



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

BLENDER ADD-ON PRO PROCEDURÁLNÍ GENEROVÁNÍ MOSTŮ

BLENDER ADD-ON FOR PROCEDURAL BRIDGE GENERATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ALŽBĚTA KUČEROVÁ

VEDOUCÍ PRÁCE

SUPERVISOR

TOMÁŠ CHLUBNA, Ing.

BRNO 2023

Zadání bakalářské práce



145492

Ústav: Ústav počítačové grafiky a multimédií (UPGM)
Studentka: **Kučerová Alžběta**
Program: Informační technologie
Specializace: Informační technologie
Název: **Blender add-on pro procedurální generování mostů**
Kategorie: Počítačová grafika
Akademický rok: 2022/23

Zadání:

1. Naučte se základy práce s 3D modelovacím programem Blender (verze 3.2 a výše).
2. Seznamte se se skriptovacím Blender Python API.
3. Nastudujte teorii o mostních konstrukcích a vyberte vhodné druhy mostů pro implementaci.
4. Navrhněte add-on, který rozšíří Blender možnost generování 3D procedurálních mostů.
5. Nastudujte existující a možná řešení dané problematiky.
6. Navrhněte uživatelské rozhraní pro výsledný add-on.
7. Add-on implementujte a demonstруйте jeho použití na různých typech dat.
8. Zdokumentujte a zveřejněte výsledný add-on.
9. Vytvořte video reprezentující výsledky vaší práce.

Literatura:

- [Blender API documentation](#)
- Chen, Wai-Fah, and Lian Duan, eds. *Bridge engineering handbook: construction and maintenance*. CRC press, 2014.
- Burke Jr, Martin P. "Bridge aesthetics: world view." *Journal of Structural Engineering* 121.8 (1995): 1252-1257.
- Leonhardt, Fritz. "Aesthetics of bridge design." *PCI Journal* 13.1 (1968): 14-31.
- Heggade, V. N. "Bridge aesthetics (case of science taken to the level of art)." *Architecture* 9 (2015): 2.
- Tang, Man-Chung. "Forms and aesthetics of bridges." *Engineering* 4.2 (2018): 267-276.
- Freiknecht, Jonas, and Wolfgang Effelsberg. "A survey on the procedural generation of virtual worlds."
- Ebert, David S., et al. *Texturing & modeling: a procedural approach*. Academic Press, 2014. ISBN 1483297020, 9781483297026

Při obhajobě semestrální části projektu je požadováno:
Body 1 až 6, experimenty vedoucí k bodu 7.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Chlubna Tomáš, Ing.**
Vedoucí ústavu: Černocký Jan, prof. Dr. Ing.
Datum zadání: 1.11.2022
Termín pro odevzdání: 10.5.2023
Datum schválení: 31.10.2022

Abstrakt

Cílem této práce je navrhnout a implementovat rozšíření pro platformu Blender, které za pomoci procedurálního přístupu usnadní tvorbu trojrozměrných modelů mostů. Oproti ostatním již existujícím řešením umožňuje generování více typů konstrukcí. Zároveň umožňuje sestavovat části mostů z objektů uživatele či z nabídky modelů poskytovaných doplňkem samotným. Procedurální tvorba geometrie je implementována systémem Geometry Nodes, který poskytuje oproti tradičnímu modelování mnoho výhod. Výsledný add-on byl publikován a je dostupný online pro volné použití.

Abstract

The aim of this thesis is the draft and implementation of a Blender add-on, which would simplify the process of modeling three-dimensional models of bridges while implementing a procedural approach. In contrast to already existing solutions, this implementation enables creation of multiple construction types. It also assembles select parts of the bridge either out of user's own objects or models provided by the add-on itself. Procedural creation of geometry is implemented using Geometry Nodes as it provides many benefits over traditional methods of modeling. The final product was published and is available for free use.

Klíčová slova

Blender, add-on, most, architektura, model ve 3D, procedurální generování, UV rozbalení, UV mapa, Python

Keywords

Blender, add-on, bridge, architecture, 3D model, procedural generation, UV unwrap, UV map, Python

Citace

KUČEROVÁ, Alžběta. *Blender add-on pro procedurální generování mostů*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Tomáš Chlubna, Ing.

Blender add-on pro procedurální generování mostů

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracovala samostatně pod vedením pana Ing. Tomáše Chlubny. Uvedla jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpala.

.....
Alžběta Kučerová
5. května 2023

Poděkování

Chtěla bych poděkovat svému vedoucímu inženýru Tomáši Chlubnovi za jeho vstřícný přístup a nápomocné rady. Stejně tak chci poděkovat svým přátelům a rodině za jejich podporu.

Obsah

1	Úvod	2
2	Problematika generování mostů na platformě Blender	3
2.1	Blender	3
2.2	Základní terminologie	8
2.3	Architektura mostů	9
2.4	Průzkum existujících řešení	16
3	Návrh řešení	17
3.1	Cílová funkcionalita	17
3.2	Uživatelské rozhraní	19
3.3	Programový návrh	21
4	Implementace	22
4.1	Generování geometrie mostu	22
4.2	UV rozbalení a texturování	36
4.3	Předpřipravené modely	39
4.4	Zprovoznění uživatelského rozhraní	46
5	Testování	47
5.1	Průběžné testování funkcionality	47
5.2	Konečné testování využití v praxi	48
6	Závěr	54
	Literatura	55
A	Implementační schémata	57

Kapitola 1

Úvod

V oblasti procedurálního generování existuje nemalý zájem o generování architektury. Jsou prozkoumány možnosti generování silnic, budov a dokonce i rozložení měst, což výrazně usnadňuje práci při budování modelů obcí nejenom ve videohrách. Mosty jsou běžný architektonický útvar, se kterým se běžný člověk setkává v každodenním životě, a proto lze na trhu nalézt nástroje na jejich generování. V realitě existuje mnoho typů mostů a k jejich generování lze přistoupit nekonečně mnoha způsoby, proto uživateli nemusí vyhovovat žádný z již existujících generátorů mostů.

Tato práce si klade za cíl implementovat add-on pro aplikaci Blender verze 3.5, který s využitím technologie Geometry Nodes umožňuje snadné vytváření modelů mostů různých typů konstrukcí na základě požadavků uživatele. Následovně je možné vygenerovaný model mostu texturovat po jednotlivých částech. Generování libovolného mostu je v praxi nereálné, cílem však bylo zařídit dostatečnou flexibilitu, proto je uživateli umožněno do tvorby mostu zakomponovat i vlastní modely. Doplněk zároveň nabízí i předem připravené modely, které jsou jako součást finální verze add-onu zveřejněny online.

Kapitola 2 slouží k uvedení do problematiky mostní architektury a je taktéž zaměřena na to, jaké možnosti nabízí prostředí Blender pro zjednodušení procesu modelování pomocí procedurálního generování. V kapitole 3 se lze dočíst o návrhu řešení – jakým způsobem lze doplněk ovládat, strukturu jeho uživatelského rozhraní a základní dekompozici funkcionality. Dále kapitola 4 je věnována implementaci samotné – procesům generování jednotlivých mostních součástí, získání UV mapy mostu. Zároveň tato kapitola uvádí souhrn všech modelů částí mostu, ze kterých lze vybírat. Nakonec testování výsledného doplňku je popsáno v kapitole 5.

Kapitola 2

Problematika generování mostů na platformě Blender

Tato kapitola slouží k teoretickému seznámení čtenáře s nástroji použitými při implementaci této bakalářské práce a stejně tak k uvedení základních znalostí o předmětu práce – mostech a jejich konstrukci. Sekce 2.1 začíná představením aplikace Blender, pro který je výsledný produkt určen, načež 2.2 navazuje a přináší krátké vysvětlení základních pojmů související s přecházejícím textem a které jsou používány dále v textu. Následující část kapitoly 2.3 se zabývá tematikou mostů. Nakonec sekce 2.4 uvádí několik již existujících generátorů mostů, ukazuje, jaké mají silné a slabé stránky, a nastiňuje, co by mohl výsledek této práce dělat jinak nebo lépe.

2.1 Blender

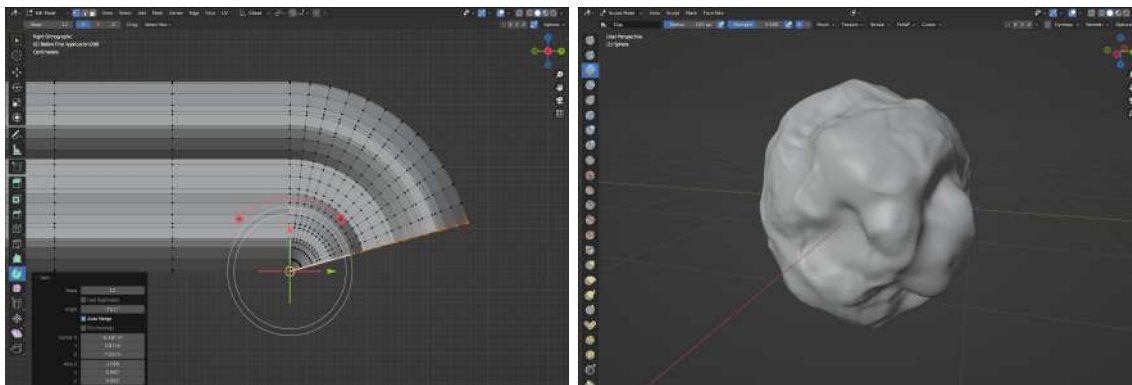
Blender¹ je bezplatný, volný (*angl. open-source*) nástroj určený pro práci s počítačovou grafikou. Mimo modelování umožňuje uživateli animovat, vytvářet a aplikovat textury, vyrábět a editovat videa, snímat pohyb ve videích a mnoho dalšího. Díky rozmanitosti si tato aplikace získala velkou popularitu a vypěstovala si velkou komunitu. Jedním z důvodů oblibenosti tohoto nástroje je fakt, že nabízí uživatelům poměrně nenáročnou možnost jeho funkcionalitu rozšiřovat ve formě add-onů.

2.1.1 Metody modelování

V případě modelování podporuje Blender mnoho přístupů ať už tradičním manuálním, nebo procedurálním způsobem. Každý má své výhody a nevýhody a hodí se pro jiné situace. Tradiční (manuální) modelování je proces ručního vytváření modelů. Typicky je prováděno s pomocí myši a klávesnice (ne však nezbytně) a v případě složitějších modelů se jedná o časově náročný proces, jehož trvání se může pohybovat v řádech desítek hodin a více. Jedním z předních zástupců modelovacích prostředí, které Blender nabízí, je editační režim (*edit mode*), ve kterém může uživatel upravovat již existující objekt postupnými operacemi nad jeho vertexy, stranami a stěnami. Oproti tomu takzvaný *sculpt mode* nabízí řadu štětců pro následnou úpravu povrchu objektu a často se používá při dodávání obličejových detailů modelům postav nebo dalším organickým materiálům. Oba tyto módy (*edit mode* a *sculpt*

¹<https://www.blender.org/>

mode), jejichž ukázky lze vidět na obrázcích 2.1, spadají do kategorie tzv. destruktivního modelování.



(a) Obrázek zobrazuje nástroj spin použitý pro ohyb plechu v *edit mode*

(b) Obrázek ukazuje model kamene vymodelovaného pomocí štětců ve *sculpt mode*

Obrázek 2.1: Ukázky manuálního modelování v Blender verze 3.5

Destruktivní modelování se vyznačuje tím, že nelze jednoduše měnit kroky prováděné v procesu modelování. Pokud by modelář v průběhu modelování zjistil, že několik kroků zpět udělal chybu nebo mohl udělat něco lépe, musel by opakovaně vracet provedené kroky, dokud by nedošel k onomu bodu, který hledal. Tímto způsobem by ztratil všechny svoje kroky, které udělal po akci, kterou potřebuje vrátit. Často je nevyhnutelné nebo dokonce vhodné provádět modelování ručně, ale v některých případech je vhodné si práci ulehčit a aplikovat procedurální přístup k modelování.

Procedurální modelování lze definovat jako generování za pomoci funkce nebo programu [2]. Ať už má být výstup jakýkoliv, jeho cílem je ušetřit čas uživateli, aby nemusel opakovaně dělat něco, co lze automatizovat. Jelikož tento přístup k modelování spočívá v tom, že model vytváří počítač na základě daného algoritmu, pak je nutné, aby programátor takový algoritmus počítači nejprve poskytl. V aktuální verzi Blenderu 3.5 je možné takový algoritmus implementovat buď pomocí skriptovacího jazyka Python, nebo pomocí nástroje Geometry Nodes, kterému se věnuje sekce 2.1.3.

2.1.2 Příklady procedurálního generování v Blenderu

Pro Blender bylo postupem času implementováno mnoho doplňků zabývajících se generováním geometrie – nejen předmětů, ale také terénu. Nejznámější add-on zabývající se vytvářením krajiny a dokonce i planet je ANT Landscape². Ten nabízí mnoho režimů používající různé typy šumů (například fraktální šum), s pomocí kterých generuje geometrii terénu. Jak bývá u generátorů poměrně časté, používá jako jeden ze svých vstupů proměnnou s názvem *Random Seed*, který do generování vnáší určitou míru náhody.

S vytvářením terénu je často spjata následná tvorba vegetace. Pro vytváření modelů stromů je v aplikaci Blender dostupný doplněk Sapling Tree Gen (ukázka na obrázku 2.2), což ale není jediný add-on kdy vytvořený pro generování vegetace. Jeden takový doplněk byl implementován jako součást bakalářské práce na jedné české univerzitě. Je jím VegGen, který je volně dostupný ke stažení online³. Je zajímavý tím, že se jedná o program založený

²https://docs.blender.org/manual/en/latest/addons/add_mesh/ant_landscape.html

³<https://eiler2.wixsite.com/treegen/home>

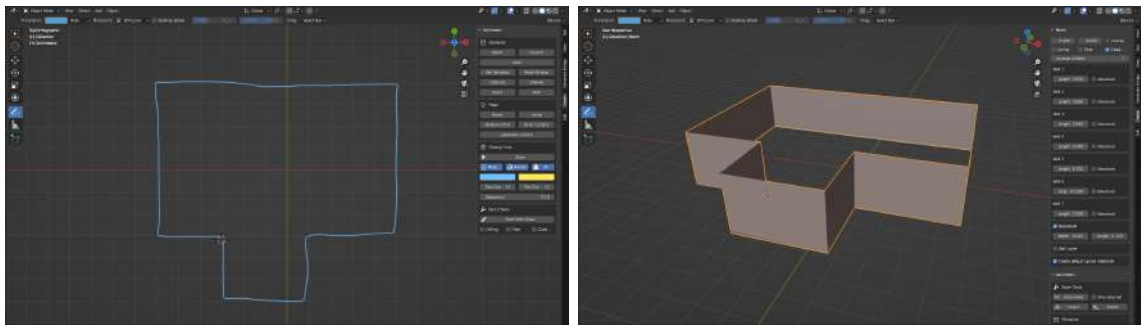
na L-systémech, který byl v roce 1968 navržen právě za účelem procedurálního generování vegetace [6, strana 7]. VegGen byl implementován pro Blender verze 2.60 a psaný v Pythonu. V nejnovějších verzích již není podporovaný.



