (při řešení konfliktu 3C_3) replikace počítají s terciárním výhledem 10 min. Všechny replikace vnořených simulací uvedené v tabulce 3 počítají do okamžiku 09:30 hod. simulačního času. Pro řešení konfliktů, které se vyskytly v replikacích na nejnižší hierarchické úrovni (např. 3C_4 a 3C_5), se již vnořené simulace nevyužívají a je použita alternativní metoda podpory rozhodování.

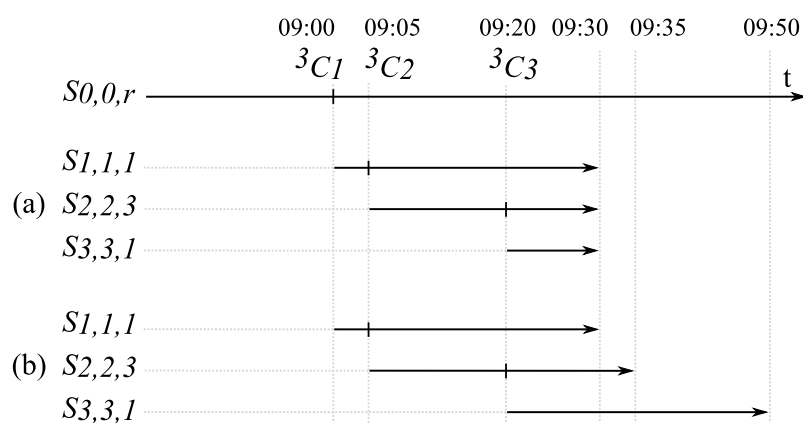
Pro názorné porovnání důsledků výše uvedených strategií jsou vybrané replikace (vycházející z tabulky 1) reprezentující 1., 2. a 3. úroveň vnoření znázorněny na obrázku 5.

Tabulka 1 – Hierarchicky redukované doby výhledů pro vybrané replikace vnořených simulací z obrázku 3, zdroj: vlastní

Replikace	Konflikt	Čas výskytu konfliktu	Délky výhledu	Aplikováno na replikace	Úroveň vnoření
$S_{0,0,r}$	3C_1	09:00	Primární doba výhledu $LD({}^3C_1) = 30$ min.	$S_{1,1,1} S_{1,1,2} S_{1,1,3}$ $S_{1,2,1} S_{1,2,2} S_{1,2,3}$ $S_{1,3,1} S_{1,3,2} S_{1,3,3}$	1
$S_{1,1,1}$	3C_2	09:05	Sekundární doba výhledu $LD({}^3C_2) = 25$ min.	$S_{2,1,1} S_{2,1,2} S_{2,1,3}$ $S_{2,2,1} S_{2,2,2} S_{2,2,3}$	2
$S_{2,2,3}$	3C_3	09:20	Terciální doba výhledu $LD({}^3C_3) = 10$ min.	$S_{3,1,1} S_{3,1,2} S_{3,1,3}$ $S_{3,2,1} S_{3,2,2} S_{3,2,3}$ $S_{3,3,1} S_{3,3,2} S_{3,3,3}$ $S_{3,4,1} S_{3,4,2} S_{3,4,3}$	3
$S_{3,3,1}$	3C_4	09:22	–	–	–
$S_{3,3,1}$	3C_5	09:28	–	–	–
$S_{1,1,1}$	3C_6	09:25	Sekundární doba výhledu $LD({}^3C_6) = 5$ min.	$S_{6,1,1} S_{6,1,2} S_{6,1,3}$ $S_{6,2,1} S_{6,2,2} S_{6,2,3}$ $S_{6,3,1} S_{6,3,2} S_{6,3,3}$	2

Legenda

$LD({}^dC_i)$ – Délky časových výhledů pro replikace související s hledáním řešení konfliktu dC_i



Obrázek 5 – Intervaly simulačních časů vybraných vnořených replikací (dle tabulky 1) při uplatnění strategií: (a) hierarchicky redukované doby výhledů, (b) konstantní doby výhledů, zdroj: vlastní

7 METODIKA PRO ZJIŠŤOVÁNÍ KAPACITY ŽELEZNIČNÍCH STANIC

Cílem dizertační práce bylo navrhnout inovovanou metodiku pro určování kapacity kolejíšť železničních stanic s využitím počítačové simulace. Tato metodika měla rovněž uplatnit metodu vnořených simulací jako techniku pro podporu rozhodování v simulacích. V následujících kapitolách budou detailně představeny základní charakteristiky metodiky, její vstupy a výstupy a přehled jednotlivých fází.

Proces vyhodnocení kapacity je proveden primárně na základě analýzy *přírůstku zpoždění* (vztaženého na jeden vlak) z množiny M provedených hlavních replikací ($M = \{S_{0,0,r} / r = 1 \dots mainReplCount\}$). Pro provedení simulace s využitím *RNS* je nutné stanovit řadu vstupních parametrů (*konfiguraci RNS*). Část těchto parametrů je nutné definovat před vlastním provedením metodiky (*stopCond*, *mainReplCount*), další parametry jsou odvozeny na základě dostupného času na provedení studie kapacity (T_{total}) a dostupných hardwarových výpočetních prostředků pro provádění výpočtů jednotlivých replikací.

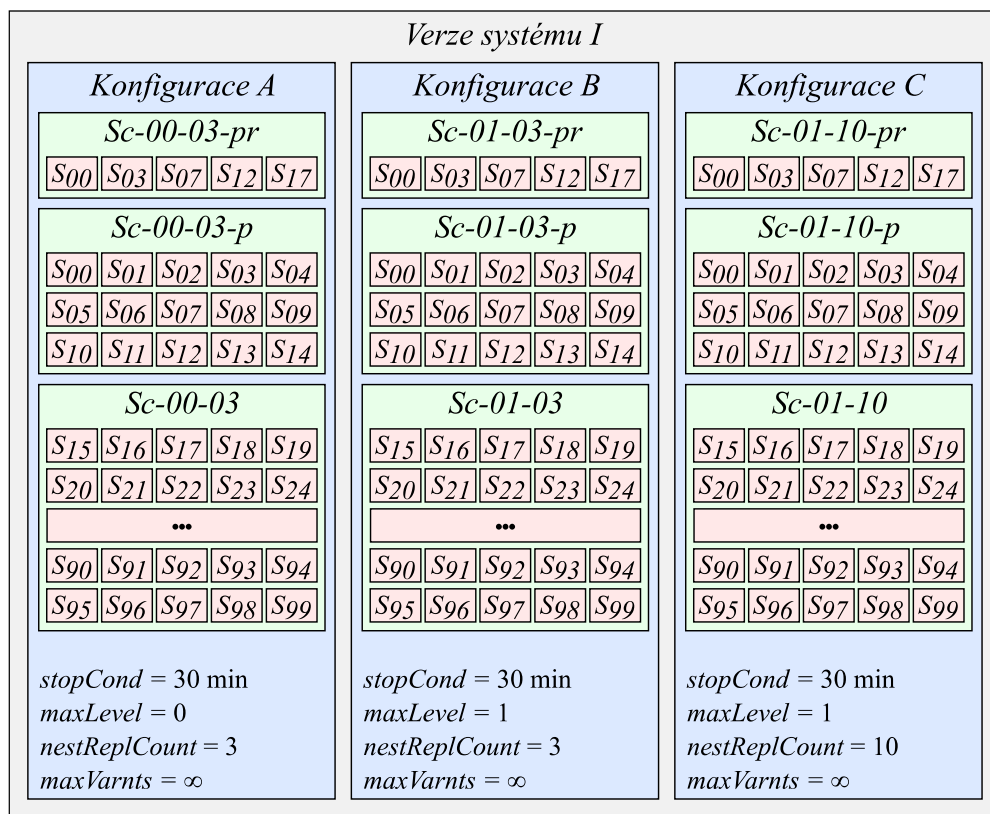
Principem metodiky je uplatnit co možná nejlepší *konfiguraci* podpory rozhodování *RNS* v definovaném časovém limitu vyhrazeném pro provádění studie kapacity. Základní myšlenkou je dosáhnout dobrých rozhodnutí při výskytu konfliktních situací. Identifikovaná řešení konfliktních situací by měla odpovídat řešení, které by bylo použito v realitě (v optimálním případě). Postup uplatněný v rámci metodiky se snaží dosáhnout kvality rozhodování na úrovni reálného dispečera železničního provozu.

Metodika je rozdělena do tří fází, na počátku (*průzkumná fáze*) je provedeno rychlé (hrubé) vyhodnocení výpočetní náročnosti (a hardwarových nároků pro provedení simulací) při různých *konfiguracích RNS*. Ve druhé fázi (*pilotní fáze*) jsou provedeny již podrobnější experimenty, které analyzují kvalitu výsledného řešení a výpočetní náročnost zvolených scénářů. Ve *finální fázi* jsou pak důsledně prověřeny, resp. dopočítány vybrané (vítězné) scénáře za účelem získání statisticky relevantních výsledků ke stanovení kapacity zkoumané žel. stanice.

7.1 Základní termíny a pojmy užívané v rámci metodiky

V následujícím textu je popsána základní terminologie užívaná v rámci metodiky. Pro popis jsou užity termíny *verze systému*, *konfigurace RNS* a *scénář*. Tyto termíny jsou použity k popisu hierarchie, kde na nejvyšší úrovni jsou definovány různé *verze* (zkoumaného) *systému*, v jejich rámci je možné aplikovat různé *konfigurace RNS*, které jsou uplatňovány pro odlišné *scénáře* (tvořící nejnižší úroveň hierarchie). Obecný příklad takové hierarchie je zobrazen na obrázku 6.

Verze systému označuje zkoumaný systém a jeho parametry. Jedná se tedy o konkrétní simulační model a danou provozní variantu se všemi definovanými parametry (zahrnující definice vlaků, tras, zpoždění, jízdní řád, ...). V kontextu jiných publikací je pro *verzi systému* běžně využíván termín *scénář*, ten je ale v rámci metodiky vyhrazen pro jiné použití (definováno níže).



Obrázek 7 – Konkrétní příklad hierarchie scénářů a jejich hlavních replikací, zdroj: vlastní

Značení scénářů

Pro značení scénářů je použit jednotný formát (tento formát je využit i dále v metodice a případové studii), který vychází ze schématu *Sc-VV-AA-BB-CC-DD-suffix*, kde jednotlivé složky jsou:

- *VV* – varianta verze systému,
- *AA* – hodnota parametru *stopCond*,
- *BB* – hodnota parametru *maxLevel*,
- *CC* – hodnota parametru *nestReplCount*,
- *DD* – hodnota parametru *maxVarnts*,
- *suffix* – definuje příponu „pr“ pro scénář užitý v průzkumné fázi, příponu „p“ pro scénář pilotní fáze a prázdnou příponu, pokud je scénář určen pro finální fázi.

V uvedeném příkladu by složky *VV*, *AA* a *DD* byly pro všechny scénáře totožné, a proto nebyly pro značení scénářů zahrnuty. Formát užitý v obrázku 7 tedy odpovídá schématu *Sc-BB-CC-suffix*. V následujícím textu je v příkladech užíváno i zjednodušené značení scénářů ve formě *ScX*, kde *X* představuje prosté číslování uvažovaných scénářů (značení je tedy bez uvádění detailů týkajících se konfigurace RNS).

7.2 Základní charakteristiky metodiky

Navrhovaná metodika představuje obecně uplatnitelnou metodiku pro provedení zjišťování kapacity železničních stanic. Základním předpokladem je dostupnost vhodného simulačního

nástroje, který je schopen provádět deterministické a stochastické simulace železničního provozu. Pro řešení konfliktních stavů vzniklých v průběhu stochastických simulací se předpokládá, že simulátor disponuje automatizovanou podporou rozhodování s využitím reflektivních vnořených simulací (s možností nastavení základních parametrů popsanych v kapitole 6).

Metodika popisuje postup, který lze uplatnit na jednu provozní variantu simulační studie (*verzi systému*). V případě, že je prováděno hromadné zkoumání více různých *verzí systému*, je potřeba metodiku celou zopakovat. V určitých případech je možné očekávat, že prvotní *průzkumná fáze* metodiky nebude muset být vykonána opakovaně. Tyto případy jsou blíže představeny v kapitole 7.4.

Metodika se nezabývá činnostmi souvisejícími s vybudováním, verifikací a validací simulačního modelu zkoumané železniční stanice. Metodika dále nedefinuje ani neupravuje, jaké parametry musí mít konkrétní zkoumaná provozní situace (jízdni řády vlaků a související parametry). V rámci metodiky jsou definovány základní vstupy, které jsou potřeba k samotnému provedení procesu zkoumání kapacity.

7.2.1 Vstupy metodiky

Základním výše uvedeným předpokladem pro použití metodiky je dostupnost vhodného simulačního nástroje. Pro kompletní použití metodiky je potřeba (a) vytvořit simulační model, (b) nastavit základní parametry souvisejících se simulační studií, (c) specifikovat hardwarové prostředky pro provádění simulací.

Simulační model

Pro samotné provádění simulační studie je nezbytné vytvořit simulační model. Ten by měl v rámci požadované úrovně podrobnosti simulace zahrnovat zásadní fakty popisující topologii kolejí železniční stanice a jejího nejbližšího okolí (např. zahrnující přilehlé traťové úseky v rozsahu do sousední železniční stanice či konce příslušné trati). Pro simulátory pracující na mikroskopické úrovni podrobnosti jsou obvykle nezbytné detailní informace o jednotlivých kolejových úsecích a jejich fyzikálním popisu – délka, sklon, obloukové poměry a maximální povolená rychlost. Dále jsou obvykle vyžadovány informace o elektrizaci trati, podrobnosti o použitém zabezpečovacím zařízení a jeho jednotlivých zařízeních včetně přesných poloh v kolejišti železniční stanice.

Pro popis provozní situace je typicky vyžadováno definování jízdniho řádu včetně detailů o jednotlivých vlacích (složení vlakových souprav, fyzikální informace týkající se dynamiky jízdy, čas vstupu do simulace a další). Dále je nutné definovat parametry ovlivňující stochastické jevy v simulaci – definování náhodných proměnných popisujících vstupní zpoždění vlaků a zpoždění generované uvnitř systému (v železniční stanici), případně náhodné proměnné popisující výskyt dalších stochastických jevů (výskyt neočekávaného vlaku, porucha zařízení, aj.). Pro provádění stochastických simulací se předpokládá, že simulační model obsahuje variantní vlakové cesty, které může automatizovaná podpora rozhodování využít při výskytu konfliktu. Pro podporu rozhodování mohou být připraveny i další varianty řešení

konfliktů – pozdržení vlaku za účelem předjetí jiným vlakem, čekání na původně plánovanou kolej, aj.

Z obecných parametrů je nutné definovat, jaké časové období je simulováno a analyzováno. Je také možné definovat náběhovou a výběhovou fázi simulace a specifikovat dobu trvání těchto fází. Samotný simulační model by měl projít standardním procesem verifikace a validace, aby bylo následně možné získané výsledky považovat za věrohodné. Simulační model a příslušná prověřovaná provozní varianta je dále označována jako *verze systému*.

Základní parametry simulační studie

Základním omezujícím faktorem při provádění simulačních studií je časové období, které je možné dané studii věnovat. Vzhledem k vyšší náročnosti techniky *RNS* je nutné tento fakt zohlednit. Metodika předpokládá rozložení času, který je k dispozici, do několika různě časově náročných fází. Celkový časový limit, který je k dispozici pro provedení kapacitní simulační studie (při uplatnění celé metodiky na vybranou *verzi systému*) je označován jako T_{total} .

Dále je nutné definovat několik základních parametrů techniky *RNS*. Je nutné zvolit vhodnou hodnotu parametru *mainReplCount*, který určuje počet prováděných replikací hlavní simulace. Pokud požadavek simulační studie nedefinuje počet replikací hlavní simulace, ale definuje limitní hodnotu pro šířku intervalu spolehlivosti sledovaných ukazatelů, pak je nezbytné provést dodatečný experiment (např. s využitím alternativní podpory rozhodování), při kterém bude analyzován vliv počtu replikací hlavní simulace na šířku intervalu spolehlivosti a bude proveden odhad potřebného počtu replikací. Při provádění metodiky se může stát, že některé specifické replikace nebude možné dopočítat z důvodu příliš vysoké výpočetní náročnosti (nedostatečná kapacita paměti na LPU nebo nedostatek času na výpočet), proto je žádoucí předpokládat, že hodnota parametru *mainReplCount* připouští vynechání několika replikací a volit hodnotu tohoto parametru např. o 10 % vyšší, než by byla standardně požadována.

Délka výhledu vnořených simulací (definovaná parametrem *stopCond*) je dalším nezbytným vstupem metodiky. Délka výhledu výrazným způsobem ovlivňuje kvalitu optimalizace, kterou poskytuje podpora rozhodování, ale také výrazně prodlužuje dobu, která je potřeba na provedení jedné replikace simulace. Volba parametru by měla reflektovat realitu a neměla by tedy být extrémně dlouhá. Velmi dlouhý výhled by pak mohl vést k optimističtějším výsledkům, než jakých je možné dosáhnout v realitě, velmi krátká doba výhledu by pak nebyla schopna efektivně řešit vzniklé konfliktní stavy. Hodnota délky výhledu by rovněž měla být uvažována v kontextu s délkou simulovaného období.

Z výše uvedeného textu vyplývá, že je potřeba stanovit hodnoty parametrů:

- T_{total} – celkový časový limit pro uskutečnění simulační studie,
- *mainReplCount* – parametr *RNS* – počet prováděných replikací hlavní simulace,
- *stopCond* – parametr *RNS* – délka výhledu vnořených simulací.

Hardwarové prostředky pro provedení simulací

Provádění simulací s využitím *RNS* může představovat výpočetně náročný proces. Za účelem efektivního zpracování velkého množství takových simulací metodika předpokládá, že

simulační studie může být vykonávána s využitím paralelizace výpočtů v prostředí výpočetního clusteru (či výpočetního gridu), který bude po provedení *průzkumné fáze* rozdělen na dílčí logické výpočetní jednotky (*LPU – Logical Processing Unit*). Výpočty je pak možné paralelizovat a v rámci každé výpočetní jednotky provádět jednu samostatnou replikaci hlavní simulace. Metodiku lze uplatnit i v rámci jediného počítače (výpočetního uzlu), který může (a nemusí) být následně rozdělen na více logických výpočetních jednotek (v závislosti na dostupných hardwarových prostředcích). Primárním důvodem využívání výpočetního clusteru o N logických výpočetních jednotkách je (téměř) N -násobné zrychlení provádění *pilotní* a *finální fáze* metodiky (oproti případu, kdyby nebyly tyto výpočetní jednotky využívány).

Pro potřeby metodiky je tak potřeba znát hardwarové vybavení všech výpočetních uzlů, které jsou k dispozici pro simulace. Jedná se zejména o znalost parametrů:

- počet fyzických výpočetních uzlů (počítačů),
- konfigurace CPU (počet fyzických a logických vláken),
- velikost operační paměti RAM.

Diskové úložiště na jednotlivých uzlech není obvykle nezbytné a lze primárně využívat různé formy síťových (centrálních) úložišť dat.

Pro provedení metodiky (*pilotní* a *finální fáze*) se předpokládá, že simulační nástroj disponuje funkcí automatického plánování provádění jednotlivých hlavních replikací na jednotlivých *LPU* s využitím plánovací metody *FCFS (First Come First Served)*. Alternativně je možnost využít externí plánovač, který je schopen dosáhnout téhož výsledku plánováním spouštění vybraných replikací v simulátoru na jednotlivé *LPU*. V obou případech se předpokládá možnost na počátku manuálně stanovit množinu replikací (a příslušných *scénářů*), která bude automatizovaně počítána (bez nutnosti dalších zásahů uživatele).

7.2.2 Výstupy metodiky

Cílem metodiky je poskytnout informace o jednom kapacitním ukazateli spojeném s provozem zkoumané železniční stanice. Tento ukazatel, který je zároveň základním výstupním ukazatelem metodiky je: *přírůstek zpoždění vztahený na jeden vlak*. Metodika dále interně využívá ukazatel *suma vážených přírůstků zpoždění (SWDI – Sum of Weighted Delay Increments)* za účelem hodnocení výsledků jednotlivých simulačních replikací.

Přírůstek zpoždění na jeden vlak

Ukazatel *přírůstek zpoždění na jeden vlak* (označovaný jako Δd) představuje základní ukazatel sloužící pro vyhodnocení kapacity zkoumané železniční stanice. Hodnota přírůstku zpoždění je vyjádřena jako rozdíl výstupního a vstupního zpoždění příslušného vlaku. Zjištěné hodnoty lze následně porovnat se stanovenými limitními hodnotami (dle směrnice ŠZDC (2019)) a zhodnotit tak vztah využití infrastruktury k předpokládané kvalitě provozu.

Suma vážených přírůstků zpoždění (SWDI)

Ukazatel *SWDI* slouží jako doplňující ukazatel výsledku jedné simulační replikace řídicí se vybraným *scénářem*. Hodnota přírůstku zpoždění pro každý vlak je změřena standardním

způsobem – jako rozdíl výstupního a vstupního zpoždění. Jednotlivé vlaky mají dále stanovenou váhu, která odráží jejich důležitost v dopravním provozu. Hodnotu váhy je doporučeno stanovit v souladu se směrnicí SŽDC (2019). Hodnota $SWDI$ je pak vypočítána jako součin hodnoty přírůstku zpoždění a váhy příslušného vlaku. Na základě ukazatele $SWDI$ je dále stanoven ukazatel $meanSWDI$.

Charakter provozu, který odráží určitý scénář ScX , lze vyjádřit pomocí statistického indikátoru $meanSWDI$. Jeho hodnota se vypočítá jako aritmetický průměr z $SWDI$ -hodnot replikací příslušejících scénáři ScX a v rámci simulačního výpočtu je získána prostřednictvím funkce $CrOptim$. Pro uvedený průměr se počítá i 95% interval spolehlivosti. Pološířku tohoto intervalu lze vyjadřovat buď absolutní hodnotou (označovanou jako $halfWidth$) anebo relativní hodnotou (označovanou jako $relatHalfWidth$). Relativní hodnota vyjadřuje poměr absolutní hodnoty pološířky daného intervalu a hodnoty korespondujícího průměru:

$$relatHalfWidth = \frac{halfWidth}{meanSWDI} \quad (1)$$

7.3 Přehled jednotlivých fází metodiky

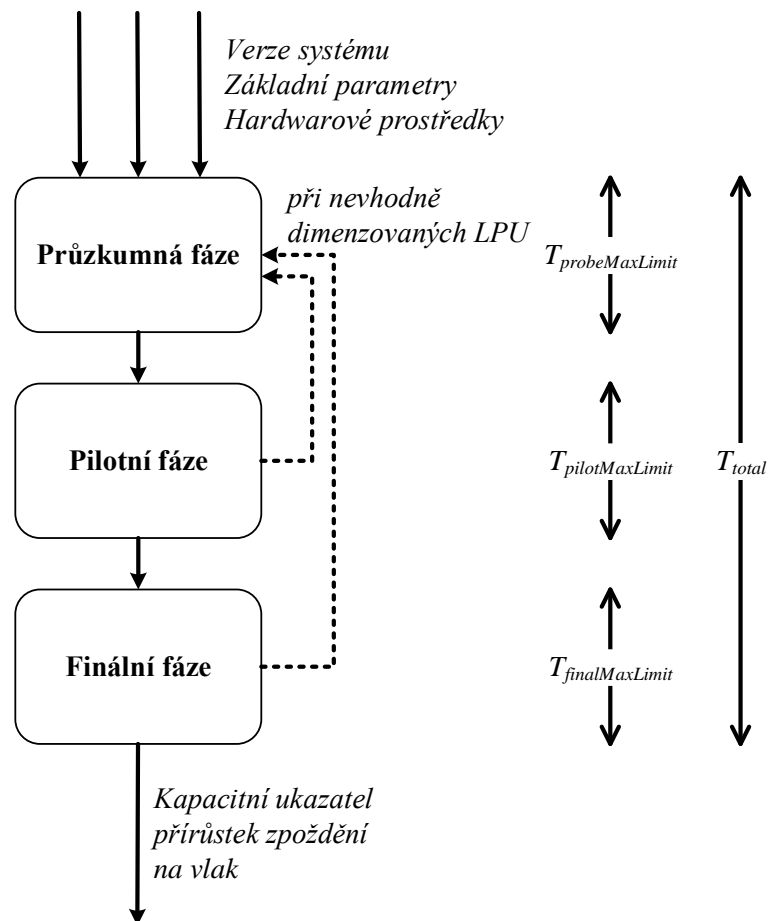
Metodika se skládá ze tří základních fází, které na sebe sekvenčně navazují. Jedná se o fáze: (a) průzkumnou, (b) pilotní a (c) finální. Na počátku simulační studie v průzkumné fázi je provedena několik základních experimentů (s různými konfiguracemi) pro zjištění náročnosti RNS v dané verzi systému. Na základě tohoto zkoumání je navrženo rozdělení výpočetního clusteru na logické výpočetní jednotky. V průběhu pilotní fáze je simulováno větší množství různých scénářů, ale omezené na menší počty replikací hlavních simulací. Na základě těchto výsledků jsou navrženy scénáře, které jsou kandidáty na kompletní výpočet v rámci finální fáze. Jednotlivé fáze a možné přechody mezi nimi jsou znázorněny na obrázku 8.

V rámci metodiky se předpokládá sekvenční průchod jednotlivými fázemi. V průběhu pilotní a finální fáze se mohou potenciálně vyskytnout situace, které nebyly na základě původně provedené průzkumné fáze očekávány a mohou být identifikovány konkrétní replikace, jejichž výpočetní náročnost výrazně převyšuje očekávání. V důsledku toho může být výpočet těchto replikací předčasně ukončen a jejich výsledek nebude dále využit.

Pokud množství neproveditelných replikací výrazně vzroste a nebude možné dokončit značnou část předpokládaných replikací, pak je nutné provést návrat do průzkumné fáze a proces zkoumání zopakovat. Zde je nutné reflektovat, že je potřebné snížit hodnotu parametru T_{total} o dosud spotřebované množství času.

7.4 Průzkumná fáze

V rámci první tzv. průzkumné fáze (*probing phase*) je proveden malý počet odlišně parametrizovaných (různých konfigurací) simulačních experimentů (s malým počtem replikací hlavních simulací) s cílem získat základní představu o výpočetní a paměťové náročnosti budoucích komplexních výpočtů. Ve zmíněných experimentech je používána definovaná doba výhledu (případně i více odlišných dob výhledů) pro RNS . Pro účely této fáze jsou stanoveny maximální časové limity jednak pro dobu výpočtu jedné replikace hlavní simulace ($T_{probeReplMaxLimit}$), a jednak pro celkovou dobu trvání této fáze ($T_{probeMaxLimit}$).



Obrázek 8 – Přehled fází metodiky pro jednu verzi systému, zdroj: vlastní

Průzkum je podkladem pro návrh koncepce technického řešení paralelizace výpočtů na dostupném výpočetním zařízení. Zmíněný návrh zohledňuje požadavky jednak na co možná nejvyšší míru paralelizace výpočtů simulačních experimentů a jednak na poskytnutí dostatečné paměťové kapacity pro tyto výpočty. Pokud je to možné, uvedené technické řešení je následně uplatňováno v rámci celé experimentální etapy (tzn. při simulačním zkoumání všech prověřovaných scénářů).

