

ŽILINSKÁ UNIVERZITA V ŽILINE

FAKULTA RIADENIA A INFORMATIKY

VYTVORENIE MIKROSKOPICKÉHO MODELU POHYBU
ENTÍT NA SCHODISKÁCH V SIMULAČNOM NÁSTROJI
POHYBU CHODCOV PEDSIM

DIPLOMOVÁ PRÁCA

28360220172209

Študijný program: Informačné systémy
Študijné zameranie: Spracovanie dát
Školiace stredisko: Katedra informatiky
Školiteľ: Ing. Michal Varga, PhD.

Žilina, 2017

Bc. Peter Kvasňovský

ABSTRAKT

KVASŇOVSKÝ, Peter, Bc.: *Vytvorenie mikroskopického modelu pohybu entít na schodiskách v simulačnom nástroji pohybu chodcov PedSim*. [Diplomová práca]

Žilinská univerzita v Žiline, Fakulta riadenia a informatiky; Katedra informatiky

Školiteľ: Ing. Michal Varga, PhD.

Stupeň odbornej kvalifikácie: Inžinier

Žilina: FRI ŽU v Žiline, 2017 - 84 s.

Predmetom práce je vytvorenie mikroskopického modelu pohybu chodcov na schodiskách v simulačnom nástroji PedSim. Práca podrobne analyzuje modelovanie rôznych typov schodísk, rámp, eskalátorov a travelátorov v iných simulačných nástrojoch. Na základe tejto analýzy navrhuje editačnú podporu na vytváranie rôznych typov schodísk v nástroji na editovanie infraštruktúry – E3it. Predstavuje štyri metódy na generovanie stupňov schodísk. Práca sa zaoberá pridaním schodísk do infraštruktúry v nástroji PedSim. Na umožnenie pohybu chodca po schodiskách je upravené jeho vnímanie okolitého sveta. Rozširuje mikroskopický model pohybu chodcov o pohyb na schodiskách. Do existujúcej agentovej štruktúry simulačného nástroja pridáva nového agenta zodpovedného za posúvanie chodcov na eskalátoroch a travelátoroch. Overuje správnosť modelovania schodísk a pohybu chodcov na schodiskách.

Kľúčové slová: pohyb chodcov, schodiská, generovanie stupňov schodísk, agentovo orientovaná simulácia

ABSTRACT

KVASŇOVSKÝ, Peter, Bc.: *Creation of microscopic model of entities movement on stairways in pedestrian movement simulation tool*. [Diploma thesis]

University of Žilina, Faculty of Management Science and Informatics, Department of Informatics

Tutor: Ing. Michal Varga, PhD.

Qualification Level: Master in Informatics.

Žilina: Faculty of Management Science and Informatics, University of Žilina, 2017 - 84 p.

The topic of this thesis is the creation of a microscopic pedestrian movement model on stairways in simulation tool PedSim. Thesis analyses in detail the modeling of different types of stairways, ramps, escalators and travelators in other simulation tools. Based on this analysis, it proposes editing tools for creating different types of stairs in the E3it infrastructure editing tool. The thesis presents four methods for generating steps of stairs. It describes the addition of stairways into PedSim's infrastructure. The pedestrian's perception of his surrounding world is modified to allow him to move on stairways. Thesis extends the microscopic model of pedestrian movement, with the microscopic model of pedestrian movement on the stairs. To the existing agent structure of the simulation tool, thesis adds a new agent responsible for shifting pedestrians on escalators and travelators. It verifies the correctness of stairway modeling and pedestrian movement on stairways.

Pod'akovanie

Rád by som sa poďakoval Ing. Michalovi Vargovi, PhD. a doc. Ing. Norbertovi Adamkovi, PhD. za vedenie práce, odborné rady a vecnú kritiku. Moje poďakovanie patrí všetkým, ktorí mi akýmkoľvek spôsobom pomohli. V neposlednom rade ďakujem svojej rodine a kolegovi Bc. Rastislavovi Vargovi za ich podporu počas celého štúdia.

Prehlásenie

Prehlasujem, že som túto prácu napísal samostatne, a že som uviedol všetky použité pramene a literatúru, z ktorých som čerpal.

V Žiline, dňa 3.5.2017

Peter Kvasňovský

Obsah

Úvod	10
1 Ciele práce	11
2 Analýza schodísk.....	12
2.1 Prvky infraštruktúry na prechod medzi poschodiami	12
2.1.1 Schodiská.....	12
2.1.2 Rampy.....	14
2.1.3 Eskalátory	14
2.1.4 Travelátory	15
2.2 Implementácia schodísk v existujúcich simulačných nástrojoch.....	16
2.2.1 AnyLogic.....	16
2.2.2 Viswalk.....	18
2.2.3 Massmotion	19
2.2.4 Pathfinder	20
2.3 Analýza schodísk – Zhrnutie.....	22
3 Editačná podpora.....	23
3.1 Modelovanie infraštruktúry.....	23
3.2 Nástroj na modelovanie infraštruktúry E3it.....	24
3.3 Modelovanie schodísk.....	25
3.3.1 Komponenty schodiska	25
4 Generovanie jednotlivých stupňov schodov	29
4.1 Vytváranie zoznamu hrán schodov v 2D	29
4.1.1 Prvá metóda generovania stupňov.....	32
4.1.2 Druhá metóda generovania stupňov	33
4.1.3 Tretia metóda generovania stupňov.....	35
4.1.4 Štvrtá metóda generovania stupňov.....	44
4.2 Vytvorenie stupňov schodov v 3D	47
5 Pohyb chodcov na schodiskách.....	49
5.1 Simulačný nástroj pre pohyb chodcov – PedSim.....	49
5.2 Reprezentácia sveta počas simulácie.....	50
5.3 Navigácia chodcov	52
5.3.1 Tvorba trasy k cieľovému prvku na úrovni	52
5.3.2 Navigácia chodca medzi úrovňami	54
5.3.3 Smerovanie chodca k cieľovému prvku na schodiskách.....	55
5.4 Chodcove vnímanie okolitého sveta počas simulácie	58
5.5 Mikroskopický model pohybu chodcov	64

5.5.1	Mikroskopický model pohybu chodcov pomocou sociálnych síl	65
5.5.2	Mikroskopický model pohybu chodcov na schodiskách.....	66
5.6	Agent na obsluhu eskalátorov a travelátorov	68
5.6.1	Architektúra ABAsim.....	68
5.6.2	Pridanie nového riadiaceho agenta eskalátorov	70
6	Overenie riešenia	73
6.1	Generovanie stupňov	73
6.1.1	Schodisko pred vstupom do budovy.....	73
6.1.2	Schodisko s medziposchodím prepájajúce dve poschodia na FRI	76
6.2	Pohyb chodcov po schodiskách.....	78
7	Záver	81
	Zoznam použitej literatúry	82
	Príloha 1: CD médium.....	85

Zoznam obrázkov

Obrázok 1: Názvy prvkov schodiska	13
Obrázok 2: Rozmery schodov.....	14
Obrázok 3: Zobrazenie ľudí na eskalátoroch s rôznymi šírkami (Kone, 2016).....	15
Obrázok 4: Zobrazenie ľudí na travelátoroch s rôznymi šírkami (Kone, 2016).....	16
Obrázok 5: AnyLogic - 3D zobrazenie testovacieho simulačného modelu.....	18
Obrázok 6: Viswalk - 3D zobrazenie testovacieho simulačného modelu	19
Obrázok 7: Massmotion - 3D zobrazenie testovacieho simulačného modelu	20
Obrázok 8: Pathfinder - 3D zobrazenie testovacieho simulačného modelu	21
Obrázok 9: Hierarchia infraštruktúry (Čadecký, 2015)	24
Obrázok 10: UML diagram tried geometrických útvarov v nástroji E3it.....	25
Obrázok 11: Geometrické komponenty schodiska	27
Obrázok 12: E3it – nástroj na editáciu infraštruktúry.....	27
Obrázok 13: UML diagram komponentov schodiska.....	28
Obrázok 14: Rôzne situácie pri kreslení schodov	30
Obrázok 15: Plynulé a neplynulé prechody medzi časťami vodiacej čiary	30
Obrázok 16: Hranice schodov.....	31
Obrázok 17: Predloha prezentovaných schodísk	32
Obrázok 18: Vygenerované stupne 1. metóda v 2D	33
Obrázok 19: Vygenerované stupne 1. metóda v 3D	33
Obrázok 20: Určovanie dĺžky hrany stupňa	34
Obrázok 21: Vygenerované stupne 2. metóda v 2D	35
Obrázok 22: Vygenerované stupne 2. metóda v 3D	35
Obrázok 23: Neupravené hranice schodov	38
Obrázok 24: Riešenie problému – zalomený roh.....	39
Obrázok 25: Riešenie problému – vzniknutá slučka 1	40
Obrázok 26: Riešenie problému – vzniknutá slučka 2	40
Obrázok 27: Riešenie problému – vzniknutá slučka, algoritmus	42
Obrázok 28: Nevhodné použitie tretej metódy	43
Obrázok 29: Vygenerované stupne 3. metóda v 2D	43
Obrázok 30: Vygenerované stupne 3. metóda v 3D (1)	44
Obrázok 31: Vygenerované stupne 3. metóda v 3D (2)	44

Obrázok 32: Modifikácia vodiacej čiary	45
Obrázok 33: Vygenerované stupne 4. metóda v 2D	47
Obrázok 34: Vygenerované stupne 4. metóda v 3D	47
Obrázok 35: Rozdiel medzi vygenerovanými stupňami schodov a časťami rampy.....	48
Obrázok 36: UML diagram reprezentácie sveta počas simulácie.....	50
Obrázok 37: UML diagram reprezentácie sveta počas simulácie so schodiskami	51
Obrázok 38: Prekážky na medziposchodí.....	52
Obrázok 39: Zobrazenie hodnôt gradientovej mapy (Čadecký, 2015).....	53
Obrázok 40: Zobrazenie smerových vektorov gradientovej mapy (Čadecký, 2015)	54
Obrázok 41: Graf medzi podlažiami.....	55
Obrázok 42: Smerové vektory na stupňoch ako kolmice na cieľové hrany stupňa.....	57
Obrázok 43: Smerové vektory na stupňoch ako kolmice na os stupňa	57
Obrázok 44: Zbieranie okolitých vrstiev podľa valca	59
Obrázok 45: Stupeň s vyznačenými dielcami a aprox. čiarami (mm).....	60
Obrázok 46: Aproximačné čiary na všetkých stupňoch schodov	60
Obrázok 47: Okolité vrstvy chodca na schodoch	61
Obrázok 48: Stanovenie maximálneho uhla medzi smerovými vektormi stupňov	62
Obrázok 49: Riešenie problému prekryvania stupňov v 2D.....	63
Obrázok 50: Vyhľadávanie okolitých vrstiev na medziposchodiach a schodoch	64
Obrázok 51: Čiastkové vektory sociálnej sily (Kormanová, 2014).....	66
Obrázok 52: Agentový model s pridaným agentom eskalátorov.....	70
Obrázok 53: ABA-graf agenta eskalátory	71
Obrázok 54: ABA-graf procesu na posúvanie chodcov na eskalátoroch a travelátoroch ...	72
Obrázok 55: Rozmery modelovaného schodiska (mm).....	73
Obrázok 56: Rozmery jednotlivých schodov modelovaného schodiska (mm)	74
Obrázok 57: Modelované schodisko.....	74
Obrázok 58: Rozmery schodiska v simulačnom nástroji (mm).....	75
Obrázok 59: Rozmery jednotlivých stupňov schodiska v simulačnom nástroji (mm).....	75
Obrázok 60: Zobrazenie schodiska v simulačnom nástroji	76
Obrázok 61: Rozmery modelovaného schodiska na FRI (mm).....	76
Obrázok 62: Modelované schodisko v budove FRI.....	77
Obrázok 63: Rozmery schodiska na FRI v simulačnom nástroji (mm).....	77
Obrázok 64: Zobrazenie schodiska na FRI v simulačnom nástroji	78

Obrázok 65: Snímka z prvého videozáznamu	78
Obrázok 66: Snímka z druhého videozáznamu	79
Obrázok 67: Snímka z tretieho videozáznamu	79
Obrázok 68: Snímka zo štvrtého videozáznamu.....	80

Zoznam tabuliek

Tabuľka 1 Rozdelenie schodov podľa sklonu (Empate, 2017).....	13
Tabuľka 2 Rôzne typy rámp podľa sklonu (Blaha, 2010)	14
Tabuľka 3 Modifikácia rýchlosti pohybu chodca na schodoch (Oasys, 2015).....	67
Tabuľka 4 Modifikácia rýchlosti pohybu chodca na rampe (Oasys, 2015).....	67

Zoznam algoritmov

Algoritmus 1: Prvá metóda generovania stupňov	32
Algoritmus 2: Druhá metóda generovania stupňov	34
Algoritmus 3: Tretia metóda generovania stupňov.....	36
Algoritmus 4: Vytvorenie neupravenej hranice schodov	37
Algoritmus 5: Riešenie problému – vzniknutá slučka	41
Algoritmus 6: Vytvorenie hrán jednotlivých stupňov	46
Algoritmus 7: Vytvorenie stupňov schodov v 3D	48
Algoritmus 8: Stanovenie počtu dielcov stupňa pre schody.....	59
Algoritmus 9: Vyhľadávanie okolitých vrstiev na schodoch	61
Algoritmus 10: Vyhľadávanie okolitých vrstiev na medziposchodí a úrovni	63

Úvod

Svet okolo nás sa neustále mení, je čoraz viac komplikovaný. Snažíme sa zvyšovať životnú úroveň na celom svete a zlepšovať existenčné podmienky ľudí. Pri tom nám pomáha aj modelovanie a simulácia rôznych systémov. Je jednoduchšie, lacnejšie a vhodnejšie využiť počítačovú simuláciu na experimentovanie so systémom, ako robiť experimenty na skutočnom systéme. Nároky na detailnosť simulácie sa zvyšujú, preto je potrebné vedieť modelovať systém detailnejšie.

Na Fakulte riadenia a informatiky sa vyvíja viacero simulačných softvérov, medzi ktoré patrí aj PedSim – simulačný nástroj pre pohyb chodcov. Tento nástroj umožňuje modelovať pohyb chodcov na rôznych infraštruktúrach, rovnako aj v budovách. V budovách chodci používajú na presun medzi poschodiami, okrem výťahov, schodiská, rampy a eskalátory. Na letiskách ľudia využívajú na rýchlejší presun travelátory (pohyblivé chodníky). V tomto simulačnom nástroji doteraz nebolo možné modelovať tieto prvky a pohyb chodcov po nich. Táto práca túto medzeru vyplňa.

Práca je rozdelená do siedmich kapitol. V prvej kapitole sú prezentované ciele práce. V druhej kapitole sú analyzované prvky infraštruktúry, ktoré slúžia na prechod medzi poschodiami spolu s tým, ako ich modelujú iné simulačné nástroje. Tretia kapitola sa venuje návrhu modelovania schodísk v editorii infraštruktúry. V štvrtej kapitole je podrobne popísaný proces generovania jednotlivých stupňov schodísk. Piata kapitola je venovaná mikroskopickému pohybu ľudí na schodiskách. Venuje sa tiež rozšíreniu simulačného sveta o schodiská. Predstavuje už existujúci mikroskopický model pohybu, ktorý rozširuje o modelovanie pohybu chodcov na schodiskách. Na konci tejto kapitoly je popísaný nový agent, ktorý posúva chodcov na eskalátoroch a travelátoroch. Predposledná kapitola je venovaná overovaniu riešenia. Poslednou kapitolou je záver, v ktorom sú zhrnuté výsledky a prínos predkladanej práce.

1 Ciele práce

Hlavným cieľom diplomovej práce je navrhnuť a implementovať mikroskopický model pohybu ľudí po schodiskách v existujúcom simulačnom nástroji pohybu a správania sa chodcov. Najväčší dôraz je kladený na to, aby sa dali modelovať rôzne typy schodísk (priame, špirálové, s medziposchodiami a iné) a aby sa chodci vedeli na takýchto schodiskách pohybovať. Na naplnenie hlavného cieľa je potrebné dosiahnuť nasledujúce čiastkové ciele:

1. *Oboznámiť sa so schodiskami v reálnom svete a ako ich modelujú iné simulačné nástroje.*

Je dôležité, aby sme preskúmali aj iné nástroje, a zistili, ako modelujú pohyb chodcov na schodiskách. Zaujímá nás tiež ako modelujú samotné schodiská, aby sme prípadne mohli ich poznatky rozviesť.

2. *Navrhnuť a implementovať modelovanie schodísk v simulačnom nástroji.*

V rámci tohto cieľa si predstavíme nástroj na modelovanie infraštruktúry, ktorý je využívaný na modelovanie infraštruktúry v simulačnom nástroji. Rozšírime ho o možnosť modelovať rôzne typy schodísk tak, aby ich bolo možné jednoducho a intuitívne definovať.

3. *Navrhnuť a implementovať generovanie jednotlivých stupňov schodiska.*

Tento cieľ je veľmi dôležitý, pretože vygenerované stupne schodísk musia zodpovedať realite. Pretože chceme modelovať rôzne typy schodísk, musíme navrhnuť a implementovať také riešenie, ktoré bude univerzálne.

4. *Rozšíriť reprezentáciu priestoru v simulačnom nástroji o schodiská.*

Do simulačného sveta pridáme schodiská, na ktorých bude možné vyhľadávať prekážky a iných chodcov. Upravíme tiež chodcove vnímanie okolitého sveta.

5. *Oboznámiť sa s existujúcim modelom mikroskopického pohybu a rozšíriť ho.*

Mikroskopický model pohybu rozšírime tak, aby sa mohol využiť aj pre pohyb chodcov na schodisku.

6. *Pridať agenta, ktorý bude riadiť obsluhu travelátorov a eskalátorov.*

Vytvorený agent bude mať za úlohu posúvať chodcov nachádzajúcich sa na travelátoroch a eskalátoroch.

2 Analýza schodísk

V našej práci sme sa rozhodli, že sa nebudeme sústrediť len na *schodiská*, ale aj na *rampy* (pre vozíčkarov a pod.), pohyblivé schody (*eskalátory*) a pohyblivé chodníky (*travelátory*). V tejto kapitole sa budeme zaoberať tým, aké jednotlivé prvky na prechod medzi poschodiami poznáme. Preskúmame, ako modelujú tieto prvky iné simulačné nástroje a na záver zhrnieme získané poznatky.

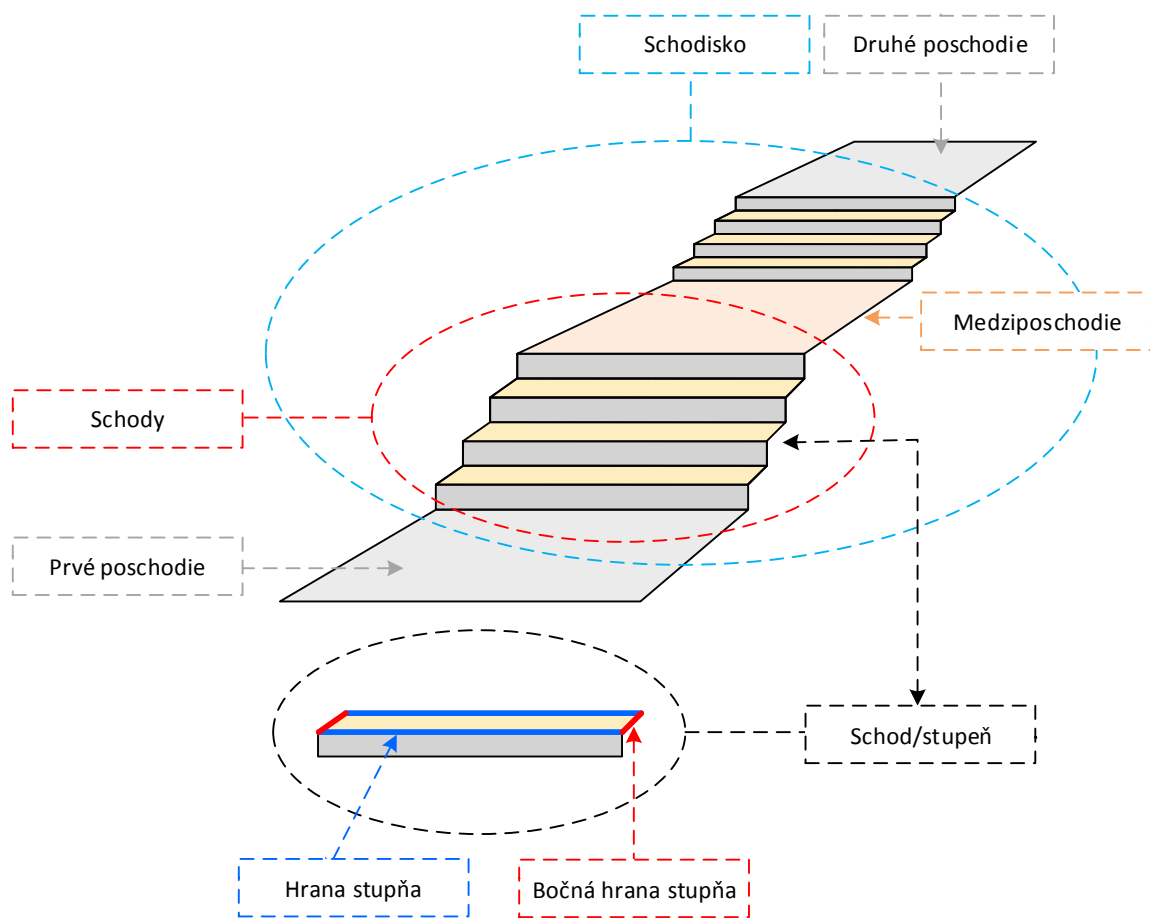
2.1 Prvky infraštruktúry na prechod medzi poschodiami

V tejto kapitole budú bližšie popísané jednotlivé prvky infraštruktúry na prechod medzi poschodiami.

2.1.1 Schodiská

Schodiská slúžia ľuďom na prekonávanie rozdielov medzi jednotlivými výškovými úrovňami napr. poschodiami. Pojmy, ktoré budeme v práci používať sú ukázané na obrázku 1. Podľa Empate (2017) sa pri schodiskách využívajú nasledovné pojmy:

- *schodisko* – stavebná konštrukcia určená na prekonávanie výškových úrovní chôdzou. Skladá sa buď z jedného schodiskového ramena, alebo zo schodiskových ramien a medziposchodí.
- *schodiskové rameno* – súvislá, bezprostredne na seba nadväzujúca postupnosť najmenej troch schodiskových stupňov, spájajúca dve výškové úrovne. My budeme pre zjednodušenie pojmov nazývať schodiskové rameno – **schody**.
- *schodiskový stupeň* – prvok schodiskového ramena, ktorého šírka a výška je v súlade s predpokladaným využitím, komfortom a bezpečnosťou chôdze osôb po schodisku. Pre zjednodušenie pojmov nazývame schodiskový stupeň – **schod** alebo **stupeň**.
- *stupnica (nástupnica)* – nášľapná, horná vodorovná plocha schodiskového stupňa.
- *medzipodesta (medziposchodie)* – odpočívadlo na schodiskovom ramene



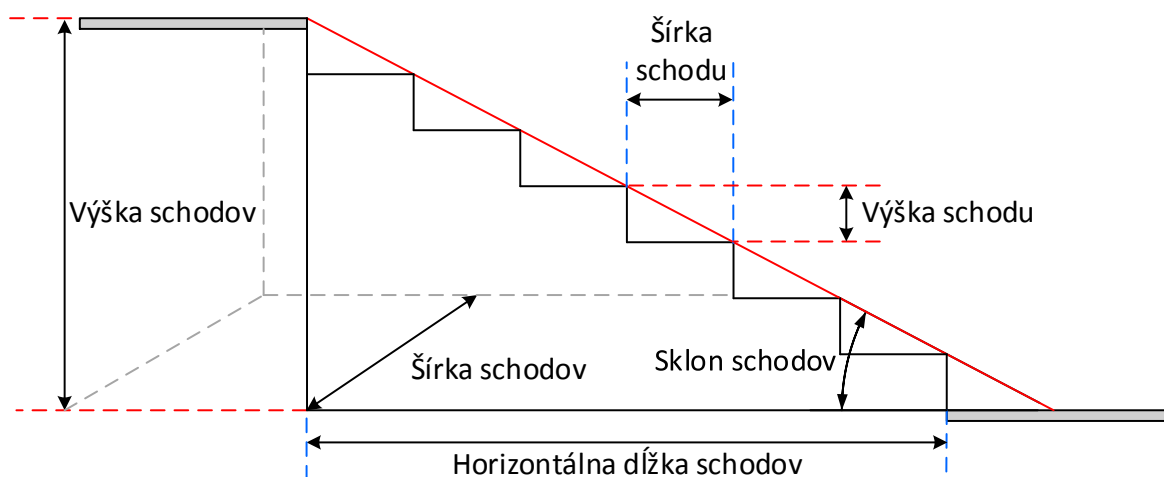
Obrázok 1: Názvy prvkov schodiska

Empate (2017) rozdeľuje schody podľa ich sklonu na 5 typov. Toto rozdelenie je zobrazené v tabuľke 1.

*Tabuľka 1
Rozdelenie schodov podľa sklonu (Empate, 2017)*

Typ	Sklon v stupňoch
Rampové	7° - 20°
Mierne	20° - 25°
Bežné/normálne	25° - 35°
Strmé	35° - 45°
Rebríkové	45° - 58°

Pomenovanie rozmerov schodov je zobrazené na obrázku 2. Pod pojmom šírka schodov rozumieme vzdialenosť od ľavej strany schodov po pravú stranu.



Obrázok 2: Rozmery schodov

2.1.2 Rampy

Rampy slúžia na prekonanie malej výšky medzi dvoma úrovňami. Používajú sa na tých miestach, kde by boli výtahy (alebo iné prostriedky slúžiace na tento účel) nepraktické. Poskytujú bezbariérový prístup pre osoby na vozíčku, ale aj matkám s detskými kočíkmi. Jedným z najdôležitejších parametrov rampy je jej sklon. Blaha (2010) rozdeľuje rampy podľa sklonu na tri typy. Tieto typy sú uvedené v tabuľke 2.

*Tabuľka 2
Rôzne typy rámp podľa sklonu (Blaha, 2010)*

Sklon v stupňoch	Pomer výšky k horizontálnej dĺžke	Maximálna horizontálna dĺžka
2,86°	1:20	9000mm
4,75°	1:12	9000mm
7,13°	1:8	3000mm

Rampa musí byť široká najmenej 1300mm. Za optimálny sklon rampy sa považuje 1:20, pretože poskytuje pohodlný pohyb osobe na vozíčku bez cudzej pomoci. Rampa, ktorá prekročí maximálnu horizontálnu dĺžku (pozri v tabuľke 2), musí byť prerušená odpočívadlom s dĺžkou najmenej 2000mm. Na úplnom začiatku a konci rampy musí byť voľná, prístupná manipulačná plocha min. 1500mm x 1500mm (Fabianyová, 2012).

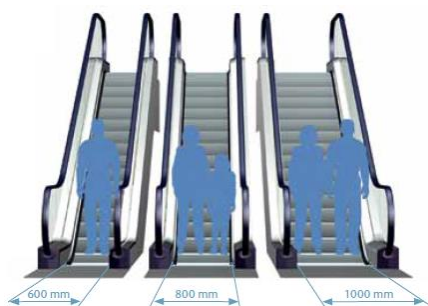
2.1.3 Eskalátory

Eskalátory slúžia na prepravu chodcov vo vertikálnom smere. Majú oproti výtahom vyššiu prepravnú kapacitu. Používajú sa tam, kde by boli výtahy nepraktické. Eskalátor má

oproti výťahu tú výhodu, že ak prestane pracovať, ľudia ho môžu používať ako normálne schody.

Eskalátory sa zvyčajne vyrábajú vo verziách so sklonmi 30° a 35° (Blaha, 2009). Za najefektívnejšie riešenie sú považované eskalátory so sklonom 35° , pretože na ich umiestnenie nie je potrebný veľký priestor a poskytujú dostatočnú prepravnú kapacitu (Schindler, 2007).

Eskalátory sú dostupné v šírkach 600mm, 800mm alebo 1000mm (Blaha, 2009). Na stupni eskalátora širokého 600mm môže stáť len jeden pasažier. Odporúčajú sa na také miesta, kde nie je možné postaviť širší eskalátor. Na stupni eskalátora so šírkou 800mm môže stáť dospelá osoba s dieťaťom alebo dospelá osoba s batožinou. Na stupeň eskalátora, ktorý je široký 1000mm sa zmestia vedľa seba dve dospelé osoby (Kone, 2016).



Obrázok 3: Zobrazenie ľudí na eskalátoroch s rôznymi šírkami (Kone, 2016)

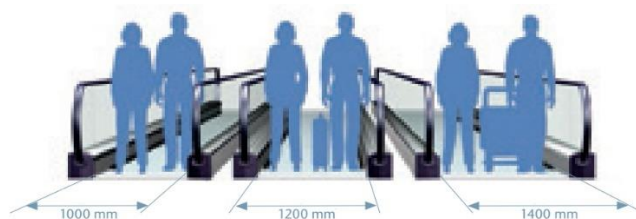
Eskalátory s rýchlosťou 0,5m/s sú najčastejšie využívané v komerčnom sektore, pretože pri tejto rýchlosti dostatočne rýchlo a bezpečne prepravujú chodcov. Na železničných staniach alebo v metre sa používajú eskalátory s rýchlosťou 0,6m/s alebo 0,65m/s. Eskalátory môžu dosahovať rýchlosť až 0,75m/s ale táto rýchlosť nie je z bezpečnostných dôvodov odporúčaná (Schindler, 2007).

2.1.4 Travelátory

Travelátory sú pohyblivé pásy, ktoré slúžia na horizontálnu prepravu osôb. Nachádzajú sa na miestach, kde ľudia potrebujú prekonávať veľké vzdialenosti s batožinou, nákupnými vozíkmi alebo s vozíkmi na batožinu napr. na letiskách. Podľa Schindler (2007) rozdeľujeme travelátory na základe ich sklonu na:

- horizontálne travelátory, ktorých sklon sa pohybuje medzi 0° a 6° .
- šikmé travelátory, ktorých sklon môže byť 10° , 11° a 12° .

Horizontálne travelátory sú dostupné v šírkach 800mm, 1000mm, 1200mm a 1400mm. Šikmé travelátory bývajú široké 800mm a 1000mm. Najširšia verzia je najviac populárna preto, lebo chodci môžu mať so sebou nákupné vozíky alebo vozíky na batožinu (Schindler, 2007). Travelátory so šírkou 800mm umožňujú, aby vedľa seba stáli jeden dospelý s dieťaťom alebo dospelý s batožinou. Travelátory široké 1000mm, 1200mm a 1400mm umožňujú stáť dvom dospelým osobám vedľa seba, pri širších môže mať jeden dospelý so sebou i batožinu (Kone, 2016).