Obrázek 2.2: Ukázka stromu vygenerovaného pomocí Sapling Tree Gen

Dalším příkladem procedurálního tvoření přírodních prvků je Ivy Gen, který je schopný umístit po povrchu 3D objektu břečtan, což může přidat architektonickým prvkům určité množství realismu. Tento doplněk má (jako velké množství dalších) základní model (list břečtanu) předem vymodelovaný a uložený ve vedlejším *.blend* souboru.

Virtuální prostředí je velmi často potřeba zaplnit lidmi vytvořenými objekty, z nichž se nejvíc nabízí stavební prvky jako domy. Stejně jako pro přírodní prvky, pro architekturu taktéž existuje mnoho doplňků. Jedním z nich je přímo v Blenderu dostupný doplněk Archimesh⁴, který slouží hlavně k modelování domů, případně pokojů. Ten uživateli poskytuje modely stěn, dveří, oken a základního vybavení domácnosti (lampy, kuchyňské skříňky apod.). Mimoto umožňuje generování pokojů podle náčrtu rozložení stěn jako ukazují snímky 2.3.



(a) Náskres polohy stěn pomocí anotace

(b) Stěny pokoje vygenerované z poskytnutého náčrtu

Obrázek 2.3: Ukázka možnosti generování stěn z náčrtu v doplňku Archimesh

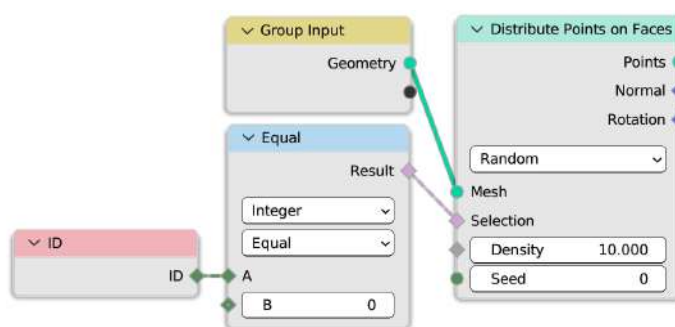
⁴https://docs.blender.org/manual/en/latest/addons/add_mesh/archimesh.html

2.1.3 Geometry Nodes

Geometry Nodes je systém zabudovaný do aplikace Blender od verze 2.90 [1], která vyšla v roce 2020. Tento nástroj umožňuje programovat i těm, kteří v životě nikdy neprogramovali, a těm, kteří neznají syntax jazyka Python, který je běžně používán při skriptování. Geometry Nodes totiž nabízí vizuální způsob programování. Vizuální programování je systém, který uživateli umožňuje specifikovat program ve dvou nebo více dimenzích (do této kategorie nespádají běžné textové programovací jazyky, protože ty zpracovávají proud dat o jedné dimenzi) [8].

Jedná se o modifikátor, který umožňuje definovat geometrii objektu pomocí 2D grafu uzlů (*angl. Node Group*). Jednotlivé uzly jsou charakterizovány názvem, parametry, a vstupy a výstupy (*angl. sockets*). Tyto uzly lze chápat jako funkce a celý graf uzlů lze brát jako určitou posloupnost volání funkcí. Platí pravidlo, že vstupy jsou umístěny na levé hraně uzlu a výstupy na pravé straně, z čehož plyne, že stromová struktura uzlů má začátek nalevo a konec napravo.

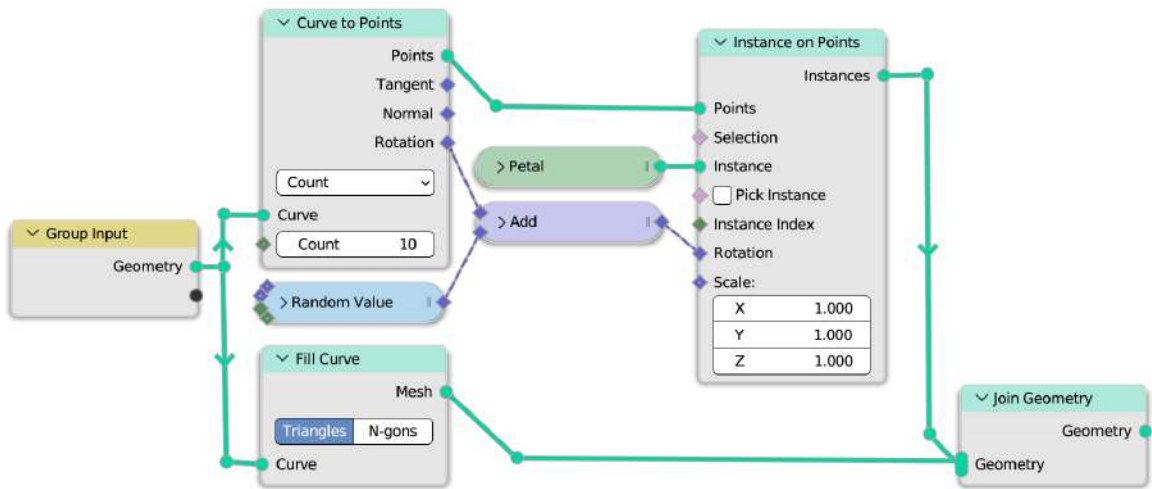
Skupina uzlů dostává na vstup skrze uzel s názvem **Group Input** geometrii objektu a na výstup dává pomocí uzlu **Group Output** výslednou geometrii. Mezi těmito dvěma uzly lze vkládat a propojovat další uzly, čímž dochází k upravování definice geometrie. Je možné přitom uzel **Group Input** zanedbat, což způsobí to, že vstupní geometrie nebude mít na chování tohoto modifikátoru žádný vliv. Vstupy a výstupy uzlů se rozlišují podle následujících dvou parametrů: tvaru a barvy. Je vhodné podotknout, že málokdy jsou uzly odlišných barev a tvarů kompatibilní. Pokud je spoj mezi vstupy a výstupy neplatný, pak se to projeví na barvě spoje – ta nabude červené barvy. Barev vstupů a výstupů je v Geometry Nodes mnoho (a jsou používány nejenom v této části aplikace Blender), ale nejčastěji používané jsou převážně šedá, zelená, tmavě modrá, růžová a tyrkysová, které v daném pořadí představují datové typy float, integer, vektor, boolean a geometrii. Geometrie může být často specifikována na konkrétní typ geometrie (např. křivka). Vstupy a výstupy jsou dvojího typu a odlišují se podle svého tvaru: kruh nebo diamant. Rozdíly mezi těmito dvěma tvary vstupů a výstupů jsou demonstrovány na příkladu (viz obrázek 2.4).



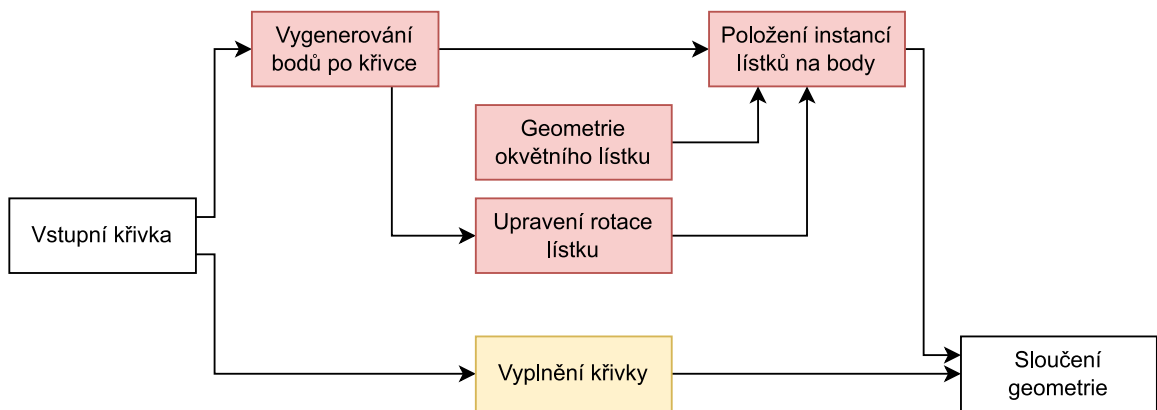
Obrázek 2.4: Ilustrační segment stromu uzlů: Na kruhový vstup uzlu **Distribute Points on Faces** jsou předávána data geometrie objektu a na diamantovém vstupu **Selection** je definováno, že efekt uzlu **Distribute Points on Faces** se má projevit pouze na straně, která má identifikátor roven nule.

Spoj mezi kruhovým výstupem a vstupem slouží k přenosu dat z výstupu do daného vstupu. Chová se jako předávání parametru funkci. Tvar diamantu představuje tzv. *field*, který neslouží k přímému předávání dat, nýbrž k definici chování uzlu, do kterého spoj vede. Nakonec existují dva další specifičtější tvary vstupů a výstupů. Diamant s tečkou je vstup,

na který je možno připojit kruhový i diamantový výstup – v závislosti na tom se bude chování uzlu měnit. Nakonec existuje zvláštní typ vstupu, který umožňuje nespecifikované množství spojů.



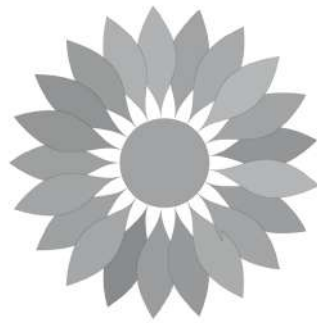
(a) Geometry Nodes diagram



(b) Zjednodušený diagram

Obrázek 2.5: Kruhová křivka na vstupu reprezentuje střed květu. Tato křivka je následovně zpracována ve dvou větvích. Ve vrchní větvi je převedena na body s pevnou délkou rozestupů mezi nimi. Dále jsou tyto body předány na vstup uzlu `Instance on Points`, který na pozice těchto bodů nasadí instance objektu okvětního lístku (výstup uzlu `Petal`). Pro každý list je vypočítána jeho rotace podle tečny (výstup `Rotation` uzlu `Curve to Points`), která je mírně odlišná pro každý list (k rotaci je připočítána náhodná hodnota), aby okvěti vypadalo přirozeněji.

Modelování pomocí Geometry Nodes může vypadat podobně, jako je vyobrazeno výše na ukázce 2.5, který zobrazuje postup modelování jednoduchého květu. Obrázek 2.5a ukazuje původní Geometry Nodes diagram. Zjednodušený pohled na tento diagram nabízí obrázek 2.5b. Výsledek modelování je k vidění na obrázku 2.6a. Barvu lze objektu následně přiřadit ručně nebo je možné využít nástroj Shader Nodes, který se velmi podobá systému Geometry Nodes v tom, že používá stromovou strukturu uzlů. Obrázek 2.6b ukazuje model po jeho vybarvení s pomocí Shader Nodes.



(a) Květ, jehož tvar byl vygenerován s pomocí systému Geometry Nodes



(b) Květ po vybarvení s pomocí systému Shader Nodes

Obrázek 2.6: Jednoduchý model květu (a) a jeho obarvená verze (b): Rozdílné odstíny šedi jednotlivých lístků neobarveného modelu udávají mírné nuance v jejich rotaci a ne jejich skutečnou barvu – model sám o sobě je bezbarvý.

2.2 Základní terminologie

Níže jsou uvedeny definice pojmů, které jsou nezbytné pro pochopení dalších částí této práce a to především kapitoly 4 zabývající se implementací. Všechny tyto termíny jsou převzaty z glosáře aplikace Blender [11] či odvozeny z ostatních sekcí dokumentace.

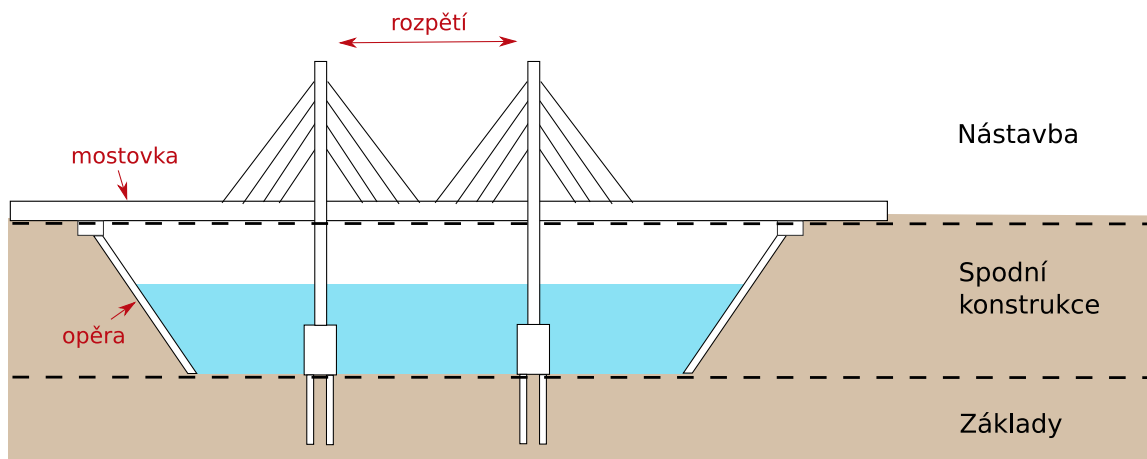
Add-on	Doplněk rozšiřující původní funkcionalitu určitého software – v tomto kontextu nástroje Blender.
Extrude	Jedná se o operaci, která duplikuje vrcholy, které nechává propojené s původními vrcholy a vytváří přitom nové stěny a hrany. Operace extrude může být provedena nad jednotlivými vertexy, hranami či stěnami.
Mesh	Způsob popisu geometrie objektu s pomocí vertexů (vrcholů), stran a hran.
Modifikátor	Modifikátor představuje nedestruktivní operaci nad daty. K nejznámějším modifikátorům patří Subdivision Surface, Mirror, Array a další.
Objekt	V kontextu práce tento pojem vyjadřuje obálku pro základní 3D data a jejich typ (mesh, křivka a další typy, které jsou mimo rámec této práce).
Operátor	Operátor je spustitelná akce, která je provedena okamžitě v době spuštění.
UV rozbalení	UV rozbalení (<i>angl. UV unwrapping</i>) je převedení objektu ve 3D do 2D reprezentace umožňující aplikování materiálů – UV mapy.
UV mapa	Termín definující vztah povrchu objektu typu mesh vůči dvourozměrné textuře.
Vertex Group	Kolekce vrcholů sloužící pro oddělení vrcholů do skupin pro odlišné zpracování.

2.3 Architektura mostů

Jednou z možných definicí mostu je, že se jedná o strukturu, která má umožnit překročení nějaké překážky [9, kapitola 1, strana 1]. Takovou překážkou mimo jiné bývá vodní tok, výškové snížení terénu nebo kombinace obojího. Na mosty je možné nahlížet z mnoha možných úhlů – z fyzikálního hlediska, estetického či stavebního hlediska, z pohledu funkcionality a tak dále. Pro potřeby této bakalářské práce se tato kapitola bude zabývat převážně jejich geometrií. Sekce 2.3.1 podá přehled o základní struktuře mostů a následovně v sekci 2.3.2 je popsáno dělení mostů podle různých aspektů a jsou shrnuty charakteristiky jednotlivých druhů mostních staveb.