Pokud se v dalších fázích vyskytne výrazně vysoké množství replikací, které není možné dokončit z důvodů nevhodně dimenzovaných logických výpočetních jednotek, je možné se vrátit zpět do *průzkumné fáze* a provést doplňující zkoumání. V takovém případě je nutné počítat s již spotřebovaným časem při provádění kapacitní studie a přizpůsobit tomu i následující fáze.

7.4.1 Vstupy průzkumné fáze

Před provedením *průzkumné fáze* je nutné definovat časové limity pro provádění experimentů a samotné fáze:

- $T_{probeReplMaxLimit}$ – časový limit pro provádění jedné replikace hlavní simulace, při překročení času je replikace okamžitě ukončena bez čekání na dokončení výpočtu (výsledek ukončené replikace je neplatný a nelze jej užít do dalšího vyhodnocování),

- $T_{probeMaxLimit}$ – časový limit definující celkovou dobu trvání *průzkumné fáze*.

U obou zmíněných časových limitů platí, že by měly nabývat hodnoty výrazně menší, než je T_{total} , pro provádění průzkumné fáze by mělo být vyhrazeno minimální množství času. Fáze by měla typicky představovat maximálně 10 % z celkového času na provedení studie.

V rámci průzkumné fáze je nutné získat výsledky z různých *konfigurací RNS* a různých hlavních replikací. Před vlastním provedením je nutné definovat zkoumané *konfigurace*. Mezi základní doporučené parametry a jejich příslušné nastavení (k prozkoumání) patří:

- *maxLevel* – povolený počet rekurzivních úrovní výpočtů, parametr výrazně ovlivňuje kvalitu výsledného řešení a výpočetní náročnost, průzkum by měl otestovat případ bez uplatnění rekurze, ale také případy s několika úrovněmi rekurze. Výpočetní složitost experimentů roste s parametrem exponenciálně, výrazně také záleží na délce výhledu (*stopCond*).
- *nestReplCount* – počet replikací pro vnořené simulace ovlivňuje kvalitu výsledku a také představuje základ pro statistické vyhodnocení výsledků. Výpočetní složitost se zvyšováním parametru roste. V rámci *průzkumné fáze* je doporučeno prozkoumat na počátku zejména malé počty vnořených replikací (jednotky replikací) a dále zkoumání rozšiřovat podle dosažených výsledků.
- *maxVarnts* – počet prozkoumaných variant se může vyplatit omezit, pokud existuje ve zkoumaném simulačním modelu mnoho alternativních variant řešení konfliktů. Omezením průzkumu jen na část možných variant je snížena kvalita výsledného řešení, pokud to situace dovolí je doporučeno neomezovat počet prozkoumávaných variant. Zkoumané *konfigurace* mohou zahrnovat omezení na jednotky až desítky variant a také *konfiguraci*, kde počet variant není omezen.

Pro jednotlivé *konfigurace RNS* je následně nutné vypočítat několik replikací hlavní simulace (*probeMainReplCount*), tyto replikace by rovněž měly v ideálním případě představovat reprezentativní vzorek různých (stochastických) variant situací, které mohou v rámci provádění studie a jejích replikací nastat. Podobně jako v případě časového limitu pro provedení fáze platí $probeMainReplCount \ll mainReplCount$.

7.4.2 Provedení průzkumné fáze

Po stanovení základních vstupů fáze je možné začít provádět vybrané experimenty. V rámci fáze je vhodné uplatnit dynamický iterativní postup a další zkoumané *konfigurace* (a jejich replikace) přizpůsobit dosud dosaženým výsledkům. Alternativně je možné rovněž na počátku stanovit statický seznam zkoumaných *konfigurací* a jejich hlavních replikací (v zásadě se jedná o *scénáře průzkumné fáze*), ale tento přístup se jeví jako méně efektivní a poskytuje méně informací pro další posuzování zkoumaného systému.

Při uplatnění dynamického iterativního způsobu průzkumu je možné pružně reagovat na průběžně dosažené výsledky. Pokud by na počátku průzkumu došlo ke stanovení finální neměnné množiny *scénářů*, které budou zkoumány, mohlo by dojít k plýtvání času s *konfiguracemi*, které nerozšiřují poznatky o chování zkoumaného systému. Takové zkoumání by rovněž bylo zbytečně časově náročné, např. pro prozkoumání všech možných kombinací při

stanovených množinách hodnot příslušných parametrů $M_{maxLevel} = \{0, 1, 2\}$, $M_{nestReplCount} = \{2, 3, 5, 10\}$ a $M_{maxVarmts} = \{3, 5, 10, 20, \infty\}$ by bylo nutné provést $3 \times 4 \times 5 \times probeMainReplCount$ simulací, ačkoliv se může ukázat, že některé konfigurace jsou v praxi neproveditelné nebo jejich výsledky nepřinášejí nové informace.

Při provádění *průzkumné fáze* může docházet k výskytům výpočetně extrémně náročných replikací (např. u replikací s vyššími hodnotami parametru *maxLevel*), které se výrazně odchyľují od standardního chování systému. Takové replikace nemusí být ve stanovených časových limitech dopočitatelné.

Po provedení *průzkumné fáze* je nutné vyhodnotit zejména časovou náročnost (vyjádřené potřebnými uplynulými dobami reálného času) a paměťovou náročnost (maximální špičkové využití operační paměti v průběhu simulace) jednotlivých konfigurací RNS. Tyto informace slouží k ustanovení logických výpočetních jednotek, a rovněž poskytují základní informaci ke stanovení scénářů, které budou dále zkoumány v průběhu *pilotní fáze*.

7.4.3 Výstupy průzkumné fáze

Na základě výsledků z *průzkumné fáze* je možné stanovit rozdělení dostupných hardwarových prostředků na logické výpočetní jednotky, které následně umožní paralelizaci výpočtů. Při definování logických výpočetních jednotek je nutné zvážit dostupný výpočetní výkon na jednu jednotku a alokovanou operační paměť na jednu jednotku. Výstupy fáze jsou:

- M_{LPURam} – množství operační paměti RAM na jednu logickou výpočetní jednotku,
- N_{LPU} – počet logických výpočetních jednotek.

Během provádění *pilotní* a *finální fáze* metodiky je možné na každou výpočetní jednotku alokovat jednu hlavní replikaci ze zkoumaných scénářů. Výpočty na jednotlivých LPU lze provádět paralelně. Pokud daná výpočetní jednotka disponuje více fyzickými nebo logickými výpočetními vlákny, mohou být dále využity pro paralelizaci výpočtů vnořených replikací při výskytu konfliktních situací a uplatnění RNS podpory rozhodování. Pokud bude výpočetní výkon (počet dostupných vláken) logické jednotky malý, dojde ke zvýšení potřebného času na provedení replikace. Pokud však bude alokováno nedostatečné množství operační paměti a prováděná replikace bude vyžadovat další paměť (např. v důsledku dalšího vzniklého konfliktu a vzniku nových vnořených simulací), dojde k předčasnému ukončení simulace a danou replikaci nebude možné dopočítat a vyhodnotit. Alokované množství operační paměti tak představuje tvrdý limit, který po jeho stanovení v zásadě definuje „maximální složitost“ prováděné úlohy. Nastavení alokované operační paměti, tak musí brát ohled na předpokládané konfigurace, které budou v rámci dalších fází simulovány.

Pokud například bude v průběhu experimentů zaznamenáno špičkové využití operační paměti 21 GB a k dispozici budou výpočetní uzly, které disponují pamětí o velikosti 128 GB, pak je možné uvažovat, že reálné špičkové využití paměti může dosáhnout 32 GB (je uplatněn pesimistický odhad) a rozdělit každý výpočetní uzel na 4 logické výpočetní jednotky ($M_{LPURam} = 32$ GB, $N_{LPU} = 4$).

Výsledky získané v *průzkumné fázi* jsou rovněž uplatnitelné pro stanovení, které *scénáře* budou dále zkoumány v *pilotní fázi*.

7.5 Pilotní fáze

Pilotní fáze (pilot phase) se soustředí na provedení zlomkového počtu replikací hlavních simulací (*pilotMainReplCount*) všech *scénářů*, které jsou naplánovány k realizaci. Po zjištění náročnosti těchto pilotních experimentů jsou spočítány příslušné časové odhady pro dokončení výpočtů každého ze scénářů vzhledem k celkovému plánovanému počtu hlavních replikací (*mainReplCount*). Na základě časových odhadů a dosažených výsledků u jednotlivých scénářů jsou vybrány perspektivní scénáře, které budou dopočítány v rámci *finální fáze*.

Z důvodu eliminace potenciálních dlouhých výpočetních dob (reálného času) některých experimentů může být stanovena příslušná horní hranice (specifikovaná hodnotou parametru $T_{pilotReplMaxLimit}$) pro dobu výpočtu jedné hlavní replikace v rámci *pilotní fáze*. Dosáhne-li doba trvání výpočtu hodnotu nastavenou parametrem $T_{pilotReplMaxLimit}$, je daný výpočet předčasně ukončen a je vyhodnocen jako neperspektivní pro další dopočítání v následující fázi. Pro celkové trvání fáze je definován časový limit $T_{pilotMaxLimit}$.

7.5.1 Vstupy pilotní fáze

Před provedením *pilotní fáze* je nutné definovat časové limity pro provádění experimentů a samotné fáze:

- $T_{pilotReplMaxLimit}$ – časový limit pro provádění jedné replikace hlavní simulace, při překročení času je replikace okamžitě ukončena bez čekání na dokončení výpočtu (výsledek ukončené replikace je neplatný a nelze jej užít do dalšího vyhodnocování),
- $T_{pilotMaxLimit}$ – časový limit definující maximální celkovou dobu trvání *pilotní fáze*.

Hodnota parametru *pilotMainReplCount*, by měla představovat zlomek ze stanovené hodnoty parametru *mainReplCount*, tak aby bylo během *pilotní fáze* možné prověřit dostatečné množství různých *konfigurací RNS*. Při nastavení parametru *pilotMainReplCount* je možné také vycházet z počtu logických výpočetních jednotek a optimalizovat rozložení výpočtů na jednotlivé výpočetní jednotky (hodnota *pilotMainReplCount* může být zvolena jako násobek hodnoty N_{LPU}).

Dále je nutné definovat množinu *scénářů*, které budou v rámci *pilotní fáze* prověřovány:

- $ScSet_{pilot}$ – množina pilotních *scénářů*.

Pro stanovení množiny pilotních *scénářů* je doporučeno využít informace o zjištěných výsledcích a výpočetní náročnosti *konfigurací* v průběhu *průzkumné fáze*.

7.5.2 Provedení pilotní fáze

Po stanovení vstupních parametrů je možné přistoupit k samotným simulačním experimentům *pilotní fáze*. Po provedení průzkumné fáze byly definovány logické výpočetní jednotky, které jsou dále využity v rámci pilotní a finální fáze. Každá jedna *LPU* umožňuje provádět simulaci jedné hlavní replikace vybraného *scénáře*. V jednom časovém okamžiku je tak možné

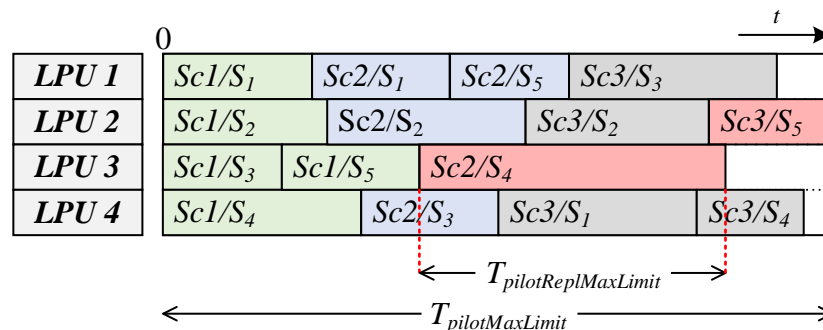
simulovat až N_{LPU} hlavních replikací. Proces přidělování LPU k replikacím je prováděn metodou *FCFS* (*First Come First Served*). *Scénáře* jsou seřazeny podle předpokládané výpočetní náročnosti od nejnižší po nejvyšší. Replikace v rámci jednoho *scénáře* jsou seřazeny vzestupně podle identifikátoru replikace (číselného pořadí či výchozí násady hlavního generátoru pseudonáhodných čísel).

Příklad provádění simulací při definovaných vstupních parametrech $N_{LPU} = 4$, $ScSet_{pilot} = \{Sc1, Sc2, Sc3\}$, $pilotMainReplCount = 5$ je uveden na obrázku 9. První čtyři replikace scénáře $Sc1$ postupně obsazují všechny LPU ve stanoveném pořadí. Replikace $Sc1/S_5$ je spuštěna až ve chvíli, kdy dojde k uvolnění některé z LPU . V uvedeném příkladu se tak nejdříve stane u $LPU 3$. Po vyčerpání všech replikací ze *scénáře* $Sc1$ jsou spouštěny replikace ze scénáře $Sc2$ a na závěr jsou prováděny replikace ze *scénáře* $Sc3$.

Na uvedeném obrázku je znázorněna rovněž situace, ve které byla replikace ukončena z důvodu vypršení časového limitu $T_{pilotReplMaxLimit}$, tato situace nastala u replikace $Sc2/S_4$. Předčasné ukončení také nastalo u replikace $Sc3/S_5$, která nemohla být dopočítána kvůli vypršení limitu $T_{pilotMaxLimit}$. Kromě těchto dvou replikací byly všechny ostatní naplánované replikace úspěšně dopočítány. Z obrázku dále vyplývá, že provedením specifikovaných scénářů byl prakticky vyčerpán stanovený časový limit $T_{pilotMaxLimit}$. Na některých LPU nebyl časový limit plně vyčerpán a mohl potenciálně být dále využit k simulaci dalších replikací. Pokud by všechny prováděné replikace byly dokončeny dříve než po uplynutí limitu $T_{pilotMaxLimit}$, pak je možné zbývající čas převést do *finální fáze* a o ušetřené časové prostředky navýšit hodnotu $T_{finalMaxLimit}$.

Při provádění pilotní fáze je rovněž možné uplatnit dynamický přístup k množině $ScSet_{pilot}$. Pokud jsou dokončeny všechny stanovené výpočty a není spotřebován veškerý vyhrazený čas ($T_{pilotMaxLimit}$), je možné do množiny doplnit další scénáře podle potřeby. Alternativně je možné zvýšit počet zkoumaných replikací $pilotMainReplCount$ a dopočítat tyto replikace za účelem zpřesnění odhadů výsledků.

Pro stanovení, v jakém pořadí je vhodné provádět jednotlivé scénáře je možné vycházet z informací zjištěných v průběhu *průzkumné fáze* a provádět *scénáře* v pořadí dle jejich očekávané výpočetní náročnosti.



Obrázek 9 – Ukázka uplatnění techniky *FCFS* při přidělování jednotlivých replikací na jednotky LPU , zdroj: vlastní

Po dokončení všech výpočtů je nutné provést statistické zpracování výsledků. Pro každý scénář jsou vypočítány následující ukazatele:

- $^{estim}T_{remainRepls}$ – hrubý odhad potřebného času pro dopočítání daného počtu replikací (*remainRepls*) při dopočtu scénáře v rámci *finální fáze* metodiky,
- *meanSWDI* – střední hodnota ukazatele *SWDI*.

Hrubý odhad potřebné doby pro dopočet scénáře

Do *finální fáze* metodiky je nutné vybrat scénář (či scénáře), které nejen poskytují dobré výsledky z pohledu podpory rozhodování, ale také je nutné přihlídnout k časovému limitu, který stanovuje maximální časový interval pro provedení *finální fáze* ($T_{finalMaxLimit}$). Hodnota $^{estim}T_{remainRepls}$ se snaží statisticky odhadnout, jak dlouho by trvalo dopočítání zbývajících replikací (*remainRepls*), které by byly simulovány v průběhu *finální fáze*, pokud by byl scénář vybrán pro *finální dopočet*.

$$remainRepls = mainReplCount - pilotReplCount \quad (2)$$

Pro stanovení časového odhadu (pro jeden scénář) se využívá extrapolace naměřených dat. Pro samotný výpočet je potřebné stanovit hodnoty:

- ^{cost}t – režijní čas (na jednu replikaci), který potřebuje plánovač v souvislosti se (i) startováním nového procesu, resp. nové replikace, (ii) sběrem výsledků procesu, (iii) ukončením procesu a (iv) prováděním samotného úkonu plánování,
- $^{upper}t_{pilotReplCount}$ – horní hranice 95% intervalu spolehlivosti průměrné doby trvání jedné hlavní replikace z množiny replikací příslušného scénáře,
- N_{LPU} – počet logických výpočetních jednotek,
- *remainRepls* – počet replikací hlavní simulace daného scénáře, které by byly počítány v průběhu *finální fáze*.

Režijní čas ^{cost}t je možné zjistit samostatně provedeným experimentem, při kterém bude celkový režijní čas změřen. Lze také vycházet ze zkušeností s daným systémem, který je užit pro plánování procesů na jednotlivé *LPU*. V případě malého až středně velkého výpočetního clusteru (řádově desítky výpočetních uzlů), který je propojen prostřednictvím standardní počítačové sítě (typu ethernet), je možné uvažovat hrubý paušální čas $^{cost}t = 1$ minuta, pokud není známa přesnější hodnota nebo není možné provést experimentální vyhodnocení režijního času.

Pro odhad doby trvání jedné replikace (^{repl}t) není využita střední hodnota doby trvání jedné replikace ($^{mean}t_{pilotReplCount}$), ale je použita bodová hodnota horní hranice 95% intervalu spolehlivosti ($^{upper}t_{pilotReplCount}$). Důvodem pro uplatnění této hodnoty je záměr aplikovat pesimistický odhad trvání jedné replikace. S ohledem na fakt, že na provedení *finální fáze* metodiky je vyhrazeno omezené množství času, je lepší předpokládat, že dopočítání vybraného scénáře bude trvat delší dobu, než očekávat kratší dobu a následně nestihnout dopočet v definovaném časovém limitu.

$$repl_t = upper_t_{pilotReplCount} + cost_t \quad (3)$$

$$estimT_{remainRepls} = \frac{remainRepls \times repl_t}{N_{LPU}} + repl_t \quad (4)$$

Výpočet statistického indikátoru *meanSWDI*

Výpočet hodnoty *meanSWDI* se provádí za účelem vyhodnocení kvality podpory rozhodování, tedy pro nalezení *scénáře*, který uplatnil nejlepší rozhodnutí ohledně řešení konfliktních situací (došlo k výběru variant řešení konfliktů spojených s minimálními hodnotami *SWDI*). Výpočet hodnoty *meanSWDI* je prováděn dle popisu uvedeného v kapitole 7.2.2.

7.5.3 Výstupy pilotní fáze

Hlavním výstupem *pilotní fáze* je množina *scénářů*, které jsou určeny pro dopočítání v závěrečné (*finální*) fázi metodiky:

- $ScSet_{final}$ – množina scénářů pro finální dopočítání.

Při stanovení množiny *scénářů* vybraných pro dopočet ve *finální fázi* je nutné vycházet ze znalosti hodnot *meanSWDI* a hrubých časových odhadů pro dopočet $estimT_{remainRepls}$. U hodnoty *meanSWDI* je cílem použít *scénáře*, kde je hodnota nejmenší. Vybraná množina scénářů musí splňovat podmínku, že součet odhadů časů potřebných k dopočítání je menší nebo roven času vyhrazenému pro provádění *finální fáze*:

$$T_{finalMaxLimit} \geq \sum_{i \in ScSet_{final}} (estimT_{remainRepls}^i) \quad (5)$$

7.6 Finální fáze

Ve *finální fázi* (*final phase*) jsou na základě vyhodnocení odhadů z předešlé fáze dopočítány vybrané *scénáře*. Výsledky finálně dokončených *scénářů* jsou podkladem pro provozní vyhodnocení kapacity prověřované *verze zkoumaného systému*.

Pokud by pro některé *verze zkoumaného systému* nevyhovovalo původně navržené technické řešení paralelizace výpočtů (například z důvodu nedostatečné kapacity paměti), potom je potřebné znovu zopakovat průzkumné experimenty a na základě jejich výsledků navrhnout jiný přístup pro realizaci paralelních výpočtů.

7.6.1 Vstupy finální fáze

Před provedením *finální fáze* je nutné definovat časové limity pro provádění experimentů a samotné fáze:

- $T_{finalReplMaxLimit}$ – časový limit pro provádění jedné replikace hlavní simulace, při překročení času je replikace okamžitě ukončena bez čekání na dokončení výpočtu (výsledek ukončené replikace je neplatný a nelze jej užít do dalšího vyhodnocování),
- $T_{finalMaxLimit}$ – časový limit definující maximální celkovou dobu trvání *finální fáze*.

Kromě uvedených časových limitů je vstupem *finální fáze* také množina *scénářů* určených k dopočítání:

- $ScSet_{final}$ – množina *scénářů* určených k finálnímu dopočítání.

Množina *scénářů* pro finální dopočítání je převzata z výsledků předchozí fáze metodiky.

7.6.2 Provedení finální fáze

Proces provedení *finální fáze* je shodný s procesem uplatněným v rámci *pilotní fáze* (viz kapitola 7.5.2). Změny oproti postupu uplatněnému v předchozí fázi jsou: (i) použití finální množiny *scénářů* ($ScSet_{final}$) a (ii) použití odlišné množiny hlavních replikací (počet dopočítávaných replikací je vyjádřen hodnotou parametru *remainRepls*).

Množina hlavních replikací se neliší pouze mohutností (počet hlavních replikací), ale také vybranými replikacemi (tak aby nebyly opakovány již provedené výpočty v *pilotní fázi*). Pokud byly v rámci *pilotní fáze* simulovány replikace $S_{0,0,k}$ ($k = 1 \dots pilotMaxLimit$), pak ve finální fázi budou simulovány replikace $S_{0,0,k}$ ($k = pilotMaxLimit + 1 \dots mainReplCount$). Dosažené výsledky z předchozí fáze budou doplněny o dopočítané replikace a vyhodnocení bude provedeno nad celou množinou hlavních replikací (o mohutnosti *mainReplCount*).

7.6.3 Výstupy finální fáze

Po provedení *finální fáze* je možné provést statistické vyhodnocení získaných dat a vypočítat hodnotu ukazatele kapacity železniční stanice:

- Δd – přírůstek zpoždění na jeden vlak, dle požadavků na vyhodnocení studie je možné vypočítat také různě agregované hodnoty vztažené na skupiny osobních/nákladních vlaků, regionální/dálkové dopravy aj.

Výsledné ukazatele (z jednotlivých *scénářů*) je možné porovnat se stanovenými limitními hodnotami (dle směrnice SM124 (SŽDC, 2019)) a formulovat závěry o zjištěné kvalitě provozu s ohledem na zkoumanou *verzi systému*.

Pokud je cílem studie provedení zjišťování kapacity ohledně více různých *verzí systému*, pak je nutné proces metodiky zopakovat od začátku. V případě, že nedochází ke změně topologie infrastruktury železniční stanice nebo nedochází k významným změnám v počtu alternativ pro řešení konfliktů je při dalším zkoumání možné vynechat *průzkumnou fázi* nebo ji omezit na nezbytně nutné minimum.

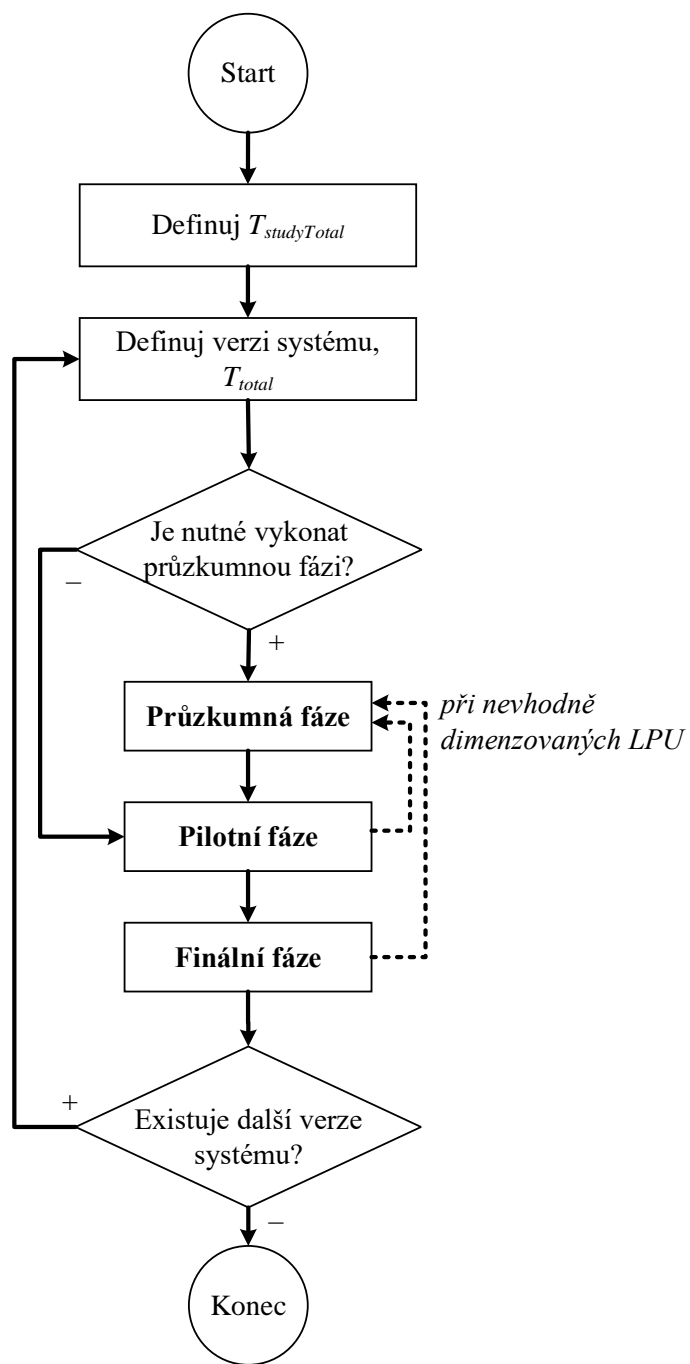
7.7 Aplikace metodiky na více různých verzí systémů

Výše uvedený postup popisuje aplikaci metodiky na jednu *verzi systému*. V případě, že je cílem kapacitní studie zkoumání více různých *verzí systému*, je v zásadě nutné celý postup iterativně opakovat. Pro vícenásobné použití metodiky je nutné definovat nový vstupní parametr:

- $T_{studyTotal}$ – časový limit pro provedení kompletní kapacitní studie (souhrnný čas využitý na zkoumání všech *verzí systému*).

Na základě souhrnného časového limitu je možné dále definovat dílčí časové limity pro jednotlivé aplikace metodiky (pro specifické *verze systému*). Dle potřeby je možné celkový časový limit rozdělit rovnoměrně nebo dle vlastních kritérií (dle priority *verze systému*, dle očekávané výpočetní složitosti, apod.).

