

# SIMULAČNÝ MODEL AUTOMATICKÉHO BLOKU V UNIVERZÁLNO M SIMULAČNOM NÁSTROJI ANYLOGIC

## SIMULATION MODEL OF AUTOMATIC BLOCK IN GENERAL PURPOSE SIMULATION TOOL ANYLOGIC

Peter Márton<sup>1</sup>

---

*Anotácia: Priepravná výkonnosť traťového úseku je ovplyvňovaná niekoľkými faktormi. Jedným z nich je aj typ traťového zabezpečovacieho zariadenia. Modernizáciou, resp. výmenou traťového zabezpečovacieho zariadenia, je možné zvýšiť priepustnú výkonnosť traťového úseku. V tomto článku je popísané budovanie simulačného modelu jedného typu traťového zabezpečovacieho zariadenia – automatického bloku.*

*Kľúčové slová: automatický blok, priepustná výkonnosť traťového úseku, univerzálny simulačný nástroj.*

*Summary: Railway line capacity is influenced by several factors. Type of interlocking and safety system is one of them. Railway line capacity can be improved by reconstruction, eventually by upgrading of the interlocking and safety system. In this paper, developing and construction of simulation model of automatic block is described. Automatic block is type of the interlocking and safety system.*

*Key words: automatic block, general purpose simulation tool, railway line capacity.*

### ÚVOD

Vo viacerých nových členských krajinách Európskej únie sa v posledných rokoch rekonštruujú železničné trate. Železničné koridory sú rekonštruované pre traťovú rýchlosť 160 km/h, prípadne viac. V niektorých štátoch sú rekonštruované tiež regionálne trate. Nedeliteľnou súčasťou rekonštruovaných tratí je traťové zabezpečovacie zariadenie.

Pri rekonštrukcii trate je dôležité dodržať holistický prístup. Obnovený traťový spodok i zvršok, nové traťové zabezpečovacie zariadenie a novo navrhnutý grafikon vlakovej dopravy by mali viesť k spokojnosti zákazníkov železničnej dopravy. Priepravnú výkonnosť tratí výrazne ovplyvňuje typ použitého traťového zabezpečovacieho zariadenia.

Tento článok je venovaný simulačnému modelu prevádzky na železničnej trati. Predstavené sú prvé výsledky vývoja a budovania simulačného modelu traťového úseku vybaveného traťovým zabezpečovacím zariadením, ktoré sa nazýva automatický blok (autoblok).

---

<sup>1</sup> Ing. Peter Márton, PhD., Žilinská univerzita v Žiline, Fakulta riadenia a informatiky, Katedra matematických metód a operačnej analýzy, Univerzitná 8215/1, 01026 Žilina, Slovenská republika, tel.: +421415134053, fax: +421415134055, e-mail: [Peter.Marton@fri.uniza.sk](mailto:Peter.Marton@fri.uniza.sk)

## 1. UNIVERZÁLNE SIMULAČNÉ NÁSTROJE

Samotný simulačný model sa tvorí zvyčajne v prostredí nejakého simulačného programu. Simulačné programy umožňujú vybudovať simulačný model skúmaného systému a následne aj experimentovanie s týmto modelom. Zjednodušene povedané, simulačný model dáva možnosť hľadať odpovede na rôzne otázky typu „čo sa stane, ak ...?“. Základným predpokladom pre úspešné používanie simulačného programu je to, že jeho používateľ ho na jednej strane dokáže správne používať a na druhej strane dostatočne pozná modelovaný systém, resp. spolupracuje pri budovaní modelu a experimentovaní s odborníkom z praxe, napr. kompetentným zamestnancom vlakovtornej stanice.