2.3.1 Části mostní konstrukce

Most se skládá ze tří vrstev: základů, spodní konstrukce a nástavby. Základy tvoří nejspodnější část mostu a mohou být považovány za součást spodní konstrukce [9, kapitola 1, strana 4]. Běžný člověk s nimi nemusí přijít v životě do styku, jelikož jsou zasazeny hluboko v půdě. Spodní konstrukce a nástavba představují nadzemní část mostu, přičemž nástavba stojí na spodní konstrukci, která ji podepírá a přenáší účinky zátěže působící na most do půdy. S ohledem na způsob konstrukce jsou mosty složeny z částí popsaných níže. Vybrané z nich jsou znázorněny na obrázku 2.7.



Obrázek 2.7: Znázornění vybraných částí mostu

- Mostovka** Tento pojem (*angl. deck, slab*) představuje tu část nástavby, na které je umístěna komunikace. Po jejím povrchu se přepravují objekty (většinou dopravní prostředky a lidé) a po stranách mívá z důvodu bezpečnosti umístěné zábradlí.
- Rozpětí** Rozpětí (*angl. span*) je část mostu vymezená dvěma podpěrami, přičemž most může mít takové segmenty různě dlouhé (několik rozpětí různé délky).
- Pilíř** Pilíře (*angl. pillar*) jsou stavební bloky, které podpírají most po jeho délce. Účelem pilíře je přenášet zátěž působící na most dolů do základů.

- Opěra** Opěry (*angl. abutment*) jsou součástí mostu umístěné na obou jeho koncích. U kratších variant mostů je možné si jich snadněji všimnout, protože působí v porovnání se zbytkem mostu výrazněji. Opěry jsou zanořeny do zeminy pod silnicí na začátku, respektive na konci mostní konstrukce, a mají zamezit sesuvu půdy, na které konce mostu stojí. Podobně jako pilíře mají převádět zátěž z nástavby pryč.
- Oblouk** Oblouk je prvek, který je tvarem buď zakřivený nebo lomený [12, strana 209].
- Kabel** Kabel je prvek nezbytný pro stavbu visutých a závěsných mostů. Běžně je jedním koncem připevněn k mostní věži a druhým upevněn k mostovce nebo k další věži.
- Ledolam** Ledolam (*angl. ice-break, ice-breaker*) je ochranná konstrukce chránící spodní části mostů, které jsou postaveny přes vodní plochu. Svoji funkci ledolamy naplňují v zimním období, když hrozí poškození mostu ledovými krami tvořícími se na povrchu hladiny vody. Tuto část mostu lze pozorovat například na pražském Karlově mostu na obrázku 2.8.



Obrázek 2.8: Ledolamy Karlova mostu v Praze, autorem je uživatel BrokenSphere⁵

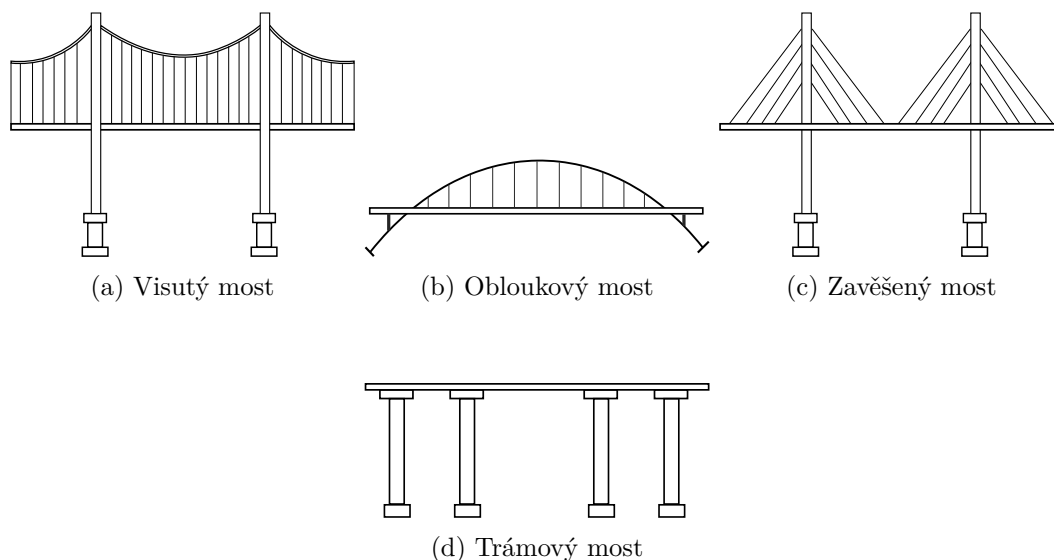
⁵https://commons.wikimedia.org/wiki/File:Charles_Bridge_arches_from_the_Vltava_River_2.JPG

Určitá kombinace výše zmíněných stavebních prvků udává konstrukci mostu a jeho všeobecný tvar. Hlubší množství detailů mostu následně přináší drobné odlišnosti a nedokonalosti jeho jednotlivých částí. Zároveň mosty mezi sebou odlišují další volitelné prvky jako pouliční lampy či jiná osvětlení, chodníky, zastřešení, římsy a další ozdobné a čistě estetické prvky jako jsou například sochy. Velký dojem dokáží budit mosty, které jsou ve večerních hodinách osvětleny.

2.3.2 Typy mostů dle konstrukce a materiálu

Základním hlediskem, podle kterého se mosty rozdělují, je materiál – dřevo, kámen, železobeton a další. Použitý materiál je přitom úzce spjatý s typem konstrukce, protože ta udává stavební prvky, které jsou při stavbě mostu použity – lano, oblouk a další prvky uvedené v sekci 2.3.1. Jednotlivé části mostu vyžadují, aby byly vyrobeny z určitého materiálu, aby mohly konat svou funkci.

Jak již bylo zmíněno v předešlé sekci kapitoly 2.3.1, tvar takového mostu záleží především na tom, jakého typu je jeho konstrukce. Nicméně neexistuje sjednocený názor na to, kolik existuje základních typů mostů, protože některé zdroje řadí určité typy mostů jako podkategorie jiných, zatímco jiné je považují za různé (a sice podobné) typy konstrukcí. Zbytek této kapitoly se zabývá vybranými základními variantami mostů [5] a popisuje čím se od sebe navzájem liší a čím se vyznačují. Obrázek 2.9 vyobrazuje typy mostů, které jsou v této kapitole probrány:



Obrázek 2.9: Typy mostů dle konstrukce

Nejběžnější a historicky nejstarší [3] typ mostu je trámový most (*angl. beam bridge*). Jak ilustruje obrázek 2.10, tento typ mostu zaujímá formu desky, která je na koncích podpírána opěrami. Pokud se jedná o delší most tvořený více než jedním rozpětím, pak je rozhraní jednotlivých rozpětí podepřeno pilíři. Nejjednodušší forma trámových mostů je například kláda nebo rovná kamenná deska položená přes potok, podepíraná na koncích jeho břehy [3]. Většina dnešních mostů jsou trámové mosty z předpjatého betonu s rozpětími o délce dvaceti až šedesáti metrů [12, strana 434].



Obrázek 2.10: Trámový most zvaný Confederation Bridge, součást Transkanadské dálnice; autor obrázku Dennis Jarvis⁶

Obloukový most (*angl. arch bridge*) je velmi různorodá kategorie mostů, protože jeho konstrukce může nabývat různých forem. Starší a všeobecně mohutnější je klenutá konstrukce, která byla používána již v době starověkého Říma [3] a která se používá dodnes. Příkladem takového mostu může být například Karlův most v Praze určený pro pěší provoz. Klenutou konstrukci používá i tzv. viadukt, což je typ mostu stavěný přes rozsáhlá údolí, a jedním z jeho zástupců může být například skotský most Glenfinnan Viaduct na obrázku 2.11a. Stavby využívající klenutou konstrukci bývají od pohledu mohutné a celistvé, obzvláště v porovnání s druhým typem konstrukce používané u obloukových mostů – oblouková konstrukce.



(a) Glenfinnan Viaduct, obloukový most s klenutou konstrukcí známý z filmu Harry Potter a Tajemná komnata (*angl. Harry Potter and the Chamber of Secrets*); autor obrázku Daniel Kraft⁷



(b) Sydney Harbour Bridge, představitel obloukových mostů s obloukovou konstrukcí; autor obrázku Navin Sigamany⁸

Obrázek 2.11: Fotografie ilustrující rozdíl mezi typy obloukových mostů dle konstrukce

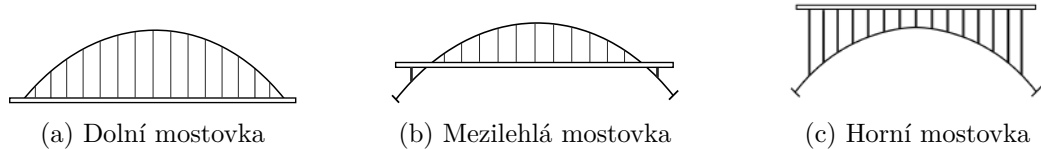
Modernějším druhem obloukových mostů jsou právě ty tvořené obloukovou konstrukcí, jakým je například most Sydney Harbour Bridge na obrázku 2.11b. Klenutá a oblouková konstrukce se kromě svého věku liší v tom, jakým způsobem je zátěž přenášena z komunikace do klenby nebo oblouku. V případě klenutých mostů je nad oblouky umístěna zemina

¹<https://www.flickr.com/photos/archer10/4937301611/>

²https://commons.wikimedia.org/wiki/File:Jacobite_on_Glenfinnan_Viaduct_1_20211024.jpg

³<https://www.flickr.com/photos/gimp/17706659740/>

(tzv. přesypávka), která u obloukových mostů není přítomná a místo přesypávky je součástí konstrukce mostovka, ze které je zátěž přenášena pomocí vzpěr nebo závěsů do oblouků po stranách mostu [10, strana 9]. U obloukových konstrukcí má velký efekt na design mostu umístění mostovky vůči oblouku (viz obrázek 2.12).



Obrázek 2.12: Dělení mostovek mostu s obloukovou konstrukcí podle jejich umístění vůči oblouku

Jak již naznačuje jejich název, mostovka zavěšených (*angl. cable-stayed bridge*) a visutých (*angl. suspension bridge*) mostů je zavěšena na lanech. Rozdíl mezi těmito dvěma typy mostů je v tom, jestli je opačný konec těchto lan upevněn v pevných bodech. V případě visutých mostů visí mezi mostními věžemi lano (typicky na každé straně mostu jedno hlavní lano), ke kterému jsou připojena lana držící mostovku (viz obrázek 2.13b mostu Golden Gate, ukázka visutého mostu). Zavěšený most žádné takové lano mezi věžemi nemá – lana držící mostovku jsou upevněna k těmto věžím přímo a jsou zarovnané vůči sobě paprskovitě, jak demonstruje fotografie mostu Stonecutters Bridge 2.13a. Primitivní provazové mosty se řadí mezi visuté mosty.



(a) Zavěšený most jménem Stonecutters Bridge ve městě Hong Kong; autor obrázku Edward Wong⁹ (b) Golden Gate Bridge, visutý most postavený ve městě San Francisco; autor obrázku Anand Dandekar¹⁰

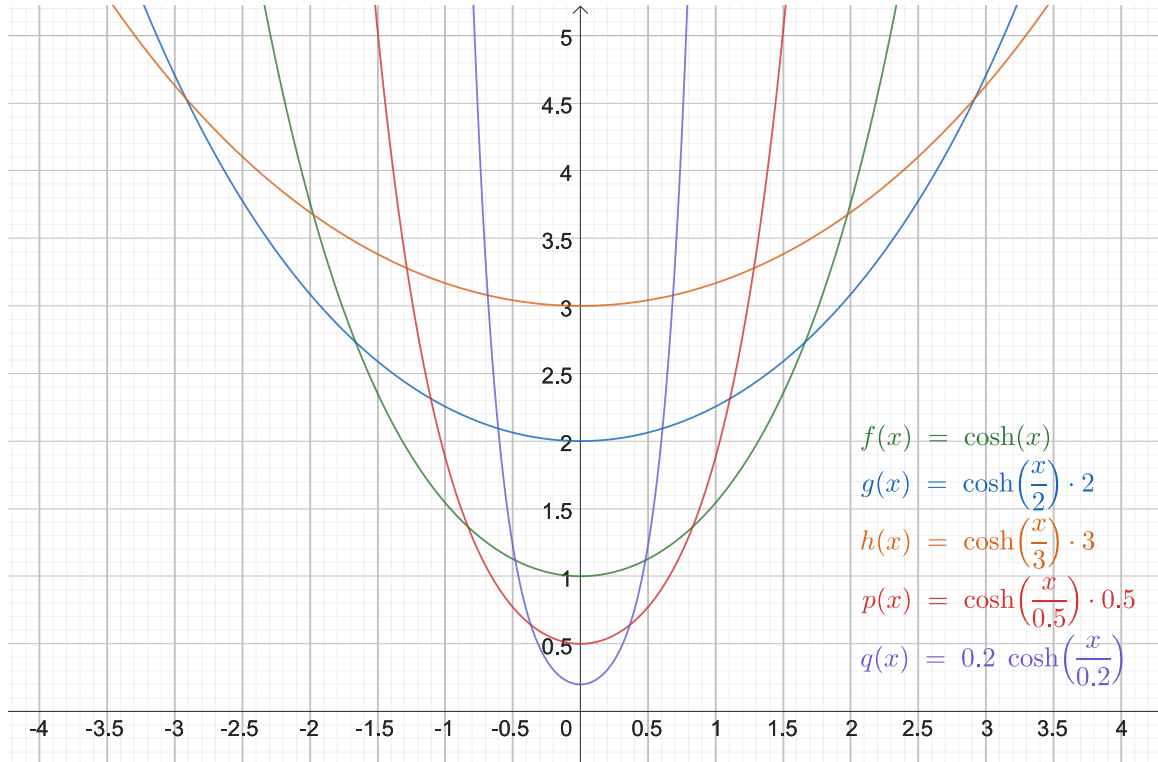
Obrázek 2.13: Fotografie ilustrující rozdíl mezi visutými a zavěšenými mosty

Jak je možné si z předešlých obrázků povšimnout, tak v případě zavěšených mostů jsou všechny kabely napnuté. To nicméně neplatí o visutých mostech. U nich jsou hlavní kabely (ty, které navzájem spojují věže a mají typicky větší poloměr než ostatní kabely) prohnuté pod vlivem gravitace a teprve z těchto hlavních kabelů jsou zavěšeny napnuté kabely držící

⁹<https://www.flickr.com/photos/edwardwong/4280136483>

¹⁰<https://www.pexels.com/photo/golden-gate-bridge-829237/>

mostovku. Prohnutí takto visícího kabelu lze popsat tzv. řetězovkou (*angl. catenary curve*, z latinského *catena*, v překladu řetěz), což je křivka funkce hyperbolického kosinu [13].