Při zkoumání první *verze systému* je nezbytné provádět *průzkumnou fázi* pro stanovení prvotních odhadů výpočetní náročnosti experimentů zaměřených na zkoumání daného systému. Ale při zkoumání dalších *verzí systému* je v zásadě možné *průzkumnou fázi* zcela vynechat nebo ji vyhradit kratší časový limit, než jaký byl k dispozici pro první zkoumanou *verzi systému*. Pokud se dá očekávat, že výpočty související s jednotlivými *verzemi systému* budou mít podobnou výpočetní náročnost nebo se jednotlivé *verze* výrazně neliší, pak je možné zvážit zkrácení nebo vynechání *průzkumné fáze* v dalších iteracích. Pokud by se takový odhad nakonec ukázal jako chybný, vždy je možné přerušit *pilotní* nebo *finální fázi* a vrátit se do *průzkumné fáze* a provést ji dodatečně. Kompletní vývojový diagram popisující aplikaci metodiky na více různých *verzích systému* je znázorněn na obrázku 10.



Obrázek 10 – Vývojový diagram uplatnění metodiky na více verzí systému, zdroj: vlastní

8 SIMULAČNÍ NÁSTROJ MESORAIL

Pro praktickou realizaci vnořených simulací byl vytvořen původní simulační nástroj *MesoRail* (Diviš, 2015), který se specializuje na provádění simulací železničního provozu na mezoskopické úrovni podrobnosti. Hlavním posláním tohoto simulačního nástroje je podpora zkoumání propustnosti/kapacity kolejových infrastruktur železničních systémů (zejména železničních stanic a uzlů). Integrované vývojové prostředí je uzpůsobeno k relativně rychlému vytváření:

- *subsystém kolejové infrastruktury* (v součinnosti s editorem *TrackEd* (Novotny, 2015)), které odrážejí jak prostorové, tak technické vlastnosti zkoumaných kolejíšť,
- *subsystém železničního provozu*, které zahrnují prověřované grafikony vlakové dopravy a příslušné typy vlaků s jejich charakteristikami,
- *subsystém řídicích postupů*, které se uplatňují zejména při řešení konfliktů v rámci stochastických simulací.

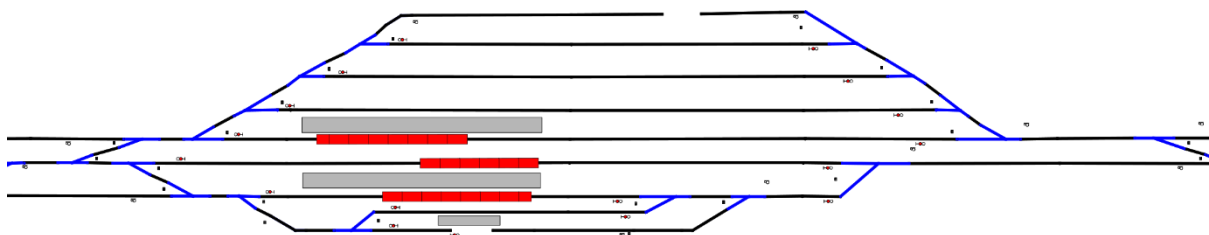
Cílový simulační model konkrétního zkoumaného systému je v rámci nástroje *MesoRail* sestaven z uvedených typů subsystémů, poté jsou prováděny příslušné simulační experimenty a následně jsou zpracovány jejich výsledky.

V porovnání s jinými simulátory železniční dopravy, které pracují na mikroskopické úrovni podrobnosti, uplatňuje *MesoRail* vyšší míru abstrakce. Prakticky to znamená, že jsou například:

- zanedbána sledování technických pracovníků, kteří pracují v terénu kolejíšť,
- zjednodušeny technologické procesy obsluh vlaků a některé úkony nejsou uvažovány (např. posuny),
- zanedbání entit cestujících,
- příslušná kolejíště jsou v průběhu simulace zobrazována ve schematické podobě, a nikoliv v měřítku (obrázek 11).

Vzhledem k zaměření simulačního nástroje a jeho aplikovatelnosti v rámci projektů v aplikační sféře byl při jeho vývoji kladen důraz na splnění hlavně následujících požadavků, které byly stanoveny ve spolupráci s odborníky z oblasti železniční dopravy:

- přemíst'ovací pohyby vlaků odrážejí jejich reálné jízdní vlastnosti (charakteristiky uplatněných výpočtů dynamiky jízd vlaků jsou uvedeny v Diviš (2014),
- řízení jízd vlaků respektuje činnost příslušných železničních zabezpečovacích zařízení,



Obrázek 11 – Ilustrace schematického zobrazování kolejové infrastruktury v rámci simulačního nástroje MesoRail, zdroj: vlastní

- jízdy vlaků dodržují platné staniční a traťové intervaly (jedná se například o následná mezidobí apod.),
- vstupní proudy vlaků do daného simulujícího systému mohou mít jak deterministický (tj. bez uplatnění zpoždění), tak stochastický charakter (s uvažovanými náhodnými zpožděními),
- jízdy vlaků jsou v průběhu simulačního výpočtu graficky animovány,
- ukládané výstupy ze simulačních experimentů jsou post-simulačně statisticky zpracovány, resp. jsou vytvořeny post-simulační grafické časové protokoly

Uvedený seznam je pouze výčtem vybraných požadavků, které jsou na činnosti simulátoru kladeny. Na jejich základě byly stanoveny úrovně abstrakce pro příslušné komponenty simulátoru. Pro stanovení vhodné míry abstrakce byly využity konzultace se železničními experty.

Pro podporu rozhodování ohledně řešení konfliktů je využívána metoda *reflektivních vnořených simulací*, která prověřuje odlišné varianty řešení. Vnořené simulace jsou implementovány pomocí rekurzivního přístupu (tj. v případě vzniku konfliktu ve vnořené simulaci je možné rekurzivně vyvolat další vnořenou simulaci). Simulátor rovněž plně podporuje využití metody *prioritního plánování*.

Pro ilustraci používaného přístupu za pomoci nástroje *MesoRail* je níže prezentována případová studie zaměřená na zkoumání železničního provozu v osobní železniční stanici menšího rozsahu. Pro vyhodnocení nasazení metody vnořených simulací jsou příslušné výsledky dosažené při uplatnění nástroje *MesoRail* porovnány s výsledky dosaženými v simulačních experimentech realizovaných v alternativním zavedeném simulačním nástroji.

V dalším textu je stručně popsána základní architektura simulačního nástroje *MesoRail* a poté následuje popis již zmíněné případové studie.

8.1 Základní předpoklady

Před zahájením vývoje simulátoru *MesoRail* a podpory rozhodování s využitím vnořených simulací byly stanoveny základní funkční požadavky na simulátor (specifikované výše) a dále prostředky potřebné pro implementaci.

Simulátor nevychází z kódu jiného simulátoru, již od počátku jeho vývoje byl plánován původní autorské dílo. Pro vývoj byl vybrán objektově orientovaný programovací jazyk *Java*, který představuje moderní vyšší programovací jazyk s podporou pro tvorbu grafického prostředí. V jeho základních knihovnách je obsažena podpora pro realizaci všech potřebných funkcionalit (např. práce se soubory, počítačovou sítí, vlákny apod.).

Datová báze simulátoru (popis infrastruktury, železničních kolejových vozidel, scénářů simulačních experimentů apod.) je uchovávána v souborech typu *XML* vlastního formátu. Původně bylo zvažováno použití standardizovaného formátu *RailML* (RailML.org, 2002), nicméně pro jeho vysokou komplexnost jej prozatím *MesoRail* nepoužívá.

8.2 Koncepce simulačního jádra

Výpočetní (simulační) jádro nástroje *MesoRail* využívá synchronizační metodu plánování diskrétních událostí. Jádro obsahuje centrální kalendář, do kterého jsou vkládány jednotlivé události a postupně jsou odebírány a zpracovávány dle jejich časového razítka. Procesy, které odrážejí kontinuální děje (pohyby vlaků aj.), jsou rozděleny na elementární aktivity, jejichž koncové události jsou periodicky (s velmi malou periodou) vkládány do kalendáře a následně zpracovávány.

Architektura simulátoru je částečně vystavěna na agentově orientované simulaci v kombinaci s využitím vlastností objektově orientovaného programování v jazyce *Java*. Jednotliví agenti (představování objekty) mohou přímo komunikovat s ostatními agenty a zasílat jim zprávy prostřednictvím volání jejich metod. V rámci simulátoru jsou výskyty událostí plánovány výhradně jednotlivými agenty.

8.3 Agentově orientovaný přístup

Simulátor využívá několik základních agentů, které lze klasifikovat jako autonomní reaktivní agenty (Bandini, 2009). Agenti jsou schopni reagovat jednak na dynamické změny ve svém okolí a jednak na zprávy od ostatních agentů. Vnitřní řídicí logika agentů se zaměřuje na ovlivňování jejich vlastního chování a/nebo na působení na jejich okolí. Základní agenti v simulaci jsou specifikováni prostřednictvím následujících tříd:

- *TrainMovementAgent* se specializuje na řízení přemístovacích pohybů jednotlivých vlaků v kolejišti.
- *TrainGeneratorAgent* je určen pro plánování vstupů vlaků do simulujícího systému podle daného jízdního řádu s potenciálním uplatňováním náhodných zpoždění.
- *TrainDispatcherAgent* se zaměřuje na celkové řízení železničního provozu včetně činnosti zabezpečovacích zařízení.
- *TrainAgent* představuje temporárního agenta, přičemž každá jeho instance popisuje vlastnosti a chování konkrétního vlaku, resp. vlakové soupravy.

8.4 Základní komponenty simulátoru

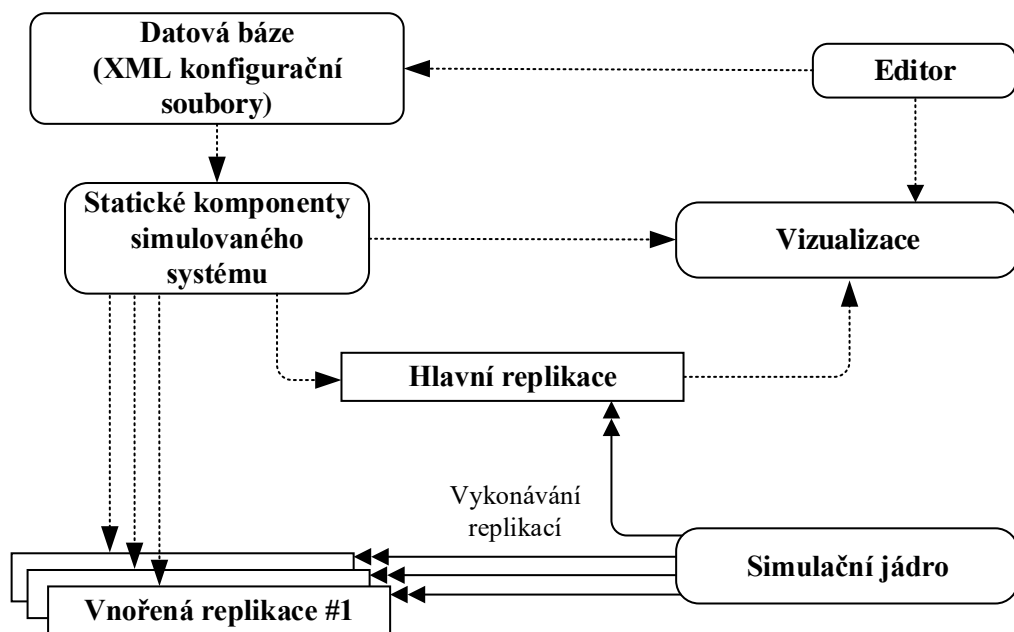
Strukturálně je softwarový simulující systém nástroje *MesoRail* vystavěn pomocí komponent, které lze rozčlenit do tří základních skupin specifikujících součásti simulátoru s odlišným zaměřením.

První skupina zahrnuje *statické součásti simulátoru* jako jsou: *datová strukturu graf* popisující metrické a topologické charakteristiky kolejové infrastruktury (včetně dalších relevantních stavebních objektů – např. nástupiště v osobních stanicích), *vektor rychlostních omezení* na infrastruktuře, množinu předdefinovaných *vlakových cest*, *katalog lokomotiv*, *vagonů* a jejich atributů. Tyto součásti jsou před zahájením simulací načteny do operační paměti z datové báze simulátoru (uložené na externím paměťovém médiu) a jejich obsah se v průběhu simulačních experimentů nemění.

Druhá skupina se skládá ze součástí podporujících *provádění běhu výpočtu replikací* simulačních experimentů. Replikace představuje jeden běh simulace (řídící se daným scénářem a příslušným nastavením parametrů) a disponuje: *výpočetním jádrem* zabezpečujícím evoluci simulačního výpočtu, *generátory pseudonáhodných čísel*, potřebnými *objekty-agenty* a *objekty-entitami*, které v průběhu simulace mohou měnit svůj stav a potenciálně vzájemně komunikovat.

Třetí skupina je složena ze součástí určených k podpoře *editace grafických objektů* a *online animací*, které graficky odrážejí měnící se stavový prostor zkoumaného systému v průběhu simulace.

Vazby mezi jednotlivými skupinami komponentů jsou znázorněny na obrázku 12.



Obrázek 12 – Vazby mezi jednotlivými skupinami komponentů nástroje MesoRail, zdroj: vlastní

8.5 Simulační replikace

Jedna *replikace hlavní simulace* je v nástroji *MesoRail* realizována jako *samostatný proces*. Na rozdíl od teoretického popisu v *Algoritmu 2* (v části *program MainSimulation*), bylo toto dělení replikací do procesů zvoleno z praktických důvodů. Při provádění hlavních simulací na výpočetním clusteru není nutné řešit distribuci výpočtů, tyto činnosti zařídí standardní plánovač úloh příslušného operačního systému.

Paralelismy ve funkcích *RunRepl* a *MultirunVariant* jsou řešeny pomocí skupiny vláken (*thread pool*). Pro jednotlivé (vnořené) replikace je následně využíváno plánování metodou *Round-Robin (RR)*. Tato metoda byla zvolena z důvodu, že dopředu není známo, ve kterých (vnořených) replikacích dojde k dalšímu vnořování. Plánování pomocí metody *First Come*

First Served (FCFS) by totiž mohlo vést k delším časovým intervalům, kdy není efektivně využívána veškerá výpočetní kapacita.

Všechny potřebné součásti replikace jsou zapouzdřeny původní komponentou *StatesPersistence*, která umožňuje zaznamenávat stav jednotlivých objektů a klonovat je. Prakticky to znamená, že prostřednictvím uvedené funkcionality je možné uchovat libovolný stav simulačního výpočtu (například při pozastavení simulace při výskytu konfliktu), ke kterému je možné se později vrátit a opět pokračovat v simulaci. Uvedený přístup je využíván pro realizaci vnořených simulací.

Činnost replikace lze rozdělit na *inicializační etapu* a *exekutivní etapu*. V rámci *Algoritmu 2* jsou činnosti spojené s uvedenými etapami prováděné podprogramy *Initialize* a *RunRepl*. V rámci *inicializační etapy* je nutno rozlišovat, zda se jedná o replikaci hlavní simulace nebo o replikaci vnořené simulace. V případě replikací hlavní simulace naplňuje agent *TrainGeneratorAgent* příjezdy prvních vlaků z každého směru do simulujícího systému. Replikace vnořených simulací, které jsou vždy vyvolané výskytem konkrétního konfliktu v hierarchicky nadřazené replikaci, prověřují příslušné varianty řešení tohoto konfliktu získané generátorem *VarntsGen*. Pro uvedené replikace jsou vždy při inicializaci stanoveny příslušné doby výhledů, které se budou počítat (blíže je tato problematika popsána v podkapitole 9.3).

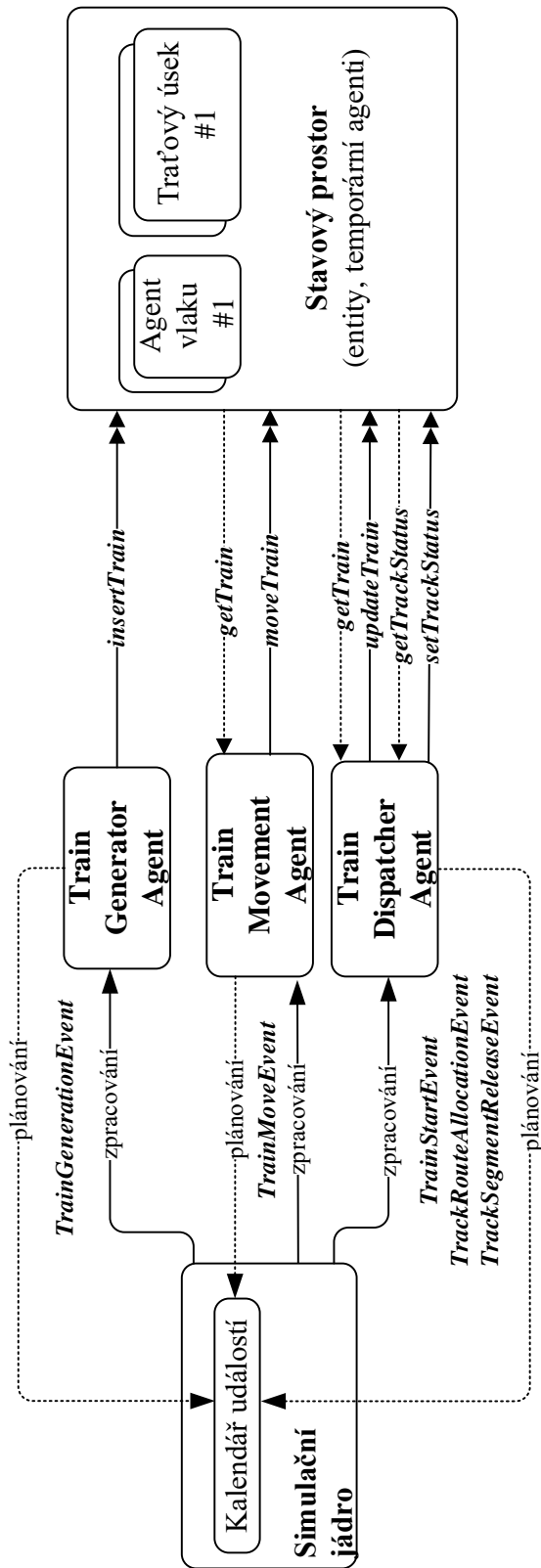
Exekutivní etapa se soustředí jednak na provádění klasického výpočetního cyklu synchronizační metody plánování diskretních událostí (prováděného rutinou *Simulate*) a jednak na ošetření výskytů konfliktů, která jsou potenciálně spojena se spuštěním vnořených simulací prostřednictvím násobných volání podprogramu *MultirunVariant*. Zjednodušená koncepce činnosti exekutivní etapy replikace je strukturálně graficky ilustrována na obrázku 13. Plánování odlišných typů událostí je dominantně prováděno agenty *TrainMovementAgent*, *TrainGeneratorAgent* a *TrainDispatcherAgent*. Při zpracování událostí (prováděném výpočetním jádrem replikace) jednak dochází ke změnám ve stavovém prostoru a jednak k potenciálnímu naplánování dalších událostí. Řídící a výkonné vazby uvedených agentů na stavový prostor jsou zobrazeny grafickými vazbami na strukturu vlaků.

8.6 Řídící logika pohybu vlaku

Vlastní simulace jízdy jednotlivých vlaků je realizována na mikroskopické úrovni podrobnosti. Jednotlivé instance agentů typu *TrainAgent* musejí v průběhu simulace reagovat na stav železniční infrastruktury a přijímat platné příkazy od agenta řídicího provoz v rámci celé simulace (*TrainDispatcherAgent*). Při simulaci jízdy vlaku se vyskytuje řada dílčích komplexních problémů zahrnujících např.:

- zjištění aktuální hodnoty zpoždění vlaku,
- udržování rychlosti vlaku dle aktuálních limitů a jízdního řádu,
- zastavení vlaku na specifické pozici na infrastruktuře (např. u nástupiště).

Při simulaci pohybu vlaku v kolejišti není možné v rámci mikroskopické úrovně podrobnosti simulace provádět skokové změny rychlosti vlaku, takové chování by neodpovídalo realitě.



Obrázek 13 – Zjednodušená koncepce činnosti exekutivní etapy replikace, zdroj: vlastní

Plynulá změna rychlosti v důsledku například zpomalení s užitím brzd pak představuje složitou funkci, která reflektuje jednak fyzikální parametry vlaku a jednak fyzikální parametry části kolejiště, po kterém se vlak právě pohybuje. Způsoby řešení výše vedených situací jsou představeny v následujících podkapitolách.

8.6.1 Zjišťování aktuální hodnoty zpoždění vlaku

Pro potřeby statistického vyhodnocení výsledků simulací, ale i pro potřeby podpory rozhodování je nezbytné, aby simulátor byl schopen určit, jaké má vlak zpoždění oproti plánovanému jízdnímu řádu. V optimálním případě by takové zjišťování mělo být prováděno v každém časovém okamžiku simulace, resp. v každé pozici vlaku při pohybu kolejištěm.

Jízdní řád obvykle vlakům definuje pouze několik diskrétních časových okamžiků, které jsou pevně svázané s určitými body na infrastruktuře železnice, např. příjezdy a odjezdy z jednotlivých zastávek a stanic. Samotný jízdní řád pak obvykle nedefinuje, v jaké poloze má být vlak v každém časovém okamžiku. Pokud by simulátor používal pouze tyto informace k určení zpoždění vlaku, pak by nemohla být informace o zpoždění vlaku dodávána včas. V důsledku neaktuálních informací o hodnotách zpoždění vlaků může docházet k suboptimálnímu rozhodování při řešení konfliktních stavů (změny velikosti zpoždění, které by nastaly mimo stanici, by zůstaly skryté až do okamžiku příjezdu vlaku do dalšího časově kótovaného bodu na železnici). Podpora rozhodování s užitím *RNS* by pak používala neaktuální hodnoty zpoždění a výsledek rozhodování by mohl být negativně ovlivněn oproti stavu se známými aktuálními hodnotami zpoždění vlaků.

V nástroji *MesoRail* se využívá dvoufázový systém pro zjišťování zpoždění. První fáze je realizována před samotnými simulačními experimenty, po vybudování simulačního modelu. V rámci této fáze je provedena série deterministických simulací, při kterých je pro jednotlivé vlaky a jejich příslušné vlakové cesty proveden záznam časových poloh a pozic v kolejišti. Druhá fáze pak představuje samotné (stochastické) simulační experimenty. Při nich jsou využity zaznamenané hodnoty a zpoždění je možné zjišťovat v prakticky každém časovém okamžiku simulace. Pro praktické použití je v nástroji *MesoRail* vzorkování jízd vlaků prováděno nikoliv v každém časovém okamžiku, ale vždy po ujetí vzdálenosti 100 m.

8.6.2 Udržování rychlosti vlaku dle platných limitů a jízdního řádu

Při pohybu vlaku v kolejišti je třeba respektovat řadu rychlostních omezení a limitů, jedná se zejména o:

- maximální povolenou traťovou rychlost,
- maximální povolenou rychlost jednotlivých kolejových vozidel,
- stanovenou rychlost vlaku dle jízdního řádu.

V simulátoru není problém určit, jaká je aktuální povolená (nebo požadovaná) rychlost vlaku v každém okamžiku simulace. Problém simulace na mikroskopické úrovni podrobnosti je dosáhnout takové rychlosti korektním způsobem (brzděním nebo akcelerací).

Aby bylo v nástroji *MesoRail* dosaženo korektní simulace jízdy vlaků, je využíván princip podobný technice *RNS*. Aby byl vlak schopen začít brzdit včas, jsou nejdříve identifikovány

všechny body na aktuální trase vlaku, kde dochází ke snížení povolené maximální rychlosti vlaku. Pro každý takový bod je následně provedena omezená vnořená simulace, která se zaměřuje pouze na daný vlak a simuluje pouze jízdu vlaku s ohledem na dodržení fyzikálních zákonitostí. Na základě těchto výhledů je možné identifikovat jednotlivé body na trase vlaku, kde je nutné začít brzdit, tak aby v bodě se sníženou rychlostí jízdy bylo dosaženo očekávané rychlosti.

8.6.3 Zastavení vlaku v definované poloze na kolejišti

Problematika zastavení vlaku v definované poloze se primárně vyskytuje ve dvou situacích. Jednak pro zastavení vlaku ve stanici v poloze u nástupiště (resp. obecně na libovolném místě) a jednak u zastavení vlaku před návěstidlem, které nedovoluje další jízdu vlaku. Problematika zastavení vlaku je totožná s problémem respektování maximální povolené rychlosti vlaku (podkapitola 8.6.2), liší se však v tom, že vlak je nutné zabrzdit do nulové rychlosti. Vlastní řešení této problematiky je totožné s již dříve uvedeným. Liší se zde pouze ve způsobu, kterým jsou definovány body na trase, kde dochází ke snížení povolené rychlosti vlaku. V tomto případě se vychází z definovaných kilometrických poloh nástupišť či návěstidel.

9 PŘÍPADOVÁ STUDIE

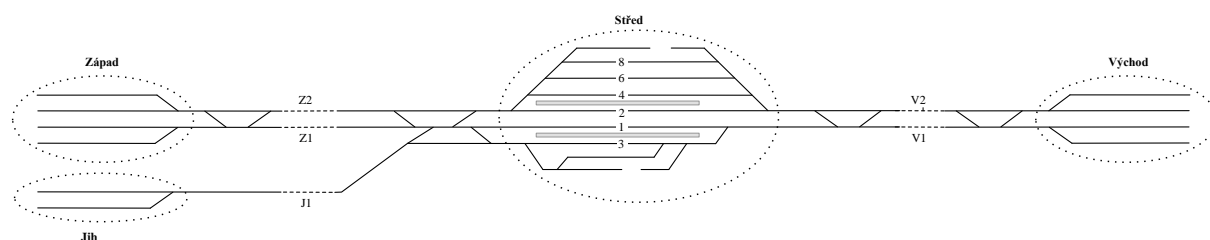
Prezentovaná případová studie navazuje na projekt smluvního výzkumu *Návrh nové metodiky propustnosti železničních stanic*, na jehož úspěšném řešení se podílel autor disertační práce. Uvedený projekt byl zpracováván pro státní společnost SŽDC v letech 2014-2016 na *Fakultě elektrotechniky a informatiky Univerzity Pardubice* (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016).

Případová studie byla zaměřena na prověřování železničního provozu v rámci menší osobní stanice *Zdice* pro konkrétně stanovený jízdní řád. V rámci simulačních experimentů byl aplikován optimalizační přístup založený na minimalizaci hodnot jednoho provozního ukazatele železniční dopravy, kterým je *úhrnný přírůstek zpoždění vlaků* v rámci systému zkoumané stanice. Uplatněná optimalizace bylo postaveno na počítačové simulaci, v jejímž rámci byla aplikována podpora rozhodování ohledně řešení konfliktů využívající reflektivní vnořené simulace (Diviš, 2017). Sledovaným typem konfliktu byl požadavek na přidělení konkrétní staniční koleje pro dané časové okno od více (potenciálně zpožděných) vlaků. V rámci případové studie byl hlavně používán simulační nástroj *MesoRail*.