Vo všeobecnosti jestvujú pre výber simulačného nástroja tri možnosti. Prvou možnosťou je použiť niektorý zo všeobecných a univerzálnych simulačných programov. Takýmito programami sú napríklad AnyLogic (4) a ARENA (5). Druhou možnosťou je použiť niektorý zo simulačných programov vyvinutých špeciálne pre tvorbu modelov v oblasti železničnej dopravy. Medzi takéto programy patria napríklad Railsys, OpenTrack a Villon (1). Tretou možnosťou je vývoj a naprogramovanie vlastného simulačného programu. Realizácia tejto možnosti je veľmi náročná z hľadiska trvania vývoja takéhoto programu a tiež znalostí programovania v niektorom programovacom jazyku. Vo všeobecnosti by sa mohlo konštatovať, že konečné rozhodnutie pre použitie univerzálneho alebo špecializovaného simulačného nástroja závisí od riešeného problému, vstupných údajov, ktoré sú k dispozícii a od požadovaných výstupných údajov a ich podoby.

V tomto článku je popísané použitie univerzálneho simulačného nástroja AnyLogic. Program AnyLogic je simulačný nástroj, ktorý podporuje všetky najpoužívanejšie metódy simulácie a to diskrétnu, spojitú a agentovo orientovanú simuláciu a tiež metódu systémovej dynamiky. Je teda dostatočne komplexný na to, aby sa ním dala vytvoriť abstrakcia ekonomického, sociálneho, výrobného či obslužného systému s ľubovoľnou úrovňou detailov. Modelový návrh v prostredí simulačnom nástroji AnyLogic je objektovo orientovaný. AnyLogic je zostavený na základe programovacieho jazyku Java, čo mu zabezpečuje bezproblémové fungovanie na mnohých platformách. Taktiež umožňuje do modelov vkladať vlastný zdrojový kód v jazyku Java, čím je zabezpečená široká flexibilita pri tvorbe modelov. Komponenty modelov (v samotnom simulačnom nástroji AnyLogic nazývané ako „objekty“) používané v tomto programe sú zoradené do niekoľkých knižníc podľa ich funkcie. Knižnice sú zamerané napríklad na jednotlivé simulačné paradigmy. Okrem nich existuje samostatná knižnica pre zobrazovanie výsledkov, prípravu animácie a tiež napríklad riadenie simulačného behu. V programe AnyLogic je možné vybudovať modely z najrôznejších oblastí, ako sú zdravotníctvo, doprava, výroba a mnohé ďalšie.

## 2. SIMULAČNÝ MODEL

### 2.1 Vstupné údaje

Pre vybudovanie simulačného modelu prevádzky na traťovou úseku sú vo všeobecnosti potrebné nasledovné vstupné údaje:

- topológia železničnej trate, t.j. technické údaje o modelovanom traťovom úseku – dĺžka, polohy oddielových návěstidiel, rýchlostné obmedzenia, atď.,
- pravidlá prevádzky a údaje týkajúce sa grafikonu vlakovej dopravy – napr. „bloková podmienka“, jazdné časy, následné medzičasy, časy vstupu vlakov do traťového úseku, atď..

Aby sa pomocou simulačného modelu získali relevantné výstupy a výsledky, je potrebné použiť presné a spoľahlivé vstupné údaje. V prípade modelovania dynamiky jazdy vlaku sú potrebné niektoré detailné technické údaje o sklonoch, polomeroch oblúkov, atď. Avšak, v modeloch popisovaných v tomto príspevku sa používajú iba jazdné časy prevzaté zo skutočného grafikonu vlakovej dopravy alebo jazdné časy odhadnuté na základe dĺžky priestorového oddielu a priemernej rýchlosti jazdy vlaku.

V tomto článku je najprv vysvetlené, ako vybudovať jednoduchý simulačný model traťového úseku bez traťového zabezpečovacieho zariadenia. Potom je vysvetlené, ako vybudovať model s niekoľkými priestorovými oddielmi, s oddielovými návěstidlami v jednom smere jazdy. Napokon je vysvetlené, ako vybudovať simulačný model s oddielovými návěstidlami pre obojsmernú prevádzku. Všetky tri simulačné modely boli vyvinuté s cieľom zobrazit' princípy fungovania automatického bloku.