Obrázek 2.14: Ukázka křivek typu řetězovka; křivky definované funkcí hyperbolického kosinu

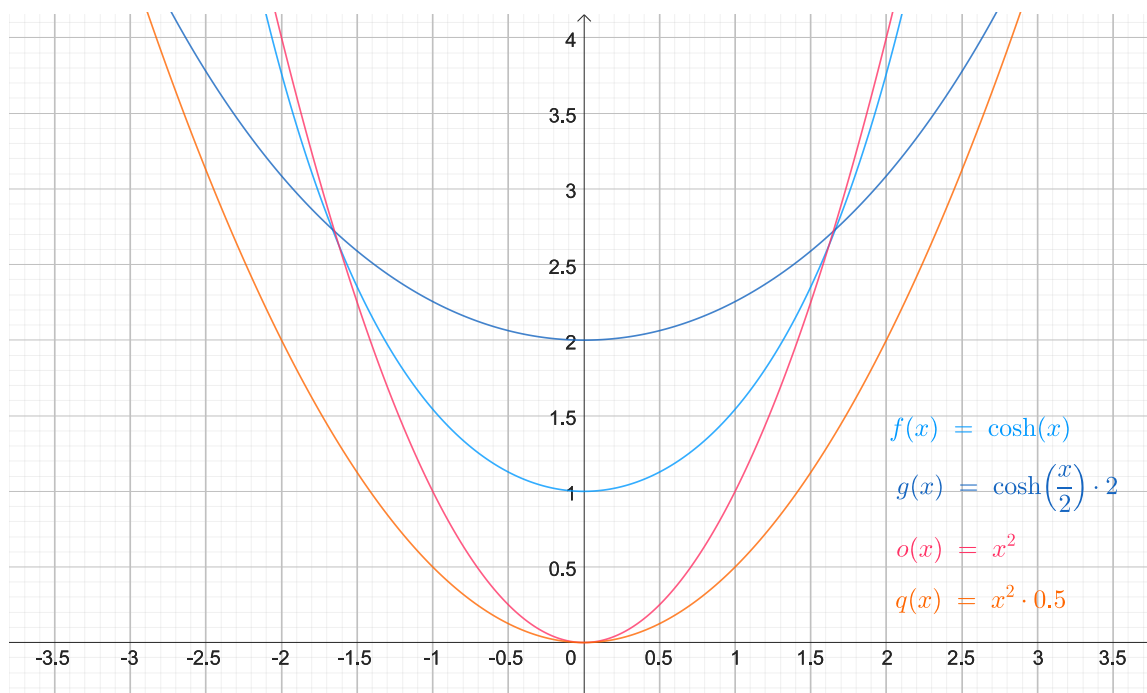
Řetězovka popisuje prohnutí řetězu, lana apod. pod svou vlastní vahou. Pro ukázkou zobrazuje graf 2.14 příklady pěti křivek tohoto typu. Následující je všeobecná rovnice, která řetězovku popisuje:

$$y = a \times \cosh \frac{x}{a} = \frac{a}{2} \times (e^{\frac{x}{a}} + e^{-\frac{x}{a}}), \text{ kde:}$$

\cosh hyperbolický kosinus
 a parametr křivky

Parametr křivky udává Y-ovou souřadnici jejího vrcholu, jak je možné si všimnout na grafu. Tmavě modrá řetězovka, jejíž parametr má hodnotu dva, je svým vrcholem umístěna v bodě, jehož souřadnice y má taktéž hodnotu dva. To stejné platí i pro ostatní křivky uvedené na grafu.

Řetězovka je nicméně vhodnější pro popis volně visících řetězů, elektrického vedení a dalších, na které není přivedena žádná další zátěž. Pokud na visící kabel působí uniformě rozložená síla v horizontálním směru, pak tvar kabelu připomíná parabolu spíše než řetězovku [7]. V případě visutého mostu tyto prohnuté kabely nevisí samy o sobě, nýbrž jsou zatíženy přibližně rovnoměrně vahou zbytku mostu, který mají za úlohu držet. Ve výsledku je tvar prohnutého kabelu visutého mostu kombinací řetězovky (kvůli vlastní váze kabelu) a paraboly (způsobeno vahou mostovky) [12].



Obrázek 2.15: Demonstrace podobnosti křivek typu řetězovka (modrá) a parabola (oranžová a červená)

Graf 2.15 slouží k porovnání tvaru paraboly a řetězovky. Lze na něm pozorovat, že řetězovka vypadá oproti parabole mírně uvolněněji, kdežto parabola působí, jako by byla prohnutá pod větší zátěží. Rozdíl je nicméně na první pohled zanedbatelný.

Kabely zavěšeného či visutého mostu musí být určitým způsobem v koncových bodech upevněny. Takovým koncovým bodem může být věž nebo mostovka po straně silnice. Takové upevnění (*angl. anchorage*) může vypadat různě. Jedním z příkladů může být obrázek pořízený na mostu s názvem Arthur Ravenel Jr. Bridge 2.16.

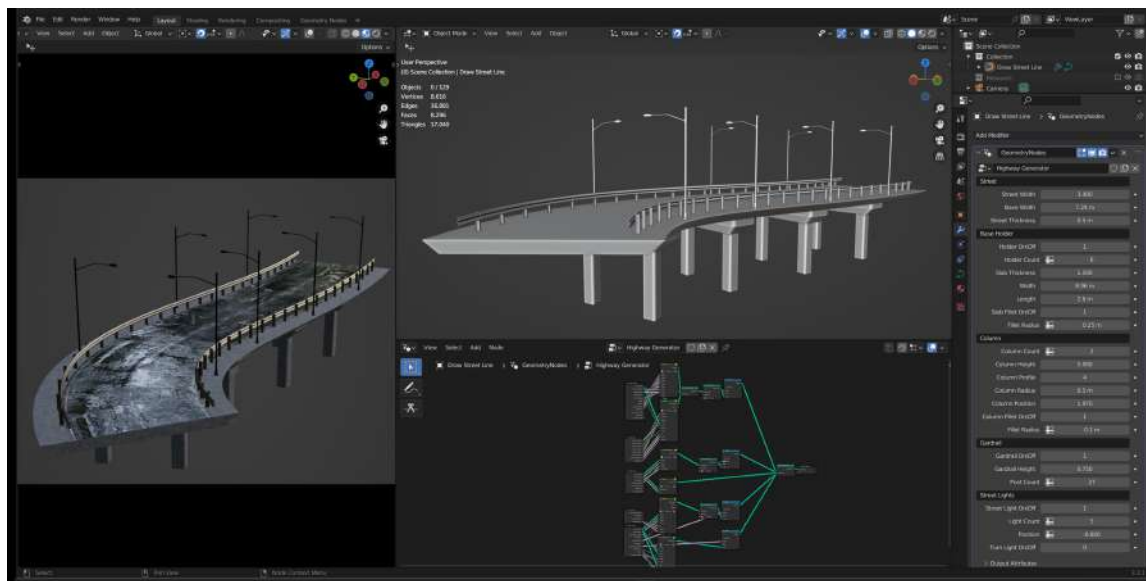


Obrázek 2.16: Kabel upevněný na mostovce mostu Arthur Ravenel Jr. Bridge¹¹

¹¹http://s3.ravenelbridge.net/large/jun_15_cable_support_1.jpg

2.4 Průzkum existujících řešení

Online lze najít malé množství doplňků dodávající funkcionalitu generování mostů. Prvním z nich je placený produkt z konce roku 2022 dostupný na Blender Market od uživatele Me Art Studio¹². Spíše než o kompletní add-on s vlastními panely se jedná o modifikátor typu Geometry Nodes, který tvoří procedurální generátor dálničních mostů (viz obrázek 2.17). Výsledkem generování je jednoduchý most stojící na pilířích, přičemž po stranách silnice jsou umístěna svodidla a pouliční lampy. Produkt má provedené UV rozbalení a uživatel má na výběr ze tří materiálů pro beton a kov.



Obrázek 2.17: Ukázka produktu Highway Bridge Generator z Blender Market

Další takovou prací je generátor od člověka pod přezdívkou Blenderesse (jménem Mietinen Jesse), který pravidelně publikuje své doplňky na svých stránkách¹³. V dubnu roku 2022 publikoval video demonstrující procedurální vytváření mostů za pomoci křivky. Generátor nemá žádný ovládací panel – nad křivkou jsou automaticky generovány obloukové mosty s obloukovou konstrukcí bez dalšího možného nastavení. Zajímavé na tomto generátoru je fakt, že generovaný most, přes nemožnost širšího nastavení, se přizpůsobuje povrchu terénu pod ním. Daná nahrávka je uvedena v krátkém článku píšící o tomto produktu¹⁴. V době psaní tohoto textu nebyl doplněk publikován a sám autor se vyjádřil v komentářích u svého demonstračního videa, že není jisté, jestli se dokončený produkt někdy objeví na jeho stránkách mezi jeho ostatními pracemi.

Oba tyto generátory implementují svou funkcionalitu využitím systému Geometry Nodes a umožňují vytváření mostů právě jednoho typu – první slouží ke generování dálničních mostů a druhý obloukové mosty s klenutou konstrukcí nebo místo toho trémové mosty (tyto typy mostů byly probrány v sekci 2.3.2).

⁹<https://www.blendermarket.com/products/procedural-highway-bridge-generator>

¹⁰<https://blenderesse.gumroad.com/>

¹¹<https://80.lv/articles/a-cool-bridge-generator-made-with-blender-s-geometry-nodes/>

Kapitola 3

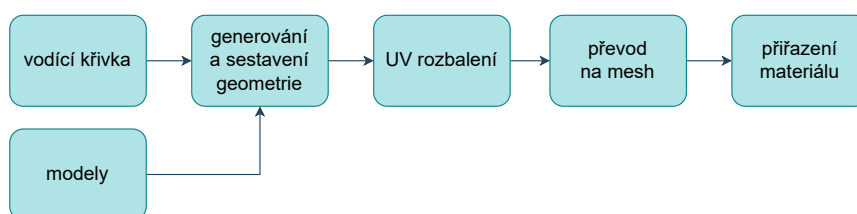
Návrh řešení

Tato kapitola se zabývá tím, jak má výsledný doplněk pro Blender verze 3.5 fungovat a jakou funkcionalitu by měl nabízet potenciálnímu uživateli, což je tematikou sekce 3.1. Ta uvede, jaké vlastnosti má návrh práce společně s již existujícími generátory mostů, co vylepšuje a co dělá jinak. Také se zaměří na to, jaké varianty mostů produkt umožňuje generovat. Způsob, kterým bude add-on komunikovat s uživatelem přes uživatelské rozhraní, je popsán v sekci 3.2 a nakonec plánované použití technologií a rozebrání programového návrhu je specifikováno v sekci 3.3.

3.1 Cílová funkcionalita

Za předpokladu, že má uživatel prostředí nebo scénu, do které by potřeboval vložit most, pak mu nemusí vyhovovat žádný již hotový statický model. Pro uživatele by bylo přínosné, kdyby mohl určit, kde most začíná, kde končí a nastavit jeho tvar, což již existující generátory řeší jednoduše: vodící křivkou. Zavedení řídicí křivky je flexibilní a pro uživatele snadno pochopitelný koncept. Uživatel nakreslí křivku a podél ní je vygenerován most. Tuto funkcionalitu je vhodné napodobit.

Samotné generování geometrie nicméně není dostačující funkcionalita. Proces vytváření modelu se skládá ze tří základních kroků: modelování, UV rozbalení modelu a následovné texturování. Add-on by měl plnit nebo přinejmenším ulehčit všechny tyto body, aby byl jeho výsledek pro uživatele přínosný a použitelný v praxi. Málokdy se stane, že by uživateli stačil neotexturovaný model postrádající UV mapu, a proto velké množství doplňků, které procedurálně generují 3D modely, dodávají své modely po předchozím UV rozbalení a případně i aplikaci textury. Schéma 3.1 zobrazuje jednotlivé kroky zamýšleného chování výsledného add-onu.



Obrázek 3.1: Diagram procesu tvoření modelu mostu: Nad křivkou je vygenerován most, je provedeno UV rozbalení a je umožněno přiřazovat materiál

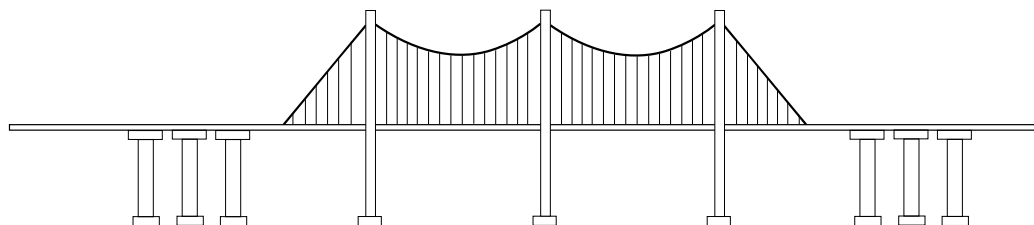
Na schématu je možné si povšimnout dvou zajímavostí. První z nich je, že mezi kroky UV rozbalení a přiřazováním materiálu je další krok – převedení na mesh. Ten je potřeba provést z toho důvodu, aby bylo možné vybírat strany hotového modelu mostu. To je zapotřebí, aby následně mohl uživatel upravovat polohu materiálu vůči UV mapě v panelu UV Editing. Druhou věcí, které je možné si na schématu všimnout, je fakt, že mimo vodící křivky jsou na vstupu generování přiváděny modely. Za tyto modely může uživatel dosadit své vlastní vymodelované objekty, což výrazně zvýší míru přizpůsobitelnosti mostu podle jeho požadavků. Ostatní generátory mostů pracují pouze s vlastními modely částí mostu a z toho důvodu je toto největší element, ve kterém se od nich bude výsledný add-on lišit.

Cílem této práce je umožnit generování mostů různých typů, aby výsledný model bylo možné zasadit do téměř jakékoliv scény. To znamená, že doplněk by měl umět generovat jak primitivnější druhy mostů (vhodné do historických prostředí) tak stavby založené na moderních technologiích a postupech. Podporované typy mostu jsou následující:

- trémové mosty
- klenuté obloukové mosty
- zavěšené mosty
- visuté mosty

První dva druhy mostů nabízí větší flexibilitu, co se týče využití jejich modelů ve virtuálním prostředí. Jsou to mosty, které se objevují napříč lidskou historií, a hodí se jak do moderního prostředí, tak například do doby středověku (s případnými úpravami na designu). Zato zavěšené a visuté mosty jsou běžnými představiteli moderních mostů. S typem mostu úzce souvisí to, jestli mostní deska má formu solidní kamenné nebo betonové masy, nebo zda se jedná o obyčejný most z dřevěných prken. To je další možnost nastavení, kterou by měl doplněk nabízet. U každé varianty mostu by mělo být možné nastavit její charakteristické parametry.

Další hledisko, kterým se produkt snaží odlišit od již existujících doplňků, je možnost rozdělení mostu na části. Most nemusí nutně využívat pouze jednoho typu konstrukce. Není neobvyklé narazit na most, jehož část stojí na sloupech a zbytek mostovky je zavěšen na lanech. Doplněk by měl tuto skutečnost reflektovat, a proto bude uživateli umožněno rozdělit most na několik sekcí, přičemž pro každou bude možné definovat typ její stavby a nastavovat její parametry pro širší přizpůsobení představám uživatele. Takové rozdělení je předvedeno na obrázku 3.2.



Obrázek 3.2: Ukázka mostu složeného z více typů konstrukcí: Střed mostu využívá visuté konstrukce, zatímco jeho konce stojí na pilířích, což je charakteristické pro trémové mosty.