Obrázok 4: Zobrazenie ľudí na travelátoroch s rôznymi šírkami (Kone, 2016)

Travelátory dosahujú rovnaké rýchlosti ako eskalátory, ale môžu dosahovať rýchlosť až 0,9m/s, ktorú sa neodporúča používať z bezpečnostných dôvodov (Schindler, 2007).

2.2 Implementácia schodísk v existujúcich simulačných nástrojoch

Predtým, ako budeme vytvárať vlastný návrh na implementáciu schodísk, preskúmame ako riešia modelovanie rámp, schodísk, eskalátorov a travelátorov iné simulačné nástroje pre pohyb chodcov. V každom nástroji vytvoríme model na testovanie, a ku každému uvedieme aj obrázok testovacieho modelu.

2.2.1 AnyLogic

AnyLogic (AnyLogic, 2017) je simulačný nástroj, ktorý podporuje rôzne paradigmy modelovania (agentovo orientované modelovanie, systémovú dynamiku, procesovo orientované modelovanie). Tento nástroj je napísaný v jazyku JAVA. Zobraziť priebeh simulácie je možné v 2D ale aj v 3D zobrazení. AnyLogic je komerčný nástroj, je tiež dostupný aj ako vzdelávacie resp. samo-vzdelávacie nástroj zdarma.

Na simuláciu chodcov v nástroji AnyLogic existuje knižnica s názvom Pedestrian Library. Táto knižnica umožňuje vytvárať rôzne modely, v ktorých sú chodci dôležitým faktorom, napr. stanice metra, nákupné centrá a pod. K dispozícii sme mali AnyLogic 7 Personal Learning Edition 7.3.6, ktorá obsahovala okrem iného aj túto knižnicu. Chodci sa pohybujú po rovinách pomocou modelu sociálnych síl.

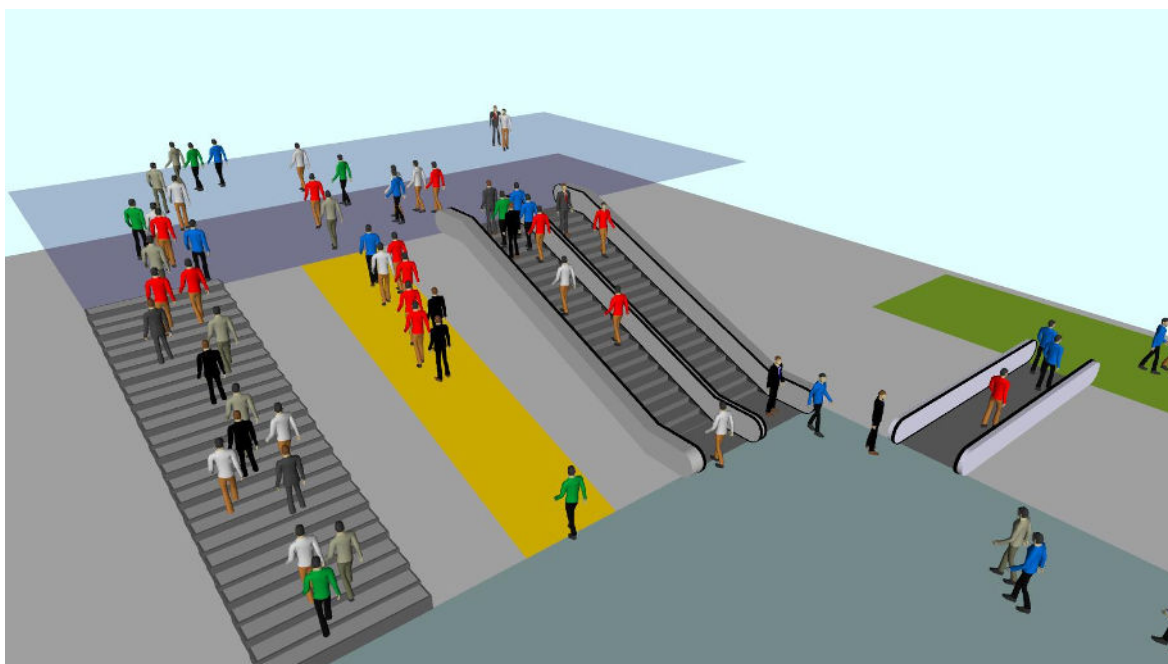
Rampa sa modeluje ako naklonená rovina. Kreslí sa pomocou obdĺžnikovej oblasti a úsečky, ktorá určuje jej sklon. Táto úsečka môže byť určená smerom a uhlom sklonu (uhol, ktorý zvierá s podlahou) alebo sa priamke zafinuje výška, ktorá určí sklon našej oblasti. Po tejto oblasti sa chodci pohybujú ako po rovine, ale AnyLogic umožňuje pomocou popisovača oblasti obmedziť alebo násobiť rýchlosť chodcov, zmeniť priepustnosť danej oblasti a pod. Keďže v našej oblasti musia chodci chodiť pomalšie, môžeme jej nastaviť to, aby spomaľovala chodcov, ktorí po nej chodia. Takto je možné docieľiť, aby chodci chodili ako po naklonenej rovine. Na to, aby sa chodci skutočne presunuli na ďalšie poschodie je nutné, aby na konci rampy (začiatku nasledujúceho poschodia) boli nakreslené dve úsečky. Tieto úsečky sa musia použiť v kombinácii s komponentom diagramu „zmena poschodia“ na to, aby chodci skutočne prešli na nasledujúce poschodie.

V nástroji sa *schodiská* modelujú rovnako ako rampa. Zmena je len v zobrazení rampy a to taká, že sa miesto rampy zobrazia jednotlivé stupne schodov. Preto počas simulácie budú zobrazené stupne schodov ale v skutočnosti sa chodci budú pohybovať po rampe.

Eskalátory sa modelujú pomocou tzv. skupiny eskalátorov. Pre každú takúto skupinu sa musí určiť, koľko eskalátorov má obsahovať. Skupina musí byť rovná (nesmie byť zatočená do strany). Jej sklon sa určuje rovnako ako pri rampe. AnyLogic pre ňu poskytuje rôzne nastavenia ako: rýchlosť, šírku stupňa, môže sa nastaviť aj strana, po ktorej majú kráčať a aké percento chodcov bude kráčať po eskalátoroch. Každý eskalátor musí mať určený smer, ktorým prepravuje ľudí – zhora nadol alebo zdola nahor. Navyše je možné eskalátor vypnúť (nebude fungovať, ale chodci budú môcť po ňom prechádzať) alebo zavrieť (chodci nebudú môcť prechádzať po eskalátore).

Pohyblivý chodník nie je samostatný komponent, ale rovnaký efekt je možné dosiahnuť tak, že začiatok aj koniec eskalátoru bude v rovnakej výške. Takýto eskalátor bude dostatočne podobný pohyblivému chodníku.

Na obrázku 5 je možné vidieť z ľavej strany – schody, rampu, dva eskalátory a pohyblivý chodník v nástroji AnyLogic.



Obrázok 5: AnyLogic - 3D zobrazenie testovacieho simulačného modelu

2.2.2 Viswalk

Simulačný nástroj Viswalk (PTV Group, 2017) je zameraný na simuláciu chodcov. Demo verzia programu PTV Viswalk 9.00-03 mala obmedzenú funkcionálnosť, ktorá sa však nevzťahovala na náš objekt záujmu. Chodci sa pohybujú pomocou modelu sociálnych síl.

Kreslenie *rampy* prebieha tak, že najskôr sa zvolí začiatkový bod, potom koncový bod a následne sa zvolí šírka. Takýmto spôsobom sa dajú vytvárať len obdĺžnikové rampy. Explicitne treba zvoliť poschodie pre obidva konce rampy. Viswalk poskytuje veľké množstvo nastavení, ale vyberieme len tie, ktoré sú pre nás zaujímavé. Možnosť nastavenia veľkosti otvoru v nasledujúcom poschodí podľa výšky stropu alebo vzdialenosťou od konca rampy. Ďalej je možné vytvorenie prekážky pod rampou, ktorá slúži na to, aby chodci nevrážali hlavami do rampy, keď popod ňu prechádzajú. Táto prekážka je automaticky vytvorená a nastavuje sa pri nastaveniach rampy buď minimálnou výškou, kadiaľ môže chodec prejsť alebo vzdialenosťou od začiatku rampy. Je možnosť nastaviť aj hrúbku samotnej rampy.

Schodisko sa vytvára rovnako ako rampa, ale má ďalšie nastavenia, ktorými je možné ho prispôbiť. Jednotlivé stupne schodov je možné definovať buď ich počtom (napr. 20), alebo stanovením výšky alebo šírky stupňa. Pri zadaní jednej vlastnosti sa ostatné automaticky prepočítajú. Prepočet je triviálny, pretože výška schodov a horizontálna dĺžka sú známe.

Eskalátor je takmer rovnaký ako schody, ale pridávajú mu ďalšie nastavenia. Je možné nastaviť smer a rýchlosť eskalátora, koľko percent chodcov a aký typ chodcov má kráčať na eskalátore, či majú stojaci chodci stáť vpravo, vľavo alebo náhodne. Taktiež sa dá nastaviť nábeh (železný koberec) a vzdialenosť, ktorú prejde eskalátor bez zdvíhania nahor resp. klesania nadol.

Pohyblivý chodník je v podstate eskalátor, ktorému je možné nastaviť len dĺžku jedného dielika. Správanie chodcov na pohyblivom chodníku je totožné s ich správaním sa na eskalátore.

Na obrázku 6 je možné zľava vidieť schody, rampu, eskalátor a pohyblivý chodník v nástroji Viswalk.



Obrázok 6: Viswalk - 3D zobrazenie testovacieho simulačného modelu

2.2.3 Massmotion

Massmotion (Oasys, 2017) je komerčný simulačný nástroj vyvíjaný firmou Oasys. Na pohyb chodcov využíva model sociálnych síl. Na tvorbu infraštruktúry poskytuje plnohodnotný 3D editor, ktorý umožňuje vytvoriť 3D modely infraštruktúry do posledného detailu. V súčasnosti je zdarma dostupná skúšobná verzia softvéru 8.5.8.0.

Chodci, ktorí prechádzajú smerom nahor po *rampe* sú spomalení na základe jej sklonu. Zaujímavé je, že chodci idúci smerom nadol nemajú modifikovanú rýchlosť vôbec. Nástroj poskytuje rôzne ďalšie nastavenia – smer, obmedzenie toku príchodu chodcov na

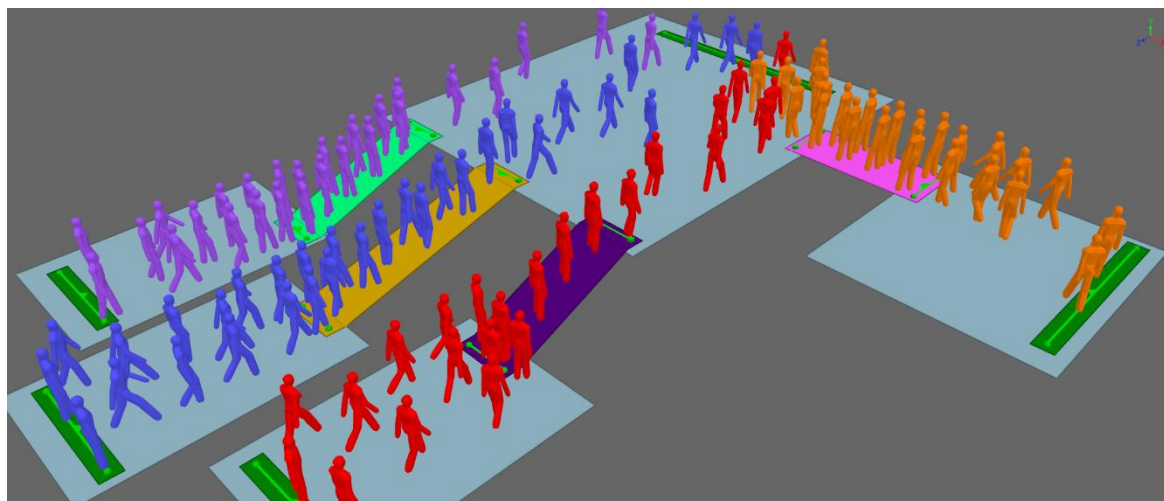
rampu, dodatočnú šírku chodca, obmedzenie rýchlosti. Tiež je možné nastaviť, ktorú stranu budú chodci preferovať, resp. ako veľmi ju budú preferovať.

Schodisko je takmer zhodné s rampou, jediným rozdielom je to, akú má prechádzajúci chodec rýchlosť. Na schodoch majú chodci obmedzenú rýchlosť viac ako na rampe. Pri stúpaní nahor sú chodci pomalší ako pri schádzaní nadol.

Pri *eskalátore* je možné nastaviť jeho rýchlosť, obmedziť tok prichádzajúcich chodcov a nastaviť dodatočnú šírku chodca. Nástroj neumožňuje, aby niektorí chodci kráčali na eskalátore, môžu len stáť.

Aj v tomto nástroji je možné modelovať *pohyblivý chodník* len ako eskalátor, ktorý má dolnú aj hornú časť v rovnakej výške – je horizontálny.

Na obrázku 7 je možné vidieť zľava schody, rampu, eskalátor a pohyblivý chodník v nástroji Massmotion.



Obrázok 7: Massmotion - 3D zobrazenie testovacieho simulačného modelu

2.2.4 Pathfinder

Komerčný simulačný nástroj Pathfinder (Thunderhead engineering, 2017) je od firmy Thunderhead Engineering. Zameriava sa predovšetkým na evakuáciu budov a simulovanie pohybu ľudí. Umožňuje 2D aj 3D vizualizáciu a je napísaný v jazyku JAVA. Model sa vytvára v 3D editore infraštruktúry, ale je možné ho aj importovať ako už existujúcu 3D infraštruktúru z iného programu. K dispozícii sme mali verziu 2016.1.1006.

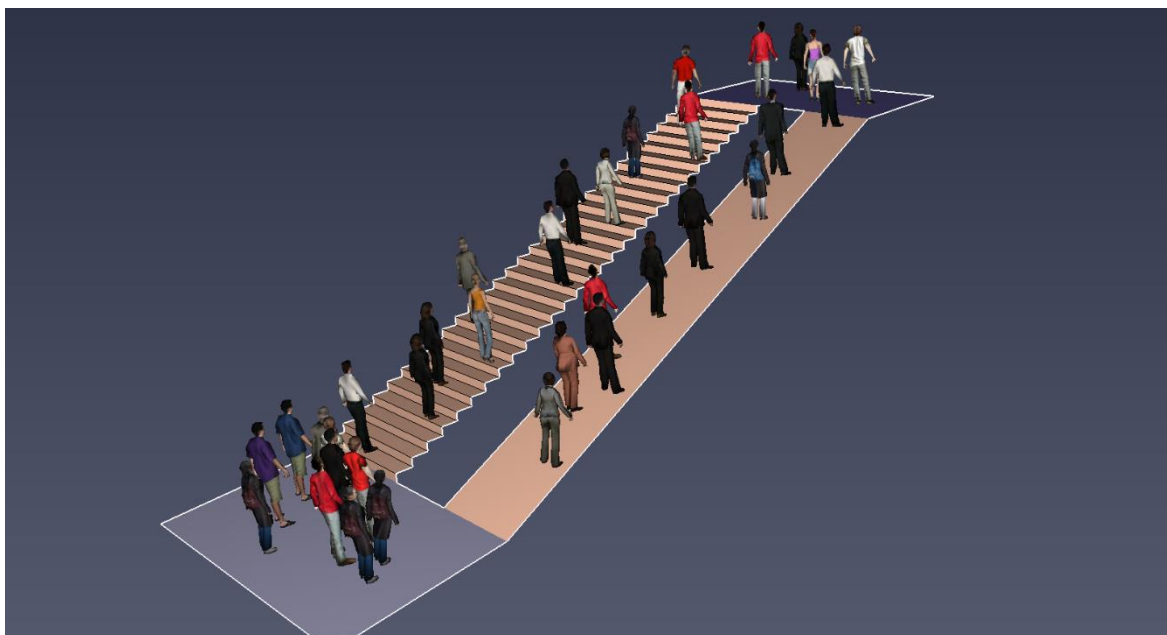
Rampu je možné nakresliť dvoma spôsobmi – zvolením miesta, kde sa má nachádzať pravý dolný roh rampy, alebo zvolením dvoch hrán, ktoré má rampa prepojiť. Sklon rampy

určuje samotný 3D objekt, je ale možné meniť jej šírku, či má byť jednosmerná a dá sa aj určiť, ako sa má na nej meniť rýchlosť pohybu chodcov.

Schodisko sa modeluje podobne ako rampa. Rozdiel je viditeľný pri vytváraní modelu v tom, ako sú zobrazené. Taktiež na schodoch sa chodci pohybujú o trochu pomalšie. Majú dve ďalšie nastavenia a to – výšku a šírku stupňa.

Pathfinder nemá priamu podporu pre *eskalátory* a *pohyblivé chodníky*, ale umožňuje ich napodobniť. Ak chceme napodobniť eskalátor, stačí zmeniť nastavenia schodiska. V prvom rade musíme určiť smer eskalátora a potom zmeniť modifikátor rýchlosti na konštantnú hodnotu. Výsledný efekt je konštantná rýchlosť chodca na eskalátore. Pri zobrazení priebehu simulácie chodec ostane stáť na eskalátore, ktorý ho odvezie správnym smerom. Týmto istým spôsobom sa dá zmeniť rampa na pohyblivý chodník. Nevýhodou takéhoto prístupu je, že sa nedá nijako určiť, či má chodec na eskalátore stáť alebo kráčať. Taktiež to znamená, že sa nedá určiť strana, kde majú chodci stáť, aby ich mohli z druhej strany obchádzať kráčajúci chodci.

Na obrázku 8 sú zobrazené vľavo schody a vpravo rampa v nástroji Pathfinder.



Obrázok 8: Pathfinder - 3D zobrazenie testovacieho simulačného modelu

2.3 Analýza schodísk – Zhrnutie

Z popisov schodísk, rámp, eskalátorov a pohyblivých chodníkov (travelátorov) sme zistili, že sklon veľmi ovplyvňuje typ schodov a použiteľnosť rámp. Taktiež sme zistili, že ako schodiská, tak aj rampy môžu mať medziposchodia vo forme odpočívadiel.

V nástroji AnyLogic je nutné pre spomalenie chodcov na schodoch manuálne nastaviť/obmedziť rýchlosť, akou sa majú pohybovať. Massmotion spomaľuje chodcov na základe sklonu schodov.

Žiaden z týchto nástrojov neumožňuje kreslenie medziposchodí (nepozná pojem medziposchodie – len poschodie), tiež neposkytuje jednoduchý spôsob kreslenia špirálových schodísk. Náš návrh budeme vytvárať tak, aby bolo čo najjednoduchšie modelovať (kresliť) rôzne schodiská spolu s medziposchodiami. Modelovaním schodísk sa zaoberá nasledujúca kapitola.

Pre pohyb chodcov na schodiskách využijeme už existujúci pohybový model. Chodcov, kráčajúcich po schodoch budeme spomaľovať na základe sklonu schodov.

3 Editačná podpora

Na to, aby sme mohli vytvoriť simulačný model, potrebujeme vytvoriť model systému. Pre nás je modelovaný systém infraštruktúra, po ktorej sa pohybujú chodci. Na vytvorenie modelu infraštruktúry používame nástroj E3it, ktorému sa budeme bližšie venovať v nasledujúcej kapitole.

3.1 Modelovanie infraštruktúry

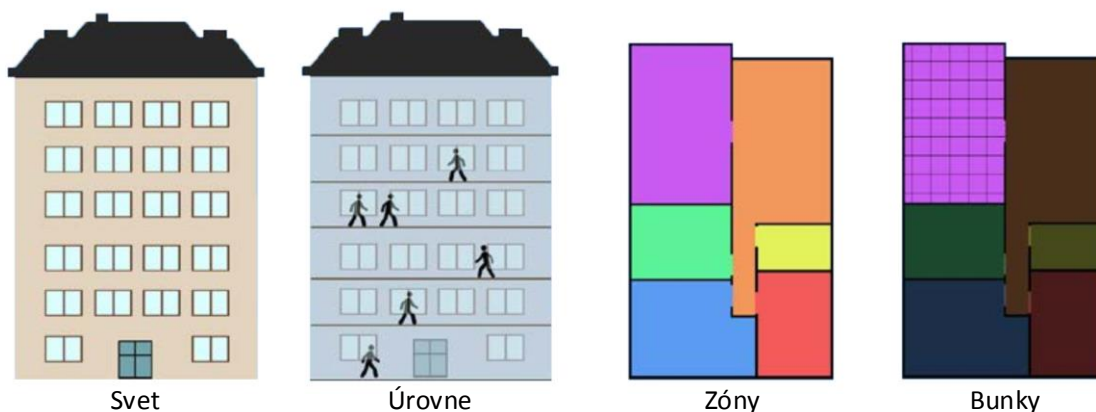
Model infraštruktúry je hierarchicky organizovaný. Každý jeho prvok definuje oblasť, ktorú zastrešuje a menšie prvky, ktoré do neho patria (Varga, 2014). Hierarchiu popíšeme od jej najväčšieho prvku po najmenší. Na obrázku 9 sú zobrazené jednotlivé prvky hierarchie.

Najväčším prvkom, ktorý reprezentuje celý modelovaný systém je *svet*. Svet je koreňom tejto hierarchie.

Úrovne rozdeľujú svet vertikálne. Každá úroveň má špecifikovanú výšku v akej sa nachádza, a preto môže reprezentovať poschodie budovy. Ďalšou funkciou úrovne je vymedzovanie plochy, po ktorej sa môžu chodci pohybovať. V každom modeli musí existovať aspoň jedna úroveň, aby sa mohli po nej chodci pohybovať.

Každá úroveň sa ďalej rozdeľuje na *zóny*. Na jednej úrovni musí existovať aspoň jedna zóna, ale môže ich byť nakreslených aj viac. Zóny umožňujú bližšie špecifikovať detailnosť modelovania a pohybový model chodcov.

Pre každú zónu sa vytvárajú rôzne *mapy*. Mapy sú bunkové siete, ktoré vznikli rozdelením zóny na nehomogénne *bunky*. Existuje viacero typov máp. Každá mapa je zložená z buniek, ktoré obsahujú dáta charakteristické pre danú mapu. Podrobnejšie sa modelu infraštruktúry, mapám a bunkám venuje Čadecký (2015).



Obrázok 9: Hierarchia infraštruktúry (Čadecký, 2015)

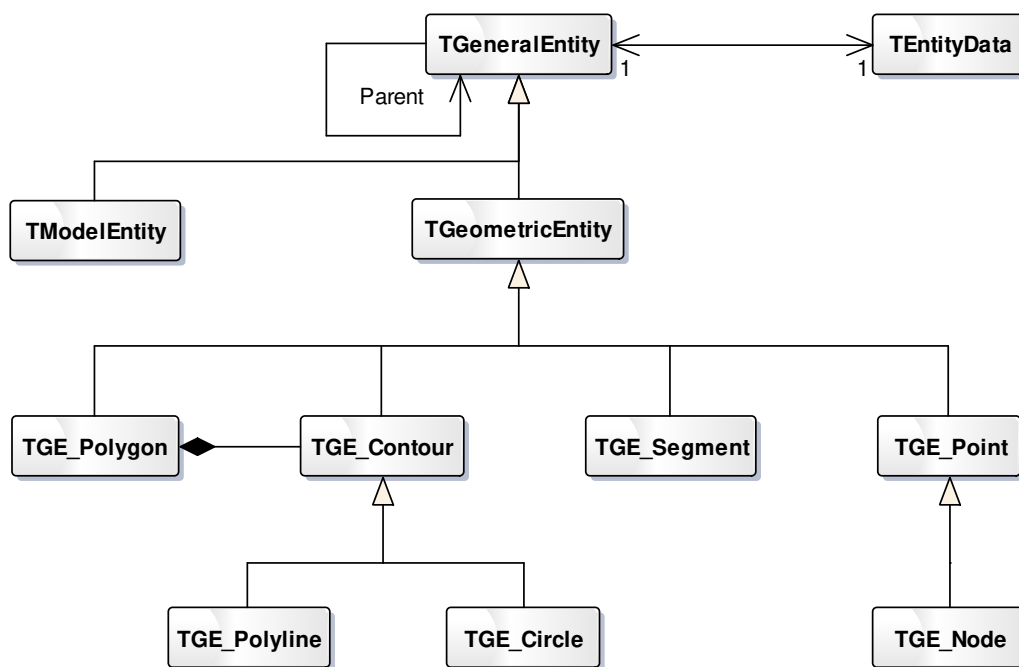
3.2 Nástroj na modelovanie infraštruktúry E3it

Nástroj E3it vyvíja firma Simcon s.r.o. E3it je rozšírený o možnosť kresliť prvky infraštruktúry, ktoré sú typické pre simuláciu chodcov (napr. obslužné miesto, výťah, steny a pod.).

Nástroj umožňuje kresliť rôzne geometrické útvary, ktoré patria do tejto hierarchie:

- Uzol – reprezentuje bod v priestore
- Segment – reprezentuje úsečku alebo oblúk, teda má začiatkový a koncový uzol
- Polyline – reprezentuje krivku, ktorá je zložená z viacerých segmentov
- Polygón – je ohraničený uzavretou krivkou a reprezentuje mnohouholník

Každému z týchto prvkov sú priradené dáta. Pomocou týchto dát určujeme, čo daná geometrická entita reprezentuje. Môžeme nakresliť mnohouholník, ktorému priradíme dáta steny a vytvorili sme mnohouholníkovú prekážku v infraštruktúre. Pre náročnejšie prvky infraštruktúry existuje ešte jeden typ geometrickej entity reprezentujúci zložené prvky. Zložené prvky sa skladajú z viacerých geometrických komponentov (geometrických útvarov). Príkladom takejto entity je výťah, ktorý sa skladá z viacerých mnohouholníkových výťahových staníc. Na obrázku 10 je možné vidieť UML diagram tried, ktorý znázorňuje vzťahy medzi jednotlivými geometrickými entitami.



Obrázok 10: UML diagram tried geometrických útvarov v nástroji E3it

3.3 Modelovanie schodísk

Na to, aby sme mohli kresliť schodiská v editorí 3D infraštruktúry (E3it-e), bolo potrebné, aby sme zdefinovali základné geometrické komponenty, z ktorých budú schodiská zložené. Schodiská, rovnako ako výťahy, sa budú skladať z viacerých geometrických útvarov. Pri navrhovaní komponentov, z ktorých budú zložené, sa dbalo hlavne na to, aby bolo možné nakresliť čo najviac typov schodísk. Schodiská budú vytvorené ako zložené entity skladajúce sa z troch hlavných geometrických komponentov.

3.3.1 Komponenty schodiska

Vytvorené komponenty schodiska zobrazuje obrázok 11. Prvým komponentom je *začiatočná/koncová hrana schodov*. Táto hrana sa kreslí na poschodia, ktoré prepájajú schody. Tento komponent je geometricky reprezentovaný ako rovná polyline (skladajúca sa z dvoch segmentov). Začiatočná/koncová hrana schodov má na mieste, kde sa stretávajú jej segmenty (stred polyline), *uzol na napojenie vodiacej čiary stupňov*, v ktorom je možné začať alebo skončiť kreslenie *vodiacej čiary stupňov*.

Vodiaca čiara stupňov reprezentuje schody a neskôr je použitá na dogenerovanie skutočných stupňov schodov. Môže byť zložená z úsečiek a oblúkov. Pri jej kreslení sa musí brať do úvahy šírka schodov. Vodiacej čiary sú priradené špecifické dáta, ktoré je možné

nastavovať. Prvým nastavením je objekt reálneho sveta, ktorý má reprezentovať – schody, rampu, eskalátor alebo travelátor. Ďalším nastavením je smer. Smer je možné nastaviť pre eskalátory a travelátory, dokonca sa dá nastaviť aj pre schody a rampy – ak chce dizajnér, aby po nich chodci chodili v špecifickom smere. Smer je možné nastaviť zhora nadol, zdola nahor a obojsmerne (obojsmerne len pre schody a rampu).

Vodiaca čiara umožňuje tiež nastaviť počet schodov. Počet schodov je možné zadať aj pomocou šírky alebo výšky jedného stupňa, čo následne automaticky prepočíta počet schodov. Dôsledkom takto zadefinovaných rozmerov stupňov je, že všetky stupne schodov budú mať rovnakú šírku aj výšku. Problém je ten, že existujú také schody, ktorých výška prvého stupňa sa výrazne líši od výšky ostatných stupňov. Schody s prvým stupňom nižším budú mať strmší sklon ako tie, ktoré majú všetky stupne rovnako vysoké.

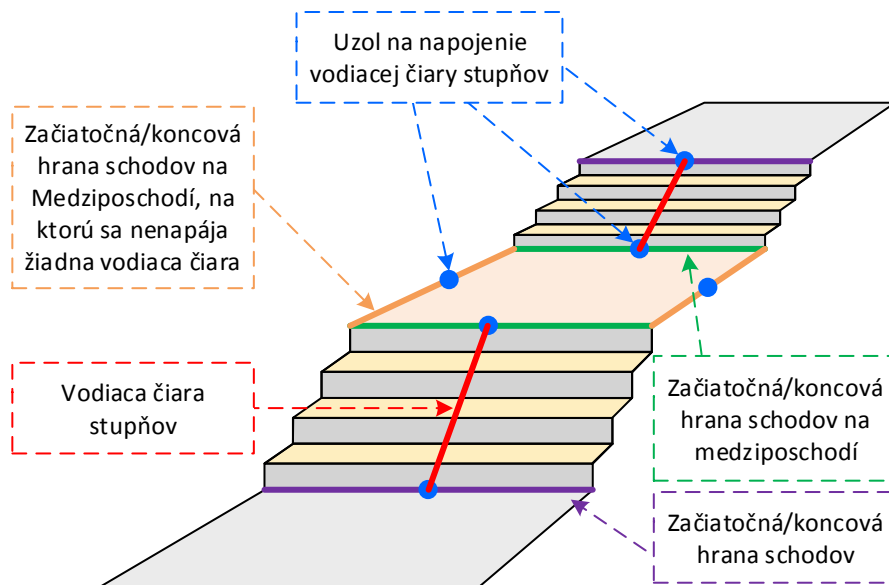
Už sme povedali, že je dôležité modelovať správny sklon schodov, preto je možné ho nastaviť v stupňoch. Hodnota tohto nastavenia sa prepočíta podľa geometrie po nakreslení vodiacej čiary, taktiež pri zmene počtu schodov. Sklon je možné manuálne zmeniť tak, aby zodpovedal modelovaným schodom.

Nastaviť sa dá aj svetlá výška (najmenšia výška medzi schodmi a stropom nad schodmi). Posledným nastavením je metóda generovania stupňov, ktorou sa budú jednotlivé stupne generovať. Po nakreslení vodiacej čiary sa automaticky nastaví metóda generovania, ktorá je pre ňu najvhodnejšia. Generovaniu stupňov a výberu najvhodnejšej metódy generovania sa budeme venovať v nasledujúcej kapitole.