9.1 Základní charakteristika železniční infrastruktury – vymezení zkoumaného systému

Submodel infrastruktury zkoumaného systému (schematicky zjednodušeně ilustrovaný na obrázku 14), který v příslušné datové vrstvě realisticky odráží sklonové a obloukové poměry jednotlivých kolejí, strukturálně zahrnoval:

- Kolejiště výše uvedené osobní železniční stanice, která bude pro potřeby této studie označována jako *Střed*. V rámci kolejiště byly pro osobní vlaky využívána zejména dvě nástupiště: *nástupiště 2* (s přiléhajícími kolejemi #1 a #3), *nástupiště 3* (s nástupištními kolejemi #2 a #4). Obě nástupiště měly délku 300 m.
- Trať vedoucí do sousedních hraničních stanic označovaných jako *Jih* (původní název stanice Lochovice), *Západ* (původní název stanice Hořovice) a *Východ* (původní název stanice Beroun). Do stanice *Jih* vedla jednokolejná trať a do stanic *Západ* a *Východ* tratě dvoukolejné (celková vzdálenost mezi hraničními stanicemi *Západ* a *Východ* činila cca 20 km).
- Zjednodušená kolejiště hraničních stanic *Jih*, *Západ* a *Východ*.



Obrázek 14 – Přehledové zobrazení kolejové infrastruktury zkoumaného systému, zdroj: vlastní

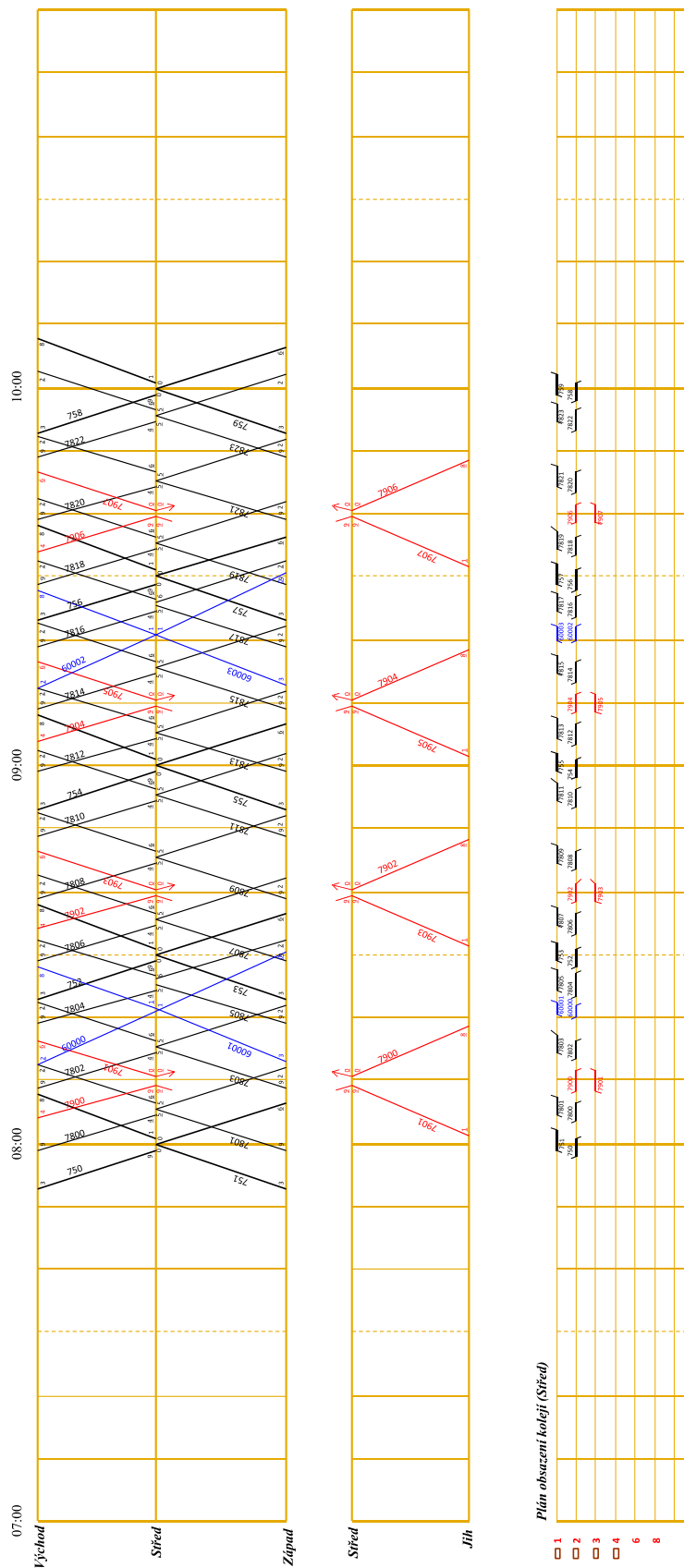
V rámci *submodelu provozu* byly specifikované dominantně *vlaků osobní dopravy* a v menším rozsahu *vlaků nákladní dopravy*. Dále byl definován příslušný *jízdný řád* (sestavený železničním expertem ze SŽDC), který charakterizoval provoz v dopravní špičce 8–10 hod. (obr. 15) a zahrnoval celkem 46 vlaků. Vzhledem k jízdnímu řádu byl pro stanici *Střed* k dispozici tzv. *statický plán obsazení staničních kolejí*, který primárně stanovil konkrétním vlakům, které staniční koleje jim měly být přiděleny (obr. 15). Tento plán počítal v ideálním případě s přidělováním kolejí #1, #2 a #3. V případě, že aktuální provozní situace neumožňovala dodržet výše zmíněný plán, bylo pro alternativní možnosti přidělování celkem určeno 6 staničních kolejí (obr. 14). Koleje #6 a #8 byly určeny zejména pro alternativní přidělování vlakům nákladní dopravy, zatímco nástupištní koleje #1, #2, #3 a #4 představovaly staniční koleje pro variantní přidělování vlakům osobní dopravy.

Při vstupu do systému byly vlaky zatíženy náhodnými zpožděními, jejichž charakteristiky byly na základě železničních metodik zadány odborníky ze SŽDC (z tohoto pohledu lze konstatovat, že drtivá většina prováděných experimentů využívala stochastické simulace). Pro každý vlak byly stanoveny množiny přípustných *alternativních vlakových cest* (vedoucích přes odlišné staniční koleje) v rámci infrastruktury stanice *Střed*, aby bylo možné realizovat jízdy vlaků odlišnými trajektoriemi při rozdílných provozních situacích.

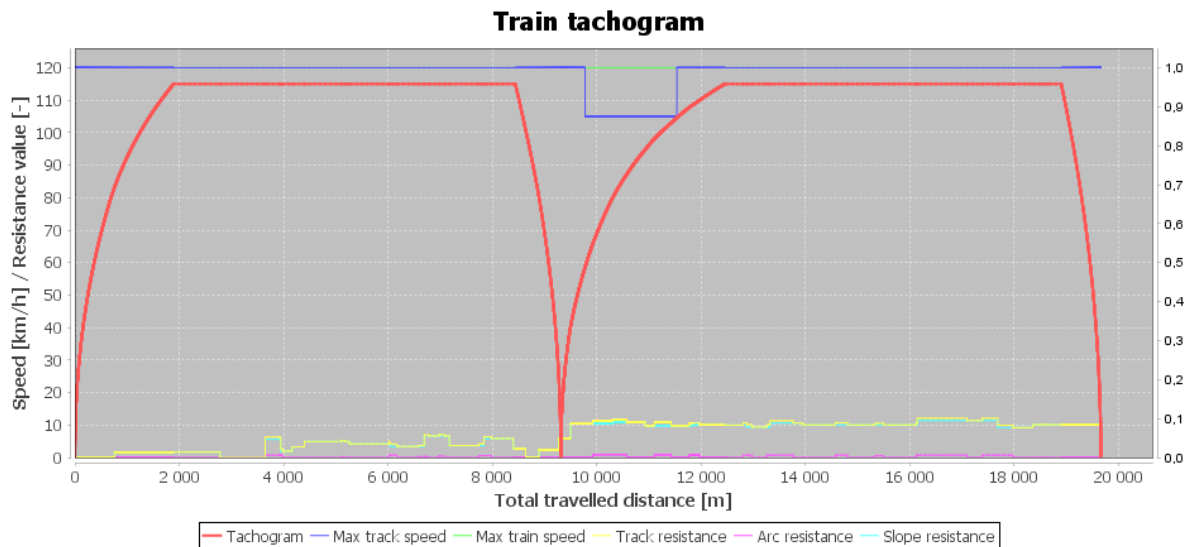
Vlaky osobní dopravy byly rozlišovány na dva typy: *osobní vlaky* (zabezpečující regionální dopravu) a *rychlíky* (zajišťující dálkovou dopravu). Základní charakteristiky vlaků jsou uvedeny v tabulce 2. Vlaky osobní dopravy zastavovaly ve stanici *Střed* na dobu jedné minuty (ukázka zaznamenaného tachogramu z deterministické simulace je na obrázku 16). Nákladní vlaky stanic pouze projížděly.

Tabulka 2 – Vybrané charakteristiky vlaků, zdroj: vlastní

Linka	Kategorie vlaku	Směr (V – Východ, S – Střed, J – Jih, Z – Západ)	Interval taktu [min]	Pravděpodobnost výskytu vstupního zpoždění (alternativní rozdělení)	Střední hodnota zpoždění (exponenciální rozdělení)
R1	dálková	$V \rightarrow S \rightarrow Z$	30	0,50	420 s
R2	dálková	$Z \rightarrow S \rightarrow V$	30	0,50	420 s
Os1	regionální	$V \rightarrow S \rightarrow Z$	10	0,33	270 s
Os2	regionální	$Z \rightarrow S \rightarrow V$	10	0,33	270 s
Os3	regionální	$V \rightarrow S \rightarrow J$	30	0,33	270 s
Os4	regionální	$J \rightarrow S \rightarrow V$	30	0,33	270 s
Pn1	nákladní	$V \rightarrow Z$	60	0,50	1800 s
Pn2	nákladní	$Z \rightarrow V$	60	0,50	1800 s



Obrázek 15 – Grafikon vlakové dopravy a plán obsazení kolejí pro zkoumaný systém v rámci dopravní špičky, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)



Obrázek 16 – Zaznamenaný tachogram podle ujeté vzdálenosti vlaku 750 (linky R1) při deterministické simulaci v nástroji *MesoRail*, zdroj: vlastní

V rámci *submodelu řídicích postupů* byla věnována pozornost zejména podpoře rozhodování spojené s řešením konfliktů spojených s přidělováním staničních kolejí vlakům přijíždějícím do stanice. K dispozici byly dva přístupy:

- (a) *Vnořené simulace*, prověřovaly pro přijíždějící vlak primárně *všechny aktuálně přípustné varianty* přidělení staniční koleje. Navíc byla simulována i varianta, ve které by přijíždějící vlak „pouze“ čekal až do uvolnění staniční koleje, která pro něj byla původně určena podle statického *plánu obsazení kolejí*. Varianta, která vykazala nejlepší výsledek (získaný pomocí funkce *CrOptim*), byla uplatněna jako řešení daného konfliktu.
- (b) *Statické prioritní seznamy*, které obsahovaly *všechny obecně přípustné varianty* přidělení staniční koleje pro konkrétní přijíždějící vlaky. Seznamy byly utříděny podle priorit prvků/variant, přičemž prioritou varianty byla tím vyšší, čím byla provozně výhodnější. Varianta s nejvyšší prioritou odpovídá řešení, které bylo v souladu s *plánem obsazení staničních kolejí*. Uvedené seznamy v rámci této případové studie byly sestaveny v souladu s doporučením železničního experta. Pro přijíždějící vlak byla ze seznamu vybrána ta aktuálně provozně proveditelná varianta, která měla nejvyšší prioritu.

Primárně byly pro řešení výše uvedených konfliktů používány vnořené simulace. Pro konflikty, které se vyskytovaly v rámci poslední povolené úrovně vnoření (definované parametrem *maxLevel*), se řešení hledaly v prioritních seznamech. Ukázka sestavených prioritních seznamů pro přidělování staničních kolejí (označení těchto kolejí ilustruje obrázek 14) ve stanici *Střed* je uvedena v tabulce 3.

Nutno podotknout, že v této případové studii byly prioritní seznamy používány dvojím způsobem:

- Výběr *jednoho* nejvhodnějšího („nesimulačního“) aktuálně proveditelného řešení daného konfliktu byl aplikován v replikacích na nejnižší hierarchické úrovni. Pro tyto případy tedy sloužily prioritní seznamy jako jediná podpora rozhodování.

- V ostatních replikacích byl nad prioritními seznamy prováděn příslušný výběr *všech* aktuálně proveditelných řešení, která byly následně prověřovány v rámci replikací vnořených simulací. Tento výběr uskutečňoval softwarový generátor variant *VarntsGen*, který byl zmíněn v podkapitole 6.1. Jako podpora rozhodování byly aplikovány vnořené simulace.

Tabulka 3 – Prioritní seznamy s alternativami přidělování staničních kolejí vlakům, zdroj: vlastní

Linka	Počet vlaků	Čísla vlaků	Směr linky		Statický prioritní seznam (staniční koleje)			
			Ze stanice (traťová kolej)	Do stanice (traťová kolej)	1.	2.	3.	4.
<i>R1</i>	5	750, 752, ..., 758	V (V2)	Z (Z2)	#2	#4	#1	#3
<i>R2</i>	5	751, 753, ..., 759	Z (Z1)	V (V1)	#1	#3	#2	#4
<i>Os1</i>	12	7800, 7802, ..., 7822	V (V2)	Z (Z2)	#2	#4	#1	#3
<i>Os2</i>	12	7801, 7803, ..., 7821	Z (Z1)	V (V1)	#1	#3	#2	#4
<i>Os3</i>	4	7900, 7902, ..., 7906	V (V2)	J (J1)	#2	#4	#1	#3
<i>Os4</i>	4	7901, 7903, ..., 7907	J (J1)	V (V1)	#3	#1	#2	#4
<i>Pn1</i>	2	60000, 60002	V (V2)	Z (Z2)	#2	#4	#6	#8
<i>Pn2</i>	2	60001, 60003	Z (Z1)	V (V1)	#1	#2	#6	#8

9.2 Základní informace o simulačních experimentech

Pro uskutečnění simulačních experimentů, zaměřených na prověřování charakteristik železničního provozu v osobní stanici *Střed* (při uplatnění definovaného jízdního řádu), je vhodné rekapitulovat základní informace týkající se simulujícího systému vybudovaného v rámci nástroje *MesoRail*.

Simulační experimenty odrážely provoz v *ranní dopravní špičce* (08:00–10:00 hod.) v osobní stanici *Střed*. V rámci zkoumaného jízdního řádu (obrázek 15) vstupovaly do simulujícího systému jako první vlaky v hraničních stanicích *Západ* a *Východ* (v čase 07:53 hod.) a jako poslední ze systému vystupoval vlak v hraniční stanici *Východ* (v čase 10:08 hod.).

Pro prvotní ověření ideálního uplatnění jízdního řádu, kdy nebyl provoz zatížen žádnými zpožděními vlaků, byl proveden jeden *deterministický simulační experiment*. Uskutečněný experiment potvrdil, že zkoumaný jízdní řád byl bezkonfliktní.

Při provádění *stochastických simulací* vstupovaly vlaky do systému podle uvedeného jízdního řádu, přičemž byly pro ně potenciálně vygenerována náhodná zpoždění. Pro určení, zda byl konkrétní vstupující vlak zpožděn či nikoliv, se využíval generátor pseudonáhodných čísel řídicí se alternativním rozdělením pravděpodobnosti (hodnoty příslušného parametru uvádí tabulka 2). Pro případ, že byl vlak zpožděn, byl dále vlaku přiřazena konkrétní hodnota zpoždění. Hodnoty zpoždění byly generovány podle exponenciálního rozdělení pravděpodobnosti (příslušné střední hodnoty jsou uvedeny v tabulce 2). Přiřazování náhodných zpoždění vlakům se řídilo principy, které uvádí *Směrnice SŽDC SM124 – Zjišťování kapacity*

dráhy (SŽDC, 2019). Kromě uvedených zpoždění nebyly v simulačních experimentech uplatňovány výskyty žádných dalších náhodných jevů.

Každá replikace hlavní simulace byla rozdělena na dvě výpočetní fáze, které aplikovaly identický (výše uvedený) jízdní řád. Nejdříve byla uplatněna pomocná *náběhová fáze* (*warming up period*), v jejímž rámci se nesbíraly údaje pro statistické vyhodnocení. Úkolem této fáze bylo náhodné zaplnění zkoumaného systému potenciálně zpožděnými vlaky. Poté bezprostředně následovala hlavní *sledovací fáze* (*tracking period*) s příslušným sběrem dat pro post-simulační vyhodnocení provozních charakteristik. V prováděných experimentech byla *náběhová fáze* zaplněna pomocí stejných vlakových souprav jako *fáze sledovací*, uplatněny byly všechny parametry uvedené v tabulce 2. Časový okamžik příjezdu jednotlivých vlaků do modelu byl posunut o dvě hodiny dopředu oproti hodnotě uplatněné ve *sledovací fázi*.

Při výskytech konfliktů byly přijíždějícím vlakům přidělovány alternativní staniční koleje a s tím spojené alternativní jízdní cesty. Každý vlak byl spojen s *jednou primární jízdní cestou* (obsahující příslušnou staniční kolej v souladu se statickým plánem obsazení kolejí) a *třemi alternativními cestami* využívajícími jiné staniční koleje (definované v prioritních seznamech, které uvádí tabulka 3).

Z důvodu možnosti získání dostačeného rozsahu výstupních dat ze simulací pro jejich následné statistické zpracování, bylo pro příslušné hlavní simulace prováděno *100 replikací* (*mainReplCount* = 100).

Další charakteristiky železniční infrastruktury a detaily zkoumaného scénáře jsou uvedeny souhrnně v příloze A.

9.3 Charakteristika podpory rozhodování v nástroji MesoRail

Jak již bylo zmíněno, v rámci simulátoru *MesoRail* byl v souvislosti s případovou studií primárně použita pro řešení konfliktů podpora rozhodování využívající vnořené reflektivní simulace. Pouze u konfliktů, které se vyskytovaly v replikacích na nejnižší hierarchické úrovni, se příslušná řešení vybíraly na základě uplatnění metody statického prioritního plánování (s využitím prioritních seznamů specifikovaných v tabulce 3).

Pro potřeby optimalizace se u každé replikace zjišťovala hodnota indikátoru *suma vážených přírůstků zpoždění* všech vlaků (*SWDI*), které byly provozně sledovány (tj. 46 vlaků). Každému vlaku byla přiřazena váha, která odrážela jeho důležitost v dopravním provozu. Váhy sledovaných typů vlaků byly nastaveny v souladu se *Směrnici SŽDC SM124* (SŽDC, 2019) následovně: $w_{dálková} = 1,8$, $w_{regionální} = 1,0$ a $w_{nákladní} = 0,2$.

Indikátor *meanSWDI* se používal pro hodnocení odlišných variant řešení konfliktu, kdy byla varianta/experiment s nejnižší hodnotou tohoto indikátoru vybírána jako optimální. Navíc se tento ukazatel rovněž používal pro vyhodnocování primárních experimentů sestávajících z replikací hlavních simulací. Pro souhrnné zhodnocení dosahovaných přesností hodnot *meanSWDI* (z více počítaných variant řešení konfliktů) byl také využíván statistický indikátor *meanRelatHalfWidth*. Tento ukazatel vyjadřoval průměr z hodnot *relatHalfWidth* příslušejících všem zpracovaným variantám řešení konfliktů.

Při výskytu konfliktu, jehož řešení bylo vybíráno na základě získaných údajů z vnořených simulací, bylo pro příslušné replikace důležité určit konkrétní hodnotu parametru *délka časového výhledu* (*LD*, resp. parametr *stopCond*). Tento parametr udával interval simulačního času, v jehož rámci byly prováděny prediktivní výpočty. Doporučení ohledně nastavení hodnoty *LD* parametru se odvíjí od typu konkrétního simulovaného systému. Pro tuto případovou studii bylo železničním odborníkem doporučeno aplikovat dobu výhledu *30 minut* na úrovni hlavní simulace. Tato hodnota výhledu realisticky odráží informovanost dispečerů o budoucím provozu (ohledně očekávaných příjezdů vlaků) v rámci železničních stanic. V případech, že hodnota *maxLevel* > 1, pak byla pro další úrovně vnoření *RNS* uplatňována strategie *hierarchicky redukováných dob výhledů* pro stanovení hodnoty *LD*.

Prověřované varianty řešení konfliktů byly generovány generátorem *VarntsGen*, který využíval definované statické prioritní seznamy staničních kolejí definované pro jednotlivé vlaky. Prověřovány byly všechny varianty, které byly v okamžiku výskytu konfliktu přípustné. Kromě variant, kdy docházelo ke změně vlakové cesty byla vždy zkoumána i varianta, kde vlak pokračoval po původní vlakové cestě. Maximální prověřovaný počet variant (*maxVarnts*) nebyl omezen.

Z pohledu uskutečněných simulací s použitím *RNS* byl ve vybraných experimentech aplikován různý počet (1, 2, 3) úrovní vnořování a při prověřování variant řešení konfliktů se testovalo uplatnění různých počtů (3, 5, 7, 10, 20) replikací vnořených simulací.

Pozn.: Pro případy simulačních experimentů, pro které platilo *maxLevel* = 0, nebyly využívány vnořené simulace a řešení konfliktů v rámci replikací hlavních simulací byla založena na výhradním používání metody prioritního plánování.

9.4 Villon

Aby bylo možné hodnotit výsledky některých výpočtů prováděných v rámci mezoskopického simulujícího systému provozovaného v nástroji *MesoRail*, byl vystavěn odpovídající simulující systém (pracující na mikroskopické úrovni podrobnosti) v rámci softwarového nástroje *Villon*. Tento systém zahrnoval stejnou kolejovou infrastrukturu, stejný jízdní řád a stejnou strategii generování zpoždění vlaků na vstupu jako simulátor *MesoRail*. Při řešení konfliktů bylo v nástroji *Villon* aplikováno výhradně statické prioritní plánování, které využívalo stejně sestavené prioritní seznamy, které byly uvedeny v tabulce 3.

Simulátor *Villon* je rutinně používán při zpracovávání simulačních studií pro velké železniční společnosti a z pohledu: (i) výpočtů jízdních dob vlaků, (ii) příslušných tachogramů ohledně jízd vlaků, (iii) modelů dopravních technologických procesů a (iv) činnosti zabezpečovacích zařízení byl expertně úspěšně validován ze strany *SŽDC*.

Byť každý z výše uvedených simulátorů pracoval na jiné úrovni podrobnosti, jevílo se jako účelné zhodnotit data o uskutečněných jízdách vlaků v nástroji *MesoRail* pomocí hrubého srovnání s příslušnými údaji získanými z validovaného nástroje *Villon*.

V obou simulačních nástrojích byly nejdříve provedeny *deterministické simulace* uplatňující zadaný jízdní řád (obr. 15). V průběhu těchto simulací došlo k zaznamenání jízdních dob všech

vlaků. Pro porovnání byl vypočítán relativní rozdíl jízdních dob jednotlivých vlaků. Maximální relativní rozdíl pro jeden vlak činil 0,634 % a průměrný relativní rozdíl pro všechny simulované vlaky činil 0,269 %. Jelikož zjištěné relativní rozdíly nepřesáhly a priori stanovenou hraniční hodnotu 5 %, byl mezoskopický model v simulátoru *MesoRail* hodnocen, že pracoval na požadované úrovni přesnosti. Porovnání výsledků ze *stochastických simulací* je uvedeno níže.

9.5 Cluster

V případové studii byly výpočty všech simulačních experimentů prováděny na počítačovém clusteru, který disponoval datovým úložištěm (o kapacitě 96 TB diskového paměťového prostoru, který byl rozčleněn na 24 HDD) a čtyřmi výpočetními uzly. Každý uzel byl osazen (i) čtyřmi 12 jádrovými procesory Intel XEON, přičemž *Hyper-Threading (HT) technologie* umožňovala paralelní vykonávání 96 vláken, (ii) paměti 384 GB RAM a (iii) SSD diskem pro operační systém o kapacitě 1TB. Cluster byl propojen 10 Gb sítí Ethernet, která sloužila pro komunikaci s úložištěm a mezi uzly navzájem. Na clusteru byl provozován operační systém *CentOS Linux 7*, jako plánovač byl použit nástroj *OpenPBS*.

9.6 Předpoklady

Pro potřeby této případové studie byla prověřována jedna *verze zkoumaného systému* (spojená s jedním konkrétním jízdním řádem a jednou konfigurací kolejové infrastruktury). Za účelem získání statisticky vypovídajících výsledků bylo požadováno simulovat *100 hlavních replikací* (*mainReplCount*) pro zvolené výsledné *scénáře*.

Dále se vycházelo z předpokladu, že při řešení reálného projektu bylo pro jednu posuzovanou verzi příslušného systému k dispozici maximálně *225 hodin pracovního (reálného) času* (T_{total}). Tato hodnota byla stanovena na základě předchozích zkušeností s obdobnými simulačními studii. Jako příklad lze uvést, že na typickou studii zahrnující 5 zkoumaných *verzí systému* mohlo být k dispozici 6 měsíců. Během této doby bylo třeba formulovat veškeré vstupní parametry, vytvořit patřičné simulační modely, provést samotné simulace, sběr a vyhodnocení získaných dat a vytvoření výstupní zprávy. Na jednu verzi systému tak připadal zhruba jeden měsíc, během kterého bylo nutné vytvořit simulační model (nebo přizpůsobit stávající), verifikovat a validovat jej a následně provést vlastní simulační výpočty a vyhodnocení dílčích výsledků. Výše uvedená hodnota ($T_{total} = 225$ hodin) pak představovala zhruba jednu třetinu měsíce. Tato doba byla k dispozici pro vlastní provedení simulačních experimentů, uvedený časový úsek tedy nezahrnoval činnosti související s vybudováním, verifikací a validací cílového počítačového simulačního modelu. Pro první *průzkumnou fázi* bylo vyhrazeno maximálně 25 hodin ($T_{probeMaxLimit}$). Na pilotní simulační výpočty bylo vyhrazeno maximálně *100 hodin* ($T_{pilotMaxLimit}$) a dalších maximálně *100 hodin* bylo plánováno na provedení vybraných finálních experimentů ($T_{finalMaxLimit}$). Poměr alokovaných časových intervalů tedy činil 1:4:4 pro jednotlivé fáze metodiky. *Průzkumná fáze* by měla představovat jen malou část z celkové alokované doby na provedení studie, pro *pilotní* a *finální* fáze bylo alokováno stejné množství času.

Posledním z parametrů, který bylo třeba specifikovat byl parametr *stopCond* určující, kdy docházelo k ukončení jednotlivých replikací vnořených simulací. Případová studie uplatňovala

parametr *stopCond* jako *dobu výhledu* (LD) a tato hodnota nebyla závislá na jiných stavech v prováděných simulacích. Pro případovou studii bylo železničním odborníkem doporučeno aplikovat dobu výhledu *30 minut* na úrovni hlavní simulace. Tato hodnota výhledu realisticky odrážela informovanost dispečerů o budoucím provozu (ohledně očekávaných příjezdů vlaků) v rámci železničních stanic. Pro uplatnění *doby výhledu* v rámci dalších úrovní vnoření rekurzivních simulací byla použita technika *hierarchicky redukováných dob výhledů*. Z experimentálních důvodů byly počítány experimenty i s jinými hodnotami dob výhledů (např. 5, 10 a 15 min.), výsledky těchto experimentů jsou uvedeny souhrnně v příloze B.