3.2 Uživatelské rozhraní

Návrh uživatelského rozhraní musí reflektovat způsob, jakým add-on funguje a v jakých krocích svou činnost vykonává. Původní návrh nepočítal s tím, že by doplněk pracoval responzivně. Uživatel měl selektovat křivku, v panelu zadat parametry mostu a výsledek by viděl až po stisknutí tlačítka pro vygenerování mostu. Z toho důvodu, jak lze vidět na nákrese návrhu na obrázku 3.3, byly součástí panelu náhledy pro výběr modelů použitých pro vytvoření mostu, protože z jejich názvů samotných by nebylo jasné, jak vypadají.

The image shows a user interface for configuring a bridge. It consists of several panels and a button:

- Type:** A dropdown menu set to "Arch Bridge".
- Height and Width:** Two input fields with values 10 and 3, both in meters (m).
- Segments:** An input field with the value 10.
- Sidewalk:** A panel with a close button (X) and two input fields: Height (0.05 m) and Width (2 m).
- Railing:** A panel with a close button (X), a dropdown menu set to "Decorative Railing 1", a "Preview" area (a grey square), and two input fields: Height (1 m) and Width (0.1 m).
- Arches:** A panel with a close button (X), a dropdown menu set to "Arch 1", a "Preview" area (a grey square), and a "Generate Bridge" button at the bottom.

Obrázek 3.3: Původní nákreš uživatelského rozhraní

Návrh uživatelského rozhraní se postupem času vyvíjel zároveň s rozšiřováním funkcionality doplňku. Po umožnění responzivního chování nebylo nutné náhledy modelů ponechat, protože uživatel může zvolené modely pozorovat v reálném čase na vygenerovaném mostě, aktualizovaném při jakékoliv změně v uživatelském rozhraní.

V předcházející sekci kapitoly 3.1 bylo uvedeno, že základem fungování doplňku je řídicí křivka udávající tvar mostu, který je nad ní generován postupným nasazením sady modifikátorů Geometry Nodes. Lze nastavovat jejich parametry a upravovat tak charakteristiku mostu. V tento moment je ale generovaný most stále objekt typu křivka. Pro umožnění uživateli pracovat s vygenerovaným modelem v panelu UV Editing, musí být prvně provedeno převedení na objekt typu mesh, čímž dojde k aplikaci modifikátorů a tvar mostu nebude možné dále upravovat jako předtím. Uživatelské panely jsou proto postaveny na tom, v jaké fázi modelování mostu se uživatel nachází, a jaký je jeho aktuálně vybraný objekt. Panel sloužící pro definování tvaru mostu lze vidět na obrázku 3.4.

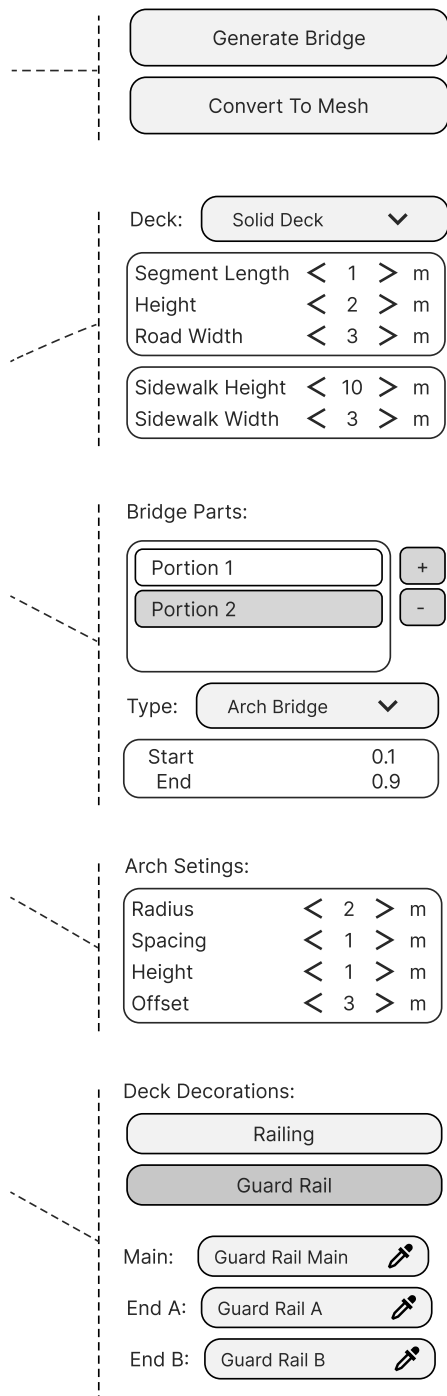
Ve výchozím stavu uvidí uživatel v hlavním panelu pouze tlačítko pro vygenerování mostu. Tlačítko je aktivní v případě, že vybraný objekt je typu křivka. Po jeho stisknutí je nad křivkou vygenerován most s výchozím nastavením a tlačítko zešedne (přestane být aktivní) a nabídka doplňku je rozšířena o tlačítko k převedení mostu na mesh, možnost vyhlazení hran a několik dalších panelů sloužící k přizpůsobení mostu.

První panel upravující geometrii mostu umožňuje vybírání, jestli most má mít pevnou mostovku, nebo jestli má být tvořen z prken. V případě solidní mostovky lze kromě základních parametrů nastavovat navíc parametry chodníku.

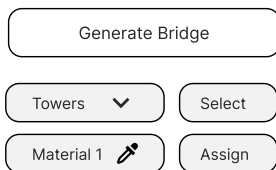
Tento panel slouží k dynamickému přidávání a odebrání částí mostu. Jinými slovy slouží k rozdělení mostu na více částí o jiné architektuře (obloukové, visuté atd.). Ke každé takové části mostu lze nastavovat její rozsah (kde začíná a končí) pomocí faktorů **Start** a **End**.

V tomto panelu lze detailněji nastavovat parametry aktuálně vybrané části mostu z panelu výše (na obrázku je touto vybranou částí **Portion 2** zvýrazněná šedě). Ku příkladu u obloukových mostů lze vybírat poloměr oblouku, vzdálenost mezi oblouky, offset apod. V případě zavěšených mostů lze vybírat modely věží a upevňovacích mechanismů, u trémových mostů měnit parametry pilířů atd.

Poslední panel slouží k přidání či odebrání zábradlí a svodidel pomocí dvou vrchních tlačítek. Pokud je tlačítko povoleno (v tomto případě je to tlačítko pro aplikaci svodidel), je pod ním rozbalena nabídka s parametry pro bližší nastavení. Svodidla lze přidat pouze na mostovku, na prkenné mosty je nelze aplikovat.



Obrázek 3.4: Nákres uživatelského rozhraní pro nastavení geometrie mostu



Obrázek 3.5: Nákres uživatelského rozhraní pro vybírání stran dle typu a přiřazení materiálu

Na obrázku 3.5 lze pozorovat konečnou fázi panelu. Po úspěšném převedení objektu na mesh je možné vybírat jednotlivé stěny mostu. Pro vybrání všech stran náležících stejné části mostu (věže, zábradlí, silnice) slouží první drop-down menu a tlačítko, které provede selekci. Následovně je možné použít vstupní pole pro výběr materiálu a tlačítkem **Assign** daný materiál přiřadit vybraným stranám. Tento panel zůstává viditelný i při přepnutí do pracovní plochy UV Editing a lze jej dále používat pro vybírání částí mostu a upravování jejich UV mapy vůči materiálu.

3.3 Programový návrh

Nejvhodnější způsob, kterým lze na cílové platformě Blender verze 3.5 zařídit procedurální generování geometrie, je nástroj Geometry Nodes, který je proto použit dále k implementaci. Při návrhu je potřeba posoudit, jak tento systém funguje, aby bylo možné provést určitou úroveň optimalizace.

V průběhu generování geometrie pomocí modifikátoru Geometry Nodes dochází k postupnému vyhodnocení celého stromu uzlů, který onu geometrii definuje. Při změně vstupního parametru modifikátoru dochází k opětovnému přepočtu všech větví stromu a to i v případě, že vstup nemá žádný vliv na definici geometrie (tj. je nevyužitý). Může docházet k tomu, že zcela zbytečně budou provedeny výpočetně náročné operace, které nezávisí na měněném parametru, a Blender může začít zamrzávat, pokud nestíhá provádět výpočty rychleji, než je daný parametr měněn. Z hlediska optimalizace je vhodné provést dekompozici na podproblémy a rozdělit generování do nezávislých funkčních bloků – rozdělit most na celky, kterou jsou na sobě nezávislé nebo závislé pouze jedním směrem.

Například vozovka není nijak závislá na svodidlech, ale naopak svodidla sedí na povrchu vozovky a řídí se jejím tvarem – existuje mezi nimi jednosměrná závislost. Při změně typu svodidel nemá smysl přepočítávat geometrii vozovky, tudíž může být pokládání svodidel separováno do modifikátoru následujícího po modifikátoru počítající tvar vozovky. Na tomto principu je doplněk rozdělen do několika modifikátorů typu Geometry Nodes.

Kapitola 4

Implementace

Tato kapitola má za účel nastínit, jakým způsobem byla problematika implementace hlouběji rozebrána na podproblémy, jaké matematické rovnice byly použity a jak byly vybrané algoritmy realizovány. Zároveň se bude kapitola zabývat tím, jaké problémy se při vývoji doplňku vyskytly a jak byly řešeny. Sekce 4.1 nabízí vhled do způsobu implementace základní funkcionality doplňku (generování mostů). Následovně sekce 4.2 popíše, jak je prováděno UV rozbalení procedurálně generovaných součástí mostu, přičemž sekce 4.3 v tomto naváže s popisem, jak byly předem připravené modely vymodelovány a jak bylo provedeno jejich UV rozbalení. Nakonec sekce 4.4 uvede a popíše problémy a řešení, které se vyskytly při implementaci uživatelského rozhraní. Implementační detaily jsou znázorněny na schématech v přílohách A. Hotový produkt je veřejně dostupný online¹.

4.1 Generování geometrie mostu

Hlavní část funkcionality add-onu je samotné generování mostu. Jak již bylo zmíněno v sekci programového návrhu 3.3, generování je prováděno pomocí sady po sobě jdoucích modifikátorů Geometry Nodes, přičemž každý se zaměřuje na jeden funkční celek mostu. Rozdělení funkcionality generování do více než jednoho modifikátoru vyvolává potřebu zajistit způsob, jakým si mohou mezi sebou jednotlivé celky předávat informace. Komunikace v hierarchii modifikátorů probíhá pomocí pojmenovaného atributu `Attachment Edge`.

0	Generické hrany mostu
1	Umístění svodidel
2	Upevnění kabelů
3	Umístění zábradlí

Tabulka 4.1: Tabulka vysvětluje význam jednotlivých hodnot atributu `Attachment Edge`: Hodnoty 1-3 označují hrany, na které mají být upevněny další části mostu a hodnota 0 představuje zbylé hrany.

¹<https://annsteelbridge.gumroad.com/1/CurveToBridge>

Modifikátor generující vozovku vytváří hrany pro umístění zábradlí, svodidel a kabelů. Atribut `Attachment Edge` slouží k odlišení hran objektu, aby například modifikátor nasažující svodidla umístil zábrany tam, kde mají být. Vysvětlení, co znamená která hodnota tohoto atributu, podává tabulka 4.1.

Každá základní část mostu se vyznačuje způsobem, jakým je generována. Velké množství částí mostu závisí na algoritmu ohybu. Jedná se o elementy mostu, které se musí řídit ohybem křivky zadané uživatelem. Zato kabely jsou v případě visutých mostů generovány metodou raycasting a výpočtem z rovnice paraboly. Kabely zavěšených mostů jsou rovné a směřují do stran, tudíž jsou implementovány vypočítáváním souřadnic konců kabelů na věži a mostu. Jednotlivé operace použité pro modelování a generování daných částí mostu jsou více rozebrány v podsekcích níže.

4.1.1 Operace ohýbání

Operace ohýbání, která je kromě implementování klenutých oblouků a mostovky také používána pro umístění zábradlí a svodidel, byla implementována zjednodušením algoritmu deformace podle křivky [4]. Porovnání efektu původního algoritmu pro ohyb a zjednodušeného algoritmu lze pozorovat na obrázcích 4.1. Na těchto ukázkách je možné pozorovat výsledky ohybu řady zcela stejných modelů sloupků. V obou případech dochází k různým deformacím jednotlivých objektů.



(a) Ukázka chování původního ohybu

(b) Ukázka chování zjednodušené verze ohybu

Obrázek 4.1: Srovnání metod ohybu na zástupu identických sloupků: Při metodě (a) se projevuje vliv normály křivky, zatímco metoda (b) zachovává vertikální hrany. Při každé metodě dochází k odlišným deformacím sloupků.

Obrázek nalevo 4.1a vyobrazuje ohyb, který bere v potaz normálu a tečnu křivky. Lze vidět, že jednotlivé objekty rovnoměrně umístěné na křivce se zdají být zarovnané s normálou v bodě křivky. V některých bodech dochází k deformaci sloupku tak, že velikost vrchní části sloupku neodpovídá velikosti podstavy. Sloupky v reálném světě typicky stojí rovně a nejsou nahnuté, k čemuž dochází kvůli zarovnání vůči normále. Z toho důvodu není pro potřeby této bakalářské práce tento efekt žádoucí a proto je používána metoda ohybu demonstrována na obrázku 4.1b, která zajišťuje, že sloupky jsou ve výsledku zarovnané rovnoběžně s osou Z. Ve skutečnosti při použití této metody dochází k zachování vertikál-

ních hran a způsobení efektu zkosení. Výpočet souřadnic bodů objektu ohýbaného tímto způsobem popisují následující rovnice:

$$\begin{aligned}x &= p_x + P_y \times n_x, \\y &= p_y + P_y \times n_y, \\z &= p_z + P_z\end{aligned}$$

kde: p_x, p_y, p_z souřadnice bodu křivky X, Y a Z
 n_x, n_y složky X a Y normály v bodu křivky
 P_y, P_z souřadnice Y a Z bodu ohýbaného objektu

Pro každý bod ohýbaného objektu je prováděn výpočet vůči určitému bodu křivky. Křivka je vzorkována od svého počátku v ose X. Pro daný bod objektu odpovídá bod křivky, který je v ose X ve stejné vzdálenosti od jejího počátku jako je vzdálenost daného bodu objektu od jeho počátku. Ohyb je následně prováděn v ose X a Y pomocí normály v bodě křivky. V ose Z je prováděn ohyb přičtením souřadnice Z k původní souřadnici Z. Klenuté oblouky jsou ohýbány modifikací tohoto algoritmu, jak je popsáno v sekci 4.1.5.

4.1.2 Sestavení mostovky (prken) podél vodící křivky

Všeobecně je mostovka či případně prkna mostu (dle zvoleného typu) tvořena určitým množstvím opakujících se segmentů. Ty se liší v tom, zda je generovaný most složen z prken či zda se jedná o solidní masu. V případě prkenného mostu je jedním segmentem právě jedno prkno. V opačném případě je segment složitější tvar tvořený střední částí pro silnici a postranními chodníky. Mezi takovými segmenty nejsou žádné rozestupy a ve výsledném objektu nebude na první pohled znát, že je z nějakých segmentů složen.