## 2.2 Model prevádzky na traťovom úseku

Pri budovaní simulačného modelu je potrebné definovať základné elementy modelovaného systému. Ide o systémové entity (zákazníci a zdroje), vstupné premenné, výstupné parametre a vzájomné závislosti a procesy medzi týmito elementmi. V našom simulačnom modeli sú systémovými entitami traťový úsek (zdroj) a vlaky (zákazníci). Čas príchodu vlaku a jazdný čas patria medzi vstupné premenné. Priemerný čas čakania pred oddielovým návěstidlom môže byť považovaný za výstupný parameter. Závislosti medzi oddielovými návěstidlami sú tiež súčasťou simulačného modelu.

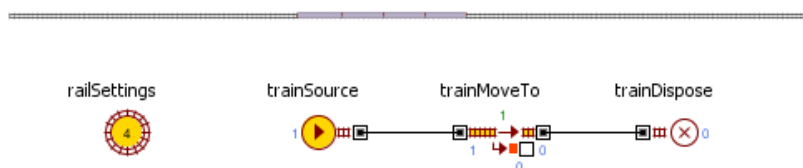
### 2.2.1 Simulačný model traťového úseku bez zabezpečovacieho zariadenia

Na vybudovanie simulačného modelu traťového úseku bez zabezpečovacieho zariadenia stačí použiť niekoľko základných objektov simulačného nástroja AnyLogic:

- railSettings: tento objekt poskytuje rozhranie pre detailné riadenie modelu na základe funkcií jazyka Java a techniky callback.
- trainSource: tento objekt sa v prostredí AnyLogic-u nachádza vždy na začiatku vývojového diagramu modelu železničných procesov. Sú v ňom generované (vytvárané) vlaky. Riadi sa ním aj to, kde sa má vlak pri vstupe do modelu umiestniť (na ktorú koľaj). Vytvorený vlak musí tvoriť najmenej jedno vozidlo. Na obrázku 1 sa nachádza vlak tvorený štyrmi vozidlami (v prostriedku koľaje).
- trainMoveTo: toto je jediný objekt, ktorým sa riadi pohyb vlaku. Vlak sa môže hýbať dopredu alebo dozadu. Vlak je možné definovať cieľovú pozíciu. V tomto objekte nie je zabudovaná kontrola toho, koľko vlakov sa na ním spravovanej koľaji nachádza.
- trainDispose: cez tento objekt opúšťajú vlaky simulačný model.

Prvý simulačný model je použitý na to, aby sa užívateľ zoznámil s prostredím simulačného nástroja AnyLogic. Použité objekty je potrebné prepojiť navzájom. Vytvorí sa tým vývojový diagram ktorý popisuje existenciu obsluhovanej entity (vlak) v simulačnom modeli:

1. Entita vlaku je vytvorená v objekte trainSource.
2. Entita vlaku je poslaná do nasledujúceho objektu (trainMoveTo). V nasledujúcom objekte sú entite oznámené inštrukcie o pohybe – má sa presunúť na koniec koľaje. Počas pohybu sa entita nachádza v tomto objekte.
3. Entita vlaku opúšťa model v poslednom objekte vývojového diagramu.



Zdroj: Autor

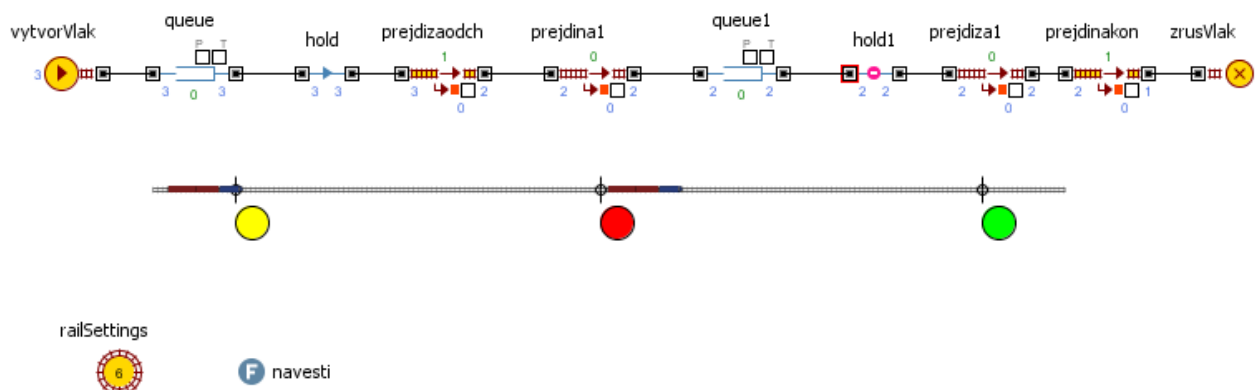
Obr. 1 – Jednoduchý model traťového úseku