Souhrnně tedy lze uvést, že byly stanoveny následující hodnoty parametrů:

- $T_{total} = 225$ hodin,
- $T_{probeMaxLimit} = 25$ hodin,
- $T_{pilotMaxLimit} = 100$ hodin,
- $T_{finalMaxLimit} = 100$ hodin,
- $mainReplCount = 100$ replikací (pro scénáře počítané ve *finální fázi*),
- *stopCond* – uplatnění *doby výhledu* 30 minut (technikou *hierarchicky redukováných dob výhledů*).

9.7 Průzkumná fáze

Cílem průzkumné fáze bylo navrhnout rozdělení dostupných hardwarových výpočetních prostředků na *logické výpočetní jednotky* pro efektivní provádění následujících fází. Bylo tedy potřeba stanovit hodnoty parametrů N_{LPU} a M_{LPURam} . Vzhledem k faktům, že pro výpočty byly k dispozici čtyři nezávislé výpočetní uzly a nástroj *MesoRail* v době provádění studie nebyl schopen efektivně výpočet distribuovat na více nezávislých uzlů, bylo možné už v rámci *průzkumné fáze* uvažovat přítomnost 4 *LPU* a provádět tedy až 4 hlavní replikace najednou. Před zahájením průzkumné fáze bylo třeba stanovit příslušné časové limity pro provádění fáze, jejich hodnoty byly následující:

- $T_{probeMaxLimit} = 25$ hodin,
- $T_{probeReplMaxLimit} = 12$ hodin.

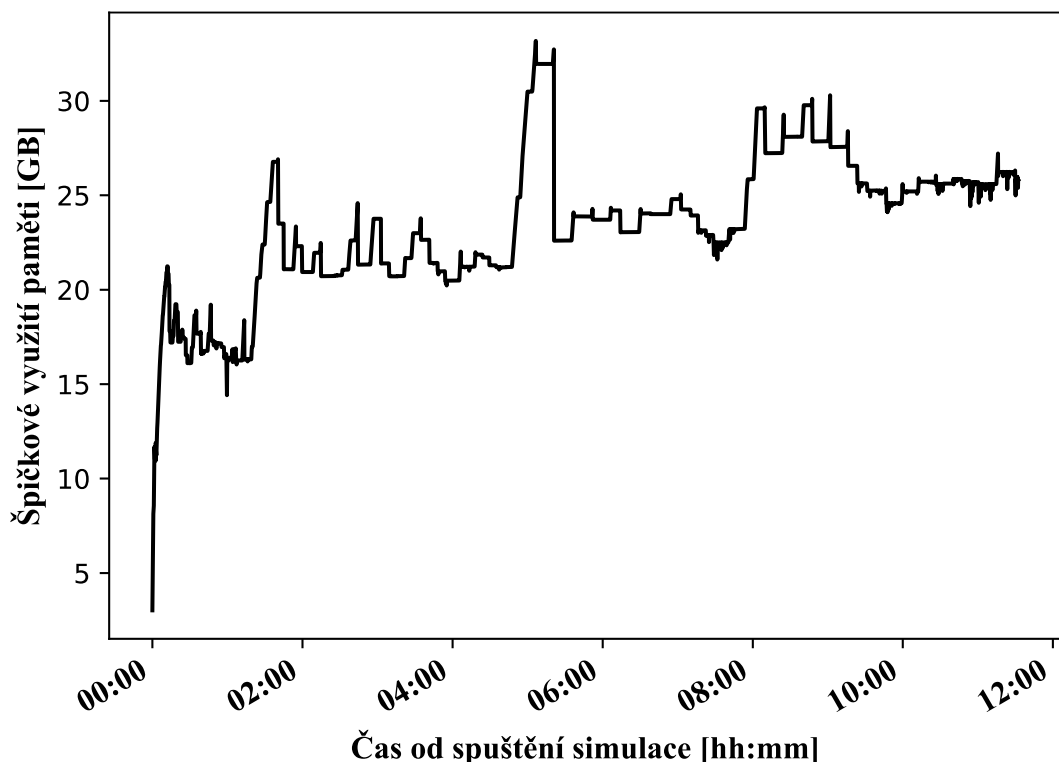
Pro provádění *průzkumné fáze* byla zvolena poměrně vysoká hodnota parametru $T_{probeReplMaxLimit}$, to ale souviselo s možností provádět až 4 replikace najednou a uplatnění dynamického iterativního přístupu k provádění *průzkumné fáze*. Testováno bylo několik různých konfigurací experimentů zahrnujících různou míru vnořování ($maxLevel \leq 3$) a různé počty prováděných replikací vnořených simulací ($nestReplCount \leq 10$). V průběhu *průzkumné fáze* bylo postupováno iterativně od potenciálně nejméně náročných replikací (malé hodnoty $maxLevel$ a $nestReplCount$). Pro každou kombinaci výše uvedených parametrů bylo simulováno několik replikací hlavní simulace. Vždy byla simulována první hlavní replikace ($ReplId = 0$) a poté případně několik dalších replikací, které byly vybrány náhodným losem.

Během průzkumných experimentů byla zaznamenána průměrná špičková paměťová náročnost okolo 32 GB na jednu replikaci hlavní simulace (hlavní replikaci), příklad průběhu špičkové alokace paměti u vybrané replikace ($ReplId = 5$, $nestReplCount = 3$, $maxLevel = 3$) je zobrazen

na obrázku 17. U vybraných hlavních replikací byly zjištěny požadavky, které vyžadovaly alokaci až 46 GB paměti. Přehled dalších základních charakteristik jednotlivých zkoumaných replikací je uveden v tabulce 4, sledované veličiny zahrnují:

- parametr *nestReplCount* (sloupec 1), parametr *maxLevel* (sloupec 2) a číslo hlavní replikace (*Repl#*, sloupec 3) – uvedené hodnoty jednoznačně určují simulovanou hlavní replikaci a její příslušnou *konfiguraci RNS*,
- počet konfliktů v hlavní simulaci ($N_{ConfMain}$, sloupec 4) – počet konfliktních stavů, které se vyskytly při provádění hlavní simulace (a nikoliv na nižších úrovních vnoření),
- celkový počet konfliktů (N_{SimTot} , sloupec 5) – počet všech řešených konfliktních stavů, tj. těch co se vyskytly v hlavní simulaci i na všech nižších úrovních vnoření v rámci řešení *RNS*,
- celkový počet simulací (N_{SimTot} , sloupec 6) – celkový počet simulovaných replikací (odpovídal mohutnosti příslušné množiny ${}^dH_b(S_{0,0,r})$),
- výpočetní doba (t , sloupec 7) – skutečný reálný čas potřebný na provedení výpočtu dané replikace.

U některých replikací došlo k předčasnému ukončení z důvodu dosažení limitní hodnoty $T_{probeReplMaxLimit}$, tyto replikace byly ukončeny bez jejich dopočítání. V průběhu provádění *průzkumné fáze* byly i tyto replikace užitečné a informace o jejich nedopočítání umožnila směřování dalších vybraných *konfigurací RNS* (a jejich replikací) k prozkoumání.



Obrázek 17 – Špičkové využití operační paměti u replikace *ReplId* = 5 (*nestReplCount* = 3, *maxLevel* = 3), zdroj: vlastní

Tabulka 4 – Přehled provedených experimentů v *průzkumné fázi*, zdroj: vlastní

1	2	3	4	5	6	7
Parametr <i>nestReplCount</i> [-]	Parametr <i>maxLevel</i> [-]	Číslo hlavní replikace <i>ReplId</i> [-]	Počet konfliktů v hlavní simulaci <i>N_{ConfMain}</i> [-]	Celkový počet konfliktů <i>N_{ConfTot}</i> [-]	Celkový počet simulací <i>N_{SimTot}</i> [-]	Výpočetní doba <i>t</i> [h]
3	1	0	14	14	105	0,56
		3	10	10	72	0,36
		6	15	15	108	0,53
	2	0	14	327	2 016	3,70
		1	14	138	921	1,67
		7	9	75	516	0,88
	3	0	—*	—*	—*	—*
		5	13	1 496	9 630	11,54
		7	9	730	5 916	4,49
5	1	0	14	14	175	0,63
		8	9	9	110	0,40
		9	8	8	100	0,32
	2	0	14	474	4 925	9,30
		1	14	225	2 515	3,89
		6	16	602	6 290	9,74
	3	0	—*	—*	—*	—*
		1	—*	—*	—*	—*
	10	0	0	14	14	350
3			10	10	240	0,59
5			12	12	260	0,77
1		0	—*	—*	—*	—*
		1	—*	—*	—*	—*
		2	8	543	11 960	10,77
		—	—	—	—	—

* Výpočet nebyl dokončen, protože v jeho průběhu vypršel časový limit $T_{probeReplMaxLimit}$.

Prováděním výše uvedených experimentů byl prakticky spotřebován celý časový limit $T_{probeMaxLimit}$, po provedení všech experimentů bylo provedeno vyhodnocení výsledků a stanovení LPU . Každá logická výpočetní jednotka by měla disponovat systémovými prostředky, které by postačovaly pro simulaci jedné hlavní replikace. Vzhledem k průzkumně naměřenému špičkovému využití paměti (46 GB na hlavní replikaci) a vzhledem k očekávanému provádění i náročnějších konfigurací RNS bylo pro jednu hlavní replikaci rezervováno 96 GB paměti. Tento požadavek vedl k rozdělení clusteru na 16 logických výpočetních jednotek (N_{LPU}). Jedna LPU tedy disponovala 96 GB paměti ($M_{LPU Ram}$) a jedním fyzickým CPU (24 vláken s využitím HT technologie). Na clusteru tedy bylo možné spustit najednou až 16 simulací hlavních replikací, přičemž každá tato simulace disponovala 24 vlákny pro paralelní provádění vnořených simulací.

Pro příslušné výpočty spojené s jedním konkrétním pilotním či finálním simulačním experimentem platilo, že na každou výpočetní jednotku byla alokována jedna odlišná výpočetní úloha reprezentovaná množinou replikací ${}^dH_b(S_{0,0,r})$, kde $r=1 \dots 100$ a hodnoty parametrů d a b byly fixovány ($d \in \{0,1,2,3\}$, $b \in \{0,3,5,7,10,20\}$). To tedy znamená, že jedna výpočetní úloha byla představována výpočtem jedné hlavní replikace včetně všech jejích hierarchicky podřízených replikací vnořených simulací.

9.8 Pilotní fáze

Před provedením *pilotní fáze* bylo nutné specifikovat časové limity pro provádění této fáze a jednotlivých replikací, ty byly stanoveny následujícím způsobem:

- $T_{pilotMaxLimit} = 100$ hodin,
- $T_{pilotReplMaxLimit} = 23$ hodin.

Na základě předcházející *průzkumné fáze* bylo stanoveno logické členění clusteru na 16 LPU, na základě této hodnoty bylo dále stanoveno, aby příslušné *scénáře pilotní fáze* počítaly 16 replikací hlavních simulací ($mainReplCount = 16$).

Před nastavením hodnoty parametru $T_{pilotReplMaxLimit}$ bylo potřebné zvážit, jaké bude mít toto nastavení důsledek pro budoucí *finální fázi*. V této souvislosti byla navržena *maximální doba* reálného času (pracovně označená jako ${}^{max}T_{84}$), která byla přípustná pro realizaci budoucího dopočítání 84 hlavních replikací každého z příslušných *scénářů* v následné *finální fázi*. Vzhledem k vyhrazeným 100 hodinám na výpočty celé *finální fáze* byla v případové studii uvedená doba nastavena takto: ${}^{max}T_{84} = 70$ hod. Motivací pro uvedené nastavení bylo umožnit ve *finální fázi* potenciální realizaci maximálně jednoho relativně časově náročného výpočtu simulačního experimentu podle příslušného *scénáře*. V případě reálného uskutečnění zmíněného „maximalistického“ výpočtu byl zbylý čas *finální fáze* potenciálně určen pro kratší výpočty dalších experimentů využívajících RNS a pro výpočet jednoho *scénáře*, který RNS neaplikoval.

Pro potřeby stanovení hodnoty parametru $T_{pilotReplMaxLimit}$ bylo možné odhad reálného času potřebného pro dopočítání 84 replikací (${}^{estim}T_{84}$) finálního scénáře alternativně vyjádřit i jako funkci závislou na odhadu doby trvání odpovídajícího pilotního výpočtu (${}^{estim}T_{16}$) provádějícího 16 replikací. Pokud byla uplatněna stejná hodnota ${}^{repl}t$ při obecném výpočtu obou zmíněných odhadů dle vztahu (4), pak bylo možné příslušnou funkci vyjádřit jako:

$${}^{estim}T_{84} = 3,125 \times {}^{estim}T_{16} \quad (6)$$

Vzhledem k požadavku ${}^{estim}T_{84} \leq {}^{max}T_{84}$ (${}^{max}T_{84} = 70$ hod.) bylo možné úpravou vztahu (4) vyjádřit odhad doby pilotního výpočtu jako: ${}^{estim}T_{16} \leq (0,32 \times {}^{max}T_{84})$, tj. ${}^{estim}T_{16} \leq 22,4$ hod. V tomto kontextu bylo tedy provedeno příslušné nastavení: $T_{pilotReplMaxLimit} = 23$ hod.

Pro tuto fázi bylo navrženo k prověření 13 kandidátských scénářů, přičemž jeden *scénář* ($Sc-00-00-p$) byl plánován bez využívání vnořených simulací a 12 *scénářů* s využitím RNS Identifikátory *scénářů* využívaly následující značení: $Sc-XX-YY-p$, kde XX a YY vyjadřují

příslušná nastavení parametrů *maxLevel* a *nestReplCount*. Množina *ScSet_{pilot}* obsahovala scénáře:

- bez použití *RNS* – *Sc-00-00-p*,
- s použitím *RNS*:
 - maximálně jedna úroveň vnoření *RNS* – *Sc-01-03-p*, *Sc-01-05-p*, *Sc-01-07-p*, *Sc-01-10-p*, *Sc-01-20-p*,
 - maximálně dvě úrovně vnoření *RNS* – *Sc-02-03-p*, *Sc-02-05-p*, *Sc-02-07-p*, *Sc-02-10-p*, *Sc-02-20-p*,
 - maximálně tři úrovně vnoření *RNS* – *Sc-03-03-p*, *Sc-03-05-p*.

S ohledem na dříve definovanou metodiku byl výše uvedený obsah množiny *ScSet_{pilot}* považován za výchozí stav a s ohledem na samotný průběh fáze bylo možné dle potřeby další scénáře doplnit. Výsledky provedených simulací jsou uvedeny v tabulce 5. V jednotlivých sloupcích jsou uvedeny hodnoty:

- *maxLevel* (sloupec 1) – maximální povolené počty úrovní vnořování *RNS*.
- *nestReplCount* (sloupec 2) – počty replikací vnořených simulací prováděných při prověřování variant řešení konfliktů.
- *mainReplCount* (sloupec 3) – počet provedených hlavních replikací.
- *meanSWDI* \pm *halfWidth* (sloupec 4) – hodnoty indikátoru *meanSWDI* včetně pološířek 95% intervalů spolehlivosti (*halfWidth*) Výpočet hodnot v tomto sloupci byl založen na zpracování dat z hlavních replikací.
- *meanRelatHalfWidth* (sloupec 5) – hodnoty *meanRelatHalfWidth* byly výsledkem zpracování dat získaných ze všech replikací vnořených simulací příslušných scénářů. Dokumentují průměrnou míru přesnosti určení hodnoty ukazatele *meanSWDI* (získávané z vnořených simulací) pro odlišné scénáře.
- *N_{ConfTot}* (sloupec 6) – průměrné počty konfliktů (zahrnující konflikty vzniklé v hlavních replikacích i na nižších úrovních vnoření *RNS*), které byly řešeny pomocí *RNS*.
- *N_{SimTot}* (sloupec 7) – průměrné počty replikací příslušející množinám ${}^dH_b(S_{0,0,r})$, kde $r = 1 \dots 16$ a hodnoty indexů d a b pro jednotlivé scénáře odpovídají hodnotám v prvním a druhém sloupci tabulky. Jako příklad lze uvést scénář *Sc-02-05-p*, kde množiny ${}^2H_5(S_{0,0,r})$, $r = 1 \dots 16$ průměrně obsahovaly 3854 replikací a 363 konfliktů, které byly řešeny s uplatněním *RNS*.
- ${}^{mean}t_{16} \pm halfWidth$ (sloupec 8) – průměrné doby trvání výpočtů jedné hlavní replikace včetně pološířek 95% intervalů spolehlivosti.
- ${}^{real}T_{16}$ (sloupec 9) – naměřené celkové doby uplynulého reálného času potřebného k provedení příslušných scénářů.
- ${}^{estim}T_{84}$ (sloupec 10) – odhady předpokládaných celkových dob reálného času spojených s potenciálním dopočítáním scénářů ve finální fázi.

Výsledky v tabulce 5 ukazují, že pilotní experimenty byly plně spočítány pro 8 scénářů a u 2 scénářů došlo k předčasnému ukončení výpočtů z důvodu vyčerpání časového limitu před dokončením příslušných experimentů. Pro 3 navrhované scénáře nebyl výpočet zahájen, neboť časové náročnosti předešlých scénářů (uplatňujících stejný počet úrovní vnoření, avšak menší

počet replikací u vnořených simulací) již vykazovaly nevyhovující výsledky. Celková časová náročnost *pilotní fáze* činila 79 hodin. Z výsledků bylo patrné, že provádění *scénářů* uplatňujících 3 úrovně vnořování nebylo proveditelné, jelikož jejich časové trvání nevyhovovalo stanovenému časovému limitu pro zpracování výpočtů.

V tabulce 6 jsou uvedeny výsledky dokončených simulačních experimentů podle těch *scénářů*, jejichž pilotní výpočty byly dle uvedených pravidel předčasně zastaveny. Tyto výsledky by se v reálném projektu již nepočítaly a jsou zde uvedeny pouze pro ilustraci.

9.9 Finální fáze

Před provedením *finální fáze* bylo nutné specifikovat časové limity pro provádění fáze a jednotlivých replikací, ty byly stanoveny následujícím způsobem:

- $T_{finalMaxLimit} = 100$ hodin,
- $T_{finalReplMaxLimit} = 70$ hodin.

Nastavení výše uvedených časových limitů bylo převzato z dříve uvedených hodnot.

Po ukončení pilotních experimentů byly pro dopočítání ($ScSet_{final}$) vybrány *scénáře* *Sc-01-20* a *Sc-02-05*. U všech plně dopočítaných pilotních experimentů aplikujících *RNS* byly hodnoty provozního ukazatele *meanSWDI*, které se počítaly z dat hlavních replikací, téměř stejné (včetně příslušných hodnot *halfWidth*). Proto byly při výběru výše zmíněných *scénářů* upřednostňovány nižší hodnoty indikátoru *meanRelatHalfWidth* (odrážející výsledky z výpočtů vnořených simulací zkoumající varianty řešení konfliktů) a dále byly zohledňovány odhadované doby pro finální dopočítání experimentů ($^{estim}T_{84}$). Pro možnost porovnání výsledků ze scénářů aplikujících *RNS* s výsledky ze scénáře, který vnořené simulace nevyužíval, byl dopočítán *scénář* *Sc-00-00*.

Ve výsledcích *finální fáze* (tabulka 7) jsou uvedeny základní hodnoty sledovaných parametrů a obsah sloupců je převzat z tabulek uvedených v rámci *pilotní fáze*. Změny ve výsledkové tabulce *finální fáze* jsou:

- $^{real}T_{84}$ (sloupec 8) – naměřené celkové doby uplynulého reálného času potřebného k provedení příslušných *scénářů*,
- $^{estim}T_{84}$ (sloupec 9), $^{real}T_{16}$ (sloupec 10) – původní odhady dob trvání *scénářů* z předchozí *pilotní fáze* (hodnoty z tabulky 5).

Pro *scénáře* (*Sc-01-20* a *Sc-02-05*) uplatňující *RNS* činila hodnota ukazatele *meanSWDI* cca 25 min. a pro scénář (*Sc-00-00*), který nevyužíval vnořené simulace, cca 33 min. (tabulka 7). Z tohoto pohledu tedy bylo možné konstatovat, že aplikace *RNS* pro podporu rozhodování přinesla provozně lepší výsledky až o 25 % oproti simulacím, které *RNS* nevyužívaly a uplatňovaly při řešení konfliktů pouze statické prioritní seznamy.

V tabulce 8 jsou pro ilustraci uvedeny výsledky *scénářů*, které nebyly po pilotní fázi vybrány pro dopočítání, ale zároveň nebyly posouzeny jako nevyhovující z hlediska doby trvání finálních výpočtů. Zmíněné výsledky by se v reálném projektu již nepočítaly.

Scénář	1	2	3	4	5	6	7	8	9	10
	<i>max</i> <i>Level</i> [–]	<i>nest</i> <i>Repl</i> <i>Count</i> [–]	<i>main</i> <i>Repl</i> <i>Count</i> [–]	<i>meanSWDI</i> ± <i>halfWidth</i> (<i>main</i>) [min]	<i>mean</i> <i>RelatHalf</i> <i>Width</i> (<i>vnoř.</i>) [–]	průměr <i>N_{ConfTot}</i> [–]	průměr <i>N_{SimTot}</i> [–]	<i>mean_{T16}</i> ± <i>halfwidth</i> (<i>main</i>) [hod]	<i>realT₁₆</i> (<i>main</i>) [hod]	<i>estimT₈₄</i> (<i>main</i>) [hod]
<i>Sc-00-00-p</i>	0	0	16	32,5 ± 7,0	–	0	1	0,10 ± 0,00	0,1	0,7
<i>Sc-01-03-p</i>	1	3	16	26,3 ± 6,5	0,13	13	84	0,43 ± 0,06	0,7	3,2
<i>Sc-01-05-p</i>	1	5	16	26,2 ± 6,5	0,08	12	138	0,47 ± 0,08	0,8	3,5
<i>Sc-01-07-p</i>	1	7	16	26,2 ± 6,6	0,06	12	193	0,63 ± 0,11	1,1	4,7
<i>Sc-01-10-p</i>	1	10	16	26,2 ± 6,5	0,05	12	276	0,84 ± 0,14	1,3	6,2
<i>Sc-01-20-p</i>	1	20	16	26,2 ± 6,5	0,03	12	554	1,83 ± 0,26	3,1	13,2
<i>Sc-02-03-p</i>	2	3	16	26,2 ± 6,5	0,08	224	1401	2,72 ± 0,75	5,2	21,8
<i>Sc-02-05-p</i>	2	5	16	26,1 ± 6,6	0,05	363	3854	7,38 ± 2,51	20,7	62,0
<i>Sc-02-07-p</i>	2	7	16	–	–	–	–	–	23,0*	–
<i>Sc-02-10-p</i>	2	10	16	–	–	–	–	–	–**	–
<i>Sc-02-20-p</i>	2	20	16	–	–	–	–	–	–**	–
<i>Sc-03-03-p</i>	3	3	16	–	–	–	–	–	23,0*	–
<i>Sc-03-05-p</i>	3	5	16	–	–	–	–	–	–**	–
Σ									79,0	

* Výpočet nebyl dokončen, protože v jeho průběhu vypršel časový limit $T_{pilotReplMaxLimit}$.

** Výpočet nebyl zahájen, protože na základě dříve spočítaných scénářů (se shodnou hodnotou *maxLevel* a nižší hodnotou *nestReplCount*) lze předpokládat, že výpočetní náročnost daného scénáře bude vyšší a nebude ho rovněž možné vypočítat v definovaném časovém limitu.

Tabulka 5 – Pilotní simulační experimenty aplikující 16 replikací hlavních simulací, zdroj: vlastní

	1	2	3	4	5	6	7	8	9	10
Scénář	<i>max</i> <i>Level</i> [-]	<i>nest</i> <i>Repl</i> <i>Count</i> [-]	<i>main</i> <i>Repl</i> <i>Count</i> [-]	<i>meanSWDI</i> \pm <i>halfWidth</i> (<i>main</i>) [min]	<i>mean</i> <i>RelatHalf</i> <i>Width</i> (<i>vnoř.</i>) [-]	<i>průměr</i> <i>N_{ConfTot}</i> [-]	<i>průměr</i> <i>N_{SimTot}</i> [-]	<i>mean_{T16}</i> \pm <i>halfwidth</i> (<i>main</i>) [hod]	<i>realT₁₆</i> (<i>main</i>) [hod]	<i>estimT₈₄</i> (<i>main</i>) [hod]
	<i>Sc-02-07-p</i>	2	7	16	26,1 ± 6,6	0,04	490	7214	17,9 ± 10,1	80,0
<i>Sc-02-10-p</i>	2	10	16	26,0 ± 6,6	0,03	704	14 791	34,6 ± 13,1	91,8	216,2
<i>Sc-02-20-p</i>	2	20	16	26,0 ± 6,6	0,02	1 394	59 085	185,0 ± 68,3	560,6	1 583,4
<i>Sc-03-03-p</i>	3	3	16	26,2 ± 6,6	0,06	2 881	18 010	31,8 ± 20,1	127,2	324,4
<i>Sc-03-05-p</i>	3	5	16	26,0 ± 6,6	0,04	7 315	74 670	185,6 ± 116,8	741,8	1 890,1

Tabulka 6 – Ilustrační ukázka výsledků dopočítání „neperspektivních“ pilotních scénářů z tabulky 5, zdroj: vlastní

Scénář	1	2	3	4	5	6	7	8	9	10
	<i>max</i> <i>Level</i> [-]	<i>nest</i> <i>Repl</i> <i>Count</i> [-]	<i>main</i> <i>Repl</i> <i>Count</i> [-]	<i>meanSWDI</i> ± <i>halfWidth</i> (<i>main</i>) [min]	<i>mean</i> <i>RelatHalf</i> <i>Width</i> (<i>vnof.</i>) [-]	<i>průměr</i> <i>N_{ConfTot}</i> [-]	<i>průměr</i> <i>N_{SimTot}</i> [-]	<i>realT₈₄</i> (<i>main</i>) [hod]	<i>estimT₈₄</i> (<i>main</i>) [hod]	<i>realT₁₆</i> (<i>main</i>) [hod]
Sc-00-00	0	0	100	32,6 ± 3,2	-	0	1	0,7	0,7	0,1
Sc-01-20	1	20	100	25,2 ± 2,5	0,03	13	567	11,9	13,2	3,1
Sc-02-05	2	5	100	24,6 ± 2,5	0,05	386	4120	56,5	62,0	20,7
Σ								69,1		

Tabulka 7 – Výsledky scénářů z *finální fáze* aplikujících 100 replikací hlavních simulací,
zdroj: vlastní

Scénář	1	2	3	4	5	6	7	8	9	10
	<i>max</i> <i>Level</i> [-]	<i>nest</i> <i>Repl</i> <i>Count</i> [-]	<i>main</i> <i>Repl</i> <i>Count</i> [-]	<i>meanSWDI</i> ± <i>halfWidth</i> (<i>main</i>) [min]	<i>mean</i> <i>RelatHalf</i> <i>Width</i> (<i>vněj.</i>) [-]	<i>průměr</i> <i>N_{ConfTot}</i> [-]	<i>průměr</i> <i>N_{SimTot}</i> [-]	<i>readT₈₄</i> (<i>main</i>) [hod]	<i>estimT₈₄</i> (<i>main</i>) [hod]	<i>readT₁₆</i> (<i>main</i>) [hod]
Sc-01-03	1	3	100	25,4 ± 2,6	0,13	13	84	2,7	3,2	0,7
Sc-01-05	1	5	100	25,2 ± 2,5	0,08	13	140	3,2	3,5	0,8
Sc-01-07	1	7	100	25,2 ± 2,5	0,06	13	197	4,1	4,7	1,1
Sc-01-10	1	10	100	25,3 ± 2,6	0,05	13	283	6,2	6,2	1,3
Sc-02-03	2	3	100	24,9 ± 2,6	0,08	236	1504	24,2	21,8	5,2

Tabulka 8 – Ilustrační ukázka výsledků finálního dopočítání nevybraných pilotních scénářů (pouze „proveditelné“ scénáře), zdroj: vlastní

V souvislosti s prověřovanou verzí zkoumaného systému se spotřebovalo 69 hodin na provedení vybraných finálních experimentů (*Sc-00-00*, *Sc-01-20* a *Sc-02-05*), jimž předcházelo 79 hodin potřebných na výpočty pilotních experimentů a 25 hodin na provedení průzkumných replikací. Celková vyhrazená doba 225 hodin (T_{total}) tedy nebyla překročena.