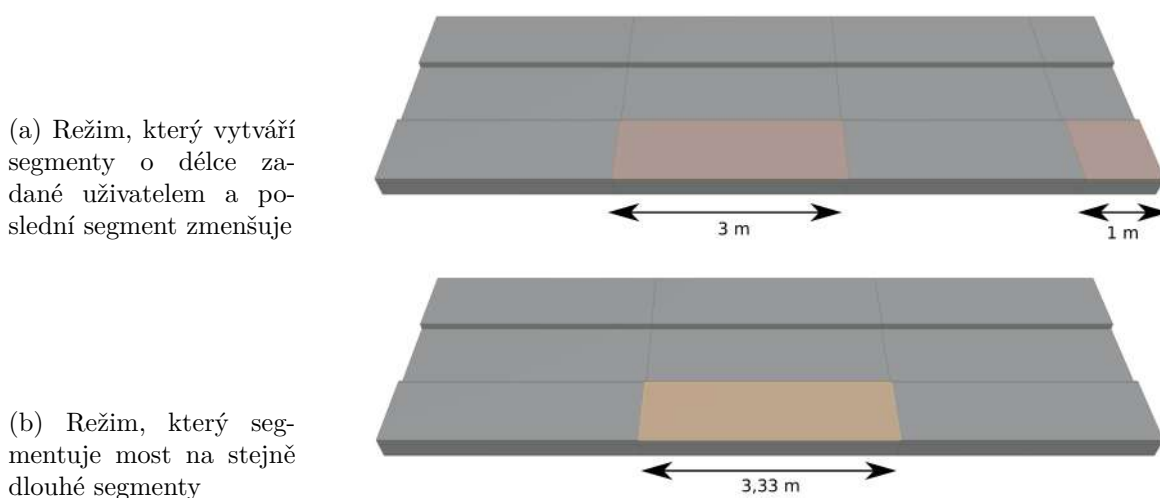


Obrázek 4.2: Modelování segmentu solidní mostovky: Šedá je původní kvádr, tyrkysová část vzniká první operací extrude, fialová opakováním této operace a tmavě modrá třetí operací.

Zatímco prkno tvoří jednoduchý tvar kvádrů, segment mostovky je potřeba dále modelovat, aby dosáhl požadovaného tvaru. Tento segment vychází z kvádřového tvaru, který tvoří plochu pro silnici bez postranních chodníků. Pro vygenerování chodníku jsou použity tři po sobě jdoucí operace extrude. Tento jednoduchý proces modelování vyobrazuje obrázek 4.2. První dvě operace extrude vygeneruje po straně silnice prostor na chodník (tyrkysová a fialová), který je ale ve stejné výšce jako silnice, která je označena šedě. První operace extrude následovaná druhou vygeneruje na chodníku hranu oddělující kraj chodníku od zbytku chodníku. V reálném světě je kraj chodníku často velmi odlišný od zbytku

chodníku a je pro něj vhodný jiný materiál, proto je chodník takto rozdělen. Nakonec zvýšení chodníku nad silnici je provedeno druhou operací extruze v pozitivním směru osy Z (označeno tmavě modře).

Totožné segmenty jsou následně poskládány do rovné linie za sebe. Jediný rozdíl mezi prkenným a solidním mostem je v tomto kroku takový, že v případě prkenného mostu jsou mezi jednotlivá prkna přidány rozestupy. Solidní segmenty jsou pokládány natěsno za sebe s tím, že překrývající se strany jsou spojeny a následovně odstraněny, aby nezůstaly žádné strany vevnitř geometrie mostu. Celá linie segmentů je v posledním kroku ohýbána podle řídicí křivky, u které je nejprve provedeno nastavení normály. Z důvodu garantování, aby silnice vždy čelila vzhůru a nemohla být převrácená nebo nějak nakloněná do strany, tak je použit uzel `Set Curve Normal` s parametrem `Z Up`.



Obrázek 4.3: Ukázková mostovka má délku deseti metrů a uživatelem je nastavena délka jednoho segmentu tři metry. Do jedné mostovky se vlezou tři segmenty a zbývá metr na poslední segment. Segmentování mostů má proto implementováno dva různé režimy (a) a (b).

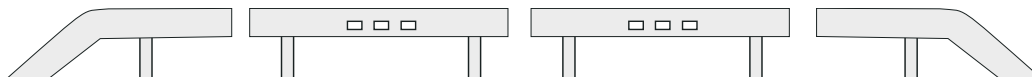
Počet segmentů mostovky (prken) mostu nepřímo ovlivňuje uživatel, který má možnost nastavit délku segmentu – čím větší délka, tím je potřeba menší množství segmentů na pokrytí celé délky mostu. V případě rovných mostů stačí nastavit velké délky segmentů, zatímco u zakřivených mostů se může přílišná délka segmentů projevit na tom, že záhyby mostu nebudou hladké a na rozmezí segmentů budou velké zlomy. Obrázek 4.3 demonstruje komplikaci, která se váže s tímto způsobem implementace, a dvě možná řešení. V prvním režimu je most rozdělen na segmenty o délce zadané uživatelem a přebývající délka mostu je tvořena zmenšeným segmentem. Všechny segmenty sdílí stejnou UV mapu a z toho důvodu se na posledním zmenšeném segmentu projeví deformace textury (smrštění). Druhý režim segmentování mostu nepoužívá pro délku segmentu hodnotu zadanou uživatelem, nýbrž rozděluje most na segmenty o stejné délce, která se pouze blíží vstupu uživatele. Toto zaručuje, že žádný segment mostu nebude mít deformovanou texturu, ale nehodí se v případě, že uživatel potřebuje přesné rozměry segmentů. Analogicky jsou tyto dva režimy segmentování implementovány i pro prkenné mosty.

Původně byla mostovka vytvářena takovým způsobem, že uživatelská křivka byla dána na vstup uzlu `Curve to Mesh`, načez byl vygenerován objekt s obdélníkovým profilem kopírující danou křivku – jednoduchá mostovka bez chodníku. Při tomto řešení vyvstaly

problémy při následovné implementaci UV rozbalení (pro podrobnější popis viz sekci 4.2, a proto muselo být ke generování vozovky přistoupeno jinak.

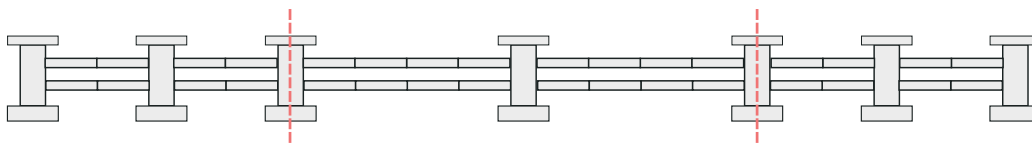
4.1.3 Položení zábradlí a svodidel

Princip pokládání zábradlí a svodidel na povrch mostu funguje v zásadě stejně jako tvoření mostovky. Opakující se segmenty zábradlí či svodidel jsou poskládány za sebe do přímé linie a následně je celý objekt složený ze segmentů ohnut podle křivky. Zásadní rozdíl v tomto procesu je v tom, čím jsou tvořeny jednotlivé segmenty ohýbaného objektu.



Obrázek 4.4: Sestavení modelů spočívá v poskládání několika opakujících se středních částí svodidel a připojení koncových částí po stranách.

Sestavování svodidel je poměrně jednoduchý proces, který naznačuje obrázek 4.4. Svodidla jsou tvořena ze tří objektů: střední sekce a dva volitelné koncové modely pro levý a pravý konec. Podle délky křivky, na které jsou svodidla umístována, je vypočítáno, kolik středních částí svodidel se na most vleze (s přihlédnutím k tomu, kolik prostoru vyžadují modely konců svodidel). Toto množství středních částí je umístěno vedle sebe do řady a po stranách jsou připevněny koncové části svodidel, čímž je celý objekt svodidel připraven.



Obrázek 4.5: Sestavení hlavních částí zábradlí a sloupků spočívá ve spojení hlavní části, která je na obrázku vyhrazena červeně, s postranními částmi. Hlavní i postranní části jsou sestaveny z dílčích modelů sloupků a segmentů mezi sloupky.

Zábradlí je možno sestavit ve dvou různých režimech. První z nich funguje stejně jako sestavení svodidel. Druhý režim připravení objektu zábradlí je složitější proces a pro jeho pochopení je nápomocný obrázek 4.5. Zábradlí je skládáno ze dvou modelů: hlavní část a volitelné sloupky. Toto rozdělení umožňuje nastavit poměr sloupků vůči hlavním částem zábradlí. Zábradlí je podobně jako svodidla tvořeno ze tří částí: střední část a koncové části, které jsou vymezeny na obrázku červenými čarami. Pro koncové části lze nastavovat jiný poměr sloupků vůči hlavním částem (typicky menší poměr než pro střední část, aby koncové části mohly zabírat menší prostor a byla pak celým zábradlím co nejeefektivněji využita celá délka křivky, na kterou je nasazováno). Rozdílné nastavení poměrů hlavních částí vůči sloupkům lze pozorovat na obrázku. Ve střední části se mezi každými dvěma sloupy opakují čtyři hlavní části. V případě koncových jsou mezi dvěma sloupky pouze dvě hlavní části. Takto je zábradlí sestaveno do jedné řady, která je poté ohnuta podle křivky, na které má být zábradlí položeno.

Zábradlí i svodidla jsou umístována na stanovené hrany po povrchu mostu. Zábradlí je umístováno na hrany, jejichž atribut `Attachment Edge` má hodnotu 3 a v případě svodidel se jedná o hodnotu 1 (pro vysvětlení viz 4.1). Umístování levého a pravého zábradlí probíhá separátně, což platí i pro svodidla.

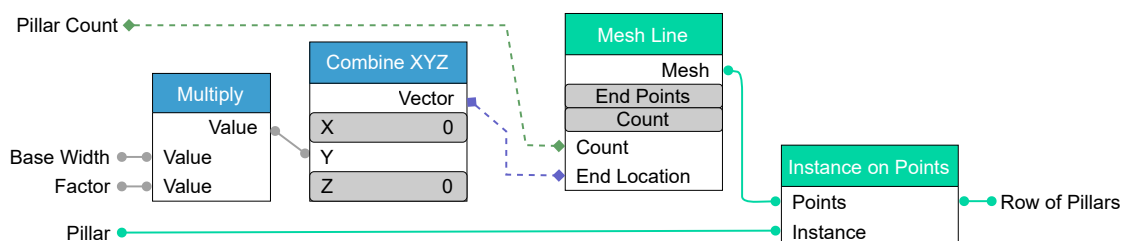
4.1.4 Sestavování a pozicování pilířů

Pilíře jsou základní část trémových mostů generována pravidelně po celé délce uživatelem vyhrazené části mostu. V kontextu funkcionality doplňku není pilíř jeden statický model, nýbrž kombinace tří separátních modelů: základny (*angl. base*), hlavní části pilíře a svršku (*angl. cap*). Toto rozdělení umožňuje sestavovat komplexnější pilíře složené z více středních částí jako ilustruje obrázek 4.6.



Obrázek 4.6: Demonstrace sestavení pilířů ze separátních modelů základny, střední části a svršku

Toto rozmístění středních částí pilířů určuje uživatel pomocí těchto vstupů: počtu středních částí pilířů a faktoru. Pilíře jsou rozmísťovány po povrchu základny a proměnná faktor určuje, kolik procent šířky základny má být použito pro rozmístění. Schéma 4.7 vyobrazuje rozložení středních částí pilířů po pomocné úsečce, jejíž délka je vyhrazena šířkou základny omezenou zadaným faktorem. Nakonec je řada hlavních částí pilíře propojena se základnou a svrškem, čímž je model pilíře sestaven.

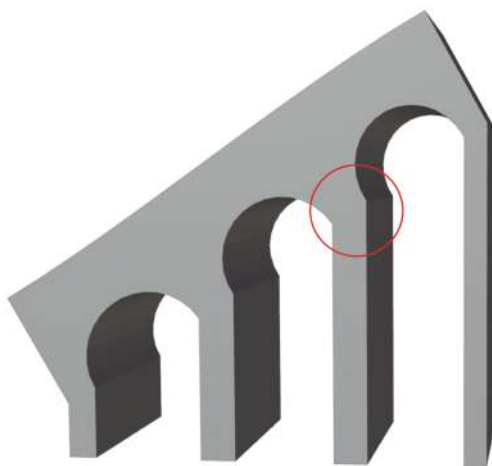


Obrázek 4.7: Zjednodušené schéma sestavení řady středních částí pilířů

Nakonec určení pozic pilířů po délce mostu je provedeno pomocí uzlu **Instance On Points** a následným ohybem. Rovná křivka o délce vodící křivky je převedena na body ve vzdálenostech určených uživatelem. Řada sloupů je nakonec ohnuta podle řídicí křivky.

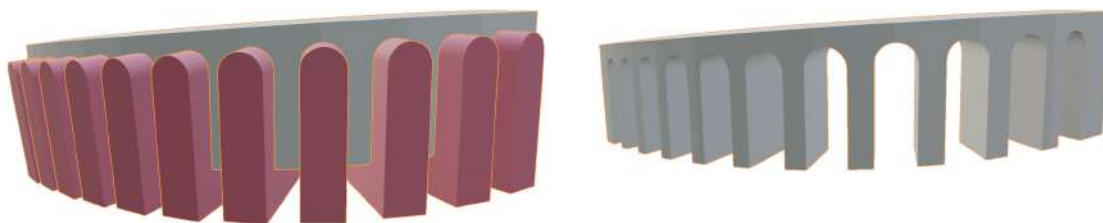
4.1.5 Umisťování klenutých oblouků

Klenuté oblouky byly implementovány ve třech různých verzích. První verze fungovala jednoduše: Program si vygeneroval model oblouku dle zadaných parametrů uživatele (poloměr a prostor mezi oblouky) a následně připravil řadu těchto modelů. V dalším kroku byla celá řada oblouků ohýbána dle uživatelské vodící křivky. Nakonec byly vybrány spodní strany oblouků a byly nataženy ve směru $(0, 0, -1)$ do země. Tento způsob implementace se ukázal být problematický. V případě mostů, které měly po celé délce nulovou změnu výšky, se tento problém neprojevil, ale u mostů, které postupně nabíraly nebo ztrácely výšku, byl nedostatek patrný. Oblouky nebyly zarovnané s osou Z. Nejzávažnější problém ale byl takový, že docházelo k tvorbě „zubů“. V místech, kde byly vytáhnuty spodní strany oblouků rovnoměrně s osou z, vznikaly ostré hrany, jak je vyznačeno na obrázku 4.8 barevně. To se dělo z toho důvodu, že oblouky nebyly zarovnané s osou Z.