Na kreslenie jednoduchých schodov by boli prezentované komponenty postačujúce, ale na presnejšie popísanie reality bol vytvorený komponent reprezentujúci medziposchodie/odpočívadlo (pozri obr. 1) s názvom *medziposchodie*. Medziposchodie je geometricky reprezentované ako planárny mnohouholník, ktorému je možné v dátach nastaviť výšku, v ktorej sa má nachádzať. Každú hranu tohto mnohouholníka (medziposchodia) môžeme považovať za začiatočnú/koncovú hranu schodov (pozri obr. 11).

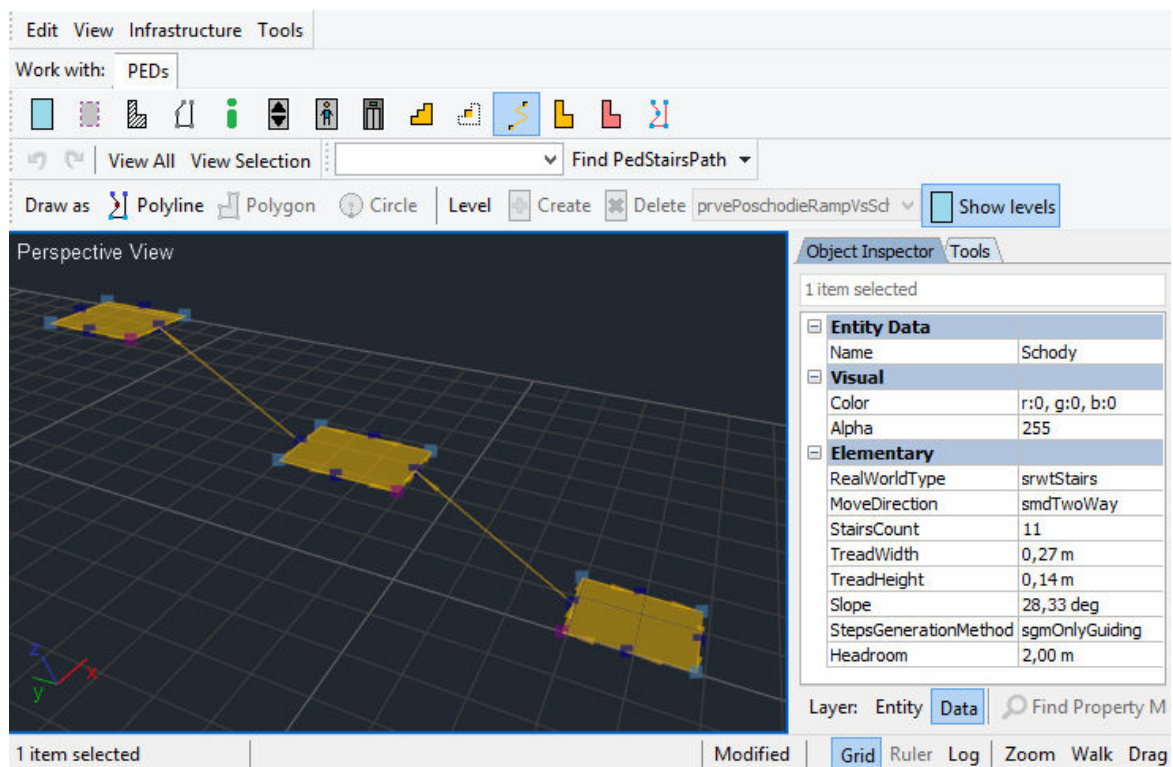
Zložený komponent, ktorý bude reprezentovať schodisko sme nazvali rovnako – *schodisko* (pozri obr. 1). Schodisko bude existovať až vtedy, keď bude obsahovať aspoň jednu vodiacu čiaru stupňov. Spolu s touto vodiacou čiarou musí obsahovať aspoň jednu začiatočnú a jednu koncovú hranu schodov, ktoré sú nakreslené na poschodiach (alebo na

rovnakom poschodí v prípade niektorých travelátorov). Medziposchodie nie je povinnou súčasťou schodiska.



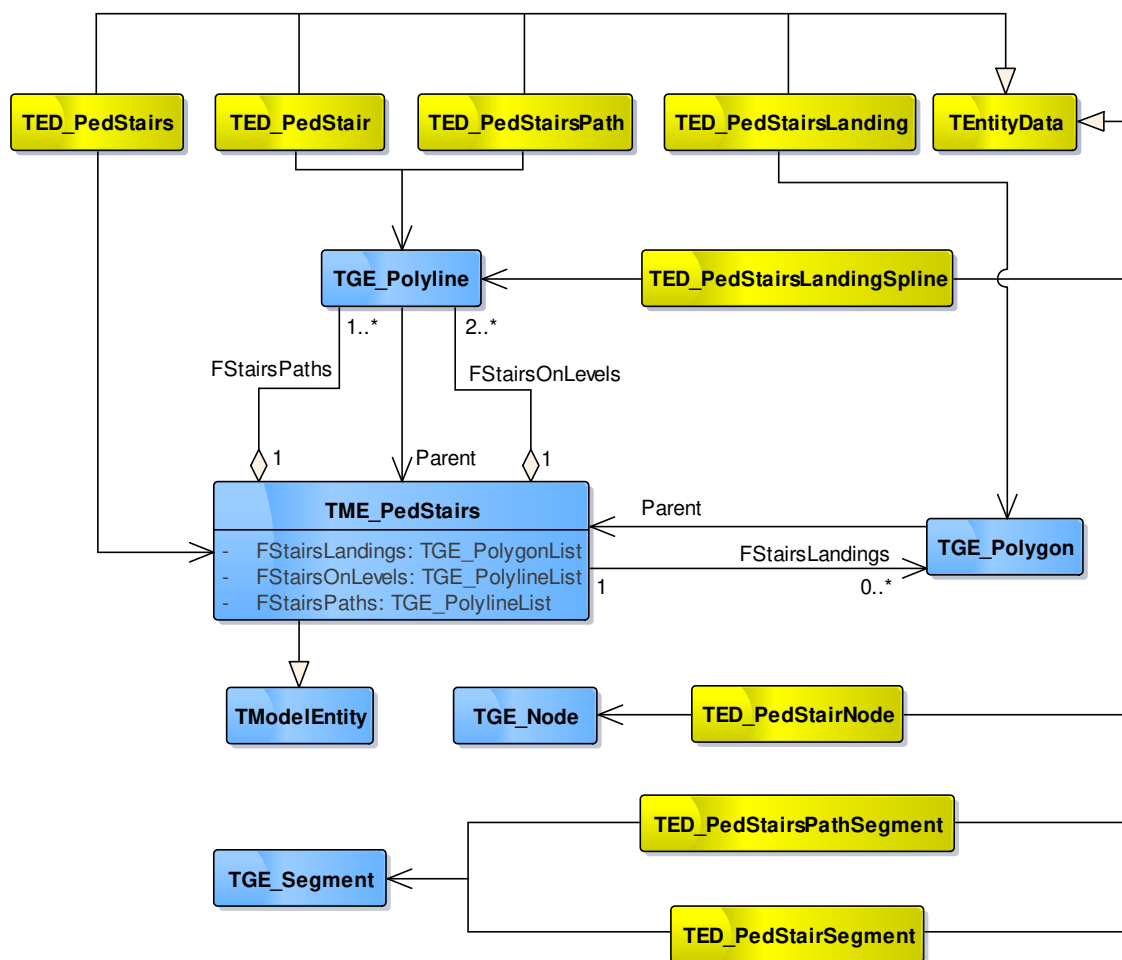
Obrázok 11: Geometrické komponenty schodiska

Na obrázku 12 je vidieť snímku z editora infraštruktúry E3it, zaintegrovaného v simulačnom nástroji PedSim, s možnosťou kresliť komponenty schodiská. Vpravo sa nachádzajú jednotlivé nastavenia pre vodiacu čiaru stupňov.



Obrázok 12: E3it – nástroj na editáciu infraštruktúry

Na obrázku 13 je zobrazený UML diagram vzťahov medzi dátami jednotlivých komponentov schodiska a ich geometrickými reprezentáciami. Je možné si všimnúť, že je zachované oddelenie geometrickej hierarchie od hierarchie dát. Modrou farbou sú označené triedy geometrických entít s prefixom *TGE* (TGeneralEntity) a *TME* (TModelEntity). Žltou farbou triedy dátových entít *TED* (TEntityData). Vidíme, že schodisko má zoznamy geometrických reprezentácií všetkých začiatkových/koncových hrán schodov, vodiacich čiar a medzioschodí.



Obrázok 13: UML diagram komponentov schodiska

4 Generovanie jednotlivých stupňov schodov

Dôležitým cieľom práce bolo navrhnúť také riešenie, ktoré bude univerzálne a jeho výstup sa bude čo najviac podobat' skutočnosti. Na to, aby sme vedeli vygenerovať stupne, potrebujeme začiatočnú a koncovú hranu schodov. Tie musia byť prepojené vodiacou čiarou stupňov. Nezáleží na tom, či začiatočná/koncová hrana schodov patrí medziposchodiu. Celé generovanie prebieha zjednodušene v 2D tak, že sa spraví priemet nakreslených komponentov schodiska do výšky 0 (pohľad zhora) a ten sa následne transformuje do 3D. Takéto riešenie je pre nás postačujúce, pretože sme ochotní zanedbať skutočnosť, že nebudeme môcť generovať tie stupne, ktoré sa budú navzájom prekrývať. Ďalším dôvodom je, že aj nákresy schodísk na plánoch budov bývajú nakreslené v 2D (pohľad zhora). Toto prevedenie nám poskytuje priestor na jednoduchšie rozmýšľanie nad problematikou generovania stupňov. Ďalším prínosom je zmenšená nepresnosť a menej komplikácií pri práci s 2D geometriou.

Nami vytvorené generovanie stupňov bude použité aj pri vytváraní rampy. Na prvý pohľad sa môže zdať, že schody sú o mnoho odlišnejšie od rampy, ale nie je to tak. Ak by sme na obyčajné priame schody položili rovnú plošinu dostaneme rampu s veľkým sklonom. Týmto faktom bolo inšpirované aj generovanie rampy. Výsledná rampa sa bude skladať z menších častí, ktoré budeme nazývať *časti rampy*. Tieto časti sú v podstate stupne schodov, ktoré majú prednú a zadnú hranu v rôznej výške.

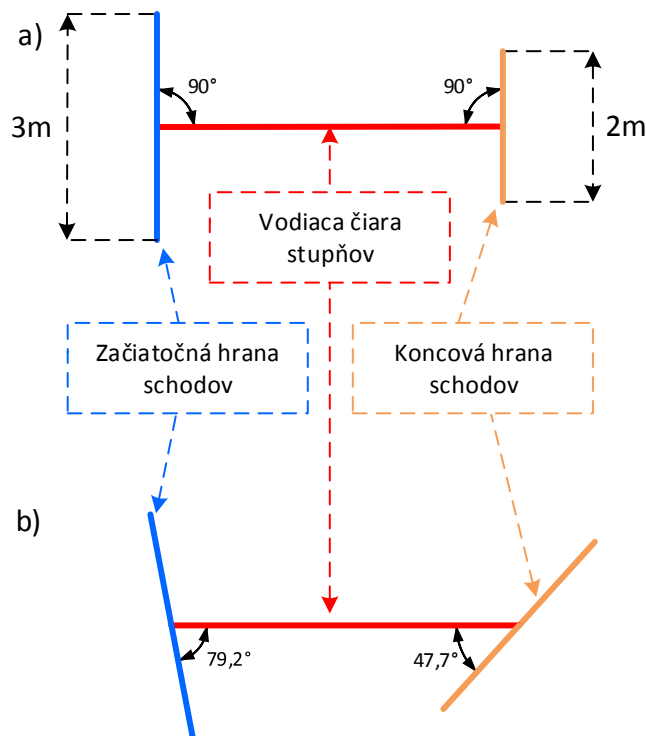
Generovanie stupňov je rozdelené na dve hlavné fázy:

- vytvorenie zoznamu hrán stupňov v 2D (rovnaký algoritmus pre schody aj rampu) a
- vytvorenie konkrétnych stupňov alebo časti rampy (podobné algoritmy pre schody a rampu) zo zoznamu hrán v 3D.

4.1 Vytváranie zoznamu hrán schodov v 2D

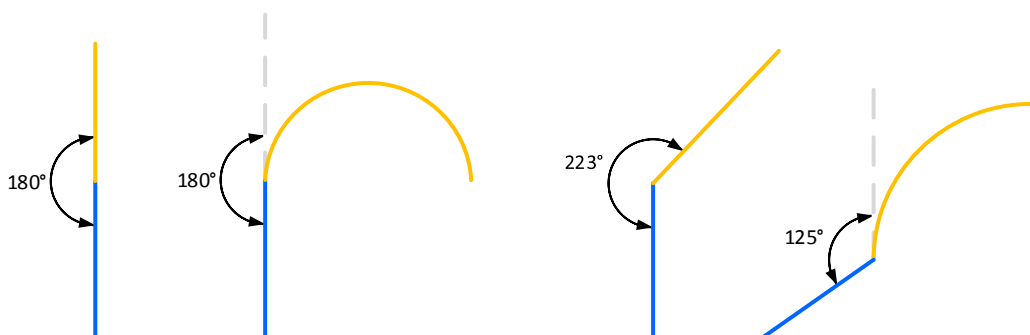
Pri generovaní jednotlivých stupňov treba brať do úvahy nasledovné situácie:

- **Začiatočná a koncová hrana schodov môžu mať rôznu dĺžku** (pozri obr. 14a).
- **Začiatočná/koncová hrana schodov a vodiaca čiara nemusia zvierat' pravý uhol** (pozri obr. 14b).



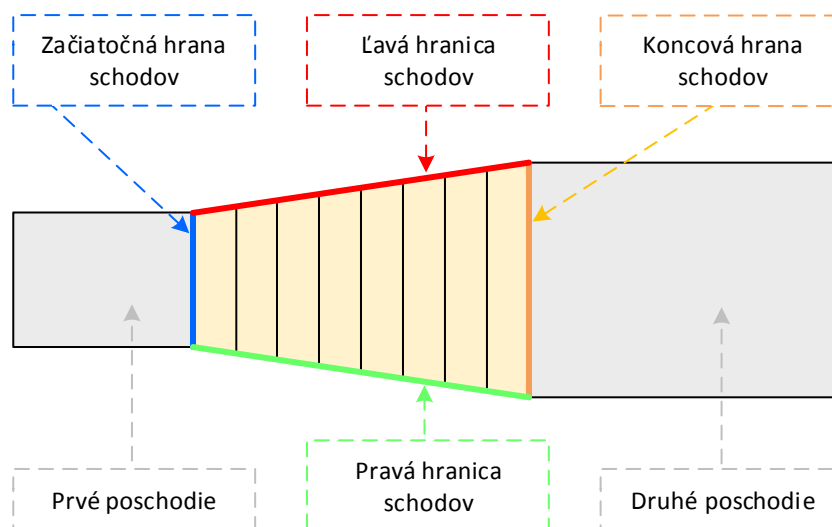
Obrázok 14: Rôzne situácie pri kreslení schodov

- Medzi jednotlivými časťami (segmentmi) vodiacej čiary nemusia byť plynulé prechody. Pod pojmom plynulý prechod je myslené to, že v bode, v ktorom sa stretávajú, zvierajú uhol približne 180° . Pri oblúkoch meriame uhol medzi predchádzajúcim segmentom a dotyčnicou ku oblúku v bode, v ktorom sa stretávajú. Na obrázku 15 je možné vidieť zľava dve úsečky, medzi ktorými je plynulý prechod, úsečku a oblúk s plynulým prechodom, dve úsečky medzi ktorými nie je plynulý prechod, úsečku a oblúk medzi ktorými nie je plynulý prechod.



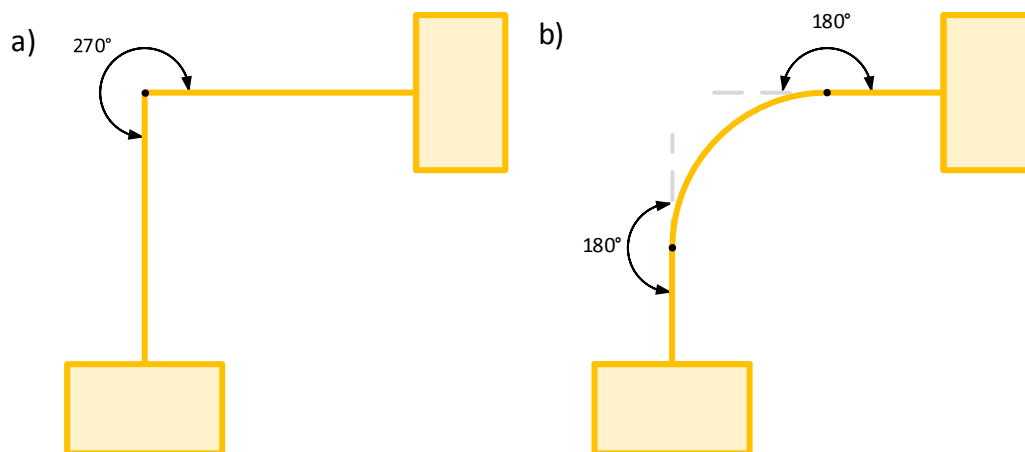
Obrázok 15: Plynulé a neplynulé prechody medzi časťami vodiacej čiary

Aby sme pokryli všetky prípady a zároveň poskytli viacero možností generovania, navrhli sme štyri metódy generovania stupňov. Pri každej metóde sa budeme stretávať s *ľavou a pravou hranicou schodov* (pozri obr. 16). Tieto hranice reprezentujú šírku schodov.



Obrázok 16: Hranice schodov

Ďalej si predstavíme jednotlivé metódy. Nie vždy sa dá metóda použiť na generovanie stupňov. To, či je ju možné použiť, závisí od toho, ako je nakreslená vodiaca čiara. Aby bolo možné danú metódu na generovanie stupňov použiť, musia byť splnené vopred určené požiadavky, ktoré sú uvedené pri popise jednotlivých metód. Pri popise každej metódy uvedieme obrázky 2D a 3D vygenerovaných stupňov pre schodiská, ktoré budú nakreslené podľa obrázka 17. Na obrázku 17a je možné vidieť len časť schodiska (aby boli obrázky s 3D vizualizáciou schodísk prehľadnejšie), pozostávajúce z dvoch medziposchodí a vodiacej čiary, ktorá obsahuje neplynulý prechod. Na obrázku 17b je tiež čiastočné schodisko, ktoré obsahuje dve medziposchodia prepojené vodiacou čiarou. Táto vodiaca čiara je vytvorená z úsečky, oblúka a ďalšej úsečky, medzi ktorými sú len plynulé prechody. V oboch prípadoch sú začiatková aj koncová hrana schodov kolmé na vodiacu čiaru.



Obrázok 17: Predloha prezentovaných schodísk

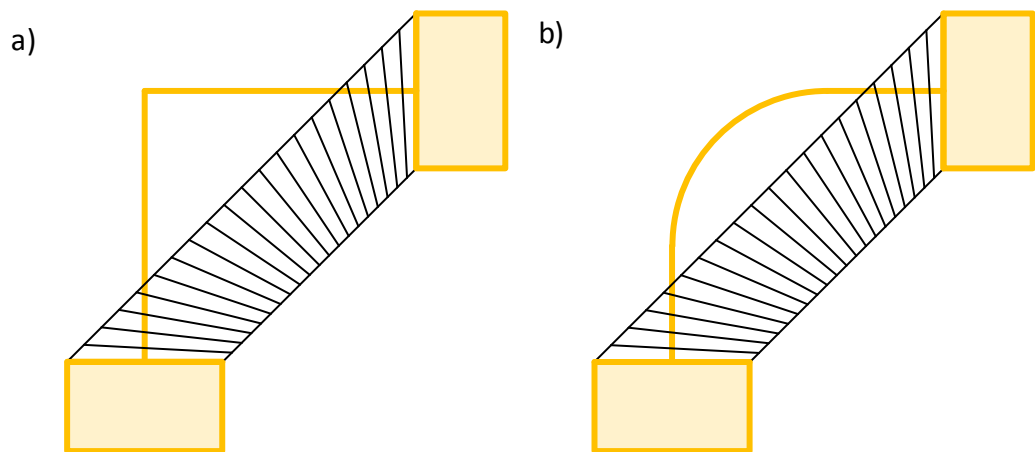
4.1.1 Prvá metóda generovania stupňov

Prvá metóda je základná metóda, ktorá nikdy nezlyhá, čo znamená, že ju je možné použiť vždy. Pri automatickom vyberaní najvhodnejšej metódy bude zvolená v takých situáciách, v ktorých nie je možné použiť žiadnu sofistikovanejšiu metódu. Taktiež je možné, aby si ju dizajnér zvolil v nastaveniach dát pre vodiacu čiaru. Túto metódu je vhodné si vybrať vtedy, ak vodiacu čiaru nepoužívame na to, aby „viedla stupne“, ale len na jednoduché prepojenie hrán schodov. Funguje nasledovne:

1. Vytvoríme ľavú a pravú hranicu schodov ako jednoduché úsečky, ktoré začínajú na začiatkovej hrane schodov a končia na koncovej. Tieto úsečky musia tvoriť spolu so začiatkovou a koncovou hranou schodov konvexný štvoruholník.
2. Hranice rozdelíme rovnomerne na toľko častí, koľko je požadovaných schodov.
3. Hrany stupňov vytvoríme spájaním protilahlých bodov na hraniciach.

Algoritmus 1: Prvá metóda generovania stupňov

Na obrázku 18 je možné vidieť ako vyzerajú vygenerované stupne v 2D. Vidíme, že pre obe schodiská metóda vytvorila rovnaké stupne. Toto je spôsobené tým, že táto metóda ignoruje vodiacu čiaru.



Obrázok 18: Vygenerované stupne 1. metóda v 2D

Na obrázku 19 sú vľavo zobrazené vygenerované stupne schodov v 3D a vpravo sú vygenerované časti rampy. Je možné vidieť, že vygenerované schody a rampa, aj keď sú netradičné, sa podobajú skutočnosti.



Obrázok 19: Vygenerované stupne 1. metóda v 3D

4.1.2 Druhá metóda generovania stupňov

Na to, aby bolo možné použiť túto metódu, a aby bola automaticky vybraná ako najvhodnejšia, musia byť obe nasledovné požiadavky naraz splnené:

- Začiatková a koncová hrana schodov a vodiaca čiara **musia** zvierat' pravý uhol.
- Prechody medzi jednotlivými časťami vodiacej čiary **musia** byť plynulé.

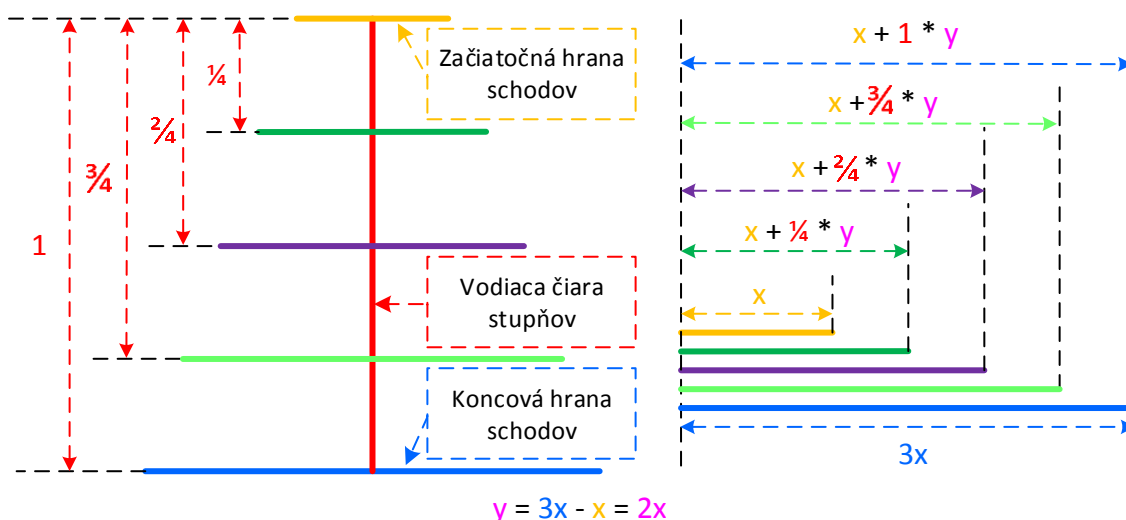
Ak sú splnené požiadavky potom metóda funguje nasledovne:

1. Rozdelíme vodiacu čiaru na toľko rovnako veľkých častí, koľko je požadovaných stupňov.
2. V každom bode, ktorý vznikol rozdelením vodiacej čiary, vytvoríme priamku ako kolmicu na dotyčnicu v aktuálnom bode, a aktuálny bod nazveme stred hrany stupňa.
3. Na vytvorenú priamku nanesieme **vypočítanú dĺžku aktuálnej hrany stupňa** tak, aby jej body boli rovnako vzdialené od stredu hrany stupňa. Táto vzdialenosť určuje šírku schodov v aktuálnom bode na vodiacej čiare.
4. Nájdene dva body sú body novovytvorenej hrany stupňa.

Algoritmus 2: Druhá metóda generovania stupňov

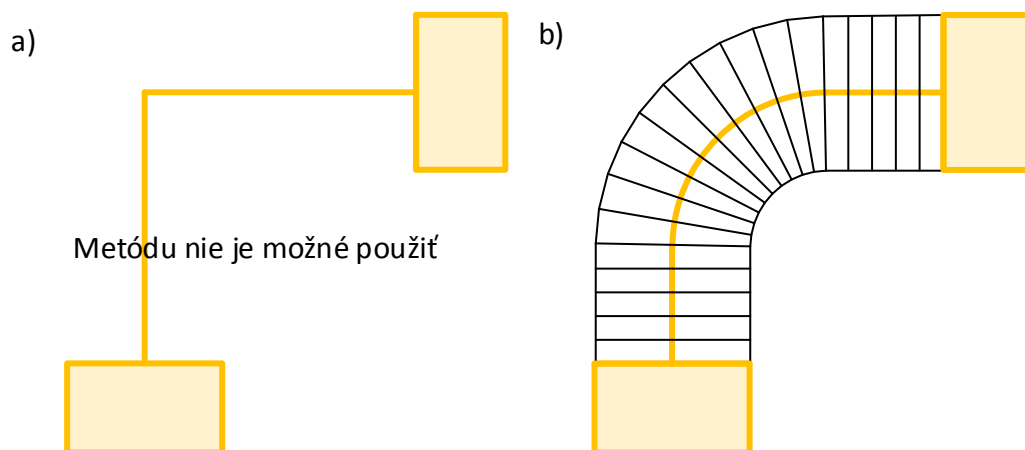
Výpočet dĺžky aktuálnej hrany stupňa

Výpočet dĺžky aktuálnej hrany stupňa je naznačený na obrázku 20. Na tomto obrázku vľavo je zobrazený pomer, ktorý vyjadruje vzdialenosť aktuálnej hrany stupňa od začiatku vodiacej čiary (dĺžka pozdĺž vodiacej čiary od jej začiatku po aktuálnu hranu stupňa) ku celkovej dĺžke vodiacej čiary. Vpravo je možné vidieť ako boli dĺžky jednotlivých hrán stupňov vypočítané. Zvolili sme takú situáciu, aby bola dĺžka koncovy hrany schodov rovná trom dĺžkam začiatkovej hrany schodov. Je možné vidieť, že rozdiel v dĺžkach jednotlivých hrán stupňov závisí od toho, ako ďaleko na vodiacej čiare sa daný stupeň nachádza. Takýmto spôsobom docielime to, že dĺžky jednotlivých hrán budú plynulo prechádzať z dĺžky definovanej začiatkovou hranou schodov do dĺžky definovanej koncovou hranou schodov.



Obrázok 20: Určovanie dĺžky hrany stupňa

Na obrázku 21a nie sú zobrazené žiadne stupne z toho dôvodu, že nie sú splnené podmienky pre fungovanie tejto metódy. Na obrázku 21b je možné vidieť, že jednotlivé hrany stupňov sú kolmé na dotyčnice k vodiacej čiare.



Obrázok 21: Vygenerované stupne 2. metóda v 2D

Tieto úspešne vygenerované stupne schodov sú zobrazené v 3D na obrázku 22 vľavo, pričom vpravo je možné vidieť vygenerovanú rampu.



Obrázok 22: Vygenerované stupne 2. metóda v 3D

4.1.3 Tretia metóda generovania stupňov

Táto metóda vznikla z prvej metódy a bude použitá na generovanie len vtedy, keď si ju dizajnér nastaví. Automatické vyberanie metódy ju nevyberie nikdy, pretože je len logickým doplnkom ostatných metód a je vhodné ju použiť len vo výnimočných situáciách.

Funguje podobne ako prvá metóda:

1. Vytvoríme ľavú a pravú **hranicu schodov podľa vodiacej čiary**.
2. Hranice rovnomerne rozdelíme na rovnaký počet častí ako je požadovaný počet stupňov.
3. Hrany stupňov vytvoríme spájaním protiľahlých bodov na hraniciach.

Algoritmus 3: Tretia metóda generovania stupňov

Vytvorenie hraníc schodov podľa vodiacej čiary

Vytvorenie týchto hraníc nie je jednoduché. Opäť si musíme uvedomiť, že sa nachádzame v 2D priestore. Hranice sa vytvárajú každá samostatne rovnakým algoritmom. Vytváranie hraníc vychádza z druhej metódy generovania stupňov. To, ako druhá metóda vytvára hrany stupňov sme vysvetlili v kapitole 4.1.2.

Vytvorené hranice budú zložené len z úsečiek. Skladajú sa z úsečiek preto, aby sme mohli jednoducho zistiť, kde priamka pretína časť hranice. Ak by boli hranice zložené aj z oblúkov, mohla by priamka preťať takýto oblúk v dvoch bodoch, čo je nevyhovujúce. Úsečku priamka pretne len v jednom alebo v žiadnom bode. Ďalšou výhodou je, že takéto riešenie bude v budúcnosti schopné transformovať na korektné hranice akékoľvek krivky, ktoré môžu pochádzať aj z iných modelovacích nástrojov. Hranice budeme vyznačovať na obrázkoch ako prerušovanú čiernu čiaru. Vytvorenie správnej hranice schodov pozostáva z dvoch častí:

- vytvorenie neupravenej hranice schodov a
- upravenie neupravenej hranice.