### 2.2.2 Simulačný model s oddielovými návěstidlami v jednom smere jazdy

Druhý simulačný model bol vytvorený tak, aby zobrazoval fungovanie automatického bloku v jednom smere jazdy (obrázok 2). Model zobrazuje jednokoľajný medzistaničný úsek so štyrmi priestorovými oddielmi. Priestorové oddiely sú ohraničené oddielovými návěstidlami. V modeli sa nachádzajú iba oddielové návěstidlá. Nie sú v ňom odchodové návěstidlo zadnej dopravne a vchodové návěstidlo prednej dopravne. Teda dva priestorové oddiely majú oddielové návěstidlá na svojom začiatku i konci. Jeden priestorový oddiel má návěstidlo iba na konci. Jeden priestorový oddiel má návěstidlo iba na začiatku.

Do tohto modelu bolo potrebné pridať ďalšie objekty. Tieto boli použité najmä na zabezpečenie toho, že v každom priestorovom oddiele sa bude nachádzať najviac jeden vlak:

- queue: tento objekt umožňuje vytvoriť front (zásobník) pre vlaky čakajúce pri oddielovom návěstidle, pokiaľ nie je jazda za neho povolená.
- hold: tento objekt zastavuje entity a púšťa ich v modeli ďalej iba ak je splnená definovaná podmienka. V tomto modeli sa objekt hold používa na zadržanie entity vlaku pred oddielovým návěstidlom s návěstou zakazujúcou jazdu.
- Function: AnyLogic umožňuje definovať vlastné funkcie. Funkcia sa používa na výpočet nejakej hodnoty podľa definovaného vzťahu. Robí tak vždy po jej zavolaní zo zdrojového kódu. V tomto modeli funkcia „navesti“ je volaná (aktivovaná) vždy pri príchode entity vlaku do ktoréhokoľvek objektu hold a pri príchode a odchode z ktoréhokoľvek objektu moveTrainTo. Vďaka tomu sa v animácii modelu zobrazujú príslušné farby oddielových návěstidiel.