Obrázek 4.8: Ukázka ohybu oblouků nevhodnou metodou

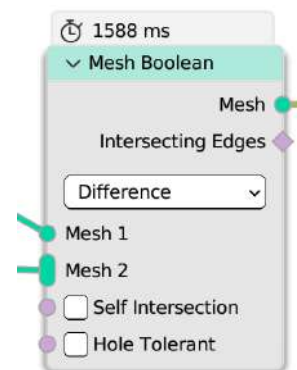
Druhé řešení přistupovalo k problému opačně. Místo vytváření oblouků a jejich následného pozicování bylo přistoupeno k metodě vyřezávání obloukových děr ve zdi mostu. K tomu bylo použito uzlu `Mesh Boolean`, který umožňuje použití booleovských operací nad geometrickými objekty. V tomto případě byla použita operace rozdíl. Do řady byla připravena sada objektů, které v průřezu měly tvar oblouku. Ty byly ohnuty podle křivky, aby protínaly zdi mostu, a následně byly odečty (byl proveden booleovský rozdíl) od geometrie mostu.



(a) Obloukové objekty nasázeny po délce mostu (b) Vyříznuté obloukové díry ve zdi mostu

Obrázek 4.9: Proces vyřezávání oblouků do stěn mostu

Tato varianta implementace byla funkční, ale vyskytl se další problém spočívající v použití booleovské operace rozdíl – jedná se o výpočetně náročnou operaci. Obrázek 4.10 ukazuje její dobu výpočtu pro obloukový most složený z většího množství oblouků (zhruba 25 oblouků), což není nijak extrémně dlouhý most a přitom jenom tento výpočet trvá více než jednu a půl sekundy. Výsledný produkt se má chovat responzivně, z toho důvodu je toto nepřijatelně dlouhá doba odezvy. Při jakékoliv operaci s vodící křivkou mostu nebo upravením jakéhokoliv parametru oblouků by došlo k zamrznutí aplikace a proto toto řešení nemohlo být použito.



Obrázek 4.10: Obrázek s časem výpočtu booleovské operace rozdíl při vyřezávání oblouků

Poslední verze generování oblouků vznikla vylepšením první verze používající ohyb. Namísto ohýbání celých oblouků v jednom běhu, čímž docházelo k jejich deformaci, byl proces ohybu rozdělen do několika částí. Svrchní strany oblouků jsou ohnuty jednoduše podle vodící křivky. Zbytek oblouků je ohýbán v druhém kroku, který se liší v určení souřadnice Z při výpočtu pozice každého bodu. Výpočet souřadnic X a Y zůstává stejný. Každý bod má uložený atribut, který vyjadřuje jeho vzdálenost od horní stěny oblouku. Pro každý oblouk je vypočítána pozice středobodu horní stěny a vůči ní jsou pozicovány zbylé body v ose Z. Následující jsou rovnice pro výpočet jednotlivých souřadnic bodů oblouku:

$$\begin{aligned}x &= p_x + P_y \times n_x, \\y &= p_y + P_y \times n_y, \\z &= p_{s_z} - (distance + 1)\end{aligned}$$

kde: p_x, p_y, p_z souřadnice bodu křivky X, Y a Z
 n_x, n_y složky X a Y normály v bodu křivky
 P_y souřadnice Y a Z bodu oblouku
 p_{s_z} souřadnice Z středobodu oblouku

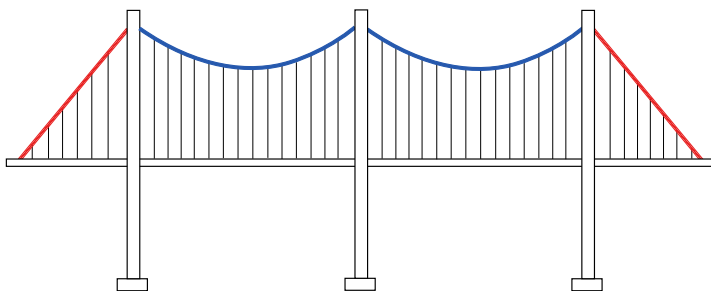
Od souřadnice Z tohoto bodu je odečtena hodnota vzdálenosti počítaného bodu, aby byl po ohybu umístěn ve stejné vzdálenosti od horní stěny jako v původní geometrii a nedošlo k deformaci modelu, jak ukazuje obrázek 4.11.



Obrázek 4.11: Ukázka úspěšného ohybu oblouků

4.1.6 Upevnění kabelů visutých mostů

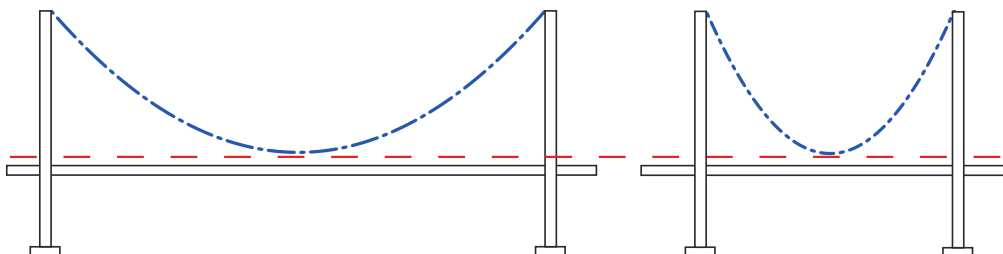
Kabely visutých mostů jsou podobně jako v případě zavěšených mostů upevněny ve vrcholech skupiny vertexů **Attachment Points** na modelu věže. Kromě způsobu jejich zpracování je rozdíl v jejich počtu. V případě zavěšených mostů je možné definovat libovolné množství upevňovacích bodů, zatímco visuté mosty podporují pouze čtyři (každý na jedné straně). Je to dáno konstrukcí z reálného světa – na věžích visutých mostů se upínají právě čtyři hlavní kabely na každé straně a teprve z těchto kabelů visí další typ tentokrát svislých kabelů, na kterých je zavěšena mostovka. Pro potřeby generování jsou hlavní kabely rozděleny do dvou skupin, které jsou počítány a vytvářeny zvlášť, jak ukazuje obrázek 4.12 – kabely mezi věžemi (modře) a postranních kabelů (červeně).



Obrázek 4.12: Odlišení kabelů mezi věžemi (modrá) a postranních kabelů (červená)

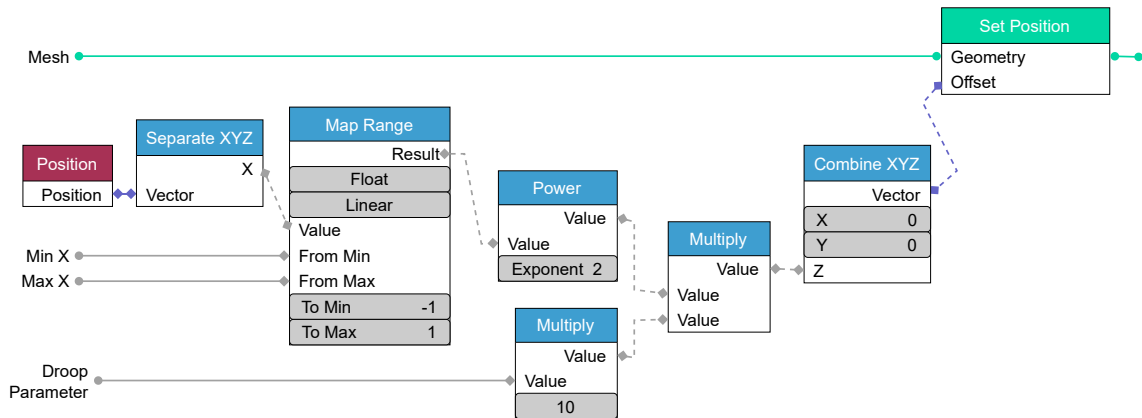
Generování kabelů mezi věžemi probíhá následovně: V každém páru věží jsou na levou věž vygenerovány dvě křivky (jedna na pravý upevňovací bod a druhá na levý). V dalším kroku je druhý konec těchto vygenerovaných křivek natažen na druhou věž. Jsou známy indexy protějších bodů na druhé věži, a proto je toto realizováno jednoduše pomocí uzlu **Sample Index**, který zjistí pozici těchto bodů.

Následovně je provedeno vypočtení prohnutí kabelů pod vlivem gravitace. Pro popis prohnutí kabelu byla vybrána rovnice paraboly. Křivka je převedena na body a pro účely výpočtu prohnutí v ose Z dojde k normalizaci bodů v ose X na interval $\langle -1; 1 \rangle$. Tato normalizace je provedena z toho důvodu, aby neohledně na měnící se vzdálenosti věží zůstávala úroveň poklesu kabelu stejná – aby paraboly tvořené kabely věží s odlišnými vzdálenostmi byly zarovnané jako na obrázku 4.13. Bez ohledu na vzdálenost věží, parabola má při stejných parametrech vrchol ve stejné výšce. To je žádoucí, protože takto může uživatel nastavovat počet věží mostu (čímž dochází k rozdílným rozestupům mezi věžemi) a nebude docházet ke změnám prohnutí kabelů. Nemůže tak docházet k tomu, že při změně počtu věží došlo k neočekávanému prohnutí kabelu pod mostovku.



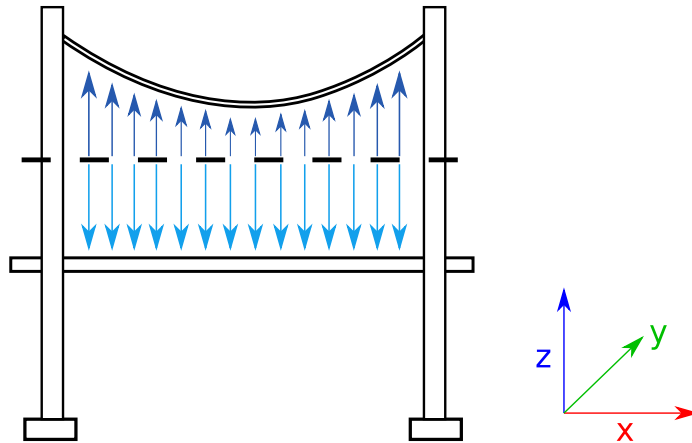
Obrázek 4.13: Demonstrace zarovnání parabol prohnutých kabelů neohledně na vzdálenosti věží

Souřadnice X a Y, bodů zůstanou nezměněné a bude měněna pouze souřadnice Z, jejíž posun je počítán pomocí normalizované souřadnice X. Normalizované souřadnice X koncových bodů (bodů upevněných ve věžích) jsou -1 a 1 a střed má normalizovanou souřadnici 0. Z těchto normalizovaných hodnot je vypočtena druhá mocnina, která je poté násobena parametrem udávajícím úroveň prohnutí kabelů (uživatelsky ovlivnitelný parametr). Pro každý bod je tímto způsobem vypočten offset v ose Z, který způsobí prohnutí. Po této operaci je křivka posunuta vůči upevňovacím bodům, proto je dále proveden posun všech bodů o konstantní vzdálenost tak, aby koncové body seděly na svých upevňovacích bodech. Tento proces ukazuje schéma 4.14.



Obrázek 4.14: Schéma algoritmu prohnutí kabelů pod vlivem gravitace

Koncové hlavní kabely jsou modelovány pomocí metody raycast. Z každého koncového upevňovacího bodu věže je vyslán paprsek pod ostrým úhlem ve směru mostovky. V bodě, ve kterém paprsek zasáhne mostovku, je zjištěna jeho poloha. Jeden konec každého kabelu je umístěn v upevňovacím bodě věže a druhý v místě střetu vyslaného paprsku s mostovkou.



Obrázek 4.15: Nákres principu použití metody raycast pro vygenerování svislých kabelů: Paprsky jsou vysílány z bodů mezi hlavními kabely a mostovkou ve směrech $(0,0,1)$ a $(0,0,-1)$.

Nakonec je potřeba vygenerovat kabely upínající se na mostovku, jak ukazuje obrázek 4.15. Všechny tyto kabely jsou svislé (směřují směrem do země) a tedy je pro jejich generování vhodné znovu použít metodu raycast. Všechny křivky hlavních kabelů jsou škálovány

v ose Z na hodnotu 0 a následně jsou převedeny na body ve vzdálenostech určených uživatelem. Následně jsou z těchto bodů paprsky směrem (0,0,1) do hlavních kabelů a ve směru (0,0,-1) do mostovky. Jeden konec kabelů získá polohu snímanou ve střetu paprsku s hlavním kabelem a druhý konec polohu v průsečíku druhého paprsku s mostovkou. Aby tato metoda fungovala, jsou tyto body umístěny mezi hlavními kabely a mostovkou. Pokud je křivka uživatele ohnuta vertikálně, pak se může stát, že část hlavních kabelů je výškově pod úrovní části mostovky, proto je potřeba, aby most byl po celé délce rovný. Nakonec vytvoření kabelů ze všech křivek je prováděno pomocí uzlu `Curve to Mesh`, který dostává na vstup jako profil kruh o poloměru zadaném uživatelem v panelu. Nakonec pouze zbývá provést UV rozbalení a generování kabelů je hotové.

Z podstaty konstrukce reálných visutých mostů platí, že musí být rovné. To je dáno tím, že hlavní kabely směřují od jedné věže přímo k druhé a po cestě se prohýbají pod vlivem gravitace. Hlavní kabely musí vést nad mostovkou, aby se svislé kabely z nich visící měly na co napojit, a jelikož musí být tyto kabely přímé, pak i samotná mostovka musí být od věže k věži rovná a nemůže se kroutit do stran. Doplněk umožňuje úspěšné vygenerování kabelů tak dlouho, dokud pod kabely se nachází mostovka a to nehledě na možné zakroucení. Místa uchycení nicméně v takovém případě nebudou pravidelná podél kraje mostu, tudíž je pořád nevhodné most zbytečně ohýbat do stran.

4.1.7 Upevnění kabelů zavěšených mostů

Generování konstrukce zavěšených mostů spočívá v rozmístění věží po délce mostu a vygenerování kabelů paprskovitě mezi věžemi a mostovkou. Hlavním kritériem je to, že je potřeba garantovat rovnoměrné rozložení kabelů jak v rámci věží, tak v rámci mostovky a to i v případě extrémně zakroucených mostů do stran a do výšky. Zároveň musí platit to, že na každou věž se musí vázat stejné množství kabelů. Je možné postupovat dvěma směry: rozmístit kabely po povrchu mostu a druhé konce připevnit na nejbližší věž nebo naopak nasadit tyto kabely na věže a jejich konce propojit s mostem.

Postup, který se osvědčil nejvíce, začíná modelováním s upevňováním kabelů na modelu věže, což vyvolává potřebu rozhodnout, v jakých místech se mají kabely na věž upínat a kolik jich má být. Tyto informace se vážou na samotný model věže a díky tomu lze uživatelem přesně definovat, která místa na věži mají sloužit k upevnění kabelů. Za tímto účelem je využita funkcionální `Vertex Group`. Pro identifikaci těchto míst je v nich potřeba vytvořit vertexy a ty zařadit do skupiny vrcholů pod označením `Attachment Points`. Vrcholy této skupiny by měly splňovat následující vlastnosti:

1. symetricky rozložené podle středu věže
2. výškově ve stejných vzdálenostech

Jsou celkem čtyři strany věže (vpravo a vlevo, vzadu a vepředu), na kterých musí být tyto upevňovací body rozmístěny stejně. Vzájemně musí být tyto body v rámci jedné strany rozmístěny rovnoměrně v ose Z – ve stejných vzdálenostech. V opačném případě nemusí být upevňování kabelů provedeno tak, jak je očekáváno. To je způsobeno procesem, který rozřazuje všechny vertexy (upevňovací body) do skupin podle stejné výšky. Nejnížší skupina bodů má přiděleno $id_h 0$, vrstva nad ní má id_h o jedno vyšší atd.