Vytvorenie neupravenej hranice schodov

Vytváranie neupravenej hranice schodov prebieha nasledovne:

1. Vytvoríme si prázdny zoznam bodov (dva za sebou nasledujúce body budú reprezentovať úsečku).
2. Postupne prechádzame jednotlivé časti (segmenty) vodiacej čiary a v prípade, že:
 - a) Aktuálny segment je úsečka, vytvoríme na jej začiatku a konci virtuálnu hranu stupňa (podobne ako pri druhej metóde), zoberieme bod tejto hrany (podľa toho či vytvárame pravú alebo ľavú hranicu,

zoberieme buď pravý alebo ľavý bod) a pridáme ho do zoznamu bodov. Virtuálnu hranu zahodíme.

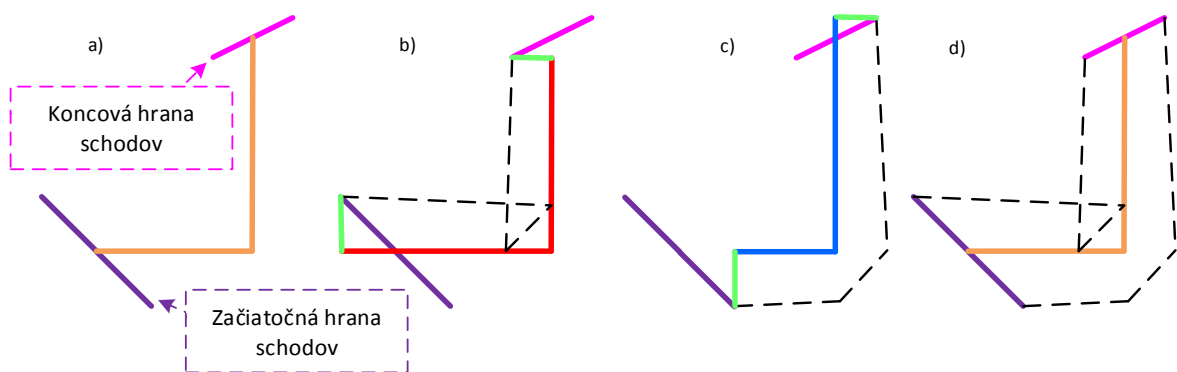
b) Aktuálny segment je oblúk, potom ho rozdelíme na rovnaké časti (dĺžku týchto častí sme určili na 25cm, pričom je možné ju zmeniť). V každom bode, ktorý vznikol týmto rozdelením, vytvoríme virtuálnu hranu stupňa a spracujeme ju rovnako ako v predošlom prípade.

3. Máme naplnený zoznam bodov, ktorý je naša hranica schodov, skladajúca sa z úsečiek. Prípadné redundantné body (body, ktoré v zozname nasledujú priamo za sebou a majú totožné pozície) odstránime zo zoznamu.

Algoritmus 4: Vytvorenie neupravenej hranice schodov

Popísaný algoritmus bude využitelný len v prípade, ak začiatková a aj koncová hrana schodov budú kolmé na vodiacu čiaru. Ak nakreslené schody nespĺňajú túto podmienku, je nutné vytvoriť nové vodiace čiary, pre každú hranicu zvlášť. Tieto nové vodiace čiary budú prispôbené tak, aby bola podmienka kolmosti splnená.

Na obrázku 23a je nakreslené schodisko, ktoré pozostáva zo začiatkovej a koncovej hrany schodov, ktoré sú prepojené vodiacou čiarou. Všimnime si, že ani jedna hrana schodov nie je kolmá na vodiacu čiaru. V tomto prípade musíme vytvoriť dve pomocné vodiace čiary, podľa ktorých budeme vytvárať hranice. Na obrázku 23b je možné vidieť červenou farbou nakreslenú novú pomocnú vodiacu čiaru na vytvorenie ľavej hranice. Zelenou farbou sú nakreslené nové polovičné virtuálne hrany schodov. Pri pôvodnej začiatkovej hrane schodov sme vytvorili polovičnú virtuálnu začiatkovú hranu schodov. Vznikla z dvoch bodov, prvý bod je ľavý bod pôvodnej hrany schodov a druhý bod je najbližší bod na dotýčnici ku vodiacej čiare v jej začiatkovom bode. Koncová polovičná virtuálna hrana vznikla taktiež z dvoch bodov, pričom prvý bod je tiež ľavý bod hrany schodu a druhý bod je najbližším bodom na vodiacej čiare. Na obrázku 23c je možné vidieť modrou farbou vyznačenú pomocnú vodiacu čiaru pre pravú hranicu spolu s jej polovičnými virtuálnymi hranami schodov, ktoré vznikli podobne ako pri predchádzajúcej pomocnej vodiacej čiare. Na obrázku 23d je možné vidieť obe neupravené hranice schodov spolu s pôvodnými hranami schodov a vodiacou čiarou. Vidíme, že vytvorené hranice začínajú aj končia v správnych miestach, pričom rešpektujú fakt, že hrany schodov neboli kolmé na vodiacu čiaru a mali rôznu dĺžku.



Obrázok 23: Neupravené hranice schodov

Upravenie neupravených hraníc schodov

Na obrázku 23d je možné vidieť, že hranice ešte nie sú pripravené na generovanie stupňov. Pri vytvorení neupravených hraníc podľa vodiacej čiary s neplynulými prechodmi vznikajú dva problémy. Tieto problémy sa vyskytujú vždy v pároch, jeden na jednej hranici a druhý na druhej hranici. Problémy sú také, že:

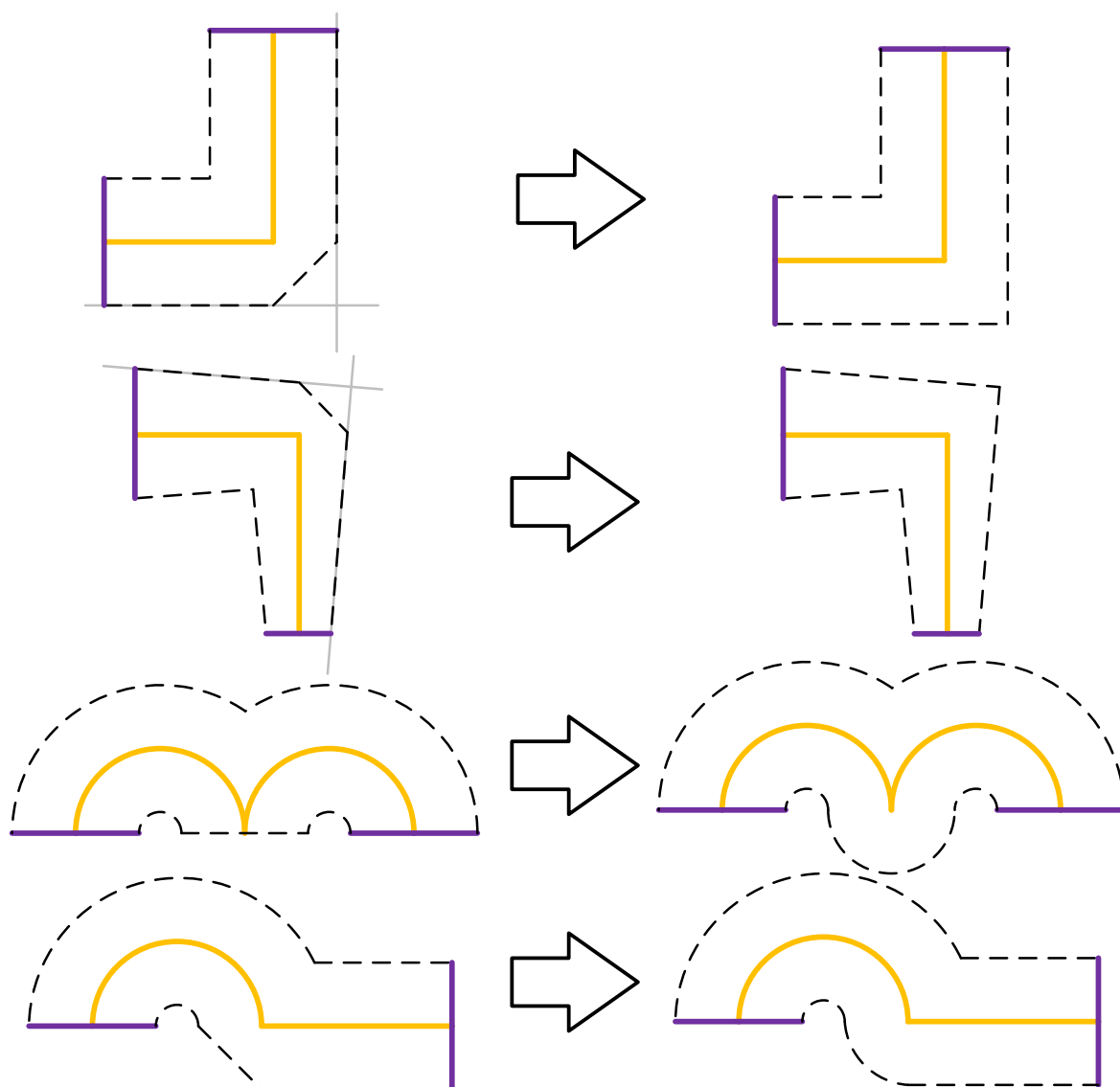
- na jednej hranici sa *zalomí roh* a zároveň
- na druhej hranici *vznikne slučka*

Riešenia týchto problémov musia byť také, aby boli použiteľné aj v prípadoch, keď začiatočná a koncová hrana schodov sú rôzne dlhé. V nasledujúcej časti sa budeme venovať riešeniu týchto problémov.

Riešenie problému – zalomený roh

Zalomenie rohu vznikne vždy pri vytváraní hranice v tom bode vodiacej čiary, kde sa stretávajú dva segmenty s neplynulým prechodom. My sme si zapamätali, ktorá úsečka hranice je zalomeným rohom, preto sa vieme pozrieť na predošlú a nasledujúcu úsečku hranice. Takže zoberieme predošlú a nasledujúcu úsečku a pre každú vytvoríme priamku, ktorá bude prechádzať bodmi danej úsečky. Tam, kde sa tieto dve priamky pretnú, sa bude nachádzať bod, ktorým musí prechádzať hrana. Ak by sa stretával na vodiacей čiare oblúk s úsečkou alebo dva oblúky, potom sa hrana upraví tak, aby miesto úsečky zalomeného rohu bol vytvorený z úsečiek oblúk. Tento oblúk vznikne úplne jednoducho keďže máme zapamätaný bod neplynulého prechodu na vodiacей čiare, ktorý bude našim stredom oblúka. Body, v ktorých bude oblúk začínať a končiť získame zo zapamätanej úsečky hranice, ktorá problém spôsobila. Na obrázku 24 je možné vľavo vidieť čiastočne upravené hranice v rôznych situáciách. Tieto čiastočne upravené hranice nemajú vyriešený problém

zalomeného rohu, ale pre jednoduchosť majú vyriešený problém vzniknutej slučky. Na tomto istom obrázku sú vpravo nakreslené hranice s vyriešenými problémami.



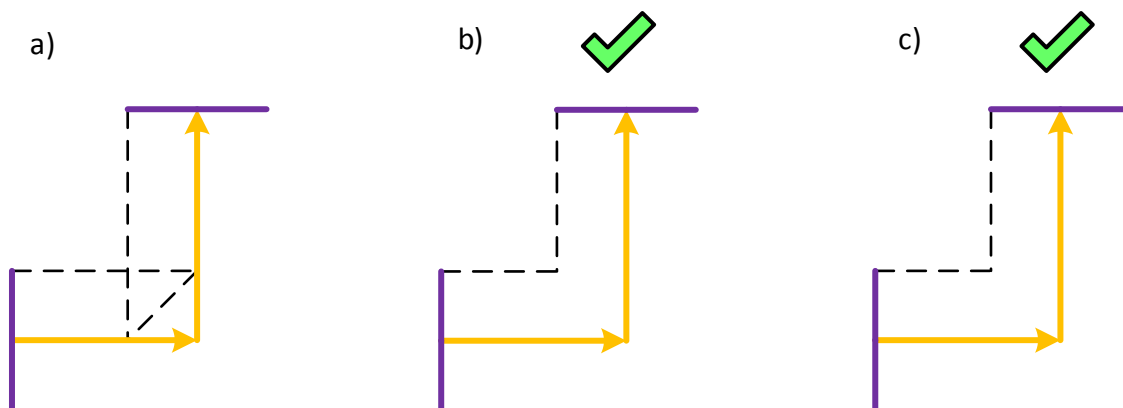
Obrázok 24: Riešenie problému – zalomený roh

Riešenie problému – vzniknutá slučka

Tento problém je možné vidieť na obrázku 23b. Slučkou nazývame tú časť, ktorá vznikne keď pri vytváraní hranice pretne hranica samú seba. Rovnako ako zalomený roh, vzniká slučka pri vytváraní hranice v tom bode vodiacej čiary, kde sa stretnú dva segmenty s neplynulým prechodom.

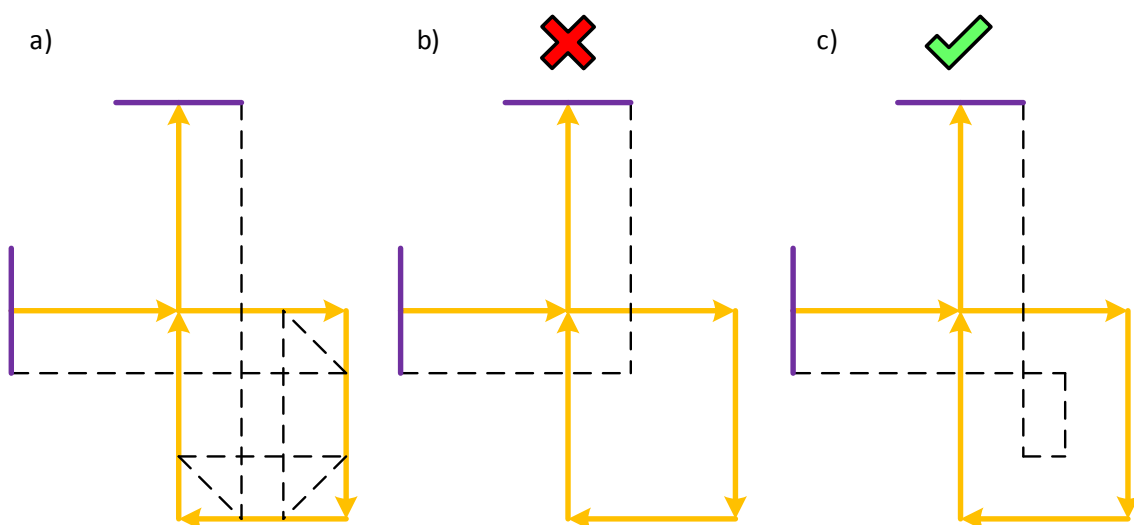
Prvé čo nás napadne je, že stačí len skontrolovať, na ktorých miestach hranica pretne samú seba a upraviť ju tak, aby sa odstránila slučka a hranica prechádzala bodom pretnutia. Na obrázku 25a je zobrazená neupravená hranica, na obrázku 25b je zobrazená hranica, ktorú

sme upravili spomínaným jednoduchým spôsobom a na obrázku 25c je zobrazená očakávaná výsledná hranica po odstránení slučky.



Obrázok 25: Riešenie problému – vzniknutá slučka 1

Lenže toto riešenie nie je správne vtedy, keď vodiaca čiara bude pretínať samú seba. To znamená, že niektoré stupne schodov sa nachádzajú nad predošlými stupňami tých istých schodov. Na obrázku 26a je zobrazené takéto schodisko spolu s jednou neupravenou hranicou, ktorá obsahuje práve tri slučky. Pre lepšie zobrazenie vodiacej čiary sú na nej nakreslené šípky, ktoré zobrazujú ako bola vodiaca čiara nakreslená. Na obrázku 26b je zobrazená upravená vodiaca čiara podľa predošlého jednoduchého návrhu a na obrázku 26c je zobrazená očakávaná, správne upravená hranica. Je možné vidieť, že tento spôsob je nepostačujúci, pretože hranicu upravil v bode pretnutia, ktorý v podstate v 3D neexistuje a je vytvorený 2D priemetom. Preto sme hľadali také riešenie, ktoré nebude trpieť týmto nedostatkom.



Obrázok 26: Riešenie problému – vzniknutá slučka 2

Naše riešenie bude vysvetlené formou algoritmu a aj pomocou obrázka 27. Označme úsečky AB, CD a BD, ktoré sú vyznačené na obrázku 27. Na odstraňovanie slučky využijeme to, že sme si pri vytváraní hranice zapamätali úsečku CD. Preto môžeme odstraňovanie slučky začať z tejto úsečky podľa nasledovného postupu:

Skontrolujeme, či sa úsečky AB a CD pretínajú:

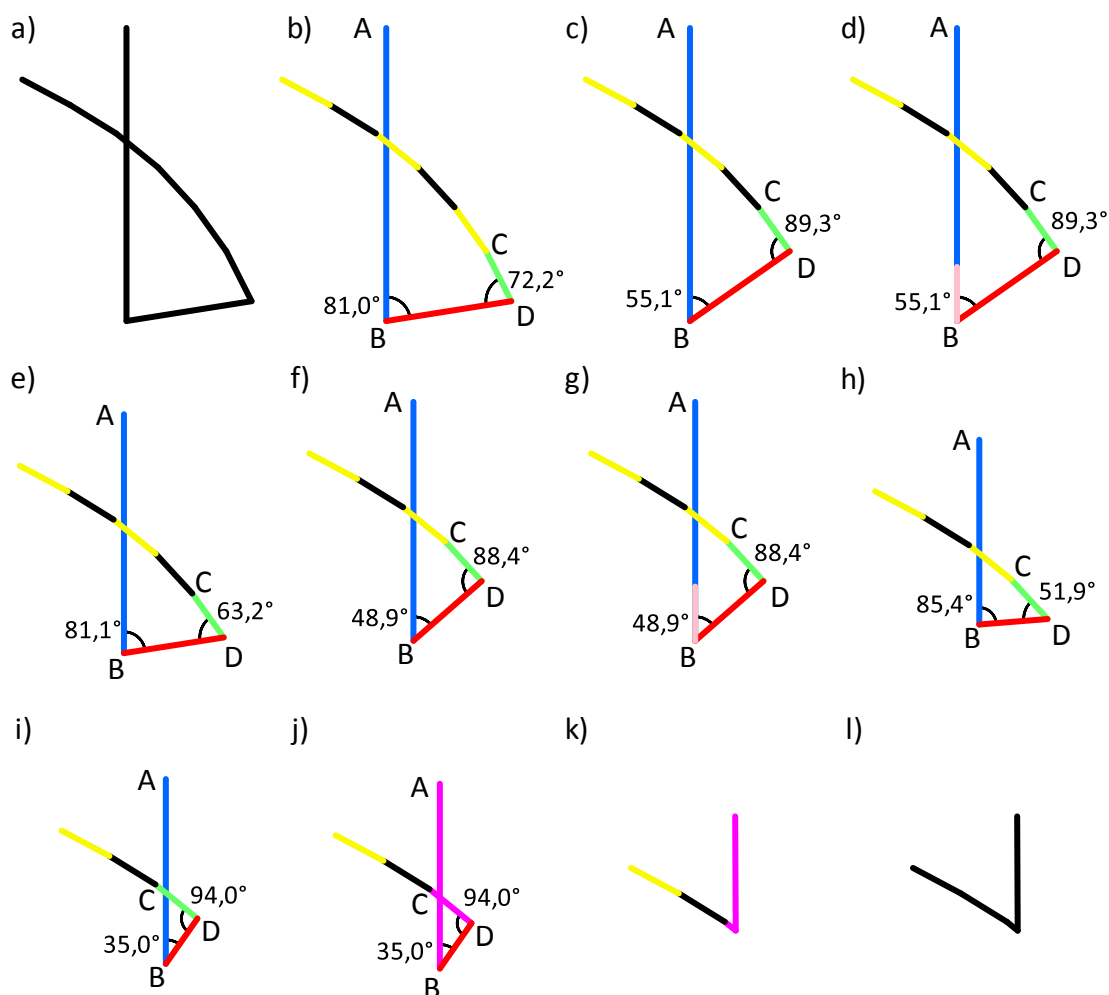
1. Ak áno, našli sme bod, v ktorom slučka zaniká, a podľa neho upravíme hranicu.
2. Ak nie, porovnáme uhly ABD a CDB:
 - a) Ak je uhol ABD menší, skrátime úsečku AB o vzdialenosť rovnajúcu sa minimu z $(|AB|, |CD|)$.
 - b) Inak to znamená, že je uhol CDB menší, čiže skrátime úsečku CD o vzdialenosť rovnajúcu sa minimu z $(|AB|, |CD|)$.
3. Ak skrátime úsečku o celú jej dĺžku (čiže táto úsečka zanikne), zoberieme ďalšiu úsečku z hranice a označíme ju príslušnými písmenami zo zaniknutej úsečky. Pokračujeme bodom 1. Takýmto spôsobom sa rovnomerne približujeme k bodu, v ktorom slučka zaniká.

Algoritmus 5: Riešenie problému – vzniknutá slučka

Na obrázku 27 je zobrazený priebeh tohto algoritmu odstraňovania slučky, ktorý si teraz prejdeme postupne:

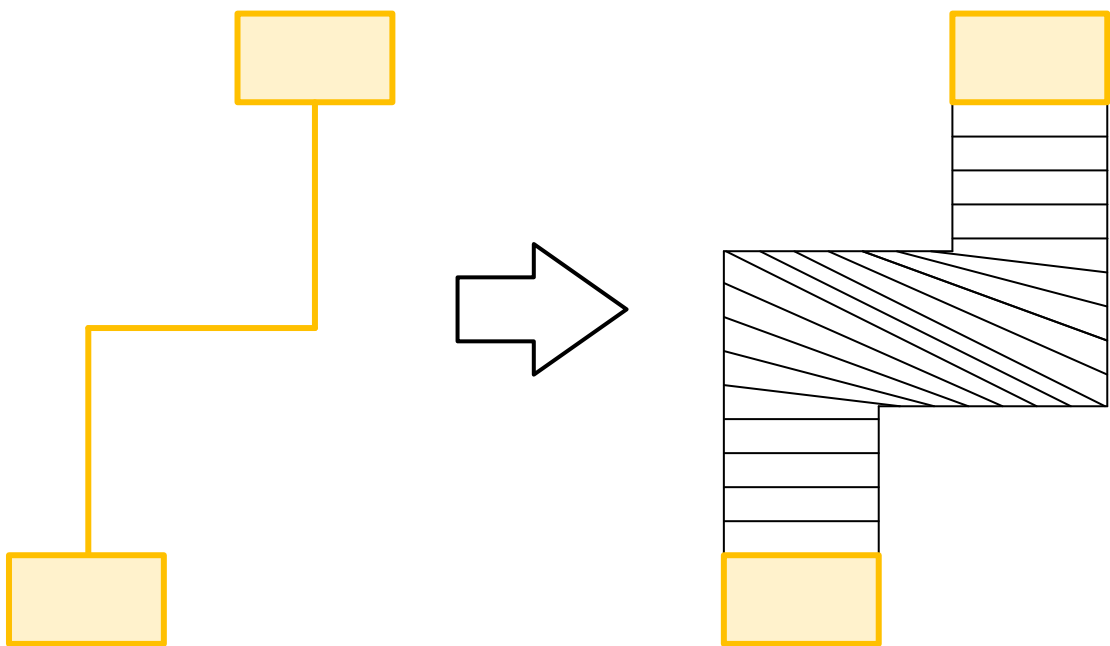
- a) Na obrázku je možné vidieť, že sme si zvolili časť hranice, ktorá obsahuje slučku, ktorú budeme odstraňovať. Táto hranica pozostáva z ôsmych úsečiek.
- b) Odteraz budeme na lepšie rozoznanie striedať žltú farbu s čiernou na označovanie neaktívnych úsečiek. Modrou farbou je vyznačená aktívna úsečka AB, zelenou farbou CD a červenou farbou BD. Taktiež budeme zverejňovať hodnoty uhlov ABD a CDB.
- c) V predchádzajúcom bode je vidieť, že uhol CDB bol menší. Zároveň úsečka CD bola kratšia ako úsečka AB, preto zanikla a aktívnou úsečkou CD sa stala nasledujúca úsečka.
- d) Vidíme, že uhol ABD je menší. Keďže je úsečka AB dlhšia ako úsečka CD, budeme skracovať úsečku AB o dĺžku úsečky CD, čo je vyznačené bledoružovou farbou.
- e) Skrátili sme úsečku AB o dĺžku CD.

- f) V predchádzajúcom bode sme zistili, že uhol CDB bol menší a spolu s faktom, že CD bola kratšia ako AB, CD zanikla. Zobrali sme nasledujúcu úsečku hranice (čiernu), prehlásili ju za aktívnu a označili ako CD.
- g) Rovnako ako v bode d).
- h) Situácia po skrátení úsečky AB o dĺžku CD.
- i) Zánik CD, prehlásenie ďalšej hrany v poradí za aktívnu.
- j) Teraz sme zistili, že sa nám obe aktívne úsečky AB a CD pretínajú.
- k) Upravili sme hranicu podľa bodu, kde sa pretáli AB a CD.
- l) Hotová upravená hranica s odstránenou slučkou.



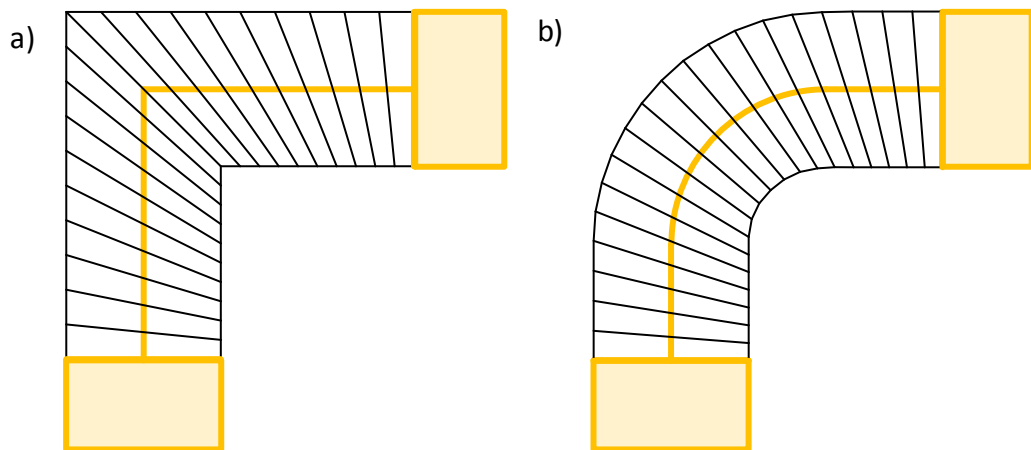
Obrázok 27: Riešenie problému – vzniknutá slučka, algoritmus

Na záver tejto podkapitoly si ukážeme jedno nevhodné použitie, ktoré znázorňuje obrázok 28. Je vidieť, že vygenerované schody pravdepodobne v reálnom svete neexistujú (možno vo veľmi výnimočnom prípade).



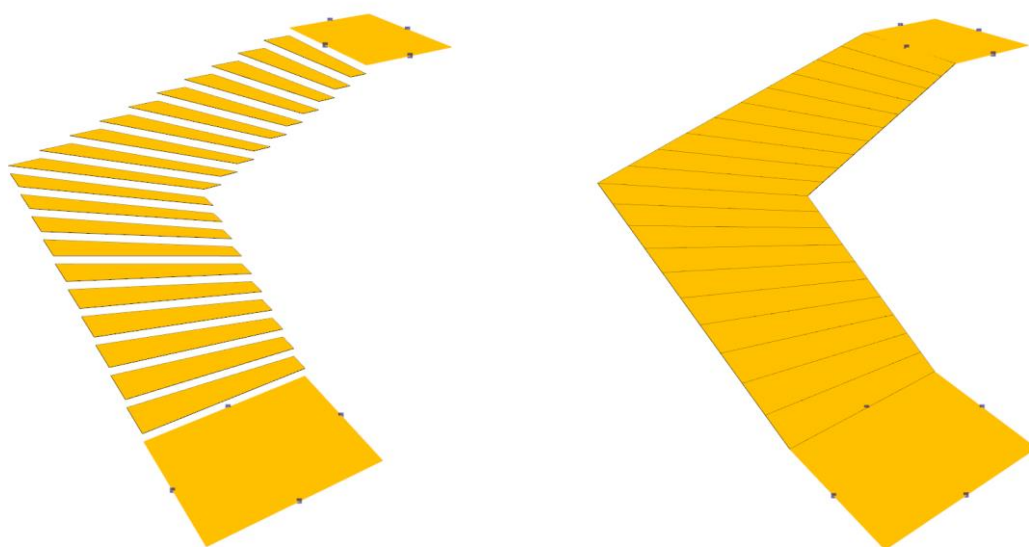
Obrázok 28: Nevhodné použitie tretej metódy

Podobne, ako pri predošlých metódach si ukážeme vygenerované stupne. Táto metóda zvládla vygenerovať stupne pri oboch schodiskách. Zobrazenie v 2D je vidieť na obrázku 29.



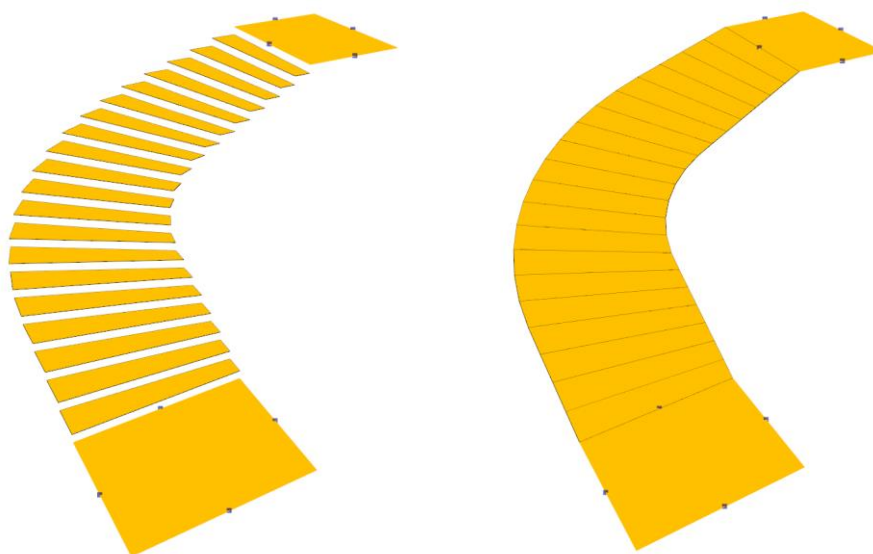
Obrázok 29: Vygenerované stupne 3. metóda v 2D

Obrázok 30 je 3D zobrazenie stupňov z obrázka 29a. Vľavo sú schody a vpravo rampa.



Obrázok 30: Vygenerované stupne 3. metóda v 3D (1)

Na obrázku 31 sú zobrazené stupne v 3D z obrázka 29b. Tiež sú vľavo schody a vpravo rampa.