Zdroj: Autor

Obr. 2 – Jednoduchý model traťového úseku s jednosmerným automatickým blokom

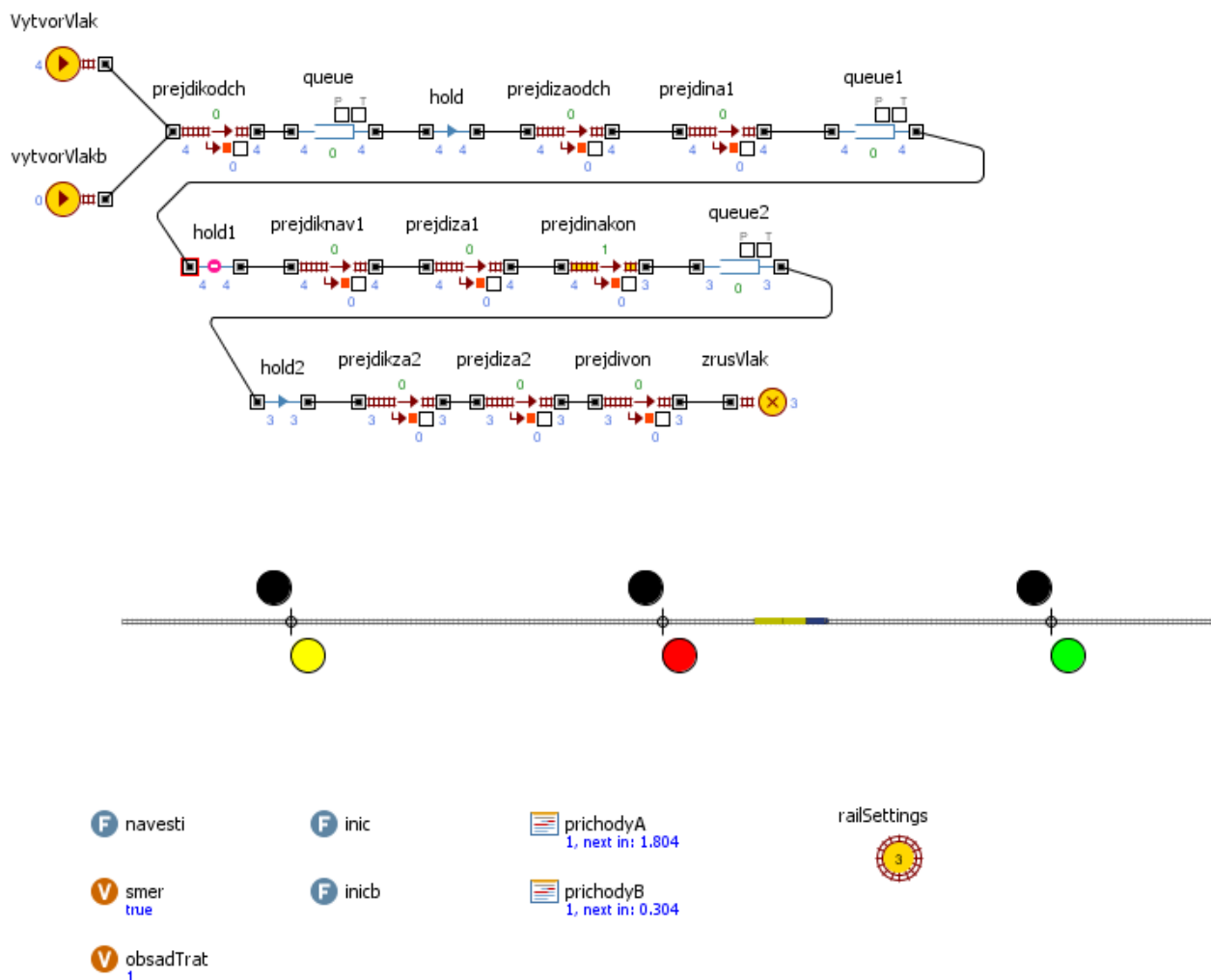
V druhom simulačnom modeli sú pre riadenie pohybu cez každý priestorový oddiel ohraničený dvoma oddielovými návěstidlami použité dva objekty trainMoveTo. Cez prvý objekt trainMoveTo je riadený pohyb vlakov od začiatku priestorového oddielu dotedy, kým sa vlak celý nenachádza za oddielovým návěstidlom. Druhým objekt trainMoveTo je riadený pohyb po koniec priestorového oddielu.

Tento model zobrazuje bežné použitie jednosmerného automatického bloku pre jazdu jedným smerom. Automatický blok je zvyčajne používaný na dvojkoľajných tratiach. Pre každý smer jazdy je zvyčajne používaná jedna traťová koľaj. Usporiadanie objektov vo vývojovom diagrame modelu je rovnaké pre každý priestorový oddiel na traťovom úseku. Existencia entít (vlakov) v druhom simulačnom modeli sa dá popísať nasledovne:

1. Všetky oddielové návěstidlá sú „zelené“. V animácii je zobrazovaná na všetkých návěst „voľno“. Symboly, ktoré v animácii zobrazujú návěstidlá, sa nachádzajú blízko hraníc priestorových oddielov.
2. Entita vlaku je vytvorená v objekte trainSource.
3. Entita vlaku je poslaná do objektu queue. Medzi objektom queue a objektom trainMoveTo sa nachádza objekt hold. Presun entity až do objektu trainMoveTo je možný iba vtedy, keď je objekt hold nezablokovaný. V prípade, že je objekt hold zablokovaný, entita vlaku musí čakať v objekte queue.
4. Entita vlaku pokračuje do prvého objektu trainMoveTo (prejdizaodch). Tento objekt riadi pohyb od momentu vstúpenia vlaku do modelu na začiatku trate dotedy, kým celý vlak neprejde za prvé oddielové návěstidlo. Vstupom entity vlaku do tohto objektu sa zapne blokovanie objektu hold. Návěstí zobrazované na oddielových návěstidlách sa zmenia podľa aktuálneho obsadenia priestorových oddielov (zavolá sa funkcia „navesti“) – prvé oddielové návěstidlo začne v animácii ukazovať návěst „stoj“.
5. Entita vlaku pokračuje do druhého objektu trainMoveTo (prejdina1). Tento objekt riadi presun vlaku po druhé oddielové návěstidlo.
6. Entita vlaku sa presúva smerom k prvému objektu trainMoveTo (prejdiza1) ďalšieho priestorového oddielu. Entita čaká v nasledujúcom objekte queue, ak je zablokovaný objekt hold príslušného priestorového oddielu. To sa stane vtedy, ak sa v nasledujúci priestorový oddiel obsadený vlakom.

7. Entita vlaku sa z objektu queue1 presúva cez objekt hold1 do objektu trainMoveTo (prejdiza1), ak je príslušný priestorový oddiel voľný. Tento objekt trainMoveTo riadi presun vlaku od druhého oddielového návěstidla, až kým celý sa celý vlak nenachádza za druhým oddielovým návěstidlom. Ďalší objekt trainMoveTo (prejdinakon) riadi presun vlaku ku koncu trate. Po vstupe vlaku do objektu trainMoveTo (prejdiza1) je odblokovaný prvý objekt hold. Blokovanie priestorového oddielu medzi prvým a druhým oddielovým návěstidlom je zrušené. Zapnuté je blokovanie priestorového oddielu medzi druhým a tretím oddielovým návěstidlom. Návesti zobrazované na oddielových návěstidlách sa zmenia podľa aktuálneho obsadenia priestorových oddielov. Na druhom oddielovom návěstidle sa zobrazuje návesť „stoj“. Na prvom oddielovom návěstidle sa podľa toho, či je priestorový oddiel za ním voľný, zobrazuje návesť „výstraha“ alebo „stoj“.
8. Entita vlaku opustí model cez posledný objekt vývojového diagramu.

### 2.2.3 Simulačný model s oddielovými návěstidlami pre obojsmernú prevádzku



Zdroj: Autor

Obr. 3 – Jednoduchý model traťového úseku s obojsmerným automatickým blokom

V treťom modeli je modelovaný obojsmerný automatický blok na jednokolažnej trati (obrázok 3). Tento model je oproti druhému modelu zmenený použitím ďalších objektov:

- objekt Variable: Tento objekt sa používa pre definovanie premenných. Premenné sú používané pre modelovanie údajov, ktorých hodnota sa počas simulácie mení. V premennej sa vždy nachádza nejaká hodnota. Hodnota premennej sa môže meniť volaním nejakej funkcie. V treťom modeli sú použité dve premenné. V prvej sa nachádza informácia o tom, pre ktorý smer jazdy je udelený traťový súhlas. To znamená, že v premennej je informácia o tom, v ktorom smere po traťovom úseku môžu jazdiť vlaky. Táto premenná je typu boolean. Hodnota „True“ znamená, že vlaky môžu jazdiť od začiatku po koniec traťového úseku (zľava doprava). Hodnota „False“ znamená, že vlaky môžu jazdiť od konca po začiatok traťového úseku. Druhá premenná slúži na sledovanie počtu vlakov, ktoré sa nachádzajú na traťovom úseku. Traťový súhlas je možné prepínať iba vtedy, keď je v tejto premennej hodnota 0.
- objekt Schedule: Tento špeciálny komponent umožňuje definovať, ako sa podľa definovaného vzoru menia v čase určité hodnoty. V tomto modeli sa objekt Schedule používa pre definovanie príchodov vlakov v stanovený čas. Použité sú dva objekty tohto typu. Jeden pre príchod vlakov z jednej alebo druhej strany traťového úseku.