9.10 Vyhodnocení zjišťování kapacity

Hlavním cílem prováděné simulační studie bylo vyhodnocení *kapacity* zkoumané železniční stanice. Zjištěný výsledek by měl poskytnout zadavateli studie (v případné reálné kapacitní simulační studii) informace o charakteru provozu dle zkoumaného jízdního řádu na příslušné železniční infrastruktuře. Jak již bylo zmíněno v podkapitole 3.1, termín *kapacita* nepředstavuje jednoznačně definovatelnou hodnotu či vlastnost, ale spíše komplexní hodnocení příslušného provozu. V následující části bude provedeno vyhodnocení *kapacity* z pohledu směrnice SM124 (SŽDC, 2019) pro případ uplatnění tzv. extenzivní simulace.

Pro vyhodnocení zjišťování kapacity v rámci případové studie, která uplatňuje mezoskopický simulační přístup, bylo nutné analyzovat *přírůstky zpoždění* pro jednotlivé kategorie vlaků (dálkové, regionální, nákladní). Hodnoty *přírůstků zpoždění* (průměrné hodnoty přepočítané na jeden vlak dané kategorie) pro výsledky z *finální fáze* jsou uvedeny v tabulce 9.

Příslušné *limitní hodnoty* vycházející ze směrnice SM124 jsou uvedeny v tabulce 10. Jelikož v případové studii bylo zkoumáno dvouhodinové období dopravní špičky, tak byly v souladu se směrnicí jednotlivé *limitní hodnoty* navýšeny o 50 %.

Tabulka 9 – *Přírůstky zpoždění* jednotlivých kategorií vlaků pro scénáře z *finální fáze*, zdroj: vlastní

Scénář	Průměrný přírůstek zpoždění na jeden vlak [s]		
	Dálková doprava	Regionální doprava	Nákladní doprava
<i>Sc-00-00</i>	26,38	47,44	-42,35
<i>Sc-01-20</i>	19,93	37,11	-44,80
<i>Sc-02-05</i>	19,71	36,77	-44,85

Tabulka 10 – *Limitní hodnoty* pro určení kvality provozu v období dopravní špičky dle hodnoty přírůstku zpoždění, zdroj: (SŽDC, 2019)

Úroveň kvality provozu	Dálková doprava [s]	Regionální doprava [s]	Nákladní doprava [s]
Optimální	≤ 0	≤ 0	≤ 90
Riziková	0–90	0–180	90–450
Nevyhovující	> 90	> 180	> 450

Na základě výše uvedených údajů bylo možné konstatovat následující fakta o zjištěné kapacitě zkoumané železniční stanice:

- Vlaky dálkové dopravy vykazovaly ve všech zkoumaných scénářích přírůstek zpoždění, který odpovídal *rizikové* kvalitě provozu.
- Vlaky regionální dopravy vykazovaly ve všech zkoumaných scénářích přírůstek zpoždění, který odpovídal *rizikové* kvalitě provozu.
- Vlaky nákladní dopravy vykazovaly ve všech zkoumaných scénářích přírůstek zpoždění, který odpovídal *optimální* kvalitě provozu.

Pro vlaky dálkové a regionální dopravy bylo zjištěno, na základě porovnání příslušných *limitních hodnot* a hodnot ukazatelů *přírůstků zpoždění*, že vykazovaly *rizikovou* kvalitu provozu. I z tohoto důvodu bylo vhodné provést další analýzu, která se zaměřila na jednotlivé vlakové linky.

V tabulce 11 jsou uvedeny hodnoty průměrných *přírůstků zpoždění* (na jeden vlak) pro jednotlivé vlakové linky pro scénáře z *finální fáze*. Z tabulky je možné rychle identifikovat potenciálně nejvíce *rizikové* linky *Os3* a *Os1*. Vyšší hodnoty *přírůstků zpoždění* byly dále spojeny s linkami *Os4* a *R1*. S ohledem na výše uváděné *limitní hodnoty* byly linky *R1*, *R2*, *Os1*, *Os2*, *Os3* a *Os4* hodnoceny jako linky s *rizikovou* kvalitou provozu. Linky *Pn1* a *Pn2* byly vyhodnoceny jako linky s *optimální* kvalitou provozu.

Tabulka 11 – Přírůstky zpoždění jednotlivých vlakových linek pro scénáře z *finální fáze*, zdroj: vlastní

Scénář	Průměrný přírůstek zpoždění na jeden vlak [s]							
	<i>R1</i>	<i>R2</i>	<i>Os1</i>	<i>Os2</i>	<i>Os3</i>	<i>Os4</i>	<i>Pn1</i>	<i>Pn2</i>
<i>Sc-00-00</i>	35,11	17,66	52,57	28,05	82,96	56,25	-8,41	-76,29
<i>Sc-01-20</i>	29,36	10,51	47,98	17,95	68,46	32,67	-16,59	-73,02
<i>Sc-02-05</i>	29,32	10,10	47,94	17,51	68,09	31,85	-16,70	-72,99

Na základě předchozích zjištění by v případě provádění reálné kapacitní studie mohlo být požadováno provedení další analýzy pro vyhodnocení *přírůstků zpoždění* pro jednotlivé vlaky. Výsledky takové analýzy by mohly identifikovat konkrétní *rizikové* vlaky.

Provedením kapacitní studie bylo možné identifikovat potenciálně „problematické“ vlakové linky, které ve zkoumaném provozu a dané železniční infrastruktuře mohou vykazovat nárůst zpoždění vlaků. U *rizikové* kvality provozu lze očekávat, že vlaky nebudou schopny zpoždění snižovat, ale naopak bude docházet k jeho navýšení (SŽDC, 2019).

Provedená provozní analýza *kapacity* železniční stanice s využitím počítačové simulace a metody *reflektivních vnořených simulací* umožnila zjistit řadu informací o *kapacitě* a charakteru provozu ve zkoumané železniční stanici. Jak již bylo dříve uvedeno, provedená studie neumožňuje popsat každý aspekt *kapacity* příslušné železniční infrastruktury, pro použití v praxi lze očekávat, že uvedená metodika bude kombinována s dalšími přístupy pro zjišťování *kapacity* (např. s uplatněním analytických metod).

9.11 Vyhodnocení výsledků v porovnání s nástrojem Villon

Nad rámec *finální fáze* bylo provedeno další posouzení výsledků z výše uvedených experimentů vzhledem k výsledkům ze simulací provedených v etablovaném simulátoru *Villon*. Jak již bylo zmíněno v podkapitole 9.4, příslušný simulující systém byl nakonfigurován stejně jako v nástroji *MesoRail*. Pro podporu rozhodování byly ve stochastických simulacích používány prioritní seznamy. Po provedení 100 replikací dosahoval $meanSWDI \pm halfWidth$ hodnot: $32,1 \pm 3,7$ min. Výsledná hodnota $meanSWDI$ se liší od odpovídajícího výsledku scénáře *Sc-00-00*, jenž byl prováděn v nástroji *MesoRail*, o méně než 2 %. Výsledek tohoto porovnání vypovídá o velmi dobré přesnosti, se kterou nástroj *MesoRail* počítá. Komparace s výsledky ze scénářů *Sc-01-20* a *Sc-02-05* svědčí o lepším optimalizačním přístupu v nástroji *MesoRail* (při uplatňování *RNS*), který dosáhl cca o 22 % lepší hodnotu ukazatele $meanSWDI$ než model vybudovaný v nástroji *Villon*.

Uvedené skutečnosti svědčí o opodstatněnosti snah o další zlepšování modelovacích možností simulátorů železničního provozu, a to například prostřednictvím aplikace konceptu *RNS* pro podporu rozhodování.

10 ZÁVĚR

V rámci zpracování disertační práce byly zadané cíle splněny. V první zásadní části práce byla detailně popsána a formalizována technika *reflektivních vnořených simulací (RNS)*. Technika *RNS* představuje obecně známý princip řešení rozhodování v rámci počítačových simulací. V práci byl proveden důsledný popis a formalizace metodiky aplikující techniku *RNS* v obecné rovině, bez vazby na konkrétní aplikační doménu.

Primárním cílem této práce byl návrh inovativní metodiky pro určování kapacity železničních stanic s využitím počítačových simulací a techniky *RNS*. V práci byla navržena a detailně popsána metodika, která staví na praktickém přístupu provádění kapacitních studií, kdy je dopředu definován časový limit, v jehož rámci je nutné provést zjišťování kapacity. Vzhledem k vyšším výpočetním nárokům metody *RNS* (v porovnání s běžně užívanou metodou prioritního plánování) metodika nabízí možnost efektivního využití dostupných hardwarových prostředků a provádění výpočtů několika hlavních replikací paralelně. Použití metody *RNS* umožňuje vylepšení rozhodování v konfliktních situacích, a umožňuje se tak přiblížit rozhodování skutečných dispečerů v provozu. V důsledku metoda nabízí výsledky, které mohou lépe odrážet běžnou realitu provozu.

Pro vlastní otestování metodiky a metody *RNS* byl navržen a implementován původní simulační nástroj *MesoRail*. Jedná se simulátor železničního provozu pracující na mezoskopické úrovni podrobnosti. Pro podporu rozhodování uplatňuje metody prioritního plánování a *RNS*. Veškeré experimenty využívající metodu *RNS* byly prováděny s užitím nástroje *MesoRail*.

Navržená metodika byla ověřena na případové studii, která vycházela z dříve realizované kapacitní studie žst. Zdice bez uplatnění techniky vnořených simulací a nástroje *MesoRail*. V rámci případové studie byl realizován totožný systém a s pomocí navržené metodiky bylo provedeno zjišťování kapacity. V souvislosti s prováděnou případovou studií bylo realizováno větší množství dalších experimentů za účelem zjištění charakteristik metody *RNS*, tyto výsledky jsou uvedeny v příloze práce.

10.1 Perspektivy dalšího vývoje

Praktickému nasazení techniky *RNS* v oblasti operativního plánování aktuálně brání poměrně značná výpočetní náročnost, která je spojena s prováděním velkého množství vnořených simulací. Toto téma bylo blíže popsáno v rámci podkapitoly 6.2, byly zde rovněž popsány autorem navrhované (heuristické) metody, které mají potenciál snížit množství prováděných vnořených simulací a snížit tak výpočetní náročnost metody *RNS*. Do této chvíle byly provedeny pouze velmi základní experimenty, které vyhodnocovaly možnou teoretickou úsporu v počtu potřebných vnořených simulací. Experimenty, které by jasně vyhodnotily, jaký přínos ke snížení výpočetní náročnosti by tyto navrhované metody (nebo i jiné metody) měly, tak zůstávají možností budoucího výzkumu.

V rámci snižování výpočetní náročnosti, resp. zefektivnění simulačních výpočtů, lze dále rozšířit simulační nástroj *MesoRail* o přímou podporu simulace s využitím cloudu (či výpočetního clusteru). Z pohledu terminologie počítačových simulací by se tak mohla

realizovat období distribuované simulace. Jednotlivé vnořené simulace by mohly být distribuovány na volné výpočetní uzly v cloudu. Realizace takových výpočtů by vyžadovala podporu pro přenos stavu simulace mezi jednotlivými uzly a realizaci komponenty manažera, který by organizoval rozdělování výpočtů jednotlivých vnořených simulací na výpočetní uzly.

V průběhu vývoje simulačního nástroje *MesoRail* a návrhu metodiky byly využívány i další výpočetní clustery: gridová infrastruktura *MetaCentrum* (MetaCentrum VO, b. r.) a výpočetní cluster *Salomon* (IT4Innovations, 2020). Výpočty *RNS* byly úspěšně aplikovány i v těchto výpočetních prostředích a jejich využití je plánováno i v rámci budoucího vývoje.

Simulátor *MesoRail* je dále vyvíjen, v rámci projektu *Softwarová simulační podpora pro určování kapacity infrastruktury železničních stanic* (grant TAČR ZÉTA, realizace 2019–2021), jehož je autor disertační práce hlavní řešitel. Nástroj je dále rozšiřován a modifikován do podoby použitelné v komerčním prostředí za účelem provádění kapacitních studií provozu železničních stanic.

BIBLIOGRAFIE

- ADAMKO, Norbert a Valent KLIMA, 2008. Optimisation of railway terminal design and operations using villon generic simulation model. *Transport*. Vilnius Gedminas Technical University, **23**(4), 335-340. DOI: 10.3846/1648-4142.2008.23.335-340.
- ALEXANDER, Fay a Eckehard AND SCHNIEDER, 1999. Knowledge-Based Decision Support System for Real-Time Train Traffic Control. In: *Computer-Aided Transit Scheduling*. Berlin, Heidelberg: Springer Berlin Heidelberg, s. 347-370. ISBN 978-3-642-85970-0.
- BAI, Lijie, Thomas BOURDEAUD'HUY, Besoa RABENASOLO a Emmanuel CASTELAIN, 2014. A Mixed-Integer Linear Program for Routing and Scheduling Trains through a Railway Station. In: *Proceedings of the 3rd International Conference on Operations Research and Enterprise Systems*. SCITEPRESS - Science and and Technology Publications, s. 445-452. DOI: 10.5220/0004863104450452. ISBN 978-989-758-017-8. Dostupné z: <http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0004863104450452>
- BANDINI, S., S. MANZONI a G. VIZZARI, 2009. Agent based modeling and simulation: An informatics perspective: An informatics perspective. *JASSS*. **12**(4).
- BARCELÓ, Jaume, 2010. *Fundamentals of traffic simulation*. New York: Springer. ISBN 978-144-1961-426.
- BAŽANT, Michael a Antonín KAVIČKA, 2009. Artificial neural network as a support of platform track assignment within simulation models reflecting passenger railway stations. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*. **223**(5), 505-515. DOI: 10.1243/09544097JRRT268. Dostupné z: <http://journals.sagepub.com/doi/10.1243/09544097JRRT268>
- BAŽANT, Michael, Antonín KAVIČKA, Roman DIVIŠ a Michal VARGA, 2019. Simulation-Based Rail Traffic Optimizations Applying Multicriterial Evaluations of Variants. *MENDEL*. Brno University of Technology, **25**(1), 139-146. DOI: 10.13164/mendel.2019.1.139.
- BONTÉ, Bruno, Raphaël DUBOZ, Gauthier QUESNEL a Jean-Pierre MULLER, 2009. Recursive simulation and experimental frame for multiscale simulation. In: *Proceedings of the 2009 Summer Computer Simulation Conference*. Society for Modeling & Simulation International, s. 164-172.
- BURGHOUT, Wilco, 2004. *Hybrid microscopic-mesosopic traffic simulation*. Dostupné z: <http://www.diva-portal.se/smash/get/diva2:14700/FULLTEXT01.pdf>
- ČD, 1965. *Směrnice SŽDC (ČD) D24 pro zjišťování kapacity železničních tratí*. Praha: České Dráhy, a.s, 40 s.
- DE LUCA CARDILLO, Dorotea a Nicola MIONE, 1998. K L-list λ colouring of graphs. *European Journal of Operational Research*. North-Holland, **106**(1), 160-164. DOI: 10.1016/S0377-2217(98)00299-9. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0377221798002999?via%3Dihub>
- DENG, Wen-tan, Hui-bing ZHAO a Hui HE, 2018. Research on Heavy Haul Railway Dispatching System Based on Fuzzy Expert System. *DEStech Transactions on Computer Science and Engineering*. DEStech Publications, (). DOI: 10.12783/dtcse/ccnt2018/24719.

- DIVIŠ, R. a A. KAVIČKA, 2014. Train movement dynamics within anylogic tool. In: *26th European Modeling and Simulation Symposium, EMSS 2014*. s. 307-312. ISBN 9788897999324.
- DIVIŠ, R. a A. KAVIČKA, 2015. Design and development of a mesoscopic simulator specialized in investigating capacities of railway nodes. In: *27th European Modeling and Simulation Symposium, EMSS 2015*. s. 52-57. ISBN 9788897999485.
- DIVIŠ, R. a A. KAVIČKA, 2016. The method of nested simulations supporting decision-making process within a mesoscopic railway simulator. In: *28th European Modeling and Simulation Symposium, EMSS 2016*. s. 100-106. ISBN 9788897999683.
- DIVIŠ, R. a A. KAVIČKA, 2017. Nested simulations supporting traffic optimizations related to railway stations. In: *29th European Modeling and Simulation Symposium, EMSS 2017, Held at the International Multidisciplinary Modeling and Simulation Multiconference, I3M 2017*. s. 135-142. ISBN 9781510847651.
- DIVIŠ, R. a A. KAVIČKA, 2019. Computational optimizations of nested simulations utilized for decision-making support. In: *31st European Modeling and Simulation Symposium, EMSS 2019*. s. 80-89. ISBN 9788885741263.
- EnerGplan: Simulation Tool, 1997. *Bombardier* [online]. [cit. 2017-11-24]. Dostupné z: <http://www.bombardier.com/content/dam/Websites/bombardiercom/supporting-documents/BT/Bombardier-Transportation-ECO4-EnerGplan.pdf>
- FAY, Alexander, 2000. Fuzzy knowledge-based system for railway traffic control. *Engineering Applications of Artificial Intelligence*. Elsevier Science Ltd, **13**(6), 719-729. DOI: 10.1016/S0952-1976(00)00027-0.
- GILMER, J.B. a F.J. SULLIVAN, 1999. Multitrajectory simulation performance for varying scenario sizes [combat simulation]. In: *Winter Simulation Conference Proceedings*. Phoenix, AZ, USA: Institute of Electrical and Electronics Engineers (IEEE), s. 1137-1146. DOI: 10.1109/wsc.1999.816832.
- GILMER, John a Frederick SULLIVAN, 2000. Recursive simulation to aid models of decisionmaking. In: *Winter Simulation Conference Proceedings*. s. 958-963. DOI: 10.1109/wsc.2000.899898.
- GORDY, Michael a Sandeep JUNEJA, 2010. Nested simulation in portfolio risk measurement. *Management Science*. **56**(10), 1833-1848. DOI: 10.1287/mnsc.1100.1213.
- HERMES Rail Simulation, 2017. *Graffica - Vision Accomplished* [online]. [cit. 2017-12-01]. Dostupné z: <http://www.graffica.co.uk/rail/>
- CHENG, Yung-Hsiang a Li-An YANG, 2009. A Fuzzy Petri Nets approach for railway traffic control in case of abnormality: Evidence from Taiwan railway system: Evidence from Taiwan railway system. *Expert Systems with Applications*. Pergamon, **36**(4), 8040-8048. DOI: 10.1016/J.ESWA.2008.10.070. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0957417408007732?via%3Dihub>
- CHUNG, Edward a Andre-Gilles DUMONT, ed., 2009. *Transport Simulation: Beyond Traditional Approaches*. First edition. EPFL Press, 213 s. ISBN 9781420095098.
- IT4Innovations* [online], 2020. [cit. 2020-08-23]. Dostupné z: <https://www.it4i.cz/>

- JÁNOŠÍKOVÁ, Ľudmila, Antonín KAVIČKA a Michael BAŽANT, 2014. Optimal operation scheduling and platform track assignment in a passenger railway station. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*. **228**(3), 271-284. DOI: 10.1177/0954409712472275. Dostupné z: <http://journals.sagepub.com/doi/10.1177/0954409712472275>
- KETTNER, Michael, Bernd SEWCYK a Carla EICKMANN, 2003. Integrating microscopic and macroscopic models for railway network evaluation. In: *Proceedings of the European Transport Conference (ETC)*. Strasbourg, France: Association for European Transport.
- KINDLER, Eugene, 2006. Nesting Simulating Agents In SIMULA. In: *20th European Conference on Modelling and Simulation*. DOI: 10.7148/2006-0526.
- KINDLER, Eugene, 2010. Nested models implemented in nested theories. In: *12th WSEAS International Conference on AUTOMATIC CONTROL, MODELLING & SIMULATION*. s. 150-159.
- KOMAYA, K. a T. FUKUDA, 1990. ESTRAC-III: An Expert System for Train Traffic Control in Disturbed Situations: An Expert System for Train Traffic Control in Disturbed Situations. *IFAC Proceedings Volumes*. Elsevier, **23**(2), 147-153. DOI: 10.1016/S1474-6670(17)52664-6. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1474667017526646?via%3Dihub>
- KOMAYA, Kiyotoshi, 1991. A new simulation method and its application to knowledge-based systems for railway scheduling. In: *Technical Papers - IEEE/ASME Joint Railroad Conference*. Publ by IEEE, s. 59-66. DOI: 10.1109/rcon.1991.160928. ISBN 0780300726.
- KRIVÝ, Ivan a Eugene KINDLER, 2006. Terminology of Nested Simulation Models. In: *International Conference on Computer Systems and Technologies - CompSysTech' 2006*.
- KŘIVÝ, Ivan a Evžen KINDLER, 2003. *Simulace a modelování 1*. Ostravská Univerzita.
- LARROCHE, Y., R. MOULIN a D. GAUYACQ, 1994. Sepia: A Real-Time Expert System That Automates Train Route Management: A Real-Time Expert System That Automates Train Route Management. *IFAC Proceedings Volumes*. Elsevier, **27**(12), 977-982. DOI: 10.1016/S1474-6670(17)47601-4. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1474667017476014?via%3Dihub>
- LUKS – *Analysis of lines and junctions* [online], 2011. VIA Consulting & Development GmbH [cit. 2017-11-42]. Dostupné z: <http://www.via-con.de/en/development/luks>
- MetaCentrum VO* [online], b. r. [cit. 2020-08-23]. Dostupné z: <https://metavo.metacentrum.cz/cs/>
- MINISTERSTVO DOPRAVY, 2019. Kapitola 5.1.1. Mezioborové srovnání přepravních výkonů osobní dopravy. *Dopravní statistika* [online]. [cit. 2020-06-01]. Dostupné z: https://www.sydos.cz/cs/rocnka-2018/rocnka/htm_cz/cz18_511000.html
- MOLKOVÁ, Tatiana, Vlastislav MOJŽÍŠ, Pavel DRDLA, Josef BULÍČEK a Pavel MAZAČ, 2010. *Kapacita železničních tratí*. 158 s.
- NASH, A. a D. HUERLIMANN, 2004. Railroad Simulation Using OpenTrack. *WIT Transactions on The Built Environment*. WIT Press, **74**. DOI: 10.2495/CR040051. ISBN 978-1-85312-715-1.

- NOVOTNY, R. a A. KAVICKA, 2015. Model of a railway infrastructure as a part of a mesoscopic traffic simulator. In: *27th European Modeling and Simulation Symposium, EMSS 2015*. s. 120-125. ISBN 9788897999485.
- NUZZOLO, Agostino a William LAM, 2017. *Modelling Intelligent Multi-Modal Transit Systems*. Boca Raton, Florida: CRC Press. ISBN 9781315351988.
- RailML.org* [online], 2002. c2002-2017 [cit. 2017-11-26]. Dostupné z: <https://www.railml.org>
- RailSys* [online], 2017. RMCon [cit. 2017-09-12]. Dostupné z: <http://www.rmcon.de/railsys-en/>
- Railway Simulation, 2017. *OpenTrack Railway Technology* [online]. [cit. 2017-09-12]. Dostupné z: http://www.opentrack.cz/opentrack_cz.html
- Railway Traction Power Analysis: Rail Power System Software | eTrax, 2017. *ETAP* [online]. [cit. 2017-11-24]. Dostupné z: <https://etap.com/solutions/S/railways>
- RICHTLINIE 405, 2008. *Richtlinie 405 – Fahrwegkapazität*. Frankfurt am Main: DB Netz, Deutsche Bahn Gruppe.
- STANDRIDGE, Charles a Duane STEWARD, 2000. Using Expert Systems for Simulation Modeling of Patient Scheduling. *SIMULATION*. **75**(3), 148-156. DOI: 10.1177/003754970007500303. Dostupné z: <http://journals.sagepub.com/doi/10.1177/003754970007500303>
- SŽDC, 2019. *SŽDC SM124 - Zjišťování kapacity dráhy*.
- TAN, C, L WAHIDIN, S KHALIL, N TAMALDIN, J HU a G RAUTERBERG, 2016. The Application of Expert System: A Review of Research and Applications: A Review of Research and Applications. *ARNP Journal of Engineering and Applied Sciences*. **11**(4), 2448-2453. Dostupné z: www.arnpjournals.com
- UIC406, 2004. *Kodex UIC 406 - Kapacita*. 1. vydání. Paříž: UIC International Union of Railways.
- UNIVERZITA PARDUBICE, FAKULTA ELEKTROTECHNIKY A INFORMATIKY, 2016. *Závěrečná zpráva k řešení projektu: Návrh nové metodiky propustnosti železničních stanic*. Pardubice.
- Villon, 2016. *Simcon* [online]. [cit. 2017-09-12]. Dostupné z: <http://www.simcon.sk/sk/tools/villon>

SEZNAM PUBLIKACÍ AUTORA

Článek v recenzím řízení (odborný časopis)

DIVIŠ, Roman a Antonín KAVIČKA. Railway-traffic models applying reflective nested simulations. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*. 2020. ISSN 1049-3301. Aktuálně v recenzním řízení.

Článek ve sborníku

DIVIŠ, R., KAVIČKA, A. Computational optimizations of nested simulations utilized for decision-making support. In *31st European Modeling and Simulation Symposium, EMSS 2019*. Janov: Dime University of Genoa, 2019. s. 80-89 s. ISBN 978-88-85741-26-3.

BAŽANT, Michael, Antonín KAVIČKA, Roman DIVIŠ a Michal VARGA. Simulation-Based Rail Traffic Optimizations Applying Multicriterial Evaluations of Variants. *MENDEL* [online]. 2019, **25**(1), 139-146 [cit. 2019-09-09]. DOI: 10.13164/mendel.2019.1.139. ISSN 2571-3701. Dostupné z: <https://mendel-journal.org/index.php/mendel/article/view/90>

BRANDEJSKY, Tomas a Roman DIVIŠ. Scalability of GPA-ES Algorithm. MATOUŠEK, Radek, ed. *Recent Advances in Soft Computing* [online]. Cham: Springer International Publishing, 2019, 2019-08-05, s. 124-133 [cit. 2019-09-09]. *Advances in Intelligent Systems and Computing*. DOI: 10.1007/978-3-319-97888-8_11. ISBN 978-3-319-97887-1. Dostupné z: http://link.springer.com/10.1007/978-3-319-97888-8_11

DIVIŠ, R., KAVIČKA, A. Nested simulations supporting traffic optimizations related to railway stations. In *29th European Modeling and Simulation Symposium, EMSS 2017*. Rende : CAL-TEK SRL, 2017, s. 135-142. ISBN 978-1-5108-4765-1.