Obrázok 31: Vygenerované stupne 3. metóda v 3D (2)

4.1.4 Štvrtá metóda generovania stupňov

Táto metóda vznikla kombináciou druhej a tretej metódy. Automaticky sa vyberie vtedy, ak je možné vytvoriť hranice a nie je možné vybrať druhú metódu. Dokáže sa vysporiadať s neplynulými prechodmi tak, aby vygenerované stupne vyzerali prirodzene rovnako, ako aj s tým, že začiatočná/koncová hrana nie je kolmá na vodiacu čiaru. Ak je možné použiť druhú metódu generovania, potom nie je možné použiť túto metódu, pretože

by vygenerovala rovnaké stupne, len pomalšie. Z tohto dôvodu sú druhá a štvrtá metóda navzájom vylúčené - teda ak ide použiť druhá, nedovolíme použiť štvrtú a naopak.

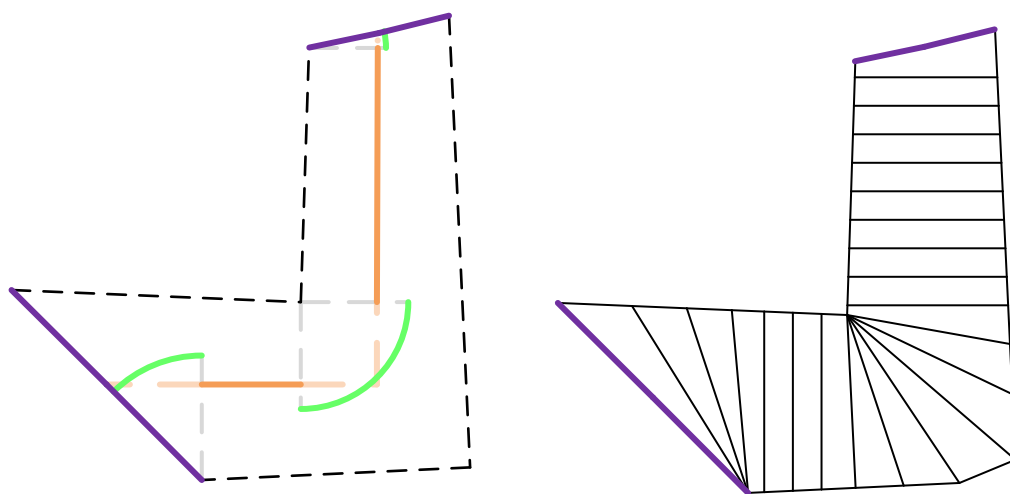
Predpokladom je, že máme nakreslenú vodiacu čiaru a brali sme do úvahy šírku schodov. Táto metóda je rozdelená na tri fázy:

1. Vytvorenie hraníc schodov podľa vodiacej čiary (rovnako ako pri druhej metóde)
2. Modifikácia vodiacej čiary aby obsahovala len priame prechody medzi jej jednotlivými časťami.
3. Vytvorenie hrán jednotlivých stupňov

Na vytvorenie hraníc sa využíva rovnaký postup ako v druhej metóde, preto rovno pokračujeme modifikáciou vodiacej čiary.

Modifikácia vodiacej čiary

Na to, aby bolo možné použiť vodiacu čiaru stupňov ako „vodiacu čiaru“ nesmú v nej existovať neplynulé prechody medzi segmentami. Riešením je nahradiť časť vodiacej čiary oblúkom s rovnakou dĺžkou. Takýmto spôsobom nenarušíme pôvodnú dĺžku nakreslenej vodiacej čiary, ktorú nakreslil dizajnér. Nie je podmienkou, aby nahradená časť bola spojená s pôvodnou, pretože šírku schodov určujú vytvorené hranice schodov. Rovnakú modifikáciu vodiacej čiary je nutné spraviť aj pri začiatkovej alebo koncovej hrane schodov, ak nezvierajú pravý uhol s vodiacou čiarou. Nahradenie spolu s vytvorenými stupňami znázorňuje obrázok 32.



Obrázok 32: Modifikácia vodiacej čiary

Vytvorenie hrán jednotlivých stupňov

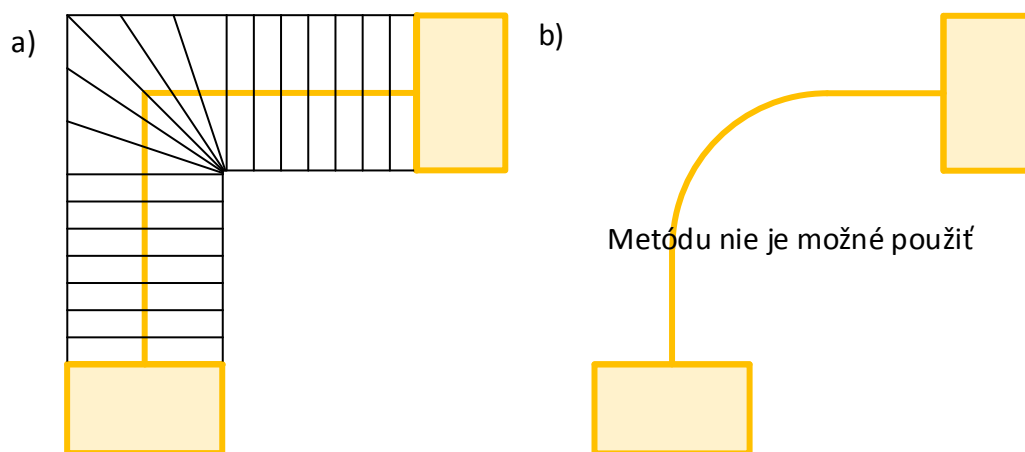
Teraz máme vytvorené hranice, ktoré sa skladajú z úsečiek a máme aj upravenú vodiacu čiaru aby medzi jej segmentami boli len plynulé prechody. To, že vodiaca čiara obsahuje len plynulé prechody znamená, že ju môžeme použiť na „vedenie“ stupňov. Netreba zabudnúť, že stále sa nachádzame v 2D projekcii 3D objektov, a preto budeme musieť aj pri vytváraní hrán stupňov prihliadať na túto skutočnosť. Priebeh algoritmu je nasledovný:

1. Začíname s jednou upravenou vodiacou čiarou a dvoma hranicami schodov.
2. Rozdelíme upravenú vodiacu čiaru na rovnaké časti, ktorých počet určuje to, koľko stupňov sa má vygenerovať.
3. Pre každý bod, ktorý vznikol týmto rozdelením:
 - a) Vytvoríme dotyčnicu k vodiacej čiare. K tejto dotyčnici vytvoríme kolmú priamku, ktorú si označíme ako P.
 - b) Postupne prechádzame zoznam úsečiek prvej hranice a testujeme, či aktuálna úsečka pretína naša priamka P. (prechádzanie tohto zoznamu vždy pokračuje tou úsečkou, ktorou predtým skončilo)
 - 1) Ak pretína, našli sme bod na hranici, ktorý bude jeden z bodov úsečky reprezentujúcej hranu stupňa.
 - c) Postupne prechádzame zoznam úsečiek druhej hranice a tiež testujeme, či aktuálna úsečka pretína naša priamka P.
 - 1) Ak pretína, našli sme bod na hranici, ktorý bude jeden z bodov úsečky reprezentujúcej hranu stupňa.
 - d) Našli sme oba body hrany stupňa, pridáme ho do zoznamu všetkých hrán stupňov a opakujeme od bodu 2 až pokiaľ nevytvoríme požadovaný počet hrán stupňov.

Algoritmus 6: Vytvorenie hrán jednotlivých stupňov

Tým, že v podstate naraz prechádzame po vodiacej čiare a zároveň aj po hraniciach sa vyhneme problému s prekrývajúcimi sa schodmi.

Na záver tejto metódy si ukážeme výsledné vygenerované stupne. Ako sme už povedali, druhá a štvrtá metóda sa navzájom vylučujú, preto neboli vygenerované stupne pre schodisko na obrázku 33b.



Obrázok 33: Vygenerované stupne 4. metóda v 2D

Na obrázku 34 vidieť vľavo vygenerované stupne schodov a vpravo časti rampy v 3D. Schody sa veľmi podobajú reálnym schodom.



Obrázok 34: Vygenerované stupne 4. metóda v 3D

4.2 Vytvorenie stupňov schodov v 3D

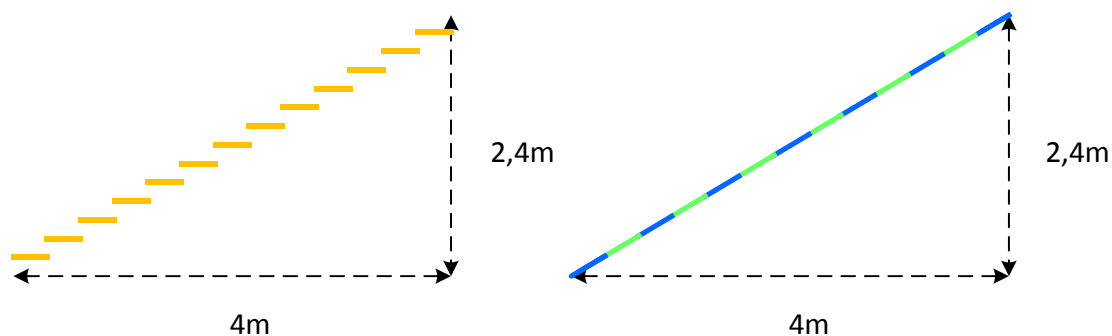
Pomocou popísaných metód sme získali zoznam hrán stupňov. Ostáva nám už len vytvoriť samotné stupne v 3D. Vytváranie stupňov schodov sa len mierne líši od toho, ako sa vytvárajú časti rampy. Odlišujú sa len v tom, v akej výške sa hrany stupňov a hrany časti rampy nachádzajú.

Postup je nasledovný:

1. Prechádzame postupne zoznam hrán, pričom si vždy predošlú hranu zapamätáme.
2. Ak máme nejakú hranu zapamätanú (teda sme už na druhej hrane, pričom prvú máme zapamätanú), vytvoríme podľa predošlej hrany prvé dva body mnohoúhelníka. Tieto body dáme do správnej výšky podľa toho či vytvárame stupne schodov alebo časti rampy (pozri obr. 35). Zostávajúce dva body vytvoríme podľa aktuálnej hrany a tiež ich dáme do správnej výšky.
3. Máme vytvorené 4 body, ktoré keď spojíme tvoria náš stupeň schodov alebo časť rampy.

Algoritmus 7: Vytvorenie stupňov schodov v 3D

Na obrázku 35 sú zobrazené jednotlivé stupne schodov (vľavo) a časti rampy (vpravo) v správnej výške.



Obrázok 35: Rozdiel medzi vygenerovanými stupňami schodov a časťami rampy

5 Pohyb chodcov na schodiskách

V tejto kapitole si najprv predstavíme simulačný nástroj pre pohyb chodcov – PedSim. Ďalej sa venujeme rozšíreniu virtuálneho sveta v simulácií o schodiská a navigovaniu chodcov na nich. Rozšírime už existujúci mikroskopický model pohybu o pohyb na schodiskách, a na záver pridáme agenta, ktorý bude zodpovedný za posúvanie chodcov na eskalátoroch a travelátoroch.

5.1 Simulačný nástroj pre pohyb chodcov – PedSim

Na Žilinskej univerzite v Žiline, Fakulte riadenia a informatiky sa vyvíja simulačný nástroj PedSim (Varga, 2014). Umožňuje modelovať správanie a pohyb chodcov na rôznych úrovniach detailnosti. Vďaka tomu je možné vytvárať modely veľkých infraštruktúr, v ktorých sa budú chodci pohybovať (Kvasňovský, 2015).

Na *mikroskopickej úrovni* detailnosti rozpoznávame chodcov ako samostatné entity. Model s touto úrovňou detailnosti nám umožňuje podrobne sledovať správanie a pohyb jednotlivých chodcov v modeli. Simulácia chodcov na tejto úrovni je náročná na výpočtové prostriedky, preto PedSim podporuje aj modelovanie na *mezoskopickej úrovni* detailnosti, ktorým sa zaoberal Čadecký (2015). Na tejto úrovni rozpoznávame chodcov ako samostatné entity, ale popisujeme ich všeobecnejšie (napr. nepoznáme presnú polohu chodca, vieme len oblasť, v ktorej sa nachádza). *Makroskopický model* (Kormanová, 2014) nerozlišuje chodcov ako samostatné entity. Entity reprezentuje (zvyčajne) ako toky, preto sa označuje tento model ako tokový model. V jednom modeli umožňuje simulačný nástroj kombinovať tieto úrovne detailnosti, čím vznikne *hybridný model* (Kormanová, 2014).

Modelovať správanie a pohyb chodca nie je jednoduché, preto Varga (2014) uvádza tri úrovne rozhodovania chodca, ktoré vychádzajú z práce Hoogendoorna a Bovyho (2004):

1. **„Strategická úroveň“** určuje činnosti a poradie ich vykonávania. Ukazuje sa, že chodci intuitívne optimalizujú poradie aktivít, pričom kľúčovú úlohu hrá „priamosť“ pri ich vykonávaní.
2. **Taktická úroveň** zahŕňa výber akcie; výber oblasti, kde sa dá daná akcia vykonať; a výber trasy do zvolenej oblasti. Kľúčovú úlohu hrá znalosť oblastí, kde sa dá daná akcia vykonať.

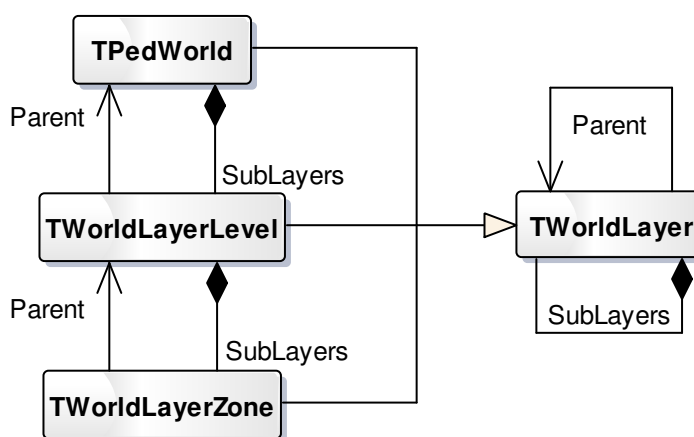
3. **Operatívna úroveň** definuje správanie sa počas chôdze, vyhýbanie sa kolíziám s prekážkami a ostatnými chodcami.“

V simulačnom nástroji pod *strategickú úroveň* spadá definovanie a výber cieľov. Ciele si chodec môže generovať sám, alebo mu môžu byť zadané. Z cieľa je možné stanoviť cieľovú entitu, ku ktorej sa má chodec dostať. Tvorba trasy k cieľovej entite spadá pod *taktickú úroveň* rozhodovania. Pod *operatívnu úroveň* rozhodovania spadá samotný pohyb chodcov k danej cieľovej entite po zvolenej trase (Varga, 2014).

Pri mikroskopickom modeli pohybu sa chodec rozhoduje v pravidelných diskretných časových okamihoch, nazývaných *Tick* (Adamko, 2013). Rovnako v týchto okamihoch prebieha aj výpočet a vykonanie kroku každého chodca.

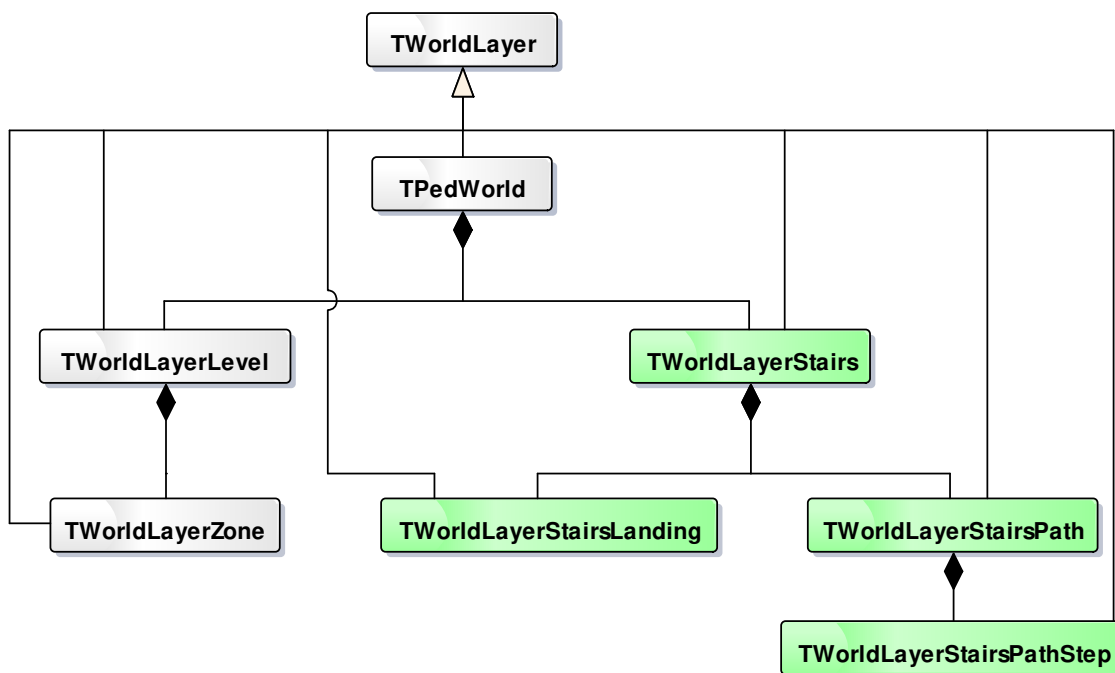
5.2 Reprezentácia sveta počas simulácie

O vytváraní modelu infraštruktúry sme si povedali v kapitole 3 *Editačná podpora*. Je potrebné tento model pretransformovať na virtuálnu infraštruktúru, po ktorej sa budú chodci pohybovať. V simulačnom nástroji je virtuálna infraštruktúra vytvorená z hierarchicky usporiadaných vrstiev. Tie sú navrhnuté podľa návrhových vzorov – Composite a Chain of responsibility (Gamma et al., 1997). Toto znamená, že vrstva nevie, pod akou vrstvou sa nachádza, ale ak nevie splniť požiadavku, prepošle ju otcovskej vrstve. Chodci, ktorí sa v simulácii nachádzajú, musia mať nastavenú vrstvu, na ktorej sa nachádzajú. Vrstvy, na ktorých sa môžu chodci nachádzať (a kráčať po nich) sú listami v tejto hierarchii. Na obrázku 36 je znázornený zjednodušený UML diagram vrstiev virtuálnej infraštruktúry.



Obrázok 36: UML diagram reprezentácie sveta počas simulácie

Do tejto hierarchie sme pridali schodiská. Celé schodisko nepatrí žiadnej úrovni, preto je zaradené pod koreň hierarchie – vrstvu svet. Schodisko sa skladá zo schodov a medziposchodí. Schody sú zložené z jednotlivých stupňov, ktoré sme vygenerovali (generovanie stupňov schodov sme podrobne popísali v kapitole 4). Na obrázku 37 je zobrazený zjednodušený UML diagram s pridanými schodiskami. Zelenou farbou sú označené pridané triedy.

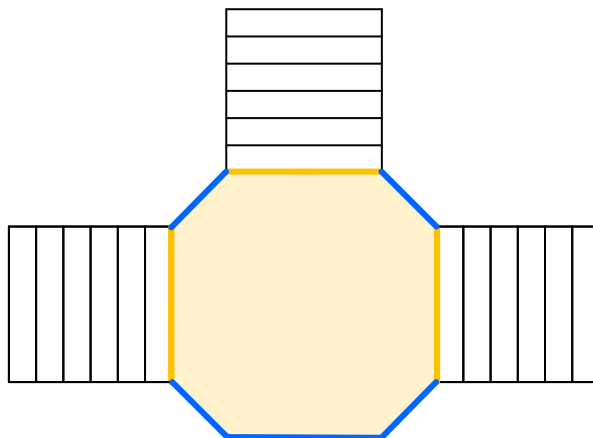


Obrázok 37: UML diagram reprezentácie sveta počas simulácie so schodiskami

Je vidieť, že listovými vrstvami pre schodiská sú medziposchodia a stupne schodov. To znamená, že chodci sa budú pohybovať po nich.

Každá vrstva, po ktorej sa má chodec pohybovať, mu musí vedieť poskytnúť (na požiadanie) zoznam okolitých prekážok a zoznam okolitých chodcov. Tieto chodec potrebuje pre svoj bezpečný pohyb. Ak vrstva nevie splniť túto požiadavku, prepošle toto volanie otcovskej vrstve. Ako to funguje si vysvetlíme, na vrstve pre zónu. Zóny sú podradené vrstve úroveň. Majú rovnakú vertikálnu výšku ako ich otcovská vrstva (úroveň). Preto nemusí mať každá zóna samostatné údajové štruktúry, ktoré urýchlia vyhľadávanie okolitých prekážok a chodcov, ale má ich jej otcovská vrstva – vrstva úroveň. To znamená, že keď sa chodec nachádza na vrstve zóny a vypýta si od nej zoznam okolitých prekážok, zóna prepošle toto volanie otcovskej vrstve – úrovni, ktorá prekážky vyhľadá za ňu (kvôli reťazi zodpovednosti). Vyhľadávanie prekážok a chodcov prebieha v 2D.

Medziposchodie v simulačnom svete môžeme považovať za špeciálny prípad úrovne s jednou mikroskopickou zónou. Preto na vyhľadávanie chodcov a prekážok môže využiť rovnaké štruktúry ako zóna resp. úroveň. Keďže medziposchodie nie je považované za otvorené priestranstvo, musíme na neobsadených hranách jeho mnohoúhelníka vytvoriť prekážky. Medziposchodie s prekážkami je vidieť na obrázku 38, kde sú modrou farbou vyznačené prekážky a oranžovou farbou medziposchodie.



Obrázok 38: Prekážky na medziposchodí

Pre jednotlivé stupne je treba tiež zabezpečiť vyhľadávanie chodcov a prekážok. Jednotlivé stupne rozdeľujú schody na menšie časti, preto ich môžeme použiť na to, aby nám vedeli poskytnúť prekážky a okolitých chodcov. Na stupni považujeme za prekážky jeho bočné hrany (pozri obr. 1). Na vyhľadávanie chodcov používame lineárny zoznam, ktorý obsahuje chodcov nachádzajúcich sa na danom stupni.

5.3 Navigácia chodcov

Chodec môže mať zadané ciele, ku ktorým sa chce dostať (definovanie cieľov chodca zabezpečuje strategická úroveň rozhodovania chodca). Na to, aby chodec prešiel k svojmu cieľu je potrebné, aby k nemu našiel trasu v infraštruktúre. Vytváranie trás patrí pod taktickú úroveň rozhodovania.

5.3.1 Tvorba trasy k cieľovému prvku na úrovni

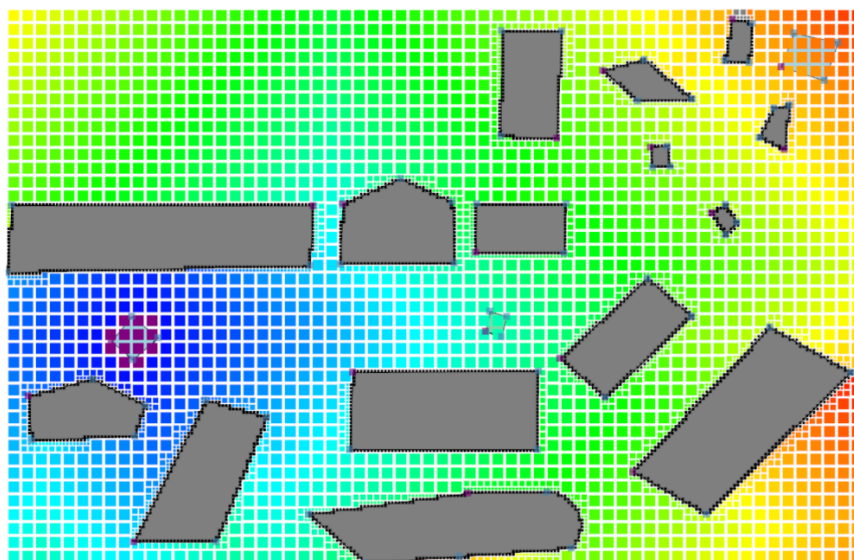
Vytvorenie trasy k cieľovej entite chodca, ktorá sa nachádza na rovnakej úrovni ako on, môže byť dosiahnuté viacerými spôsobmi, napr. gradientovými mapami, grafmi priamej viditeľnosti, Voronoiovými diagramami a pod. Tieto spôsoby podrobnejšie rozoberá Varga

(2014). My priblížime gradientové mapy, pretože sme ich princíp využili pri riešení navigácie chodcov na schodoch.

5.3.1.1 Gradientové mapy

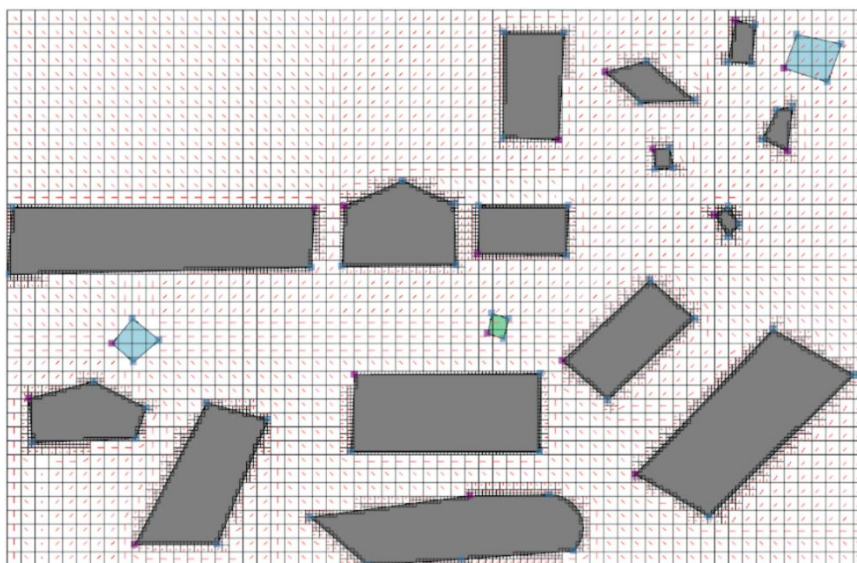
Gradientové mapy sa využívajú ako pomocná štruktúra pri navigácii chodca. Každá bunka tejto mapy obsahuje dve informácie – hodnotu gradientu a smerový vektor. Gradientové mapy sa počítajú pre každý prvok infraštruktúry, ktorý môže byť cieľom pre chodca, napríklad obslužné miesto, začiatočná/koncová hrana schodov atď.

Výpočet prebieha tak, že sa najprv inicializujú bunky pod cieľovou entitou na hodnotu -1 a hodnoty buniek, ktoré sa nachádzajú pod prekážkami sa prehlásia za blokové. Nakoniec sa flood algoritmom (Čadecký, 2015) dopočítajú hodnoty ostatných buniek. Hodnota gradientu reprezentuje určitú vzdialenosť aktuálnej bunky od našej cieľovej entity. Čím ďalej sa bunka nachádza od zdroja gradientu, tým bude jej hodnota gradientu vyššia. Keďže chodec sa chce približovať k svojmu cieľu, bude sa snažiť pohybovať smerom k bunkám s nižšou hodnotou gradientu. Príklad gradientovej mapy znázorňuje obrázok 39. Na tomto obrázku sú bunky pod cieľovou entitou vyznačené fialovou farbou. Taktiež je možné vidieť, že bunky s rovnakými hodnotami gradientu vytvárajú vrstevnice.



Obrázok 39: Zobrazenie hodnôt gradientovej mapy (Čadecký, 2015)

Pre každú bunku sa ďalej vypočíta smerový vektor. Smerový vektor vždy ukazuje smerom na bunky s nižšou hodnotou gradientu. Prínosom je, že chodec pri svojom pohybe nemusí vždy vyhľadávať bunky s nižšou hodnotou gradientu, ale použije smerový vektor aktuálnej bunky. Smerové vektory gradientovej mapy je vidieť na obrázku 40.



Obrázok 40: Zobrazenie smerových vektorov gradientovej mapy (Čadecký, 2015)

Nevýhodou gradientových máp je, že trasa, ktorú podľa nich vytvoríme, bude vždy najkratšou trasou k cieľovému prvku. Zároveň bude takáto trasa takmer vždy tesne obopínať prekážky, preto sa využíva ako pomocná forma navigácie chodca (Varga, 2014).

5.3.2 Navigácia chodca medzi úrovňami

Vtedy, keď chodec stojí na inej úrovni ako jeho cieľová entita, je potrebné zabezpečiť navigáciu chodca na úroveň tejto cieľovej entity. V simulačnom nástroji je na navigáciu medzi úrovňami vytvorený graf medzi podlažiami, ktorý popisuje Varga (2014) takto:

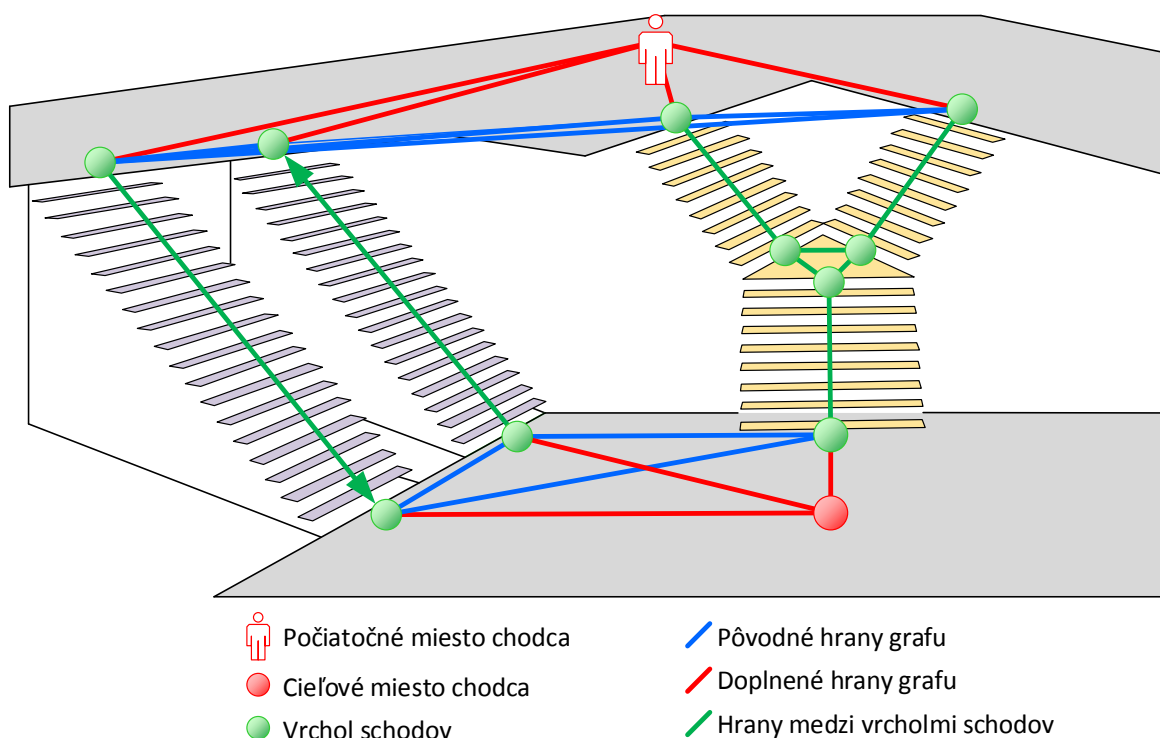
„Jeho vrcholy v predstavujú prvky, ktoré slúžia na prepojenie jednotlivých úrovní. Tými sú schody (či už statické alebo pohyblivé) a výťahy. Každý takýto prvok generuje toľko vrcholov grafu, do koľkých úrovní zasahuje (schody teda vždy generujú dva vrcholy, výťahy toľko vrcholov, koľko má staníc). Hrany v takomto grafe môžeme rozdeliť na dva druhy:

1. *Medzi vrcholmi toho istého prvku sú vytvorené orientované hrany (pohyblivé schody neumožňujú prechod v protismere). Sú ohodnotené na základe chodcovho odhadu časového trvania presunu medzi úrovňami s použitím daného prvku.*
2. *Hrany medzi vrcholmi rôznych prvkov v rámci tej istej úrovne tvoria úplný graf pre každú úroveň. Sú ohodnotené na základe časového trvania presunu medzi jednotlivými prvkami.“*

Každý chodec má k dispozícii vlastný graf medzi podlažiami. Chodec si v ňom môže upraviť hodnoty hrán na základe svojich skúseností a preferencií (Varga, 2014).