V treťom modeli je upravených niekoľko zjednodušení, ktoré sa nachádzajú v prvom a druhom modeli. Vo vývojovom diagrame tretieho modelu sú tieto vylepšenia:

- Aby bolo možné zobrazit' všetky možné stavy oddielových návestidiel, vo vývojovom diagrame simulačného modelu sa nachádzajú tri objekty hold. Tretie oddielové návestidlo bolo v druhom modeli modelované zjednodušene.
- Pre modelovanie pohybu okolo každého oddielového návestidla sú použité tri objekty trainMoveTo.
- Na začiatku vývojového diagramu sa nachádzajú dva objekty trainSource. Každý pre generovanie entít vlakov prichádzajúcich z jednej strany traťového úseku.

V treťom modeli môžeme existenciu entít vlakov interpretovať takto:

1. Pri spustení simulácie sú „rozsvietené“ oddielové návestidlá v tom smere, v ktorom je povolená jazda vlakov. Toto je spustené zavolaním funkcie „inic“. Oddielové návestidlá v smere, v ktorom jazda nie je povolená, sú zhasnuté (čierny).
2. V jednom z dvoch objektov trainSource je na základe údajov v jednom z objektov Schedule vytvorená entita vlaku.
3. Entita vlaku vstúpi do objektu trainMoveTo (prejdikodch). V tomto objekte je entite odovzdaná inštrukcia pre presun po prvé oddielové návestidlo. Hodnota premennej sledujúcej počet vlakov v traťovom úseku je zvýšená o jednu.
4. Entita vlaku je poslaná do objektu queue. Pokiaľ je objekt hold blokovaný, musí v objekte queue čakať. Je tomu tak vtedy, ak je v priestorovom oddiele za prvým návestidlom nejaký vlak. Ak je tento priestorový oddiel voľný, entita vlaku sa presúva do objektu trainMoveTo (prejdizaodch). Toto je druhý objekt trainMoveTo, ktorý patrí k prvému oddielovému návestidlu. Jeho úlohou je riadiť presun od prvého návestidla, kým prvé vozidlo vo vlaku neprejde za toto návestidlo. Tretím objektom trainMoveTo (prejdina1) sa riadi presun vlaku k druhému oddielovému návestidlu.

5. Entita vlaku je po dosiahnutí druhého oddielového návestidla (opustenie objektu `prejdina1`) poslaná do objektu `queue1`. Čaká v ňom, ak je objekt `hold1` zablokovaný. Objekty `queue1` a `hold1` prislúchajú k časti vývojového diagramu, ktorou sa modeluje činnosť druhého oddielového návestidla. Ich činnosť je rovnaká ako pri objektoch `queue` a `hold` použitých pre modelovanie činnosti prvého oddielového návestidla.
6. Entita vlaku prechádza tromi objektmi `trainMoveTo`, ktoré prislúchajú k druhému oddielovému návestidlu. Prvý z objektov (`prejdiknav1`) slúži na presun vlaku kúsok za druhé oddielové návestidlo (kým za toto návestidlo neprejde prvé vozidlo vlaku). Vstúpením do tohto objektu sa zablokuje objekt `hold1`. Druhý z objektov `trainMoveTo` (`prejdiza1`) sa používa na presun celého vlaku za druhé oddielové návestidlo – tým sa uvoľní celý priestorový oddiel medzi prvým a druhým návestidlom. Po opustení objektu `prejdiza1` sa odblokuje modul `hold` patriaci k prvému oddielovému návestidlu. Tretí objekt `trainMoveTo` (`prejdinakon`) riadi presun vlaku k tretiemu oddielovému návestidlu.
7. Pre modelovanie jazdy vlaku okolo tretieho oddielového návestidla sa entita vo vývojovom diagrame modelu opäť posielala cez postupnosť objektov `queue`, `hold` a `trainMoveTo` (`queue2`, `hold2`, `prejdikza2`, `prejdiza2`, `prejdivon`). Činnosť týchto objektov je rovnaká ako rovnakých objektov použitých pri druhom oddielovom návestidle.
8. Entita vlaku opúšťa model cez posledný objekt vývojového diagramu (`zrusVlak`). Hodnota premennej, v ktorej sa sleduje počet vlakov na traťovom úseku, sa zmenší o jedna.