DIVIŠ, R., KAVIČKA, A. The method of nested simulations supporting decision-making process within a mesoscopic railway simulator. In *28th European Modeling and Simulation Symposium, EMSS 2016*. Janov : University of Genoa, 2016, s. 100-106. ISBN 978-88-97999-68-3.

DIVIŠ, R., KAVIČKA, A. Design and development of a mesoscopic simulator specialized in investigating capacities of railway nodes. In *27th European Modeling and Simulation Symposium, EMSS 2015*. Janov : University of Genoa, 2015, s. 52-57. ISBN 978-88-97999-48-5.

DIVIŠ, R., KAVIČKA, A. Train movement dynamics within AnyLogic tool. In *Proceedings of the European Modeling and Simulation Symposium 2014*. Genova : Università di Genova, 2014, s. 307-312. ISBN 978-88-97999-32-4.

Software

DIVIŠ, R. Mesoskopický simulátor železniční dopravy MesoRail se základní podporou řízení provozu s podporou rozhodování pomocí vnořených simulací. Software. 2016.

DIVIŠ, R. Knihovna TrainDyn pro podporu dynamiky jízdy vlaku v nástroji AnyLogic. Software. 2014.

Výzkumná zpráva

KAVIČKA, A., BAŽANT, M., RAK, J., DIVIŠ, R. *Optimalizace nakládky prázdných obalů*. ŠKODA AUTO a.s. Výzkumná zpráva. 2017.

KAVIČKA, A., BAŽANT, M., VESELÝ, P., BULÍČEK, J., DIVIŠ, R. *Návrh nové metodiky propustnosti železničních stanic – závěrečná zpráva 2016*. Správa železniční dopravní cesty, s.o. Výzkumná zpráva. 2016.

PŘÍLOHY

Příloha A – Popis infrastruktury a zkoumané verze systému případové studie.....	100
Příloha B – Výsledky všech provedených experimentů	106

PŘÍLOHA A – POPIS INFRASTRUKTURY A ZKOUMANÉ VERZE SYSTÉMU PŘÍPADOVÉ STUDIE

Případová studie je založena na reálné infrastruktuře železniční stanice Zdice a jejího nejbližšího okolí (schematicky zjednodušeně ilustrovaná na obrázku PA.1), infrastruktura zahrnuje:

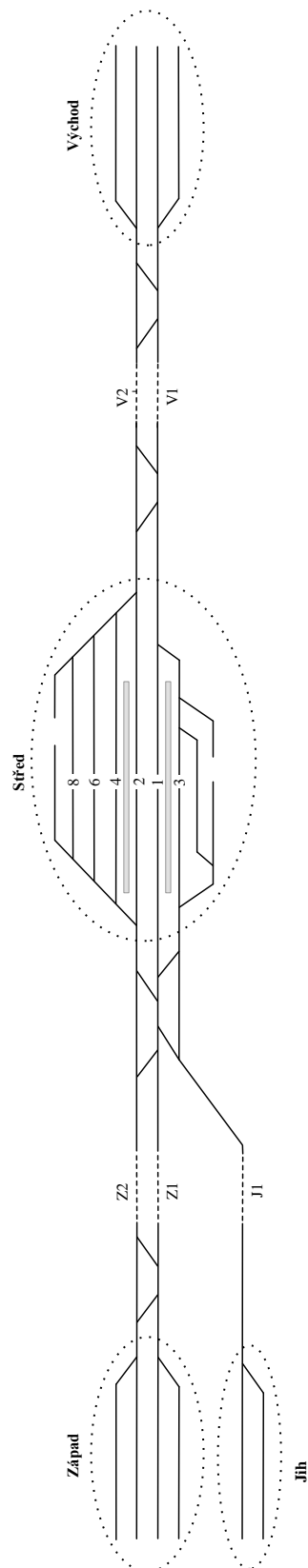
- Kolejiště výše uvedené osobní železniční stanice, která bude pro potřeby této studie označována jako *Střed*. V rámci kolejiště jsou pro osobní vlaky využívána zejména dvě nástupiště: *nástupiště 2* (s přiléhajícími kolejemi #1 a #3), *nástupiště 3* (s nástupištními kolejemi #2 a #4). Obě nástupiště mají délku 300 m.
- Trať vedoucí do sousedních hraničních stanic označovaných jako *Jih* (původní název stanice Lochovice), *Západ* (původní název stanice Hořovice) a *Východ* (původní název stanice Beroun). Do stanice *Jih* vede jednokolejná trať a do stanic *Západ* a *Východ* tratě dvoukolejné (celková vzdálenost mezi hraničními stanicemi *Západ* a *Východ* činí cca 20 km).
- Zjednodušená kolejiště hraničních stanic *Jih*, *Západ* a *Východ*.

Model infrastruktury odpovídá stavu kolejiště z roku 2016 s tím, že všechny přilehlé stanice jsou modelovány zjednodušeným způsobem, každá traťová kolej je rozvětvena do dvou staničních kolejí ukončených zarážedlem (obr. PA.2).

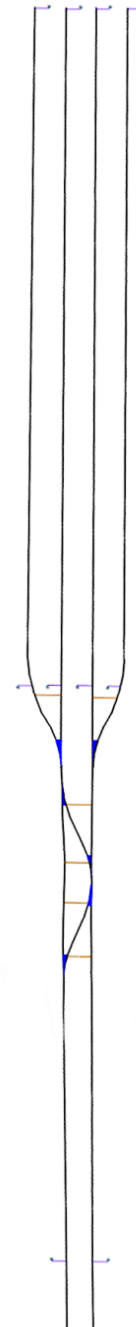
Jednotlivé prvky, sklonové a rychlostí poměry jsou uvedeny vzhledem ke vzdálenosti k referenčnímu bodu – návěstidlu **S1**. Vzdálenost je měřena po jednotlivých kolejových úsecích a představuje skutečnou vzdálenost, kterou musí vlak ujet od referenčního bodu na dané místo na trati.

Polohy a typy návěstidel v modelu infrastruktury odpovídají stavu návěstidel dle skutečnosti.

V následujících částech přílohy jsou uvedeny detailní informace o charakteristice infrastruktury ve zkoumané stanici a jejím okolí (tabulky PA.1 – PA.6). Popis se dále věnuje informacím o aplikovaných zabezpečovacích zařízeních a technologických dobách. Uvedeny jsou rovněž informace týkající se jednotlivých vlakových linek a jejich parametrů (tabulky PA.7 – PA.8).



Obrázek PA.1 – Přehledové zobrazení kolejové infrastruktury zkoumaného systému, zdroj: vlastní



Obrázek PA.2 – Ukázka infrastruktury přilehlé stanice, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Sklonové poměry

Tabulka PA.1 – Sklonové poměry směr *Východ*, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Od [km]	Do [km]	Sklon [‰]
0	0,183	-2,6
0,184	0,827	-5,5
0,828	1,431	-3,66
1,432	1,888	-6,66
1,889	2,543	-3,4
2,544	3,347	-4,19
3,348	4,187	-4,94
4,188	4,728	-3,29
4,729	5,103	-5,73
5,104	5,968	0
5,969	7,911	-1,7
7,912	8,804	0

Tabulka PA.2 – Sklonové poměry směr *Západ*, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Od [km]	Do [km]	Sklon [‰]
0	0,051	-2,6
0,052	0,306	0
0,307	0,686	-2,5
0,687	0,917	-6
0,918	1,715	-10,4
1,716	2,346	-10,8
2,347	3,256	-9,8
3,257	4,288	-10
4,289	4,737	-9,4
4,738	5,359	-10,6
5,36	7,556	-10
7,557	9,106	-11,3
9,107	9,787	-9,2
9,788	10,68	-10

Tabulka PA.3 – Sklonové poměry směr *Jih*, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Od [km]	Do [km]	Sklon [‰]
0	0,051	-2,6
0,052	0,306	0
0,307	0,686	-2,5
0,687	0,917	-6
0,918	1,118	-9
1,119	1,27	-8
1,271	1,881	-11,1
1,882	2,18	-3,3
2,181	3,2	-6,6
3,201	4,6	-6,7
4,601	4,985	-3,4
4,986	5,484	-9,7
5,485	6,587	-11,8
6,588	8,455	-10,1
8,456	8,997	0

Rychlostní poměry – uvažován pouze jeden rychlostní profil „N“

Tabulka PA.4 – Rychlostní poměry směr *Východ*, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Od [km]	Do [km]	Rychlost [km/h]
0	8,804	120

Tabulka PA.5 – Rychlostní poměry směr *Západ*, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Od [km]	Do [km]	Rychlost [km/h]
0	1,198	120
1,199	2,834	105
2,835	10,68	120

Tabulka PA.6 – Rychlostní poměry směr *Jih*, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Od [km]	Do [km]	Rychlost [km/h]
0	0,917	120
0,918	1,351	50
1,352	1,753	70
1,754	8,149	75
8,15	8,997	60

Vstupy pro výpočet technologických dob

- Traťové zabezpečovací zařízení:
 - *Východ* – automatický blok,
 - *Západ* – automatický blok,
 - *Jih* – automatické hradlo.
- Staniční zabezpečovací zařízení – elektronické zabezpečovací zařízení:
 - Doba přípravy vlakové/posunové cesty: 11 s.
 - Doba rušení vlakové/posunové cesty:
 - zhlaví s uvažováním výhybky v posledním úseku: 6 s,
 - koleje: 3 s.

Charakteristiky scénáře a vlaků

V tabulce PA.7 jsou uvedeny další charakteristiky jednotlivých vlakových linek, které nebyly uvedeny v podkapitole 9.2. Hodnota zpomalení při použití brzdy činí pro:

- vlaky osobní dopravy – 0,55 m/s,
- vlaky nákladní – 0,3 m/s.

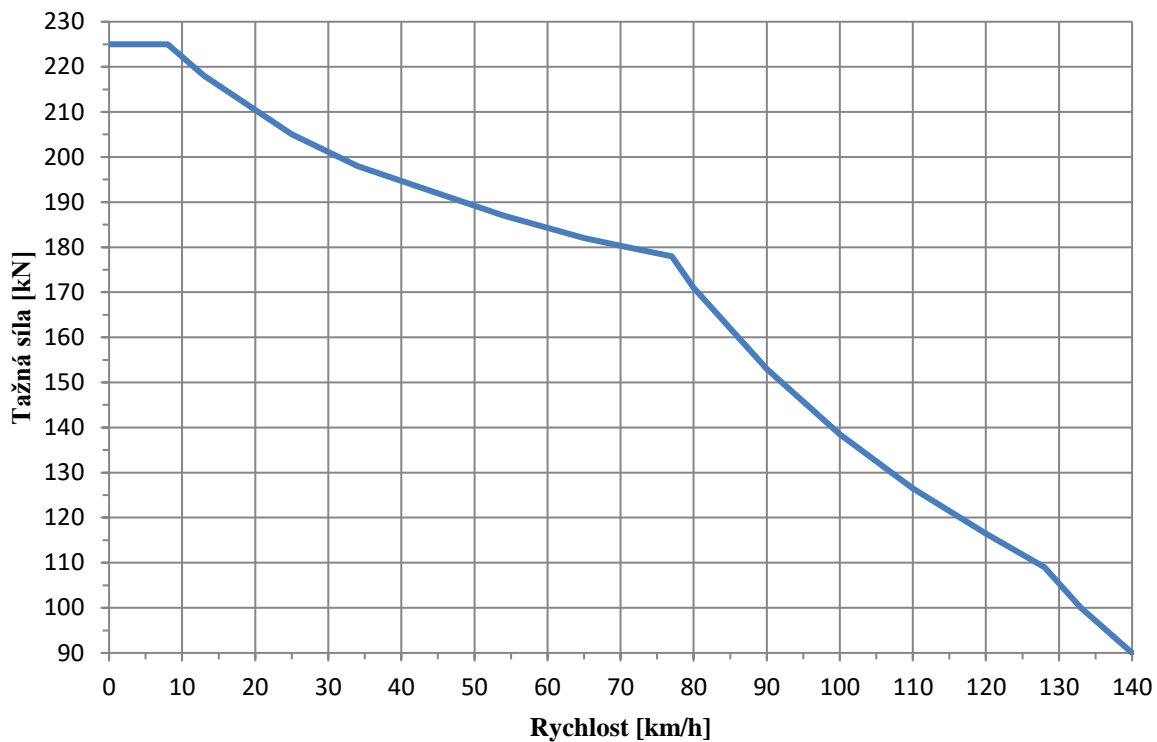
Ve scénáři jsou použity dva druhy hnacích vozidel, jejich základní charakteristiky jsou uvedeny v tabulce PA.8.

Tabulka PA.7 – Další charakteristiky vlakových linek, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

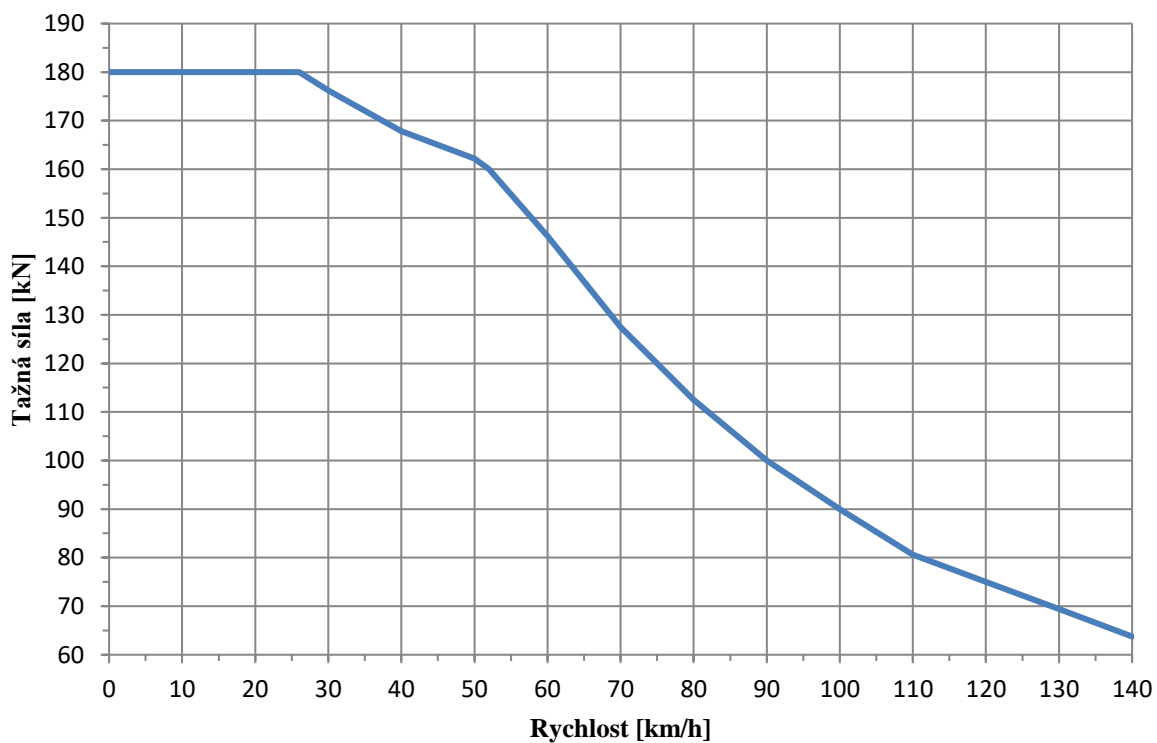
Linka	Počet lokomotiv / počet vagónů	Délka vlaku	Příjezd prvního vlaku do modelu	Standardní rychlost [km/h]	Maximální povolená rychlost vlakové soupravy [km/h]
<i>R1</i>	1 / 7	202 m	07:52:48	115, 115	120
<i>R2</i>	1 / 7	202 m	07:53:20	115, 115	120
<i>Os1</i>	2 / 4	158 m	07:58:47	115, 115	120
<i>Os2</i>	2 / 4	158 m	07:59:09	115, 115	120
<i>Os3</i>	1 / 2	79 m	08:03:47	115, 70	140
<i>Os4</i>	1 / 2	79 m	08:01:27	70, 115	140
<i>Pn1</i>	1 / 22	336 m	08:12:57	80	120
<i>Pn2</i>	1 / 22	336 m	08:13:01	80	120

Tabulka PA.8 – Charakteristiky hnacích vozidel, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

Typ lokomotivy	Délka [m]	Hmotnost [t]	Max. rychlost [km/h]	Linky	Trakční charakteristika
362	16,8	86,0	120	<i>R1, R2, Pn1, Pn2</i>	(obrázek PA.3)
471	26,4	62,7	140	<i>Os1, Os2, Os3, Os4</i>	(obrázek PA.4)



Obrázek PA.3 – Trakční charakteristika lokomotivy typu 362, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)



Obrázek PA.4 – Trakční charakteristika lokomotivy typu 471, zdroj: (Univerzita Pardubice, Fakulta elektrotechniky a informatiky, 2016)

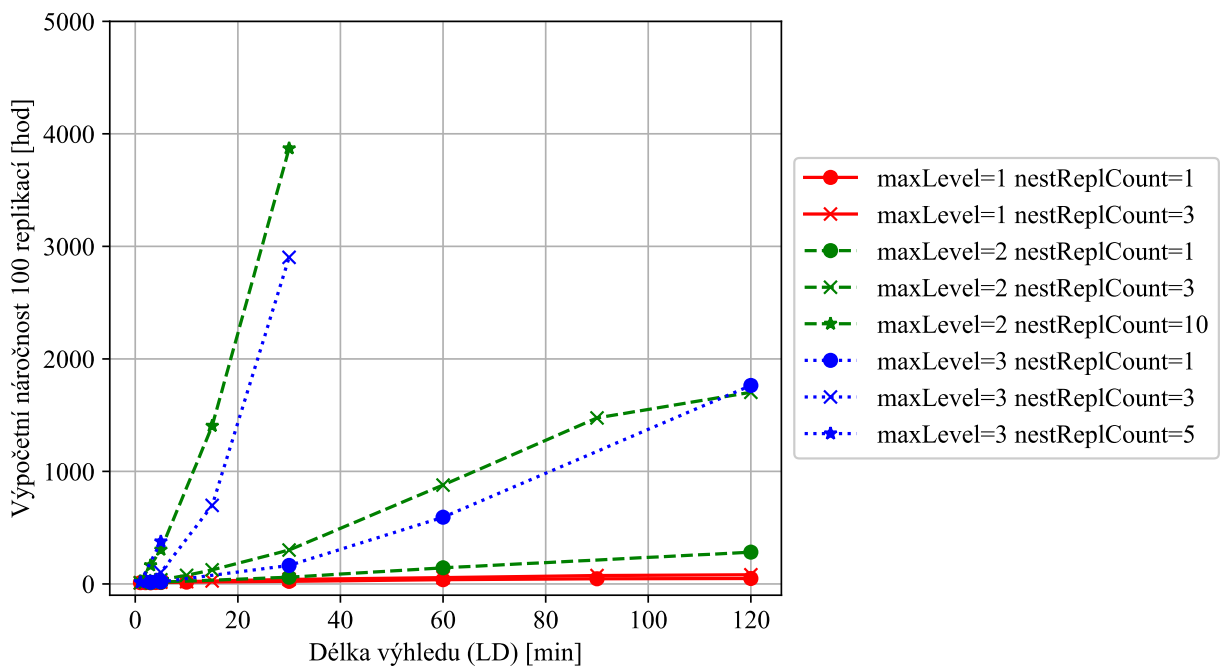
PŘÍLOHA B – VÝSLEDKY VŠECH PROVEDENÝCH EXPERIMENTŮ

Nad rámec případové studie bylo provedeno větší množství experimentů, kde byly aplikovány různé konfigurace RNS za účelem zjištění hodnot přírůstků zpoždění vlaků, hodnoty SWDI a výpočetní náročnosti jednotlivých experimentů. Výsledky jsou uvedeny v tabulce PB.1, každý řádek popisuje výsledek experimentu (odpovídajícího scénáři z finální fáze), který zahrnuje 100 hlavních replikací ($mainReplCount = 100$). V jednotlivých sloupcích tabulky se nachází hodnoty:

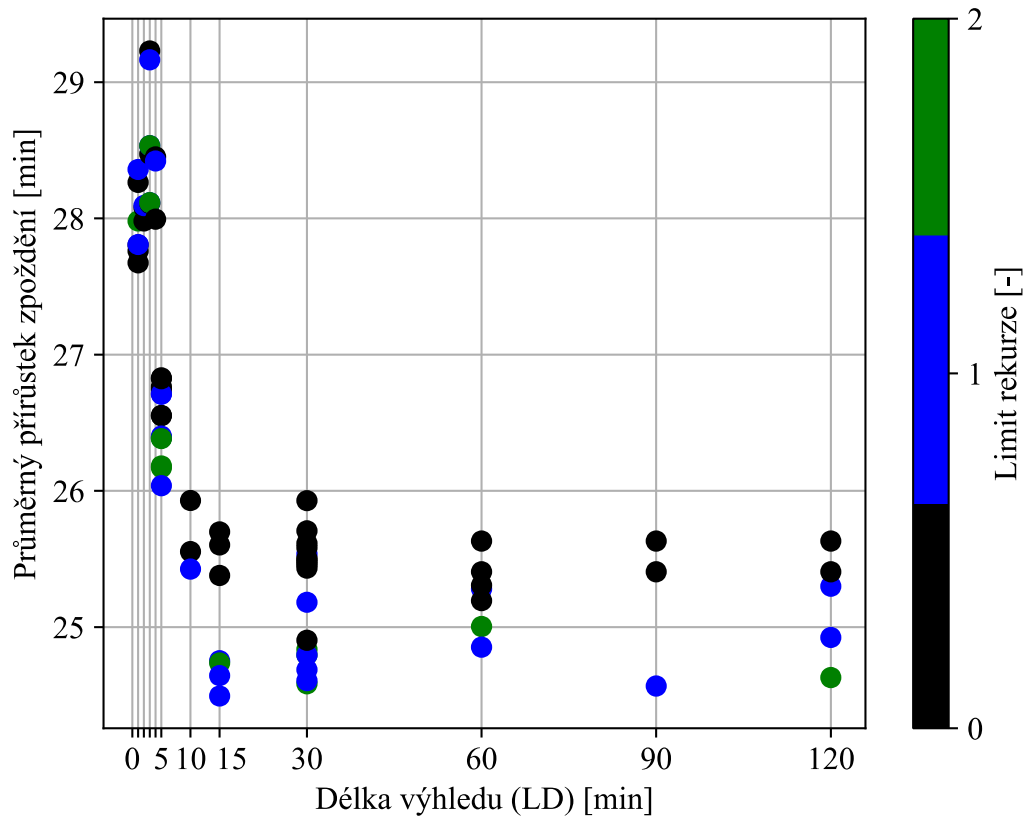
- *Doba výhledu (LD)* – doby výhledů (v minutách) použité pro příslušné experimenty.
- *nestReplCount* – počty replikací vnořených simulací prováděných při prověřování variant řešení konfliktů.
- *maxLevel* – maximální povolené počty úrovní vnořování RNS.
- *meanSWDI ± halfWidth* – hodnoty indikátoru *meanSWDI* včetně pološířek 95% intervalů spolehlivosti (*halfWidth*).
- *meanRelatHalfWidth* – hodnoty *meanRelatHalfWidth* byly výsledkem zpracování dat získaných ze všech replikací vnořených simulací příslušných scénářů. Dokumentují průměrnou míru přesnosti určení hodnoty ukazatele *meanSWDI* (získávané z vnořených simulací) pro odlišné scénáře.
- Δd_R – průměrné přírůstky zpoždění na jeden vlak pro vlaky z kategorie dálkové dopravy (v sekundách).
- Δd_{Os} – průměrné přírůstky zpoždění na jeden vlak pro vlaky z kategorie osobní dopravy (v sekundách).
- Δd_{Pn} – průměrné přírůstky zpoždění na jeden vlak pro vlaky z kategorie nákladní dopravy (v sekundách).
- $N_{ConfTot}$ – průměrné počty konfliktů (zahrnující konflikty vzniklé v hlavních replikacích i na nižších úrovních vnoření RNS), které byly řešeny pomocí RNS.
- N_{SimTot} – průměrné počty replikací příslušející množinám ${}^dH_b(S_{0,0,r})$, kde $r = 1 \dots 16$ a hodnoty indexů d a b pro jednotlivé scénáře odpovídají hodnotám v prvním a druhém sloupci tabulky. Jako příklad lze uvést scénář $Sc-02-05-p$, kde množiny ${}^2H_5(S_{0,0,r})$, $r = 1 \dots 16$ průměrně obsahovaly 3854 replikací a 363 konfliktů, které byly řešeny s uplatněním RNS.
- ${}^{sum}T$ – vypočtené doby uplynulého reálného času potřebného k provedení příslušných scénářů, pokud by výpočet byl realizován na 1 LPU (bez paralelizace výpočtů hlavních replikací).
- ${}^{real}T_{100}$ – naměřené celkové doby uplynulého reálného času potřebného k provedení příslušných scénářů (čas potřebný pro výpočet 100 hlavních replikací na clusteru uvedeném v případové studii).
- ${}^{mean}T$ – průměrné doby uplynulého reálného času potřebného na výpočet jedné hlavní replikace příslušného scénáře.
- ${}^{max}T$ – maximální doby uplynulého reálného času potřebného na výpočet jedné hlavní replikace příslušného scénáře.

Data z tabulky PB.1 jsou rovněž vizualizována na obrázcích. Hrubá výpočetní náročnost v závislosti na parametrech $maxLevel$, $nestReplCount$ a LD je zobrazena na obrázku PB.1. Na grafu je patrné, že u vybraných kombinací parametrů RNS (např. $maxLevel = 2$, $nestReplCount = 10$) docházelo při zvyšování *doby výhledu* (LD) téměř k exponenciálnímu nárůstu výpočetní náročnosti. Dále základní grafické trendy mezi hodnotami jednotlivých parametrů *konfigurace RNS* a dosaženým výsledkem jsou zobrazeny na obrázcích PB.2–PB.5.