„Pri hľadaní cesty medzi jednotlivými úrovňami je potrebné vziať do úvahy začiatočnú polohu a cieľové miesto chodca. Tie sú definované ako dočasné vrcholy na príslušných úrovniach a opäť doplnené hranami tak, aby na týchto úrovniach tvorili úplný graf. Hraný sú ohodnotené na základe vzdialenosti konkrétnej polohy vrcholu k jednotlivým vrcholom danej úrovne. Výsledná cesta je upravená tak, aby v nej ostali len vrcholy, ktoré sú zviazané s rozdielnymi prvkami. Chodec sa následne túži daným prvkom dostať na príslušné poschodie.“ (Varga, 2014).

Graf medzi podlažiami sme rozšírili o medzипoschodia. Veľkým prínosom pre nás je to, že tento graf vyrieši problém s navigáciou chodca na komplikovaných schodiskách (napr. s viacerými medzипoschodiami). Na obrázku 41 je zobrazený graf medzi podlažiami. Vľavo sú dva eskalátory, kde jeden smeruje nahor, druhý nadol. Vpravo je schodisko s trojuholníkovým medzипoschodím.



Obrázok 41: Graf medzi podlažiami

5.3.3 Smerovanie chodca k cieľovému prvku na schodiskách

Z kapitoly 5.2 vieme, že na schodiskách môže chodec kráčať po medzипoschodiach a po samotných stupňoch schodov. Tvorbu trasy k cieľovému prvku na medzипoschodiach (v tom prípade, keď sa chodec nachádza na rovnakom medzипoschodí ako cieľová entita) riešime rovnako ako smerovanie k cieľovému prvku na úrovniach, pretože, ako sme už

povedali (v kapitole 5.2), medzi poschodiami môžeme považovať za špeciálnu úroveň s jednou mikroskopickou zónou.

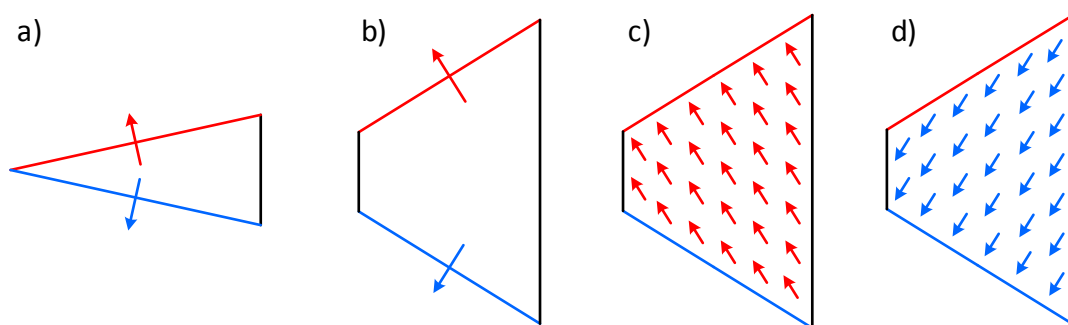
Ostáva nám vyriešiť ako budeme vytvárať trasu k cieľovým prvkom na schodoch. Na schodoch existujú dve cieľové entity – začiatková a koncová hrana schodov. Tak, ako bunky rozdeľujú zónu na menšie diskkrétne časti, podobne aj samotné stupne schodov rozdeľujú schody. Na to, aby chodec prišiel k cieľovej entite stačí, aby smeroval na taký stupeň, ktorý je k cieľu bližšie. Pre zjednodušenie tohto smerovania sme sa rozhodli, že môžeme pre každý schod určiť dva vektory, kde jeden bude smerovať na stupeň bližší k jednému cieľu a druhý k druhému. Tieto smerové vektory môžeme vytvárať dvoma spôsobmi:

- smer budú určovať kolmice na cieľové hrany stupňa, alebo
- smer budú určovať kolmice na os stupňa

Teraz si tieto dva spôsoby priblížime a vyberieme jeden, ktorý budeme používať. Ku každému uvidíme obrázok, na ktorom bude červenou farbou vyznačená jedna cieľová hrana stupňa a k nej bude rovnakou farbou vyznačený smerový vektor pre jej cieľ. Modrou farbou bude vyznačená druhá cieľová hrana stupňa a jej smerový vektor bude vyznačený tiež modrou farbou. Nebudeme sa zaoberať obdĺžnikovými stupňami, pretože v tomto prípade budú mať oba spôsoby rovnaký výstup.

Smerové vektory na stupňoch ako kolmice na cieľové hrany stupňa

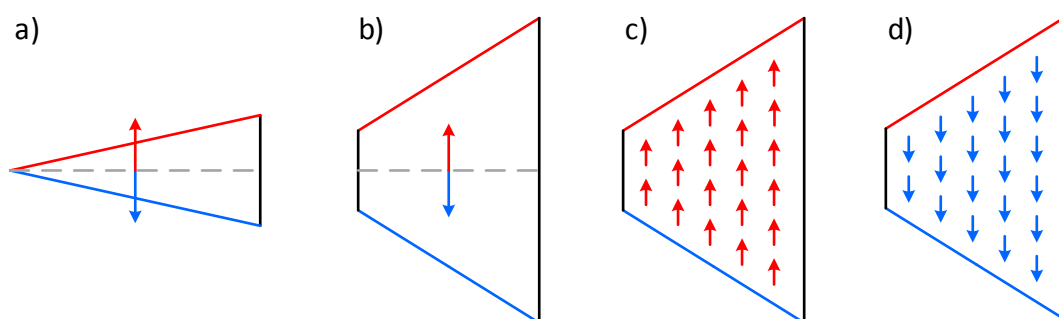
Za cieľovú hranu stupňa považujeme tú hranu stupňa, ktorá sa nachádza bližšie k cieľovej entite na schodoch. Smerové vektory budeme vytvárať ako kolmice na cieľové hrany stupňa. Obrázok 42a znázorňuje, ako by smerové vektory vyzerali pre trojuholníkový stupeň. Ďalej na obrázku 42b je vidieť, ako by vyzerali smerové vektory vytvorené týmto spôsobom pre lichobežníkový stupeň. Na jednom stupni bude vypočítaný vždy len jeden vektor pre daný cieľ, ale pre lepšiu vizualizáciu nakreslíme tieto vektory po celom stupni (na celej ploche stupňa budú mať rovnaký smer) ako sú zobrazené pre prvú cieľovú hranu na obrázku 42c a pre druhú cieľovú hranu na obrázku 42d.



Obrázok 42: Smerové vektory na stupňoch ako kolmice na cieľové hrany stupňa

Smerové vektory na stupňoch ako kolmice na os stupňa

Zostrojenie kolmice na os stupňa je jednoduché. Je to priamka, ktorá prechádza stredmi bočných hrán stupňa. Smerové vektory vytvoríme ako kolmice k osi stupňa, ktoré budú smerovať smerom k cieľovej hrane stupňa. Rovnaký výsledok by sme dostali aj keby sme sčítali kolmé vektory k oboj cieľovým hranám v jednom smere, a potom spravili opačný vektor k nemu na vytvorenie druhého smerového vektora. Na obrázku 43a je vidieť šedou prerušovanou čiarou vyznačenú os trojuholníkového stupňa. Ďalej je na obrázku 43b vidieť lichobežníkový stupeň spolu s jeho smerovými vektormi a na obrázkoch 43c a 43d vidíme tieto smerové vektory na celom stupni.



Obrázok 43: Smerové vektory na stupňoch ako kolmice na os stupňa

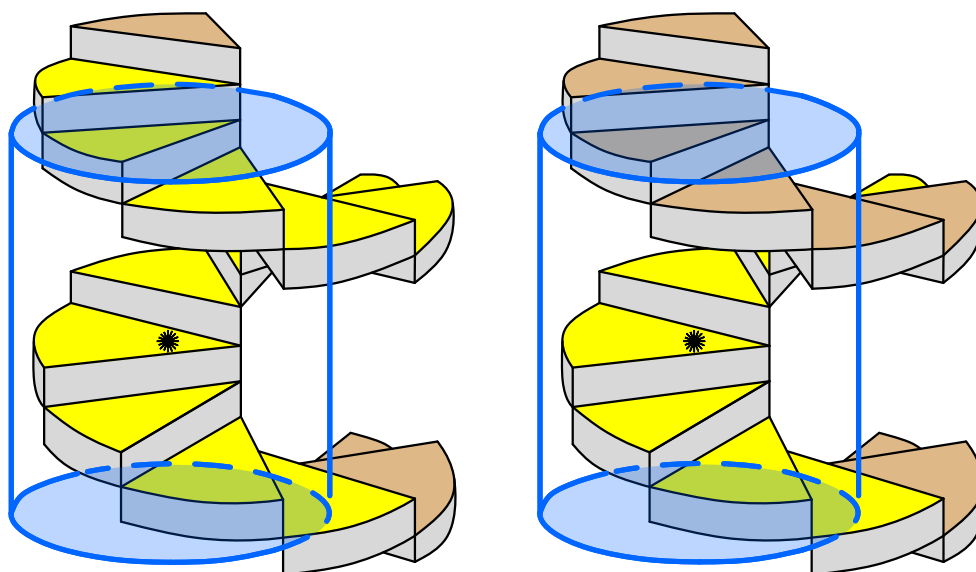
Pri prvom spôsobe sme si mohli všimnúť, že smerové vektory neukazovali výhradne na cieľové hrany stupňa, ale čiastočne ukazovali aj na jeho bočné hrany. Smerové vektory vytvorené druhým spôsobom netrpia týmto nedostatkom. Pre nás je preto viac vyhovujúci druhý spôsob vytvárania smerových vektorov, nakoľko menej ovplyvňuje chodcov smer vtedy, keď sa bude používať ako pomocný vektor pri smerovaní chodca bližšie k cieľu na schodoch.

5.4 Chodcove vnímanie okolitého sveta počas simulácie

V kapitole 5.2 sme uviedli, že chodec potrebuje pre svoj pohyb zoznamy prekážok a chodcov zo svojho okolia. Chodec takýmto spôsobom vníma svoje okolie. Vzdialenosť, na ktorú chodec dovidí (resp. je schopný vnímať svoje okolie) nazývame *rozhľad chodca*. Spomenuli sme tiež, že vyhľadávanie chodcov a prekážok na každej vrstve prebieha v 2D. Algoritmy na výpočet kroku chodca tiež pracujú v 2D. Problémom je, že schodiská sú trojrozmerné objekty, ktoré môžu prekryvať sami seba v ich priemete do 2D priestoru. Preto upravíme chodcove vnímanie okolitého sveta v zónach so schodiskami a aj na samotných schodiskách.

Na to, aby sme mohli chodcovi poskytnúť zoznamy okolitých chodcov a prekážok na schodoch, je potrebné, aby sme najprv zistili, na ktorých vrstvách stupňov schodov (alebo aj iných vrstvách) ich budeme vyhľadávať. Tento proces budeme nazývať – *zbieranie okolitých vrstiev chodca v 2D*.

Pri zbieraní týchto vrstiev musíme prihliadať aj na časovú zložitosť algoritmu pre vyhľadávanie okolitých vrstiev, na rýchlosť vyhľadávania okolitých vrstiev. Bolo by možné vyhľadávať okolité vrstvy chodca podľa valca, ktorý by reprezentoval chodcov rozhľad v 3D, a následne zozbierať vrstvy, ktoré tento valec prekrýje. Takéto riešenie by vyžadovalo vopred skonštruovať pre každú vrstvu množinu okolitých vrstiev, ktoré by sa neprekrývali v 2D, pretože valec by mohol zasahovať aj do stupňov schodov, ktoré sa nachádzajú nad stupňom, na ktorom chodec stojí. Pri veľkej infraštruktúre by vybudovanie takejto štruktúry, ako aj samotné vyhľadávanie v nej, bolo pomalé. Na obrázku 44 je nakreslené na oboch stranách rovnaké točité schodisko. Vľavo chodec zbiera vrstvy podľa valca, bez štruktúry, ktorá by zaručila to, že sa nebudú zozbierané vrstvy prekryvať v 2D. Vpravo je rovnaká situácia, je ale zabezpečené to, že zozbierané vrstvy sa nebudú prekryvať v 2D. Valec je vyznačený modrou farbou, žltou farbou sú vyznačené zozbierané stupne schodov. Nezbierané stupne majú hnedú farbu. Čiernou hviezdikou je vyznačený stred chodca.



Obrázok 44: Zbieranie okolitých vrstiev podľa valca

Na schodiskách nemusíme za rozhl'ad považovať kružnicu s polomerom rovným rozhl'adu. Môžeme za rozhl'ad považovať vzdialenosť po čiare opisujúcej tvar schodov, ktorú budeme nazývať *aproximačná čiara*. Keďže jednotlivé stupne na seba nadväzujú, môžeme začať zbierať okolité vrstvy z aktuálneho stupňa, na ktorom chodec stojí a postupne prechádzať nadväzujúce stupne, pokiaľ nezobierame všetky vrstvy v chodcovom rozhl'ade.

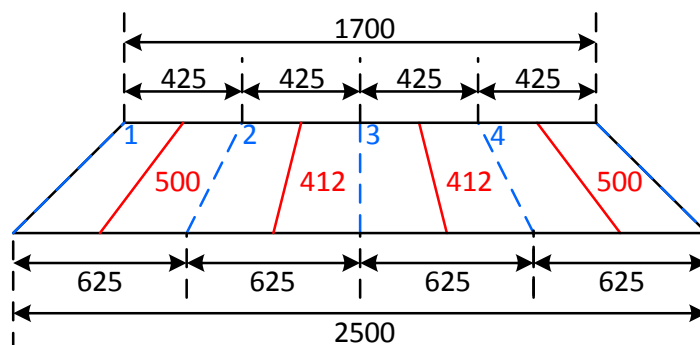
Každý stupeň schodov rozdelíme na menšie časti – *dielce*. Pre každú takúto časť budeme vytvárať aproximačnú čiaru. Začneme tým, že vypočítame na koľko dielcov budeme stupne schodov rozdeľovať:

1. Nájdeme najdlhšiu hranu stupňa zo všetkých stupňov schodov.
2. Jej dĺžku podelíme hodnotou 700mm (Teknomo (2002) uvádza pre šírku chodca rozpätie 500 - 900mm, my berieme do úvahy priemer týchto hodnôt) a výsledok zaokrúhlime nahor.
3. Výsledok určuje, na koľko častí (dielcov), budeme rozdeľovať jednotlivé stupne.

Algoritmus 8: Stanovenie počtu dielcov stupňa pre schody

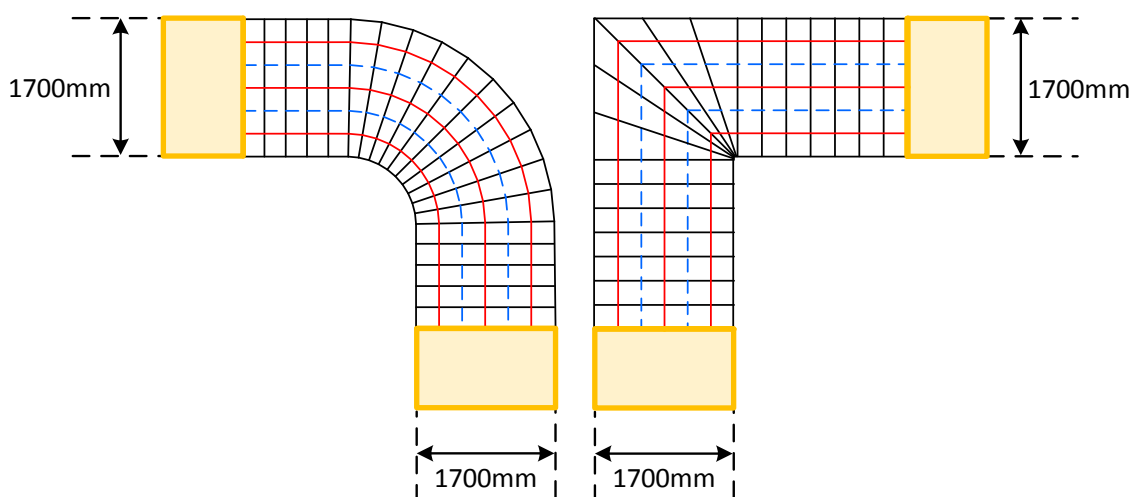
Pre každý vzniknutý dielec stupňa nájdeme stredovú úsečku, ktorej dĺžka bude aproximovať šírku stupňa v danom dielci – aproximačnú čiaru. Každému dielcu priradíme index (poradové číslo v rámci stupňa), aby sme neskôr vedeli, po ktorých aproximačných čiarach budeme pokračovať v prehľadávaní nasledujúcich stupňov v smere vyhl'adávania. Na obrázku 45 je zobrazený stupeň, ktorý má jednu hranu stupňa dlhšiu ako druhú.

Jednotlivé dielce stupňa sú zvýraznené modrou farbou, rovnako ako aj ich index. Červenou farbou sú zvýraznené aproximačné čiary a ich dĺžky. Rozmery sú uvedené v milimetroch.



Obrázok 45: Stupeň s vyznačenými dielcami a aprox. čiarami (mm)

Na obrázku 46 sú zobrazené dva typy schodov so zvýraznenými dielcami a aproximačnými čiarami, ktoré im patria. Modrou prerušovanou sú vyznačené hranice dielcov a červenou aproximačné čiary.



Obrázok 46: Aproximačné čiary na všetkých stupňoch schodov

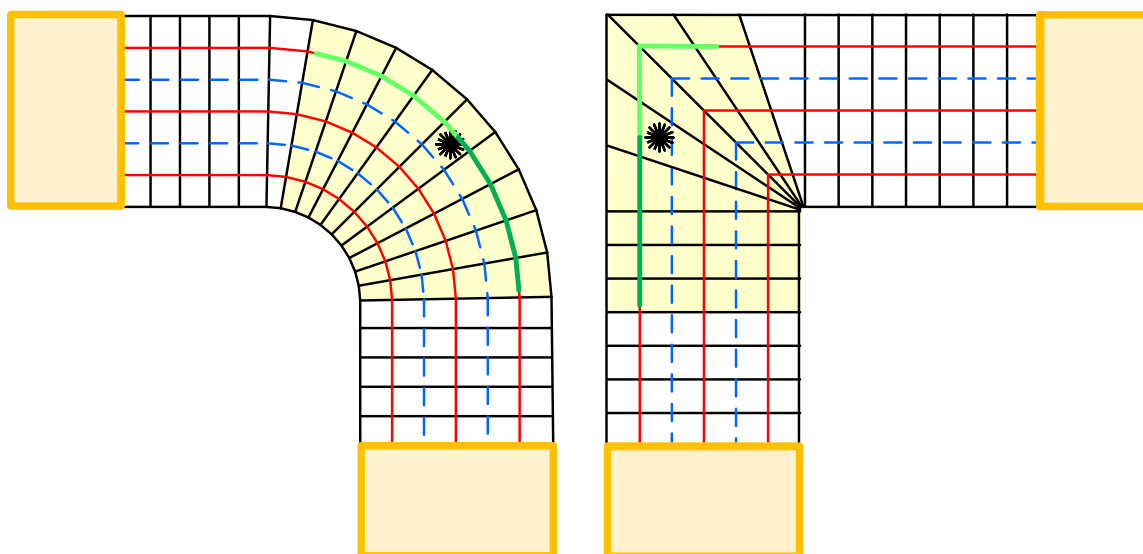
Aproximačné čiary máme vytvorené, preto môžeme vyhľadávať okolité vrstvy v rozhl'ade chodca. Vrstvu, na ktorej sa chodec nachádza budeme nazývať – zdrojová vrstva. Vyhľadávanie okolitých vrstiev na schodoch prebieha nasledovne:

1. Na zdrojovej vrstve nájdeme dielce, v ktorom sa nachádza stredový bod chodca. Index dielca si zapamätáme. Zdrojovú vrstvu pridáme do zoznamu okolitých vrstiev (výsledného zoznamu).
2. Nasledovný postup vykonáme najprv pre jeden, potom pre druhý smer vyhľadávania:

- a) Nájďme najbližší bod na aproximačnej čiare od stredu chodca. Táto aproximačná čiara patrí dielcu z bodu 1.
- b) Vypočítame vzdialenosť od nájdeného najbližšieho bodu aproximačnej čiary, koniec aproximačnej čiary v smere vyhľadávania. Túto vzdialenosť odpočítame od pôvodnej hodnoty rozhladu, čím dostaneme hodnotu zostávajúceho rozhladu.
- c) Postupne prechádzame nasledujúce stupne v smere, v ktorom vyhľadávame, pokiaľ zostávajúci rozhlad je menší alebo rovný nule, alebo nasleduje vrstva medziposchodia alebo úrovne. Vyhľadávanie okolitých vrstiev na nich popisuje algoritmus 10.
 - 1) Spracovávanú vrstvu pridáme do výsledného zoznamu.
 - 2) Odpočítame od hodnoty zostávajúceho rozhladu dĺžku aproximačnej čiary. Táto čiara patrí tomu dielcu, ktorého index sme si zapamätali na začiatku.
3. Oba smery sme prehľadali a naplnili sme výsledný zoznam okolitých vrstiev chodca.

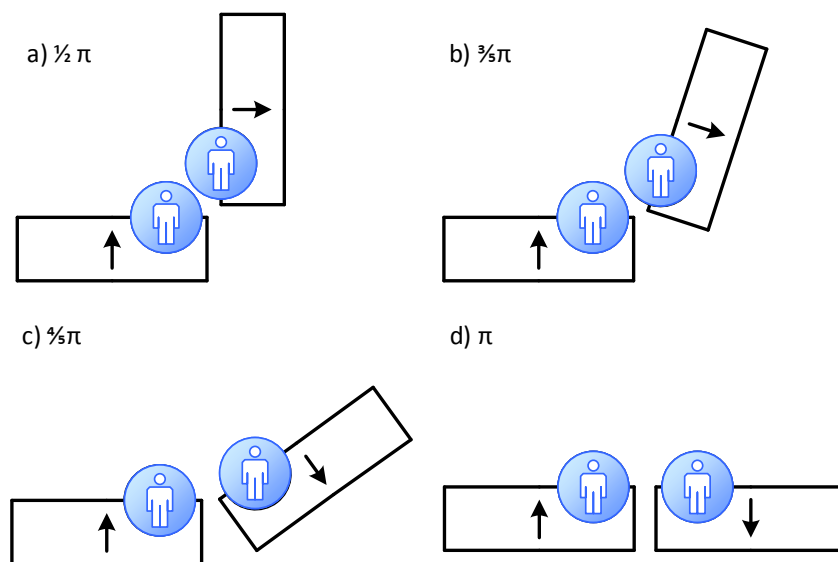
Algoritmus 9: Vyhľadávanie okolitých vrstiev na schodoch

Na obrázku 47 sú zobrazené schodiská z prechádzajúceho obrázka, na ktorých sú vyznačené bledožltou farbou okolité vrstvy chodca v jeho rozhlade. Čiernou hviezdikou je vyznačený stred chodca. Bledozelenou a tmavozelenou farbou sú vyznačené aproximované vzdialenosti rozhladu v jednom a druhom smere od zdrojovej vrstvy.



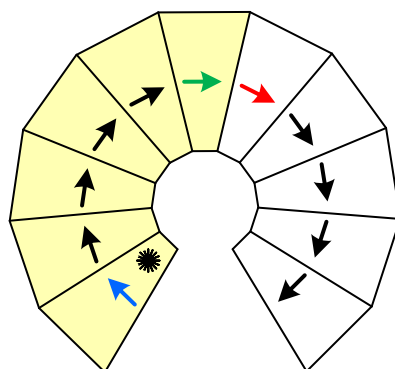
Obrázok 47: Okolité vrstvy chodca na schodoch

Ďalší problém je to, že sa zozbierané stupne môžu prekryvať v 2D. Pri rozhodovaní, či spracovávanú vrstvu do zoznamu okolitých vrstiev pridáme alebo nie, môžeme využiť smerové vektory na stupňoch. Budeme porovnávať uhol medzi smerovým vektorom zdrojovej vrstvy a smerovým vektorom spracovávanej vrstvy. Smerové vektory budeme porovnávať len tie, ktoré sú určené pre rovnaký smer ako je náš smer vyhľadávania. Obrázok 48 zobrazuje jednotlivé situácie, ktoré môžu nastať. V prípadoch a), b), c) sa chodci navzájom vidia, v prípade d) sa chodci navzájom nevidia. Na základe tejto vizualizácie sme stanovili, že ak uhol, ktorý zvierajú smerový vektor zdrojového stupňa so smerovým vektorom aktuálne spracovávaného stupňa prekročí hodnotu 144° ($\frac{4}{5}\pi$, pozri obr. 48c), potom aktuálne spracovávaný stupeň do zoznamu okolitých vrstiev nepridáme. Toto porovnávanie je zakomponované do algoritmu 9 ako prídavná podmienka na zastavenie vyhľadávania v danom smere.



Obrázok 48: Stanovenie maximálneho uhla medzi smerovými vektormi stupňov

Na obrázku 49 je možné vidieť časť točitého schodiska. Žltou farbou sú vyznačené vrstvy, ktoré sa nachádzajú v okolí chodca s neobmedzeným rozhľadom. Čiernou hviezdikou je vyznačený stred chodca. Smerové vektory sú označené čiernou farbou. Modrou farbou je označený smerový vektor stupňa, z ktorého sme začínali zbierať okolité vrstvy. Modrý vektor so zeleným vektorom (zelený je smerový vektor z naposledy pridaného stupňa) zvierajú uhol $135,5^\circ$. Uhol medzi modrým a červeným vektorom (červený je smerový vektor zo stupňa, ktorý nasleduje po naposledy pridanom stupni) je $162,6^\circ$, preto sa vrstva s červeným smerovým vektorom nepridala do vrstiev okolia chodca.



Obrázok 49: Riešenie problému prekryvania stupňov v 2D

Ak má chodec v svojom rozhlade medziposchodie alebo úroveň, potom sa musia okolité vrstvy vyhľadávať aj na nich. Takáto situácia je znázornená na obrázku 50. Vyhľadávanie prebieha nasledovne:

1. Rozoznávame dve situácie:

- a) Ak sa chodec nachádza na stupni, vyhľadávanie prebieha podľa algoritmu 9 až pokiaľ sa vyhľadávanie okolitých vrstiev nedostane na medziposchodie. Bod, z ktorého budeme vyhľadávať, nastavíme na posledný bod aproximačnej čiary dielca posledného stupňa pre vyhľadávaný smer. A polomer vyhľadávania nastavíme na hodnotu zostávajúceho rozhladu z predošlého algoritmu.
- b) Ak sa chodec nachádza na medziposchodí alebo úrovni, nastavíme polomer vyhľadávania na hodnotu rozhladu. Bod, z ktorého budeme vyhľadávať, nastavíme ako stred chodca.

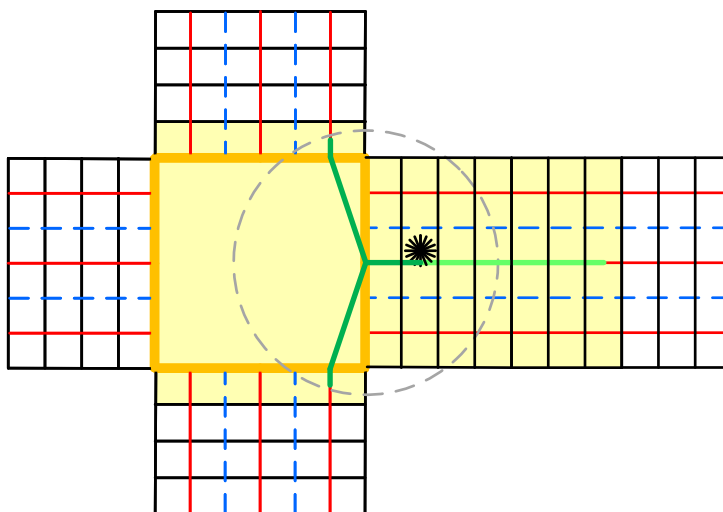
2. Vyhľadáme najbližšie stupne podľa kružnice. Kružnica je daná bodom, z ktorého vyhľadávame a polomerom vyhľadávania.

3. Na nájdených stupňoch nájdeme najbližšiu aproximačnú čiaru k bodu, z ktorého vyhľadávame. Zoberieme jej bližší koncový bod, podľa ktorého sa bude vyhľadávať na daných schodoch. Rozhľad, pre ktorý budeme na schodoch vyhľadávať nastavíme na hodnotu zostávajúceho rozhladu.

Algoritmus 10: Vyhľadávanie okolitých vrstiev na medziposchodí a úrovni

Na obrázku 50 sú zelenými farbami vyznačené aproximované vzdialenosti rozhladu v jednom a druhom smere. Šedá kružnica má polomer o veľkosti zostávajúceho rozhladu a jej stred je na konci aproximačnej čiary dielca posledného stupňa z vyhľadávania na schodoch. Vidíme, že v dosahu tejto kružnice sa nachádzajú aj iné schodiská, na ktorých

sme našli najbližší dielec a využili jeho aproximačnú čiaru, čo spôsobilo prídanie prvých vrstiev stupňov obidvoch týchto schodísk.