Tretí simulačný model obsahuje zmenu traťového súhlasu. To znamená, že sa v ňom mení povolený smer jazdy vlakov cez traťový úsek. Aby sa traťový súhlas zmenil iba vtedy, keď sa na traťovom úseku nenachádza žiadny vlak, v modeli sa nachádza premenná „`obsadTrať`“. Zmena traťového súhlasu je možná iba vtedy, keď je jej hodnota rovná nule. Iba vtedy sa hodnota v premennej „`smer`“ môže zmeniť z `true` na `false` alebo naopak. Symboly oddielových návestidiel sú aktivované vždy po generovaní príchodu vlaku do traťového úseku.

## ZÁVER

Univerzálne simulačné nástroje (ako napríklad `AnyLogic`) ponúkajú zvyčajne po skončení simulácie rôzne štatistické vyhodnotenia výstupných údajov. Bežné je napríklad vyhodnotenie priemerného času stráveného entitou v systéme, priemerné trvanie obsluhy v zdrojoch, priemerná dĺžka frontov a priemerné časy čakania vo frontoch, tiež priemerné využitie zdrojov.

V nástroji `AnyLogic` sa nachádza niekoľko objektov, ktoré je možné použiť pre zapisovanie výstupných údajov. Sú nimi objekty `TimeMeasureStart` a `TimeMeasureEnd`. Dvojica týchto objektov ohraničuje vo vývojovom diagrame modelu úsek, v ktorom sa sleduje pobyt entity. Ďalším užitočným objektom je objekt `TextFile`. Tento umožňuje zápis do súboru z niektorého objektu modelu, napr. z objektu `queue`. Možné je tiež zapisovať vývoj stavu niektorej premennej.



V prípade modelu automatického bloku by bolo možné zapísané výstupné údaje použiť pre vyhodnocovanie polohy oddielových návěstidiel, ich počtu a v konečnom dôsledku pri skúmaní priepustnej výkonnosti traťového úseku (2), (3).

*Príspevok bol pripravený v rámci riešenia projektov VEGA 1/0296/12 „Verejné obslužné systémy s férovým prístupom k službe“ a APVV-0760-11 „Navrhovanie férových obslužných systémov na dopravných sieťach“.*

## **POUŽITÁ LITERATURA**

- (1) ADAMKO, N., KLIMA, V., MÁRTON, P. Designing railway terminals using simulation techniques. *International Journal of Civil Engineering*, 2010, roč. 8, č. 1, s. 57-67, ISSN 1735-0522.
- (2) GAŠPARÍK, J., ŠIROKÝ, J., PEČENÝ, L., HALÁS, M. Methodology for assessing the quality of rail connection on the network. *Communications : scientific letters of the University of Žilina*, 2014, roč. 16, č. 2, s. 25-30, ISSN 1335-4205.
- (3) GAŠPARÍK, J., ZITRICKÝ, V. Aspects of railway capacity and occupation time estimation. *Journal of civil engineering and architecture*, roč. 8, č. 3, s. 322-331, ISSN 1934-7367.
- (4) GRIGORYEV, I. *AnyLogic 6 in Three Days*. AnyLogic North America, USA. 162 s. ISBN 978-0615705675.
- (5) KELTON, W.D., SADOWSKI, R.P., SWETS, N.B. *Simulation with Arena, 5/e*, McGraw-Hill, USA, 656 s. ISBN 978-0-07-352341-5.