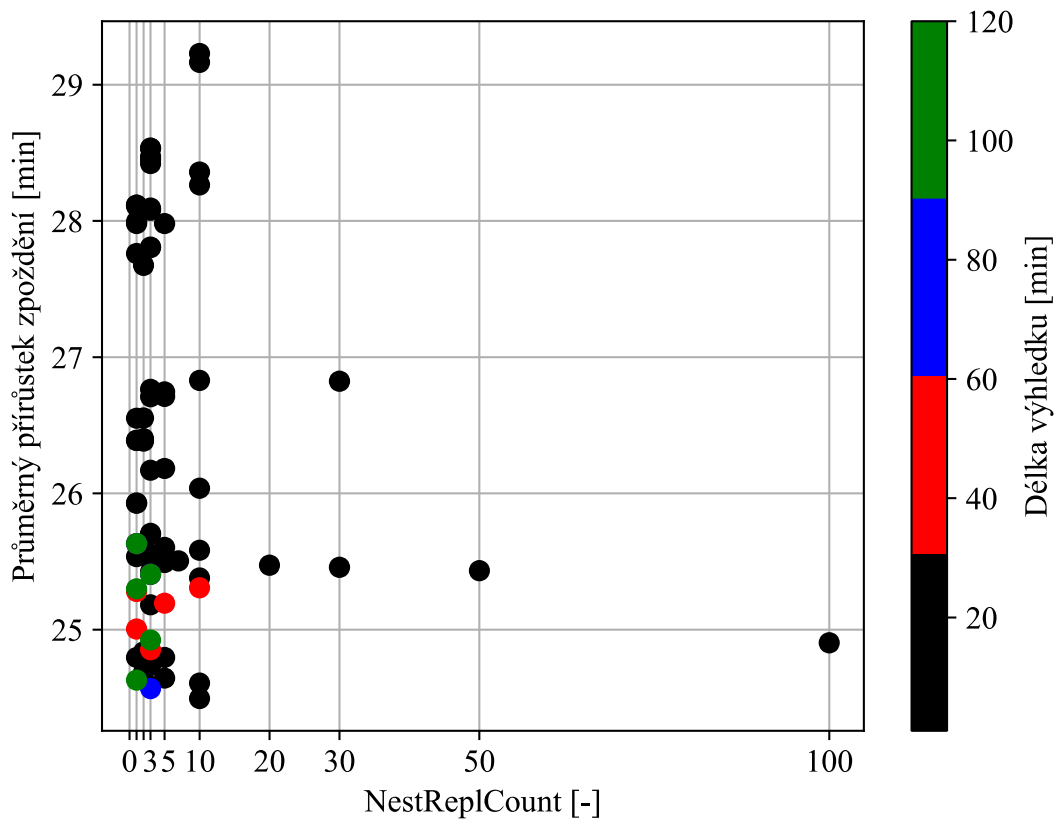
Na obrázku PB.6 jsou vizualizovány jednotlivé hodnoty *přírůstků zpoždění* pro jednotlivé vlaky v každé hlavní replikaci. Na ose x jsou jednotlivé vlaky, seřazené podle čísla vlaku vzestupně. Na ose y jsou jednotlivé hlavní replikace. Barva každého pole pak určuje konkrétní hodnotu *přírůstku zpoždění*. Zelené pole označují záporný *přírůstek zpoždění*, červené pak kladný *přírůstek zpoždění* a žlutá pole mají nulovou hodnotu *přírůstku zpoždění*. Na obrázku jsou zobrazeny 3 scénáře, které byly vybrány do *finální fáze* případové studie. Na první pohled se vizualizace *přírůstků zpoždění* pro uvedené scénáře může zdát jako totožná, to je dáno faktem, že všechny scénáře simulují totožné hlavní replikace a metoda RNS poskytovala velmi podobná řešení konfliktů. Pro lepší znázornění odlišností je podobná vizualizace na obrázku PB.7, kde jsou vizualizovány odpovídající rozdíly mezi těmito scénáři.



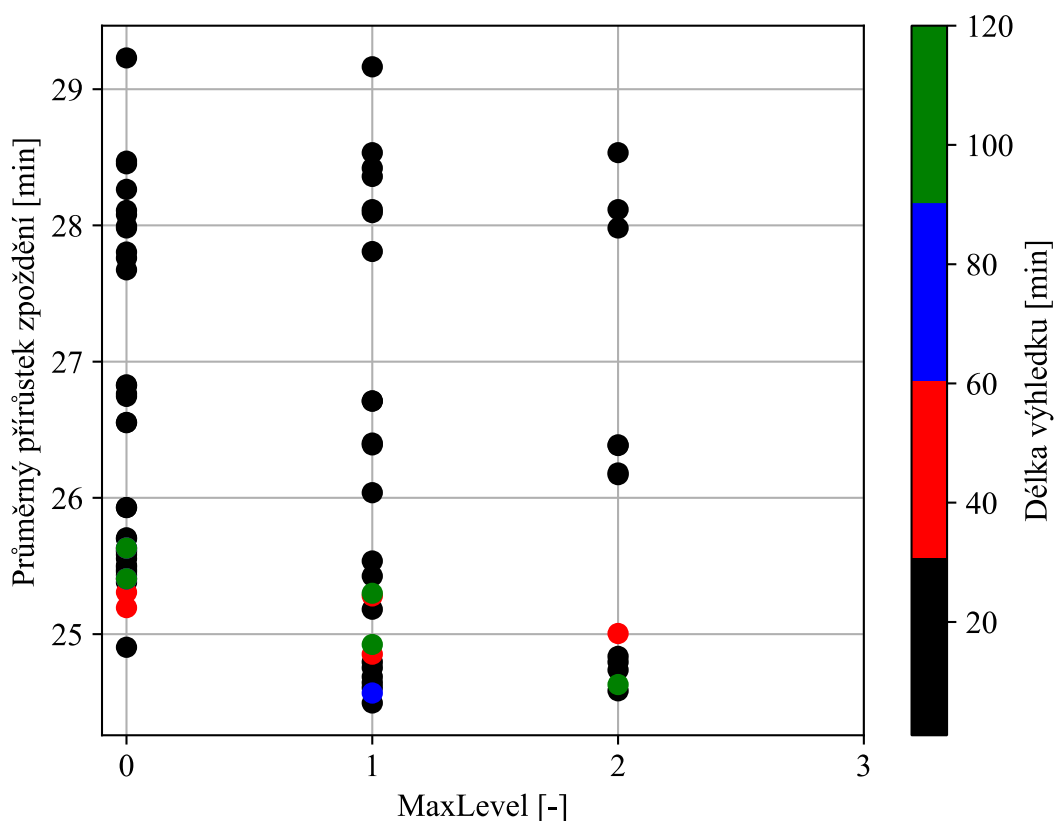
Obrázek PB.1 – Vztah výpočetní náročnosti ($^{sum}T_{100}$) na *konfiguraci RNS*, zdroj: vlastní



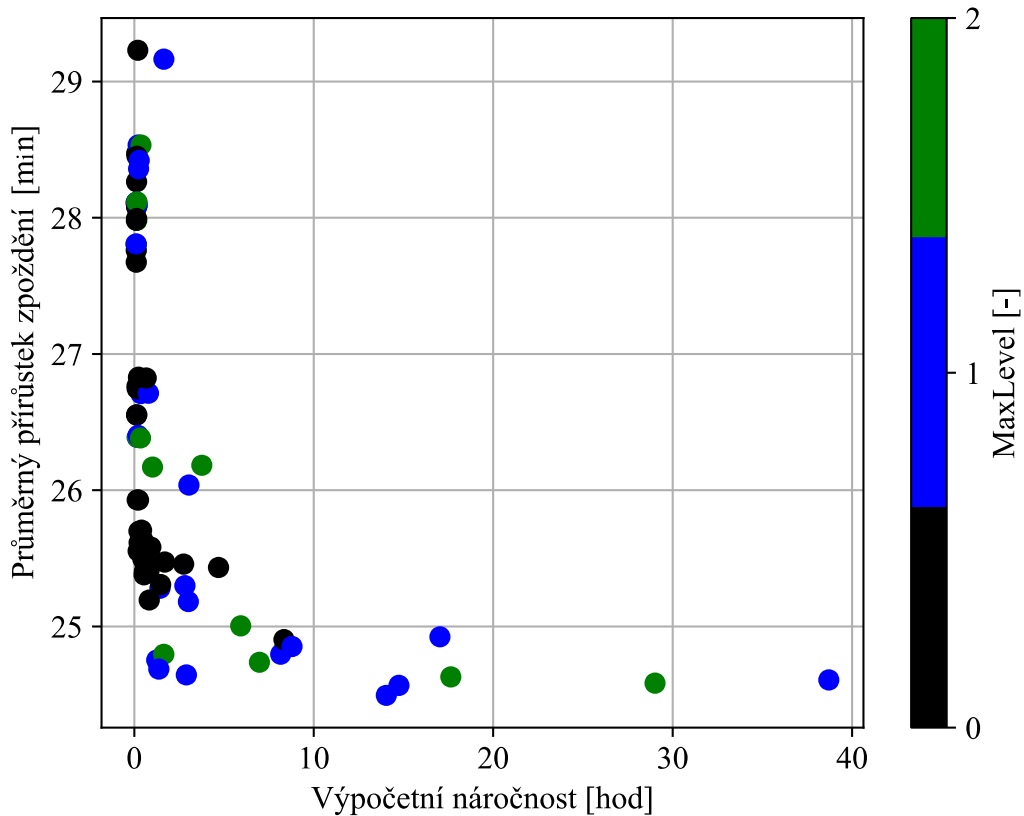
Obrázek PB.2 – Průměrný přírůstek zpoždění v závislosti na délce výhledu, zdroj: vlastní



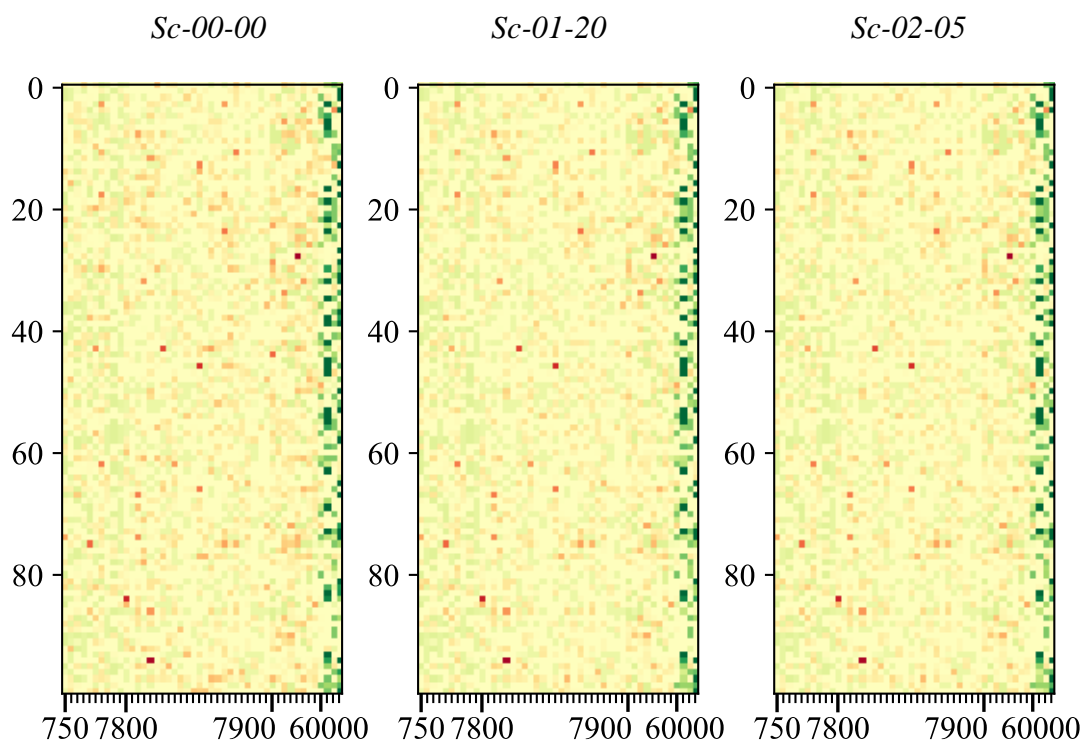
Obrázek PB.3 – Průměrný přírůstek zpoždění v závislosti na počtu vnořených replikací (parametr *nestReplCount*), zdroj: vlastní



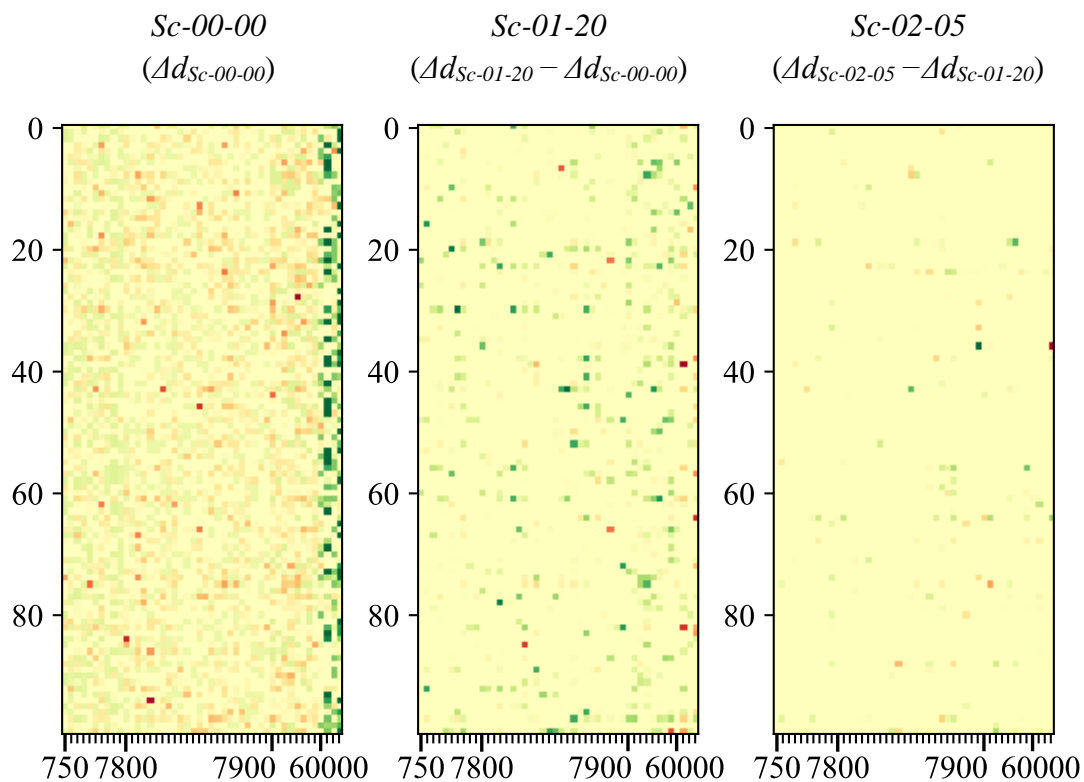
Obrázek PB.4 – Průměrný *přírůstek zpoždění* v závislosti na maximální možné úrovni rekurze (parametr *maxLevel*), zdroj: vlastní



Obrázek PB.5 – Průměrný *přírůstek zpoždění* v závislosti na výpočetní náročnosti ($\sum T_{100}$) a parametru *maxLevel*, zdroj: vlastní



Obrázek PB.6 – Hodnoty přírůstků zpoždění pro jednotlivé vlaky a jednotlivé hlavní replikace pro scénáře z finální fáze, zdroj: vlastní



Obrázek PB.7 – Hodnoty přírůstků zpoždění a jejich změny (v porovnání s předchozími scénáři) pro jednotlivé vlaky a jednotlivé hlavní replikace pro scénáře z finální fáze, zdroj: vlastní

<i>LD</i> [min]	<i>Nest Repl Count</i> [-]	<i>Max Level</i> [-]	<i>Mean SWDI</i> [min]	<i>Half Width</i> [min]	<i>Mean Relat Half Width</i> [-]	<i>AdR</i> [s]	<i>AdoS</i> [s]	<i>AdPn</i> [s]	<i>Průměr N_{ConfTot}</i> [-]	<i>Průměr N_{SimTot}</i> [-]	<i>sumT₁₀₀</i> [hod]	<i>realT₁₀₀</i> [hod]	<i>meant</i> [hod]	<i>maxy</i> [hod]
0	0	0	32,65	3,25	-	26,38	47,44	-42,35	0	0	9,9	0,8	0,1	0,1
1	1	1	27,76	2,57	-	23,19	40,13	-45,04	13	29	11,1	0,9	0,1	0,1
1	2	1	27,67	2,59	0,11	23,19	39,98	-45,38	13	58	11,2	0,9	0,1	0,2
1	3	1	27,80	2,61	0,02	23,20	40,22	-45,25	13	87	10,8	0,9	0,1	0,2
1	3	2	27,81	2,61	0,01	23,20	40,22	-45,09	29	194	10,2	0,8	0,1	0,2
1	5	3	27,98	2,63	0,01	23,30	40,48	-45,10	44	500	14,1	1,3	0,1	0,4
1	10	1	28,26	2,71	0,01	23,32	41,00	-44,97	13	290	12,7	1,1	0,1	0,3
1	10	2	28,36	2,75	0,01	23,32	41,18	-44,82	67	1 498	24,0	2,3	0,2	1,0
2	1	1	27,98	2,63	-	22,38	40,95	-43,04	13	28	11,5	0,9	0,1	0,1
2	3	1	28,08	2,65	0,03	22,59	41,03	-43,31	13	85	12,9	1,0	0,1	0,2
2	3	2	28,09	2,65	0,03	22,59	41,05	-43,30	55	328	16,4	1,4	0,2	0,3
3	1	1	28,11	2,72	-	21,74	41,54	-42,43	13	29	10,4	0,8	0,1	0,1
3	1	2	28,12	2,73	-	21,76	41,54	-42,44	32	64	12,7	1,1	0,1	0,2
3	1	3	28,12	2,73	-	21,76	41,54	-42,44	52	95	14,6	1,3	0,1	0,3
3	3	1	28,47	2,72	0,04	22,12	42,00	-42,40	13	87	13,9	1,1	0,1	0,2
3	3	2	28,53	2,73	0,03	22,12	42,12	-42,41	72	412	21,7	2,0	0,2	0,6
3	3	3	28,53	2,73	0,02	22,12	42,12	-42,41	243	1 234	37,6	4,5	0,4	1,9
3	10	1	29,23	2,80	0,01	21,91	43,53	-41,96	13	289	19,7	1,8	0,2	0,6
3	10	2	29,16	2,80	0,01	21,92	43,40	-41,97	209	3 891	164,9	18,1	1,6	6,8
4	1	1	27,99	2,74	-	20,96	41,73	-41,15	13	29	13,0	1,0	0,1	0,2
4	3	1	28,45	2,84	0,04	20,84	42,65	-41,02	13	87	15,3	1,2	0,2	0,2
4	3	2	28,42	2,82	0,04	20,84	42,59	-40,77	88	488	27,5	2,5	0,3	0,7
5	1	1	26,55	2,64	-	21,02	39,12	-46,14	12	27	13,0	1,1	0,1	0,2

Tabulka PB.1 – Přehled výsledků pro všechny testované konfigurace v případové studii, zdroj: vlastní

<i>LD</i> [min]	<i>Nest Repl Count</i> [-]	<i>Max Level</i> [-]	<i>Mean SWDI</i> [min]	<i>Half Width</i> [min]	<i>Mean Relat Half Width</i> [-]	<i>Adn</i> [s]	<i>Ados</i> [s]	<i>Adrn</i> [s]	<i>Průměr N_{ContTot}</i> [-]	<i>Průměr N_{SimTot}</i> [-]	<i>sum T₁₀₀</i> [hod]	<i>real T₁₀₀</i> [hod]	<i>meant</i> [hod]	<i>maxt</i> [hod]
5	1	2	26,39	2,62	-	21,08	38,78	-46,20	40	76	16,0	1,4	0,2	0,3
5	1	3	26,39	2,62	-	21,08	38,78	-46,20	80	137	18,5	1,7	0,2	0,5
5	2	1	26,55	2,69	0,15	21,27	38,99	-46,73	12	55	14,1	1,1	0,1	0,2
5	2	2	26,40	2,64	0,13	21,25	38,70	-45,98	67	250	21,2	1,9	0,2	0,5
5	2	3	26,38	2,64	0,12	21,22	38,68	-45,98	225	728	34,7	3,7	0,3	1,3
5	3	1	26,57	2,62	0,05	20,92	39,23	-46,91	12	83	15,2	1,3	0,2	0,3
5	3	2	26,61	2,66	0,04	21,07	39,19	-46,29	95	525	35,9	3,3	0,4	1,0
5	3	3	26,61	2,66	0,03	21,07	39,19	-46,30	454	2 147	101,4	12,4	1,0	5,4
5	5	1	26,56	2,61	0,03	21,31	38,98	-46,85	12	139	16,4	1,4	0,2	0,3
5	5	2	26,43	2,66	0,02	21,32	38,73	-46,24	151	1 377	77,8	8,3	0,8	3,2
5	5	3	26,44	2,66	0,02	21,32	38,74	-46,24	1 149	8 936	387,1	42,2	3,9	17,8
5	10	1	26,67	2,64	0,02	21,29	39,19	-46,18	12	277	23,1	2,1	0,2	0,5
5	10	2	26,41	2,63	0,01	21,13	38,78	-46,02	288	5 191	308,4	31,3	3,1	11,0
5	30	1	26,65	2,62	0,01	20,96	39,34	-46,37	12	836	65,3	5,5	0,7	1,6
10	1	1	25,93	2,64	-	19,87	38,57	-45,16	13	28	16,2	1,3	0,2	0,2
10	3	1	25,55	2,58	0,06	19,98	37,81	-45,18	13	84	21,2	1,8	0,2	0,4
10	3	2	25,43	2,67	0,05	19,88	37,63	-45,65	125	720	76,2	8,7	0,8	3,5
15	3	1	25,40	2,58	0,07	19,76	37,64	-45,20	13	84	26,4	2,3	0,3	0,5
15	3	2	25,08	2,56	0,06	19,43	37,22	-44,82	156	943	126,0	13,5	1,3	4,8
15	3	3	25,03	2,57	0,05	19,23	37,22	-44,77	1 240	6 909	711,8	91,9	7,1	45,1
15	5	1	25,32	2,56	0,04	19,87	37,41	-44,66	13	140	29,3	2,6	0,3	0,6
15	5	2	25,04	2,59	0,03	19,54	37,08	-44,57	257	2 564	292,6	30,3	2,9	10,9
15	10	1	25,38	2,57	0,02	19,73	37,60	-44,35	13	284	54,3	5,1	0,5	1,4
15	10	2	24,49	2,56	0,02	19,79	35,93	-45,29	468	9 301	1 404,3	156,3	14,0	74,0

Tabulka PB.1 (pokračování) – Přehled výsledků pro všechny testované konfigurace v případové studii, zdroj: vlastní

LD [min]	Nest Repl Count [-]	Max Level [-]	Mean SWDI [min]	Half Width [min]	Mean Relat Half Width [-]	Δd_R [s]	Δd_{Os} [s]	Δd_{Ph} [s]	Průměr $N_{ConfTot}$ [-]	Průměr N_{SimTot} [-]	sum T_{100} [hod]	real T_{100} [hod]	mean t [hod]	max t [hod]
30	1	1	25,63	2,58	-	19,87	38,01	-44,83	13	28	23,2	2,0	0,2	0,5
30	1	2	25,29	2,61	-	19,67	37,48	-45,13	92	195	58,2	5,8	0,6	1,9
30	1	3	25,25	2,60	-	19,35	37,58	-45,15	425	866	162,1	21,0	1,6	10,1
30	2	1	25,30	2,61	0,35	20,00	37,31	-44,88	13	56	27,2	2,4	0,3	0,7
30	2	2	24,94	2,55	0,24	19,70	36,79	-44,44	173	747	136,6	14,8	1,4	4,9
30	2	3	24,84	2,54	0,18	19,16	36,92	-45,21	1 305	5 383	835,4	135,0	8,4	79,9
30	3	1	25,41	2,57	0,13	20,00	37,52	-45,32	13	84	40,5	3,4	0,4	0,9
30	3	2	24,93	2,56	0,08	19,60	36,83	-44,85	236	1 504	292,7	29,4	2,9	11,0
30	3	3	24,53	2,55	0,06	19,16	36,32	-44,62	2 757	17 021	2 927,2	334,1	29,3	168,8
30	5	1	25,19	2,55	0,08	20,05	37,08	-44,98	13	140	47,7	4,0	0,5	1,1
30	5	2	24,93	2,55	0,05	19,71	36,77	-44,85	396	4 212	818,3	84,6	8,2	27,1
30	7	1	25,21	2,54	0,06	19,96	37,17	-44,81	13	197	61,7	5,2	0,6	1,5
30	10	1	25,31	2,57	0,05	19,95	37,35	-44,74	13	283	91,1	7,5	0,9	1,8
30	10	2	24,61	2,57	0,03	19,62	36,19	-43,49	758	16 003	3 870,4	345,8	38,7	94,4
30	20	1	25,18	2,55	0,03	19,93	37,11	-44,80	13	567	167,6	15,0	1,7	4,1
30	30	1	25,16	2,54	0,03	19,95	37,08	-44,91	13	845	272,3	23,4	2,7	7,2
30	50	1	25,14	2,54	0,02	19,89	37,08	-44,80	13	1 410	467,6	42,0	4,7	11,7
30	100	1	25,16	2,54	0,01	19,92	37,09	-44,96	13	2 818	821,2	67,5	8,2	18,9
60	1	1	25,63	2,58	-	19,87	38,01	-44,83	13	28	37,6	3,1	0,4	0,8
60	1	2	25,28	2,60	-	19,68	37,46	-45,13	140	306	143,3	14,7	1,4	5,7
60	1	3	25,23	2,53	-	19,42	37,47	-43,72	1 137	2 601	591,8	71,3	5,9	35,3
60	3	1	25,41	2,57	0,18	20,00	37,52	-45,32	13	84	56,1	4,4	0,6	1,2
60	3	2	24,87	2,54	0,10	19,62	36,68	-43,49	386	2 577	886,2	98,0	8,9	43,9
60	5	1	25,19	2,55	0,05	20,05	37,08	-44,98	13	140	83,5	6,9	0,8	1,9

Tabulka PB.1 (pokračování) – Přehled výsledků pro všechny testované konfigurace v případové studii, zdroj: vlastní

<i>LD</i> [min]	<i>Nest Repl Count</i> [-]	<i>Max Level</i> [-]	<i>Mean SWDI</i> [min]	<i>Half Width</i> [min]	<i>Mean Relat Half Width</i> [-]	<i>Δdk</i> [s]	<i>Δdos</i> [s]	<i>Δd_{pn}</i> [s]	<i>Průměr N_{ConfTot}</i> [-]	<i>Průměr N_{SimTot}</i> [-]	<i>sum T₁₀₀</i> [hod]	<i>real T₁₀₀</i> [hod]	<i>mean_t</i> [hod]	<i>max_t</i> [hod]
60	10	1	25,31	2,57	0,03	19,95	37,35	-44,74	13	283	147,0	12,5	1,5	3,8
90	1	1	25,63	2,58	-	19,87	38,01	-44,83	13	28	46,0	3,8	0,5	0,9
90	3	1	25,41	2,57	0,21	20,00	37,52	-45,32	13	84	74,6	6,3	0,7	1,6
90	3	2	24,89	2,54	0,12	19,42	36,84	-43,79	494	3326	1 491,4	160,1	14,9	57,5
120	1	1	25,63	2,58	-	19,87	38,01	-44,83	13	28	48,8	3,9	0,5	1,3
120	1	2	25,30	2,60	-	19,64	37,52	-45,23	192	431	282,3	25,3	2,8	10,0
120	1	3	24,53	2,42	-	18,69	36,57	-43,26	1 868	4220	1 766,4	144,9	17,7	79,3
120	3	1	25,41	2,57	0,24	20,00	37,52	-45,32	13	84	81,6	6,8	0,8	1,8
120	3	2	24,15	2,50	0,12	18,70	35,82	-42,47	530	3595	1 651,3	146,5	16,5	41,2

Tabulka PB.1 (pokračování) – Přehled výsledků pro všechny testované konfigurace v případové studii, zdroj: vlastní