Obrázok 50: Vyhľadávanie okolitých vrstiev na medziposchodiach a schodoch

Okolité vrstvy chodca máme zozbierané. Na každej z nich vyhľadáme prekážky a chodcov (ako bolo popísané kapitole 5.2). Teraz si môže chodec vypočítať svoj krok (pohybom, resp. výpočtom kroku chodca sa zaoberáme v nasledujúcej kapitole). Ostáva len zistiť, na ktorej vrstve bude chodec stáť po uskutočnení kroku. Bod, do ktorého je potrebné chodca po uskutočnení kroku premiestniť, máme k dispozícii. Postupne budeme prechádzať zoznam okolitých vrstiev, pokiaľ nenájdeme tú, do ktorej tento bod patrí. Problém je, že bod môže patriť dvom vrstvám – úrovni (resp. zóne) a zároveň jednému stupňu schodov. Táto situácia nastáva, keď sa chodec nachádza na rozhraní úrovne a schodov. Preto sme zaviedli priority, podľa ktorých sa vyberie vrstva, na ktorú chodec prejde. Najvyššiu prioritu majú vrstvy stupňov schodov pretože, chodec buď vstupuje na schody alebo z nich odchádza. Zvyšné vrstvy majú priradené rovnaké priority. Na koniec sa vykoná krok chodca a nastaví sa mu správna nadradená vrstva.

5.5 Mikroskopický model pohybu chodcov

Ako už bolo spomenuté, na mikroskopickej úrovni detailnosti je každý chodec samostatnou entitou. Modeluje sa pohyb a individuálne správanie sa chodcov. Pohyb chodcov patrí pod operatívnu úroveň rozhodovania.

Existuje viacero modelov pohybu na mikroskopickej úrovni, ktoré spomína vo svojej práci Kormanová (2014). My sa budeme zaoberať mikroskopickým modelom pohybu s využitím sociálnych síl.

5.5.1 Mikroskopický model pohybu chodcov pomocou sociálnych síl

Kormanová (2014) uvádza, že koncept tohto modelu využíva pôsobenie sociálnych síl (Helbing a Molnar, 1995), (Helbing et al., 2002), (Helbing et al., 2005). Tieto sily nevznikajú pôsobením okolia na chodca, ale sú mierou vnútornej motivácie chodca urobiť krok.

Kormanová (2014) uvádza, že v modeli na chodca pôsobia tri základné vplyvy:

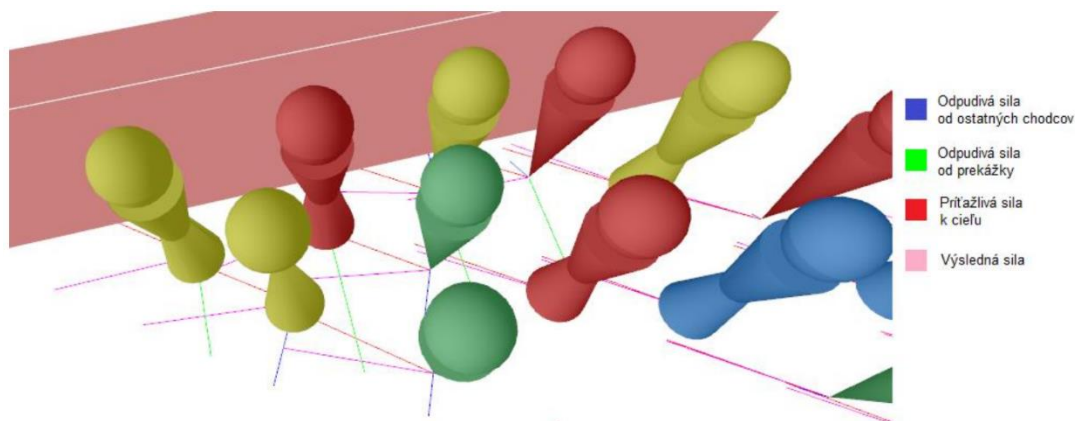
- „*Snaha o zrýchlenie pohybu pre dosiahnutie požadovanej rýchlosti chodca.*
- *Prispôsobovanie vzdialenosti chodca vo vzťahu k iným chodcom a ohraničeniam, a to počas pohybu.*
- *Príťažlivé pôsobenia.*“

Tieto vplyvy sú vyjadrené ako vektory. Prvý vektor bude vyjadrovať spomínanú snahu o zrýchlenie pohybu. Smer tohto vektora závisí od definovaného bodu, ku ktorému sa snaží chodec priblížiť. Tento bod môže byť určený ako bod na trase k jeho cieľu alebo najbližším bodom na cieľovej entite, ku ktorej kráča a pod. Veľkosť požadovanej rýchlosti chodca je definovaná ako preferovaná rýchlosť kráčania chodca. Prvý vektor potom vzniká ako rozdiel vektora požadovanej rýchlosti chodca od vektora aktuálnej rýchlosti chodca (vektor, ktorý bol vypočítaný v predchádzajúcom kroku chodca alebo nulový vektor ak chodec stál). Tento vektor je ešte upravený vzhľadom na to, ako rýchlo môže chodec meniť svoj smer.

Druhý vektor vzniká zložením vektorov odpudivých síl od okolitých chodcov a od prekážok. Týmto silám je možné nastavovať účinnosť a dosah (čím nižší, tým sa bude môcť chodec viac priblížiť k inému chodcovi alebo prekážke a naopak).

Posledný vektor je príťažlivá sila k ostatným osobám (napr. priateľom, rodine a pod.) alebo k objektom (napr. informačným tabuliam). Modelujú sa podobne ako odpudivé vplyvy, ale zvyčajne ich sila s časom klesá, pretože chodec stráca záujem o objekt, ku ktorému je príťahovaný.

Pôsobenie týchto síl závisí aj od toho, či sa jedná o situácie vzniknuté v smere požadovaného pohybu. Preto situácie, ktoré vzniknú za chodcom ho ovplyvňujú menej ako tie pred ním. Sčítaním vektorov základných vplyvov dostaneme vektor sociálnej sily. Tento vektor sčítame s vektorom aktuálnej rýchlosti chodca, prispôsobíme ho tak, aby sa bral do úvahy čas, ktorý uplynul medzi predchádzajúcim a aktuálnym krokom. Nakoniec chodec vykoná výsledný krok.



Obrázok 51: Čiastkové vektory sociálnej sily (Kormanová, 2014)

5.5.2 Mikroskopický model pohybu chodcov na schodiskách

V predchádzajúcich kapitolách sme stanovili, čo sú prekážky na schodiskách, ako sa chodci môžu navigovať na schodiskách, ostáva nám len vytvoriť pohybový model chodcov na schodiskách.

Z analýzy sme zistili, že viacero nástrojov pristupuje k pohybu na schodiskách tak, že nechajú rovnaký (alebo mierne upravený) pohybový model, ktorý využívajú na rovinách a len znížia rýchlosť chodcov. Preto aj my použijeme už existujúci model pohybu s využitím sociálnych síl popísaný v predchádzajúcej kapitole.

Na schodoch sa chodci pohybujú pomalšie. Preto sme zaviedli dodatočnú modifikáciu rýchlosti chodcov pri chôdzi. Vychádzame z toho – že chodec vie akou rýchlosťou by sa chcel pohybovať na rovine, a toho – aký je sklon schodov, na ktorých sa chodec aktuálne nachádza. Koeficient, ktorým budeme násobiť cieľovú rýchlosť chodca závisí od toho, či sa nachádza na schodoch alebo rampe, či ide smerom hore alebo dole a od sklonu schodov/rampe. Spomalenie na schodoch je uvedené v tabuľke 3.

Tabuľka 3
Modifikácia rýchlosti pohybu chodca na schodoch (Oasys, 2015)

Smer pohybu chodca	Sklon X (v stupňoch)	Percentá z cieľovej rýchlosti chodca na rovine
Smerom nahor	$0 < X < 27$	42,5%
Smerom nahor	$27 \leq X \leq 32$	Rovnomerne rozdelené medzi 42,6% a 37,8%
Smerom nahor	$32 < X$	37,8%
Smerom nadol	$0 < X < 27$	57,4%
Smerom nadol	$27 \leq X \leq 32$	Rovnomerne rozdelené medzi 57,4% a 49,8%
Smerom nadol	$32 < X$	49,8%

Spomaľovanie chodcov na rampách nie je tak výrazné ako na schodoch a jeho prehľad je vidieť v tabuľke 4.

Tabuľka 4
Modifikácia rýchlosti pohybu chodca na rampe (Oasys, 2015)

Smer pohybu chodca	Sklon X (v stupňoch)	Percentá z cieľovej rýchlosti chodca na rovine
Smerom nahor	$0 < X < 5$	100%
Smerom nahor	$5 \leq X < 10$	88,5%
Smerom nahor	$10 \leq X \leq 20$	Rovnomerne rozdelené medzi 88,5% a 75%
Smerom nahor	$20 < X$	75%
Smerom nadol	Ľubovoľný	100%

Chodci, ktorí sa pohybujú po schodiskách chodia bližšie k prekážkam a okolitým chodcom, ako keď sa pohybujú po otvorenom priestranstve. Toto je spôsobené tým, že na schodiskách je menší priestor na pohyb a taktiež sa na nich chodci pohybujú pomalšie. Z tohto dôvodu sme zmenšili dosah odpudivých síl tak, aby sa chodci mohli viac približovať k prekážkam a iným chodcom. Kalibrácii pohybového modelu sme sa v diplomovej práci nevenovali.

5.6 Agent na obsluhu eskalátorov a travelátorov

Simulačný nástroj PedSim využíva princípy *agentovo orientovanej simulácie*, ktorú popisuje Varga (2014) nasledovne:

„Agentovo orientovaná simulácia identifikuje autonómne prvky skúmaného systému - agentov. Agenti vykonávajú v systéme zadané činnosti, môžu prostredie skúmať, pružne reagovať na zmeny v ňom prípadne prispôbovať svoje úlohy na základe aktuálnej situácie.“

Existuje viacero agentových architektúr. Simulačný nástroj PedSim je postavený na architektúre ABAsim.

5.6.1 Architektúra ABAsim

Agentová architektúra ABAsim (Adamko, 2013) sa vyvíja na Fakulte riadenia a informatiky niekoľko rokov. Poskytuje mnoho prvkov, pomocou ktorých umožňuje vytvárať agentovo orientované simulačné modely (Varga, 2014).

5.6.1.1 Agenti v ABAsim

Agent je základný prvok agentovo orientovanej simulácie. Jennings (2001) definuje agenta takto:

„Agent je zapuzdrený počítačový systém, vložený do nejakého systému, schopný v tomto systéme pružne a autonómne konať tak, aby naplnil stanovené ciele.“

ABAsim rozlišuje dva typy agentov, ktoré Varga (2014) popisuje nasledovne:

„Riadiaci agent je komponent zodpovedný za špecifickú časť modelovaného systému, riadi jeho entity (napr. výťahy, eskalátory) a udalosti (napr. príchod nového človeka). Je súčasťou nemennej hierarchickej štruktúry, ktorá ako celok spolupracuje za účelom dosiahnutia systémového cieľa. Pomocou riadiacich agentov je možné popísať správanie sa rôznych systémov. Ich štruktúra do značnej miery kopíruje štruktúru modelovaného systému, pričom jej hierarchické usporiadanie pochádza z hierarchickej povahy systému. Takúto organizáciu je možné badať hlavne v obslužných systémoch. Riadiaci agent nemusí mať fyzickú reprezentáciu v modelovanom systéme. Každý takýto agent môže spravovať skupinu dynamických agentov.“

Dynamický agent predstavuje autonómnu inteligentnú entitu pôsobiacu v modeli, schopnú nezávislého riadenia vlastného správania. Dynamický agent je vždy podriadený

jednému riadiacemu agentovi. Jeho ciele môžu byť zadané z riadiacej vrstvy agentov (tieto ciele sú pre dynamického agenta záväzné) alebo si môže generovať vlastné (pokiaľ tieto nebudú v rozpore s cieľmi od riadiacich agentov). Dynamickí agenti nie sú nútení medzi sebou spolupracovať pre dosiahnutie svojich cieľov a nesprávajú sa k ostatným dynamickým agentom nepriateľsky.“

Agent je zložený z manažéra a asistentov. Manažér je považovaný za „mozog agenta“. Poznáme *promptných asistentov* (akcia, dotaz, poradca), ktorých činnosť sa vykoná okamžite, nemá časové trvanie a *kontinuálnych asistentov* (proces, monitor, plánovač), ktorých činnosť trvá definovaný časový interval. Manažér v spolupráci s asistentmi riadi činnosť agenta. Je tiež zodpovedný za komunikáciu s asistentami vlastného agenta a s inými agentmi v agentovej hierarchii. Komunikácia v rámci architektúry prebieha prostredníctvom správ, ktoré majú priradené rôzne parametre, ako napr. kód, odosielateľ adresát, atď. Tejto problematike a architektúre ABAsim sa podrobnejšie venuje Adamko (2013).

5.6.1.2 ABAGraf

Manažérovi a asistentom agenta je potrebné zdefinovať ako budú vykonávať svoju činnosť. Pre manažéra to znamená, ako bude spracovávať prijaté správy a ako má riadiť činnosti agenta. Asistentom sa definuje ako majú vykonávať svoje úlohy. Správanie manažéra a asistentov môže dizajnér zdefinovať dvoma spôsobmi:

1. pomocou programovacieho jazyka,
2. pomocou formalizmu

Architektúra ABAsim má vlastný formalizmus s názvom ABA-graf (Kavička, 2003). Jedná sa o podtriedu farebných petriho sietí s určitými modifikáciami, ktoré popisuje Čadecký (2015) takto:

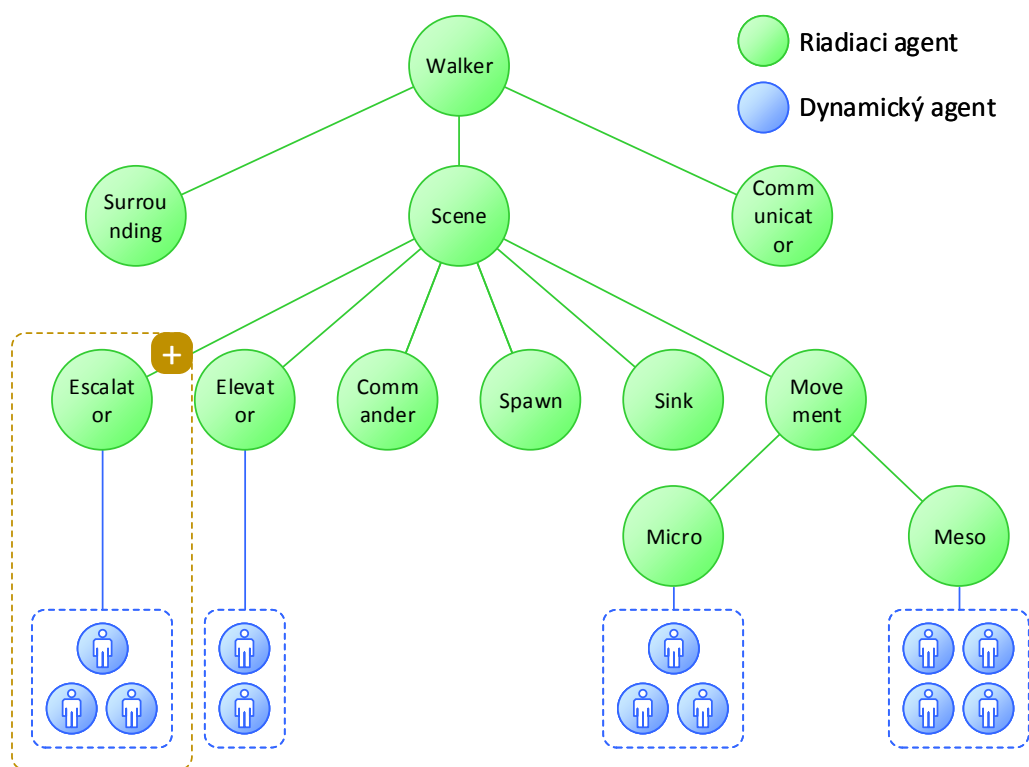
- „Značka je v sieti reprezentovaná správou.
- *Agenti a kontinuálni asistenti reprezentujú vstupné a výstupné miesta siete. Ide o odosielateľov a prijímateľov správ. Vnútorne miesta siete reprezentujú promptných asistentov. Ak značka (správa) prejde miestom, dôjde k vyvolaniu činnosti príslušného promptného asistenta. V sieti môžeme nájsť ešte jedno špeciálne vstupné miesto (simulačné jadro) a jedno špeciálne výstupné miesto (absorpciu), ktoré reprezentuje zánik správy.*

- Rozlišuje dva druhy prechodov – jednoduchý prechod a rozhodovací prechod. Jednoduchý prechod spája dve miesta v sieti. Rozhodovací prechod na základe definovanej podmienky zvolí hranu, po ktorej má poslať značku (správu).
- ABA-graf umožňuje spojiť dva za sebou idúce prechody. Medzi dvomi spojenými prechodmi sa automaticky vytvorí fiktívne miesto a hrana medzi nimi sa označí ako fiktívna.“

Výhodou využitia tohto formalizmu je to, že umožňuje rýchlo a jednoducho meniť správanie manažéra alebo kontinuálneho asistenta (promptní asistenti vykonávajú len jednoduché úlohy, ktoré sa musia naprogramovať).

5.6.2 Pridanie nového riadiaceho agenta eskalátorov

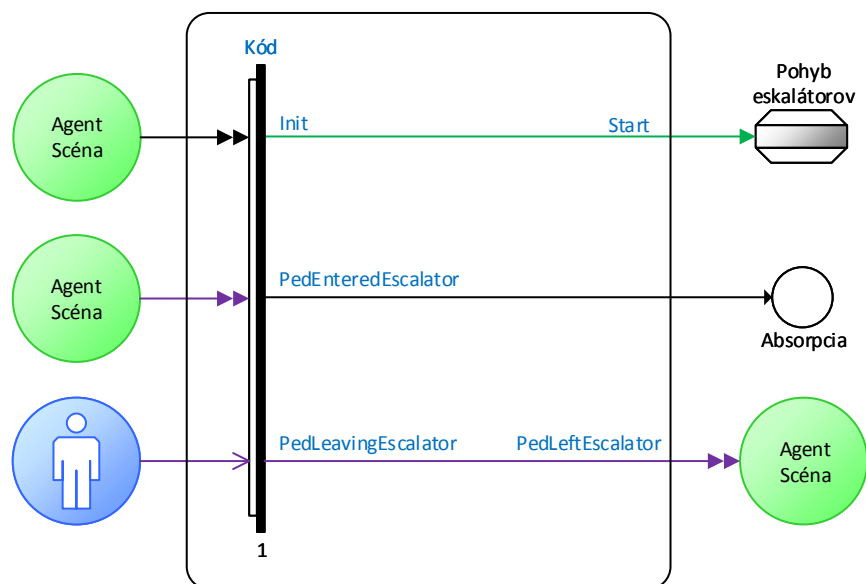
Pohyb chodcov po schodiskách sme navrhli tak, aby sa vedeli po nich pohybovať samostatne, bez nutnosti riadiaceho agenta. Chodcov, ktorí sa nachádzajú na eskalátoroch a travelátoroch je potrebné posúvať. Na vykonávanie tejto činnosti sme vytvorili nového riadiaceho agenta. Tohto agenta budeme nazývať **agent eskalátorov**, aj keď bude obsluhovať aj travelátory. Agentu eskalátorov sme zaradili pod agenta Scéna, pretože eskalátory a travelátory posúvajú chodcov v scéne. Nový agent spolu s agentovou hierarchiou je zobrazený na obrázku 52.



Obrázok 52: Agentový model s pridaným agentom eskalátorov

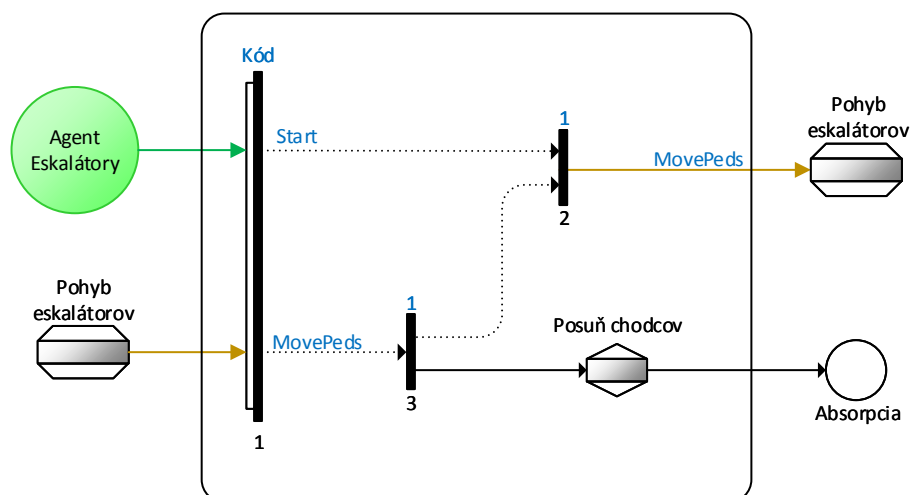
Agentovi eskalátorov sme vytvorili manažéra, ktorý vie spracovávať nasledujúce správy:

- *Init* – po prijatí tejto správy s týmto kódom sa naštartuje proces na posúvanie vlastných podriadených dynamických agentov.
- *PedEnteredEscalator* – správa s týmto kódom bude obsahovať aj dynamického agenta, ktorého náš agent prevezme a pridá ho do zoznamu vlastných podriadených dynamických agentov.
- *PedLeavingEscalator* – správu s týmto kódom odosiela dynamický agent keď opustí eskalátor, na ktorom sa práve nachádzal. Po jej prijatí, daného dynamického agenta odstránime zo zoznamu podriadených dynamických agentov a prepošleme ju agentovi scéna s kódom *PedLeftEscalator*.



Obrázok 53: ABA-graf agenta eskalátory

Vieme, že pohyb chodcov sa vykonáva v diskretných časových okamihoch. Na konci každého okamihu posunieme naraz všetkých chodcov, ktorí sa nachádzajú na eskalátoroch a travelátoroch. Na posúvanie chodcov sme vytvorili kontinuálneho asistenta – proces. Tento proces bude mať po jeho spustení za úlohu posúvať všetkých chodcov, ktorí sa nachádzajú na eskalátoroch alebo travelátoroch. Chodci, ktorí sa nachádzajú na eskalátoroch alebo travelátoroch sa nachádzajú v zozname vlastných podriadených dynamických agentov nášho agenta eskalátorov. ABA-graf tohto procesu je zobrazený na obrázku 54.



Obrázok 54: ABA-graf procesu na posúvanie chodcov na eskalátoroch a travelátoroch

Akcia posuň chodcov (pozri obr. 54) postupne prechádza zoznam agentových podriadených dynamických agentov a posunie chodca podľa rýchlosti eskalátoru alebo travelátoru, na ktorom sa nachádza.

6 Overenie riešenia

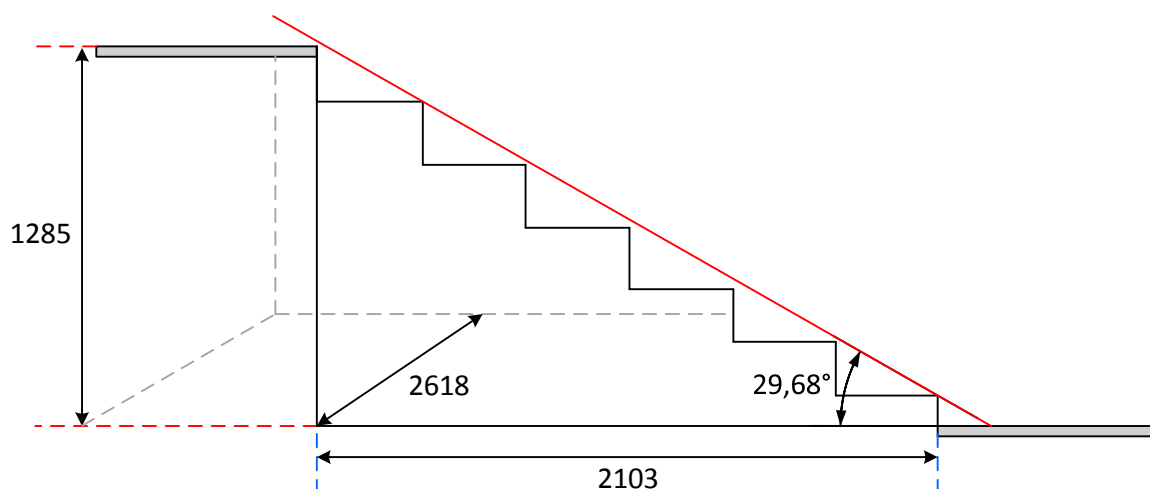
V tejto kapitole ukážeme funkčnosť generovania stupňov na modeloch reálnych schodísk. Taktiež sa budeme venovať overovaniu funkčnosti pohybu chodcov po viacerých typoch schodísk.

6.1 Generovanie stupňov

Na overenie generovania stupňov sme si vybrali dve schodiská. Obe schodiská sme vybrali tak, aby boli prístupné na podrobnejšiu analýzu a osobné premeranie rozmerov. Snažili sme sa čo najpresnejšie zmerať jednotlivé schody. Rozmery sú uvedené v milimetroch.

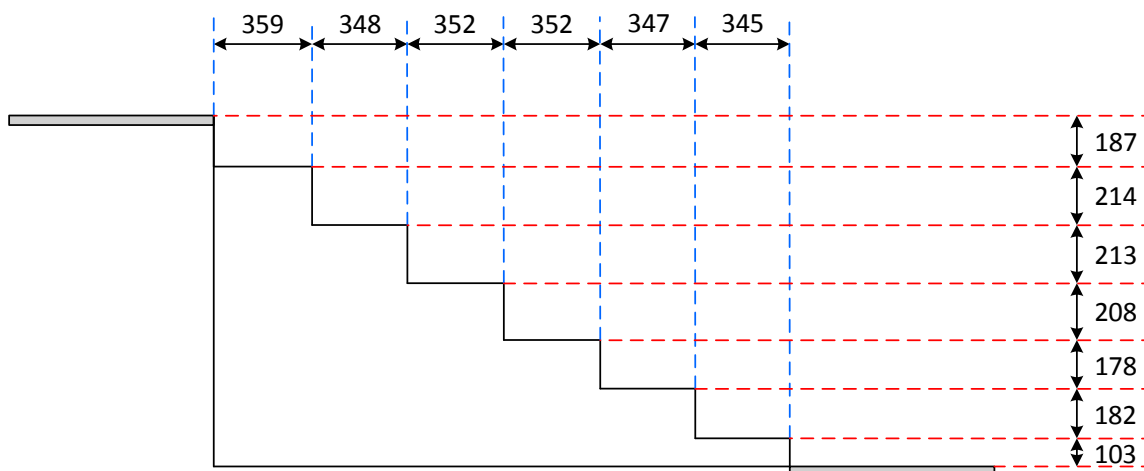
6.1.1 Schodisko pred vstupom do budovy

Ako je vidieť na obrázku 55, výška schodov je 1285mm, šírka 2618mm a horizontálna dĺžka je 2103mm. Taktiež je zobrazený sklon schodiska, ktorý má 29,68°.



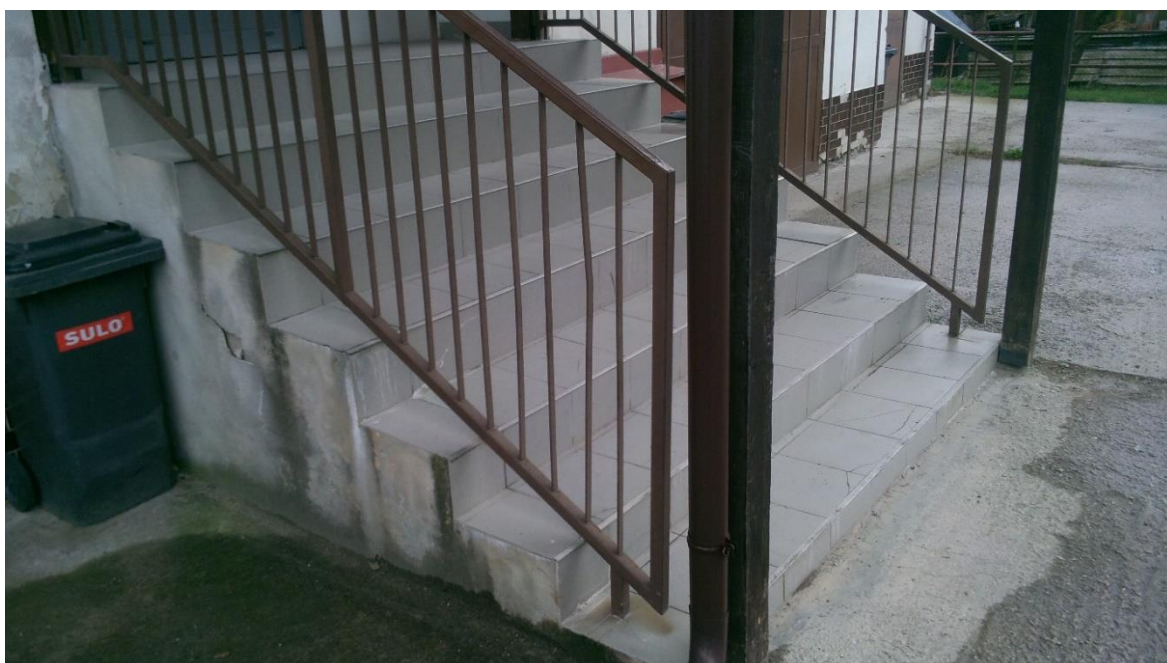
Obrázok 55: Rozmery modelovaného schodiska (mm)

Rozmery jednotlivých výšok a dĺžok schodov je možné vidieť na obrázku 56.



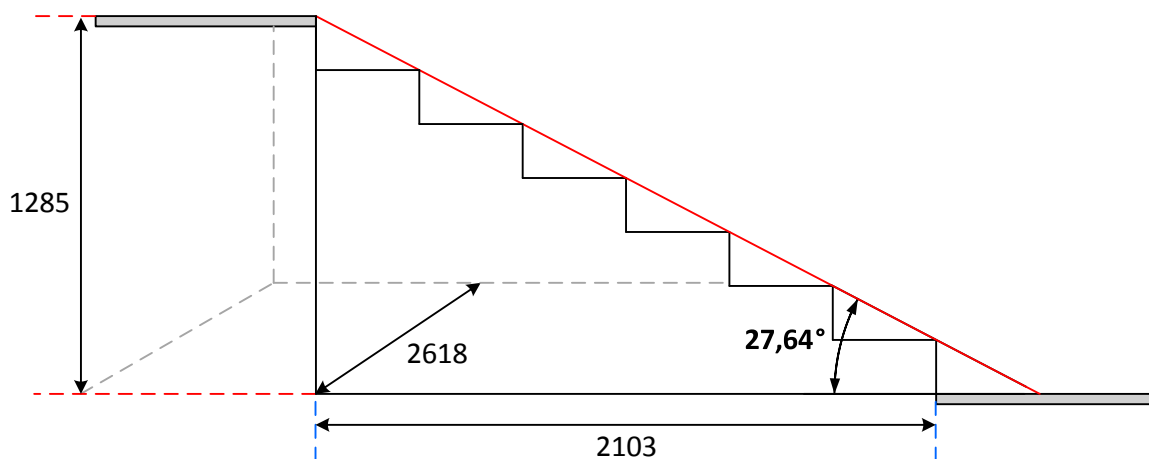
Obrázok 56: Rozmery jednotlivých schodov modelovaného schodiska (mm)

Na predchádzajúcom obrázku sme si mohli všimnúť, že jednotlivé schody nemajú rovnakú výšku. Taktiež výška najnižšieho schodu sa výrazne líši od ostatných. Rozdiely medzi rozmermi jednotlivých schodov mohli byť zapríčinené pri stavbe schodiska, pokladaní dlaždíc na schody alebo menšími odchýlkami pri meraní. Najnižší schod je zámerne nižší, aby mohli mať ostatné schody rovnaký rozmer, a aby sa dosiahol požadovaný sklon schodov. Na obrázku 57 je možné vidieť fotografiu modelovaného schodiska.



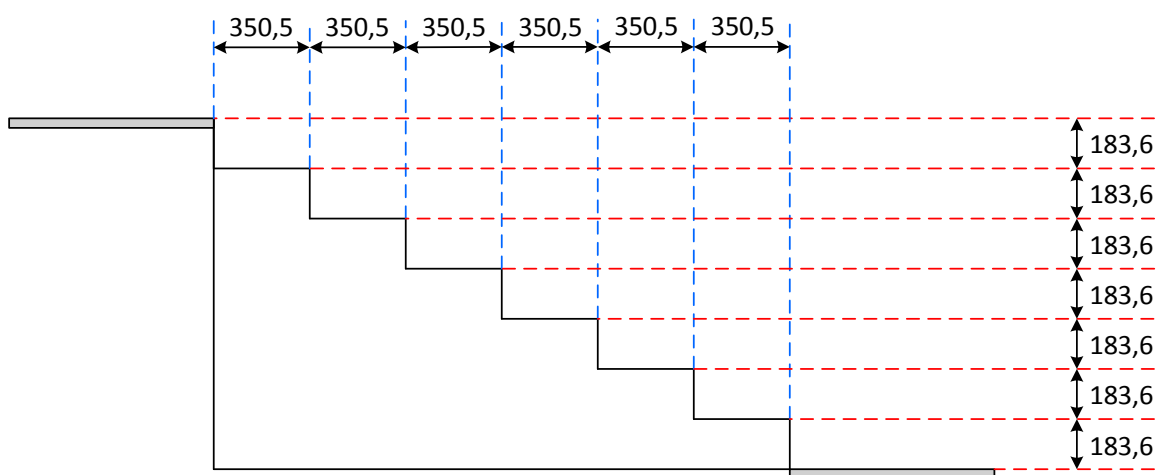
Obrázok 57: Modelované schodisko

Podľa nameraných rozmerov sme v simulačnom nástroji vytvorili model schodiska a následne sme dali vygenerovať jednotlivé stupne. Na obrázku 58 vidieť rozmery schodiska v simulačnom nástroji.



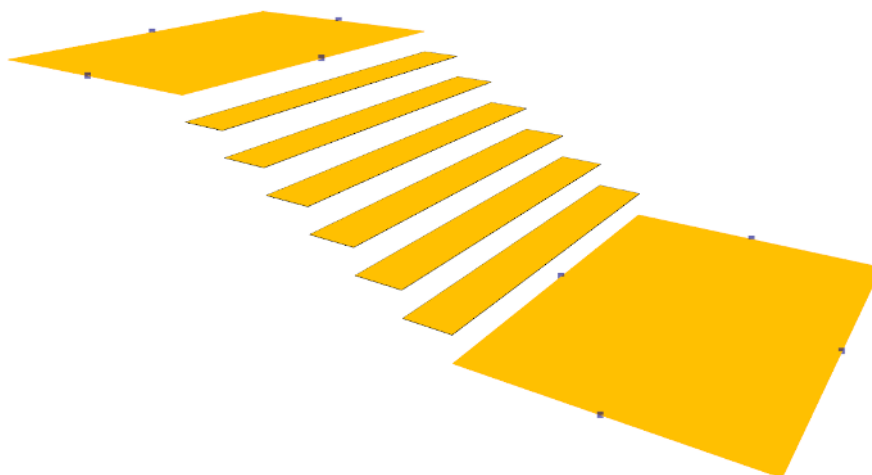
Obrázok 58: Rozmery schodiska v simulačnom nástroji (mm)

Vidíme, že všetky rozmery sú totožné s modelovanými schodmi, okrem sklonu. Uhol sklonu sme merali rovnako ako pri modelovaných schodoch. Schody v simulačnom nástroji majú menší sklon o $2,04^\circ$. Tento rozdiel je dosť výrazný. V takejto situácii môžeme využiť to, že sa dá pri kreslení schodov nastaviť ich sklon, ktorý majú v skutočnosti, aj napriek tomu, že nebude súhlasiť s ich geometrickou reprezentáciou v simulačnom nástroji. Na obrázku 59 je možné vidieť rozmery jednotlivých schodov schodiska v simulačnom nástroji.



Obrázok 59: Rozmery jednotlivých stupňov schodiska v simulačnom nástroji (mm)

Mohli sme si všimnúť, že jednotlivé schody majú rovnaké rozmery. Toto je spôsobené tým, ako sa generujú jednotlivé stupne. Obrázok 60 znázorňuje vygenerované schodisko v simulačnom nástroji.

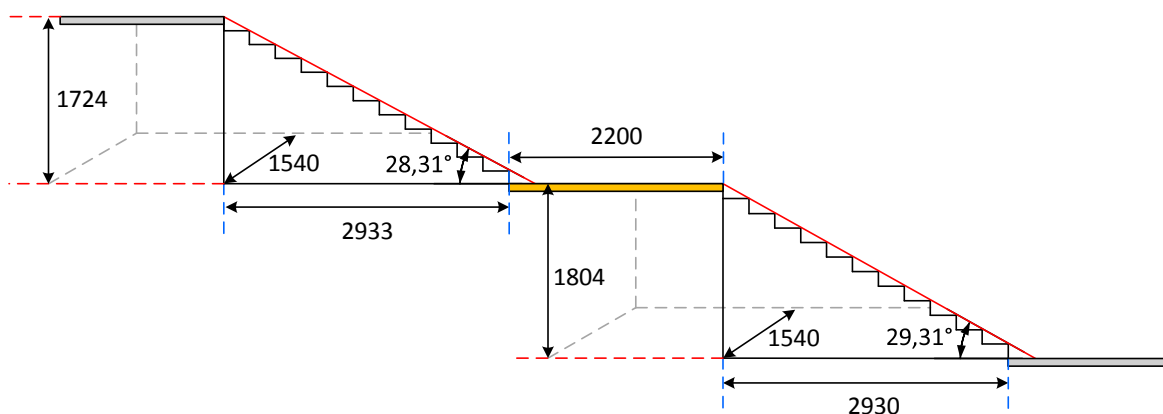


Obrázok 60: Zobrazenie schodiska v simulačnom nástroji

6.1.2 Schodisko s medziposchodím prepájajúce dve poschodia na FRI

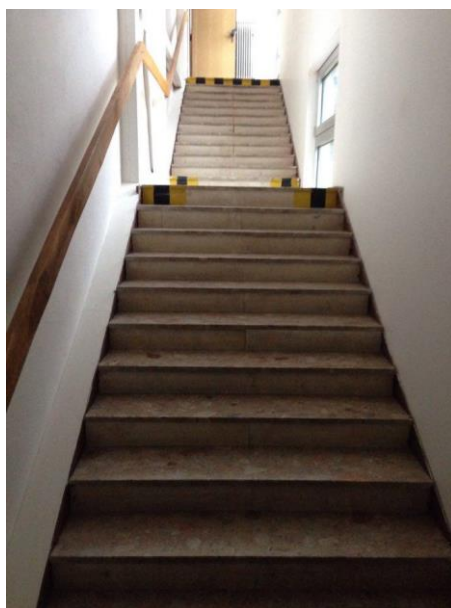
Toto modelované schodisko sa nachádza na Fakulte riadenia a informatiky Žilinskej univerzity, kde v budove A prepája druhé a tretie poschodie. Jeho súčasťou je jedno medziposchodie, ktoré je rovnako široké ako naň napájajúce sa schody. Budeme rozlišovať horné schody – tie, čo sa nachádzajú nad medziposchodím a dolné schody – nachádzajúce sa pod ním.

Na obrázku 61 je vidieť, že výška horných schodov je 1724mm, šírka 1540mm, horizontálna dĺžka je 2933mm a ich sklon je $28,31^\circ$. Výška dolných schodov je 1804mm, šírka 1540mm, horizontálna dĺžka je 2930mm a ich sklon je $29,31^\circ$. Dĺžka medziposchodia je 2200mm.



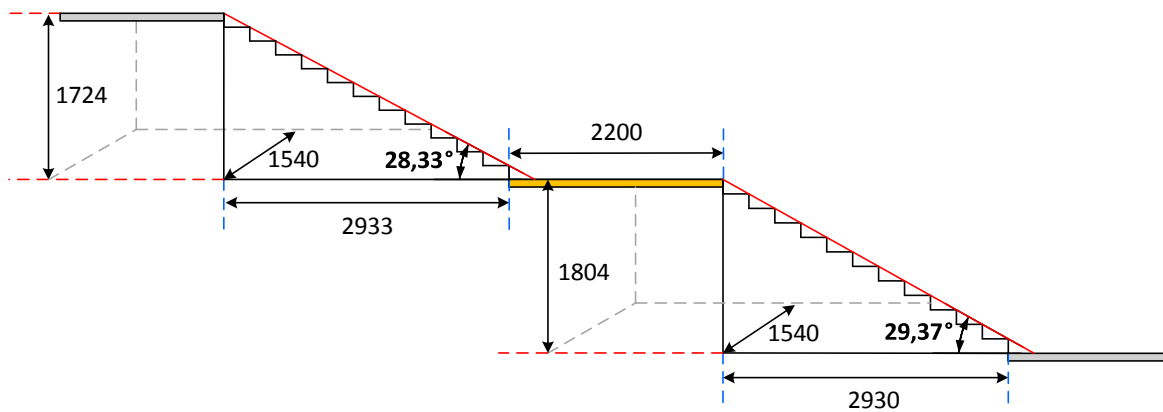
Obrázok 61: Rozmery modelovaného schodiska na FRI (mm)

Na hornom aj dolnom schodisku majú jednotlivé stupne veľmi podobné rozmery. Na obrázku 62 je fotografia modelovaného schodiska.



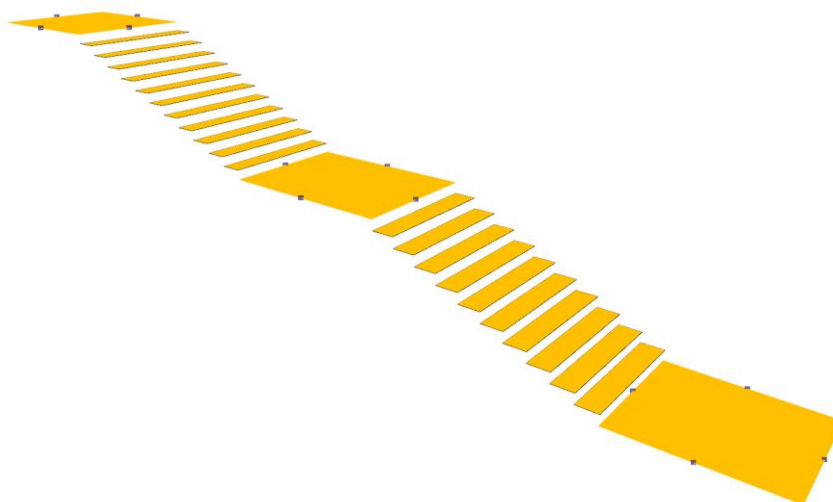
Obrázok 62: Modelované schodisko v budove FRI

Keďže jednotlivé stupne mali podobné rozmery, vygenerované stupne budú skoro presne zodpovedať skutočným stupňom schodiska. Na obrázku 63 vidieť, že sklon horných schodov v simulačnom nástroji sa od modelovaných líši len o $0,02^\circ$ a sklon dolných schodov o $0,06^\circ$. Takýto malý rozdiel môžeme zanedbať.



Obrázok 63: Rozmery schodiska na FRI v simulačnom nástroji (mm)

Na obrázku 64 je možné vidieť 3D náhľad schodiska v simulačnom nástroji.

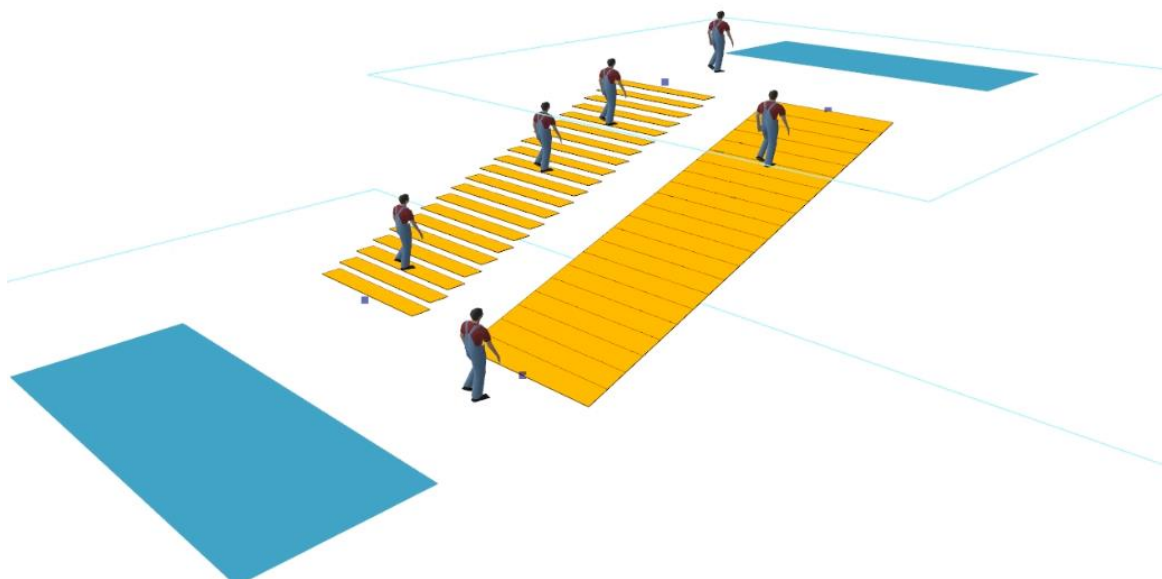


Obrázok 64: Zobrazenie schodiska na FRI v simulačnom nástroji

6.2 Pohyb chodcov po schodiskách

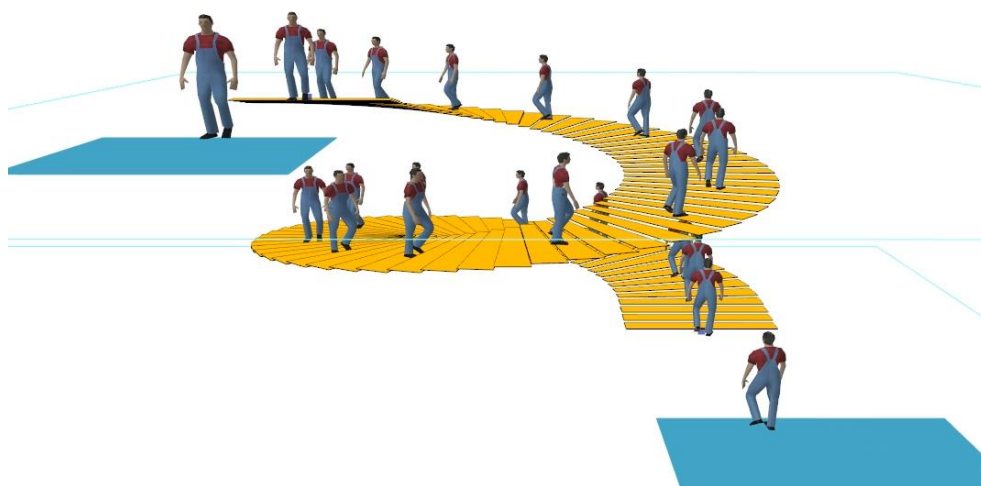
Jedným z cieľov práce bolo, aby sa chodci vedeli pohybovať po rôznych typoch schodísk. Túto úlohu sme splnili a pohyb chodcov po viacerých typoch schodísk sme zaznamenali na videozáznamy, ktoré je možné nájsť v prílohe.

Prvý videozáznam ukazuje rozdiely v pohybe chodcov po schodoch a rampe, ktorých rozmery sú rovnaké. Vľavo na videozázname sa nachádzajú schody a vpravo rampa. Ak sa budeme lepšie pozerať, je možné si všimnúť, že aj napriek podobnému sklonu schodov a rampy, sa chodci na schodoch pohybujú pomalšie. Toto je spôsobené tým, ako obmedzujeme rýchlosť chodcov v závislosti od sklonu a toho, či sa nachádzajú na schodoch alebo rampe.



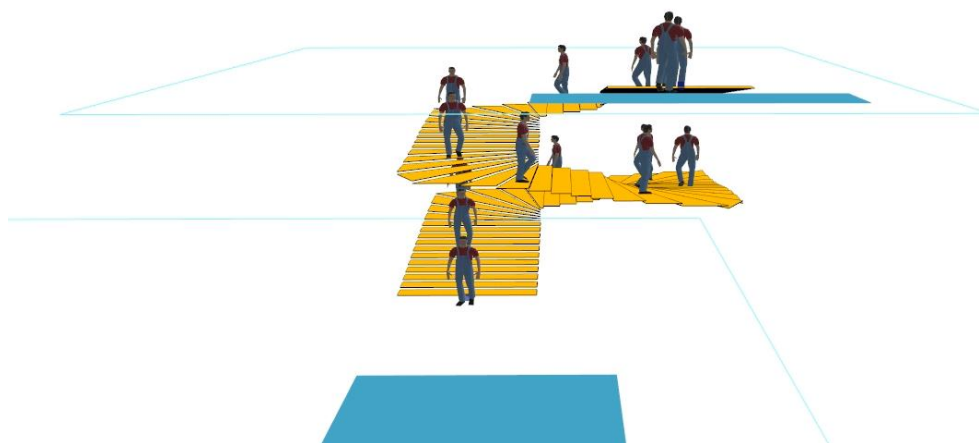
Obrázok 65: Snímka z prvého videozáznamu

Druhý videozáznam zaznamenáva pohyb chodcov na točitom schodisku. Je zreteľne vidieť to, že chodci, ktorí sa nachádzajú jeden nad druhým sa nijako neovplyvňujú, resp. o sebe nevedia. Toto je zabezpečené tým, ako chodci vnímajú svoje okolie na schodiskách. Taktiež je možné vidieť, že chodci plynulo zatáčajú popri tom ako idú po jednotlivých stupňoch, čo je spôsobené smerovými vektormi na nich.



Obrázok 66: Snímka z druhého videozáznamu

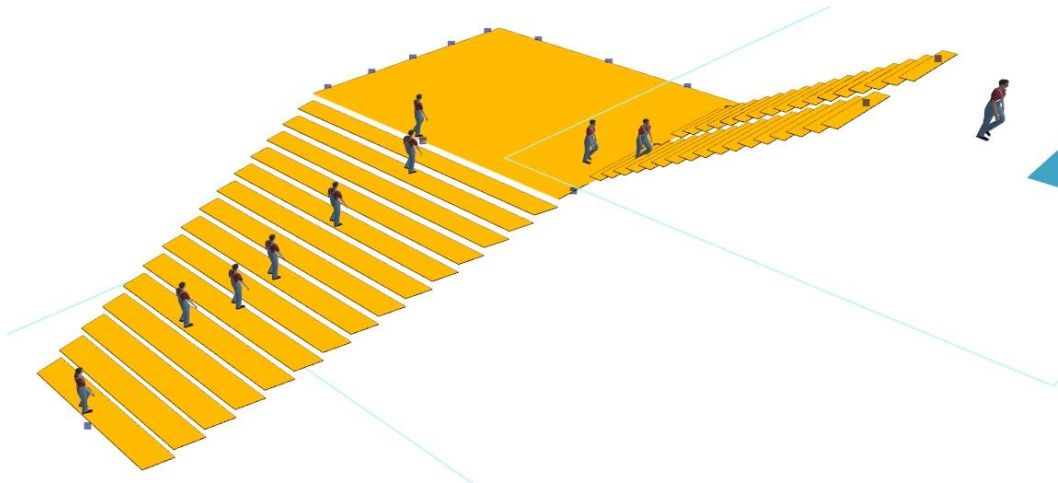
Na treťom videozázname je zachytený pohyb chodcov na hranatom točitom schodisku. Je možné vidieť, že aj na takýchto schodoch sa vedú chodci pohybovať. Aj na tomto videu je možné vidieť to, že chodci, ktorí sa nachádzajú nad sebou sa navzájom neovplyvňujú.



Obrázok 67: Snímka z tretieho videozáznamu

Na poslednom, štvrtom videozázname je možné vidieť schodisko, zložené zo schodov, ktoré vedú z prízemnia na medziposchodie, a dvoch eskalátorov, ktoré vedú z medziposchodia na prvé poschodie. Tieto eskalátory vytvárajú pár, kde jeden má smer

nastavený zhora nadol a druhý zdola nahor. Je možné vidieť, že chodci si vyberajú správny eskalátor na základe toho, kam sa chcú dostať. Tí chodci, ktorí sa nachádzajú na eskalátoroch sa pohybujú omnoho rýchlejšie. Oba eskalátory majú nastavenú vyššiu rýchlosť, aby bol lepšie vidieť rozdiel rýchlosti chodcov na nich a mimo nich. Tiež vidíme, že sa chodci vedia správne orientovať na medzипoschodí.



Obrázok 68: Snímka zo štvrtého videozáznamu

7 Záver

Cieľom práce bolo vytvoriť mikroskopický model pohybu entít na schodiskách v existujúcom simulačnom nástroji pre pohyb chodcov.

Najprv bolo potrebné oboznámiť sa so schodiskami v reálnom svete, rovnako ako aj s rampami, eskalátormi a travelátormi. Následne boli preskúmané iné simulačné nástroje pre pohyb chodcov. Na základe získaných skúseností pri vytváraní testovacích modelov v týchto simulačných nástrojoch, boli opísané jednotlivé prvky na prechod medzi poschodiami.

Bol predstavený nástroj na editovanie infraštruktúry – E3it, ktorý bol na základe predchádzajúcej analýzy rozšírený o kreslenie schodísk tak, aby v ňom bolo možné modelovať rôzne typy schodísk (točité, schodiská s medziposchodiami a pod.).

Dôležitou súčasťou bolo vytvorenie štyroch metód na generovanie stupňov schodov pre pohyb chodcov. Tieto metódy umožňujú generovať také stupne schodov, ktoré sa čo najviac približujú k realite.

Pre potreby mikroskopického modelu pohybu po schodiskách bol rozšírený simulačný svet o schodiská. Ďalej sa riešila navigácia chodca na schodiskách. V zónach so schodiskami a na schodiskách bolo upravené chodcove vnímanie okolitého sveta tak, aby sa mohol v takomto prostredí pohybovať. Bol predstavený mikroskopický model pohybu chodcov pomocou sociálnych síl, ktorý bol rozšírený tak, aby umožňoval pohyb chodcov aj na schodiskách. Vytvoril sa nový agent – agent eskalátorov, ktorý je zodpovedný za posúvanie jednotlivých chodcov nachádzajúcich sa na eskalátoroch a travelátoroch.

Overené bolo modelovanie schodísk a pohyb chodcov po schodiskách. Boli vytvorené modely dvoch schodísk, jedny mali začiatkový stupeň výrazne nižší ako ostatné stupne, druhé schodisko obsahovalo medziposchodie. Priebeh simulácie bol zachytený na videozáznamoch, na ktorých je vidieť pohyb chodcov na rôznych typoch schodísk (priamych, točitých a iných).

Všetky ciele práce boli naplnené. Dôraz bol kladený na generovanie stupňov schodísk, časovú zložitosť algoritmu pre vyhľadávanie okolitých vrstiev a funkčnosť pohybového modelu. Do budúca je možné modelovať špecifické správanie chodcov, napr. aby sa na schodiskách pohybovali po pravej strane, alebo aby niektorí chodci na eskalátoroch a travelátoroch stáli a niektorí kráčali.

Zoznam použitej literatúry

ADAMKO, N. 2013. *Agentovo orientovaná simulácia zložitých obslužných systémov*. Habilitačná práca. Žilinská univerzita v Žiline, Fakulta riadenia a informatiky, 2013, 28360020135002

ANYLOGIC. 2017. *AnyLogic* [online]. 2017 [cit. 2017-01]. Dostupné na internete: <<http://www.anylogic.com/>>

BLAHA, P. 2009. *Eskalátory* [online]. 2009 [cit. 2017-01]. Dostupné na internete: <<http://www.archiportal.sk/2009/11/14/eskalartory/>>

BLAHA, P. 2010. *Ako správne navrhovať rampy pre osoby so zníženou pohyblivosťou* [online]. 2010 [cit. 2017-01]. Dostupné na internete: <<http://www.archiportal.sk/2010/04/06/ako-spravne-navrhovat-rampy-pre-osoby-so-znzenou-pohyblivostou/>>

ČADECKÝ, M. 2015. *Vytvorenie mezoskopického modelu pohybu entít v simulačnom nástroji pohybu chodcov*. Diplomová práca. Žilinská univerzita v Žiline, Fakulta riadenia a informatiky, 2015, 28360220152371

EMPATE. 2017. *Technické normy* [online]. 2017 [cit. 2017-01]. Dostupné na internete: <<http://www.empate.sk/technicke-normy/>>

FABIANYOVÁ, I. 2012. *Bezbariérové riešenie budov občianskej vybavenosti a ich cena*. Bakalárska práca. Bankovní institut vysoká škola Praha, Zahraničná vysoká škola Banská Bystrica, Katedra ekonómie a oceňovania majetku, 2012, Dostupné na internete: <https://is.bivs.cz/th/16079/bisk_b/Bezbarierove_riesenie_budov_obcianskej_vybavenosti_a_ich_cena_b.p.pdf>

GAMMA, E., JOHNSON, R., VLISSIDES, J. a HELM, R. 1997. *Design Patterns: Elements of Reusable Object-Oriented Software*. Boston: Pearson Education (US), 1997, ISBN 978-0-201-63361-0

HELBING, D., BUZNA, L., JOHANSSON, A. a WERNER, T. 2005. *Self-Organized Pedestrian Crowd Dynamics: Experiments, Simulations, and Design Solutions*. Transportation science, roč. 39, č. 1.

- HELBBING, D., FARKAS, I. J., MOLNAR, P. a VICSEK, T. 2002. *Simulation of pedestrian crowds in normal and evacuation situations*. Pedestrian and evacuation dynamics, roč. 21, s. 21-59.
- HELBBING, D. a MOLNÁR, P. 1995. *Social force model for pedestrian Dynamics*. Dostupné na internete: <<http://vision.cse.psu.edu/courses/Tracking/vlpr12/HelbingSocialForceModel95.pdf>>
- HOOGENDOORN, S. P. a BOVY, P. H. L. 2004. *Pedestrian route-choice and activity scheduling theory and models*. Transportation research, Part B: Methodological. 2004, 38.2, s. 169-190.
- JENNINGS, N.R. 2001. *An agent-based approach for building complex software systems*. Communications of the ACM, 2001
- KAVIČKA A. 2003. *Petriho sít' s rozhodovacími přechody aplikovaná v rámci ABAsim architektúry simulačného modelu*. Ostrava: Modelling and Simulation of Systems, MARQ, 2003, ISBN 80-85988-89-0
- KONE. 2016. *Planning guide for kone escalators & autowalks* [online]. 2016 [cit. 2017-01]. Dostupné na internete: <http://www.kone.us/Images/KONE-Escalator-AutoWalk-Planning-Guide_tcm25-18783.pdf>
- KORMANOVÁ, A. 2014. *Hybridné simulačné modely pohybu a správania sa chodcov s nehomogénnou granularitou*. Dizertačná práca. Žilinská univerzita v Žiline, Fakulta riadenia a informatiky, 2014, 28360020143009
- KVASŇOVSKÝ, P. 2015. *Využitie výpočtového výkonu grafických kariet pre urýchlenie zvolených algoritmov v simulačnom nástroji pre pohyb chodcov*. Bakalárska práca. Žilinská univerzita v Žiline, Fakulta riadenia a informatiky, 2015, 28360220151428
- OASYS. 2015. *Massmotion* [online]. 2015 [cit. 2017-01]. Dostupné na internete: <http://www.oasisy-softwre.com/media/Manuals/Latest_Manuals/MassMotion_Flow.pdf>
- OASYS. 2017. *Massmotion* [online]. 2017 [cit. 2017-01]. Dostupné na internete: <<http://www.oasisy-softwre.com/products/engineering/massmotion.html>>
- PTV GROUP. 2017. *Viswalk* [online]. 2017 [cit. 2017-01]. Dostupné na internete: <<http://www.ptvgroup.com/en/solutions/products/ptv-viswalk/>>

SCHINDLER. 2007. *Planning guide for escalators and moving walks* [online]. 2007 [cit. 2017-01]. Dostupné na internete: <http://www.schindler.com/content/nz/internet/en/mobility-solutions/products/moving-walks/_jcr_content/rightPar/downloadlist/downloadList/70_1368592441298.download.asset.70_1368592441298/Planning%20Guide%20for%20Escalators%20and%20Moving%20Walks_en.pdf>

TEKNOMO, K. 2002. *Microscopic pedestrian flow characteristics: Development of an image processing data collection and simulation model*, dizertačná práca, Tohoku university Japan, 2002, 141s, dostupné na internete: <<http://people.revoledu.com/kardi/publication/Dissertation.pdf>>

THUNDERHEAD ENGINEERING. 2017. *Pathfinder* [online]. 2017 [cit. 2017-01]. Dostupné na internete: <<http://www.thunderheadeng.com/pathfinder/>>

VARGA, M. 2014. *Modelovanie získavania informácií a ich využitie pri inteligentnom rozhodovaní v agentovo orientovaných simulačných modeloch*. Dizertačná práca. Žilinská univerzita v Žiline, Fakulta riadenia a informatiky, 2014, 28360020143006

Príloha 1: CD médium

Priložené CD médium obsahuje:

- Práca v elektronickej podobe (formát PDF)
- 4 videozáznamy pohybu chodcov na schodiskách