Toto rozřazení skupin vrcholů podle výšky zaručuje uzel `Map Range` v stupňovitě lineárním módu (*angl. stepped linear*). Tento uzel mapuje výšky vrcholů na atribut id_h . Hranice, ze kterých je prováděno mapování, jsou nejnížší a nejvyšší souřadnice Z všech bodů. Nejnížší hranicí, na kterou probíhá mapování je 0, protože identifikátory id_h jsou počítány

od nuly. Nejvyšší hranicí je celkový počet upevňovacích bodů na straně věže snížen o jedničku. Uzel `Map Range` nemusí provádět přepočítání výšky na id_h správně v případě, že jsou mezi body různé výškové vzdálenosti a proto je doporučeno zajistit rovnoměrné rozmístění bodů v ose Z. V praxi je rovnoměrné rozmístění těchto bodů typické, a proto toto není žádné zásadní omezení.

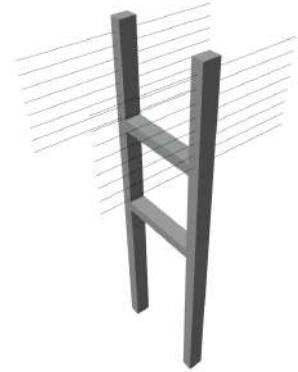
V prvním kroku umístování kabelů jsou na upevňovací body věže instanciovány křivky rovnoběžné s osou Y, jak je ukázáno na obrázku 4.16a. Následně jsou druhé konce kabelů, zarovnané do stejné výšky a v dalším kroku rozmístěny v ose Y (obrázek 4.16b). Toto rozmístění je realizováno s pomocí id_h , které byly určeny podle výšek bodů. Rozmístění konců kabelů dále ovlivňují dva uživatelské vstupy: vzdálenost sousedních kabelů a offset kabelů od věže samotné (při nulovém offsetu by kabel nejbližší věži začínal ve středu strany věže). Výpočet souřadnice Y konců křivek je následující:

$$y = (id_h \times spacing + offset) \times sign(n_x), \text{ kde:}$$

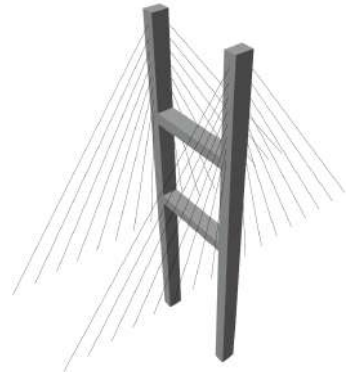
y	souřadnice Y konce kabelu
id_h	identifikátor výškové skupiny
$offset$	offset kabelů od věže
$spacing$	vzdálenost mezi sousedními kabely
n_x	X-ová složka normály v bodě na věži

Každý konec kabelu je od středu věže posunut o $offset$ a je dále posunut o parametr $spacing$ tolikrát, kolik je jeho hodnota id_h . Celá hodnota posunu bodu od věže je násobena kladnou či zápornou hodnotou jedna (výstupem funkce $sign(n_x)$) – to zajišťuje, že kabely budou posunuty podle normály stěny, ze které vychází. Kabely vycházející z přední stěny budou posunuty dopředu a ty vycházející ze zadní se posunou směrem dozadu. Souřadnice X konců kabelů zůstává stejná a souřadnice Z je nastavena u všech na nulu, což způsobí jejich zarovnání. V tento moment je model věže připravený na umístění po délce mostu.

Rozmístování věží probíhá stejným způsobem jako umístování pilířů – pomocí `Instance on Points`. V této fázi vypadá most, jako je vidět na obrázku 4.17a. Zbývá už jenom kabely posunout v ose X a Z, aby seděly na mostovce, aniž by došlo k porušení rozestupů. Nejprve je však potřeba určit, na kterou část mostovky se mají kabely upínat – tento prostor je vyhrazen hranami označenými atributem `Attachment Edge` s hodnotou 2 (pro vysvětlení všech hodnot viz tabulka 4.1). Proces připevňování pravých a levých kabelů probíhá zvlášť, aby se zabránilo tomu, že by se pravý kabel připevnil k levé straně mostu a naopak. Je tedy potřeba připevnit pravé kabely k pravé upevňovací hraně a levé kabely k té levé.

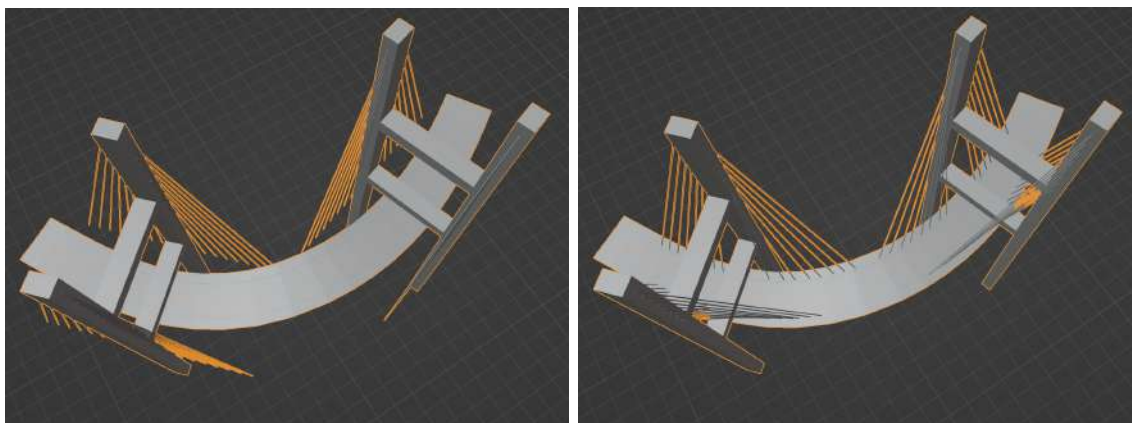


(a) Připojení kabelů v upevňovacích bodech věže



(b) Rozmístění konců kabelů v ose Y

Obrázek 4.16: Připojení a připravení kabelů po stranách modelu věže

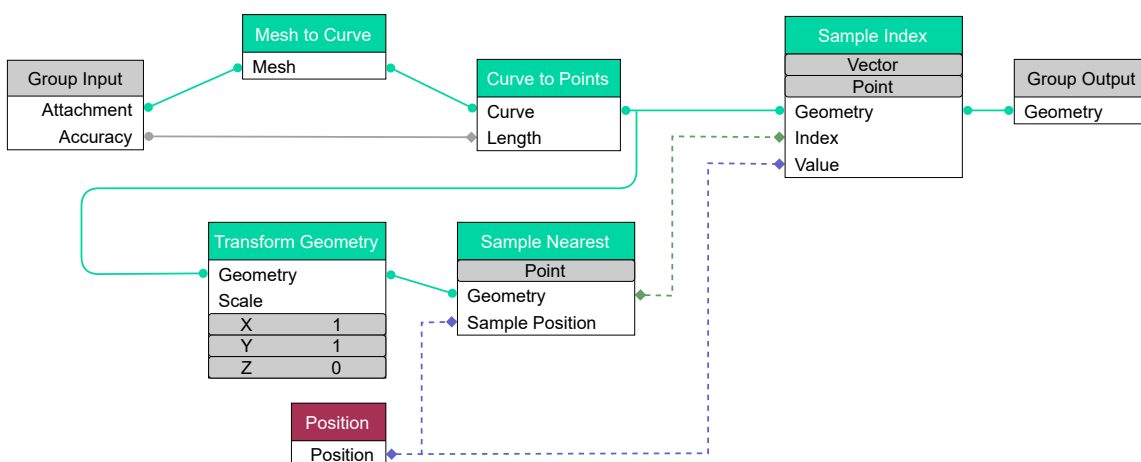


(a) Stav po rozmístění věží

(b) Stav po posunu konců kabelů v ose X a Z

Obrázek 4.17: Proces umísťování věží po délce mostu a upevnění kabelů

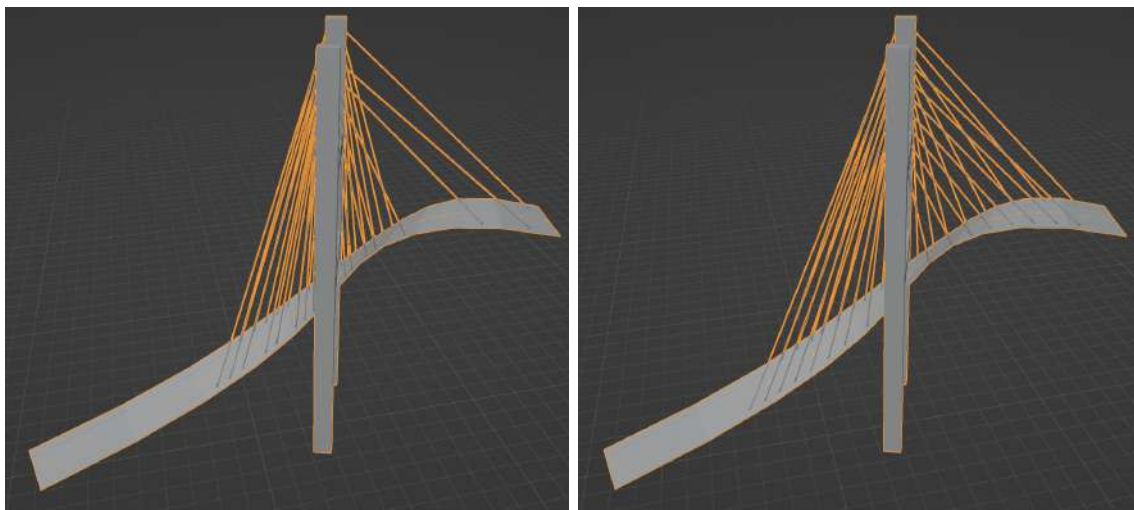
Hlediskem, podle kterého probíhá umísťování kabelů na jejich náležité hrany, je vzdálenost – každý kabel je upevňován v nejbližším bodu hrany. K tomu je použit uzel **Sample Index** v kombinaci s uzlem **Sample Nearest**. Uzel **Sample Nearest** nalézá index nejbližšího bodu a předává ho uzlu **Sample Index**, který získá pozici bodu na daném indexu. Tento algoritmus popisuje schéma 4.18.



Obrázek 4.18: Schéma algoritmu vzorkování polohy nejbližšího bodu na straně mostovky

Jak si lze na schématu výše povšimnout, jeden ze vstupů nese název **Accuracy** – přesnost. Při hledání nejbližšího bodu je nejprve potřeba získat body, ve kterých bude hledán ten nejbližší. Při převodu křivky na body je potřeba určit, v jaké mají být tyto body vzdálenosti. Tato vzdálenost udává přesnost vypočtení místa k upevnění kabelu – čím nižší vzdálenost, tím větší přesnost, ale zároveň to znamená vyšší výpočetní náročnost. Druhá věc, které je možné si na schématu všimnout, je, že body, na které mají být kabely upevněny, jsou škálovány v ose **z** a dochází k jejich výškovému zarovnání. To je z důvodu, aby zakřivení mostu do výšky nemělo vliv na rozmístění kabelů. Obrázky 4.19 demonstrují, jak se program chová v situaci, kdy výškové zarovnání je a není provedeno. Levý obrázek ukazuje situaci, ve které je zachován vliv převýšení na umísťování kabelů. Oproti tomu obrázek

napravo vykazuje daleko lepší výsledek, kdy došlo k zachování pravidelného rozprostření kabelů.



(a) Upevnění kabelů bez předešlého provedení transformace bodů

(b) Upevnění kabelů po předešlém provedení transformace bodů

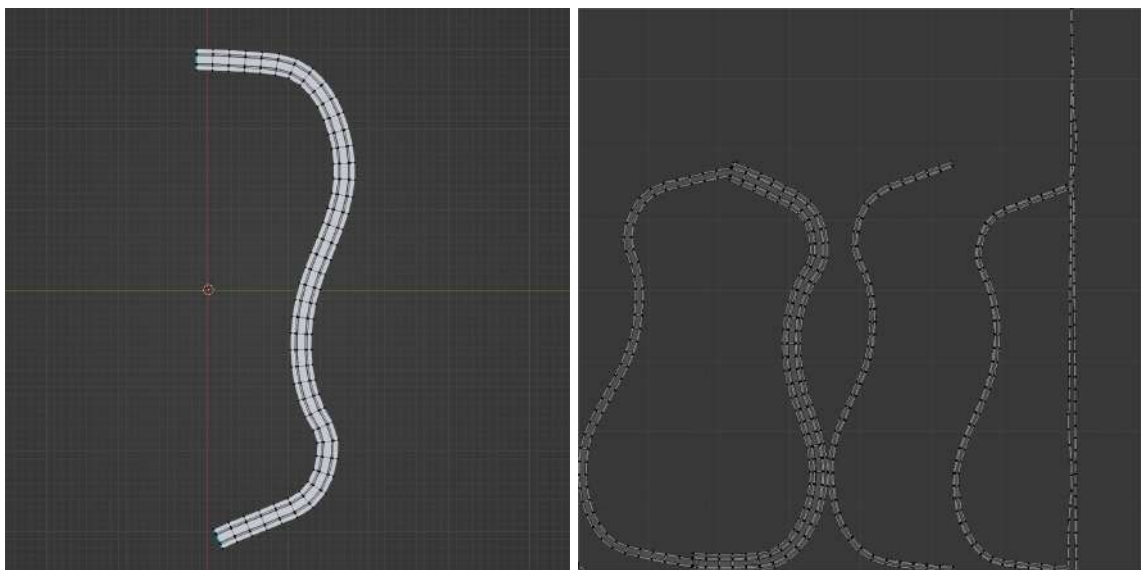
Obrázek 4.19: Demonstrace nutnosti transformace vzorkovaných bodů do 2D na extrémním příkladu zakřiveného mostu

4.2 UV rozbalení a texturování

Provedení UV rozbalení jednotlivých částí mostu se liší v tom, zda je daná část procedurálně generována nebo zda se jedná o předem vymodelovaný, statický model. V případě statických modelů doplněk sám o sobě neprovádí žádné UV rozbalování a očekává, že na vstupy dostává již rozbalené a případně i otexturované modely. Z toho důvodu jsou všechny modely dále uvedené v sekci 4.3 již rozbalené. To znamená, že pokud chce uživatel použít vlastní model, musí si jej nejprve připravit. Z toho důvodu se tato část textu bude zabývat UV rozbalením generovaných částí mostu. Mezi ně patří následující:

- mostovka
- prkna
- trámy
- kabely
- oblouky

Jak bylo zmíněno v předcházející sekci 4.1, generování mostovky muselo být kvůli problémům s UV rozbalením přepracováno. UV rozbalení totiž bylo prováděno nad celým objektem mostovky a ve výsledku docházelo k tomu, že UV mapa byla příliš velká a redundantní. Největším problémem bylo to, že zakřivení mostovky do stran se promítlo i do vygenerované UV mapy, jak lze vidět na obrázku 4.20. To byl problém v tom smyslu, že to zbytečně zatěžovalo uživatele, který by musel manuálně hrany zarovnat.



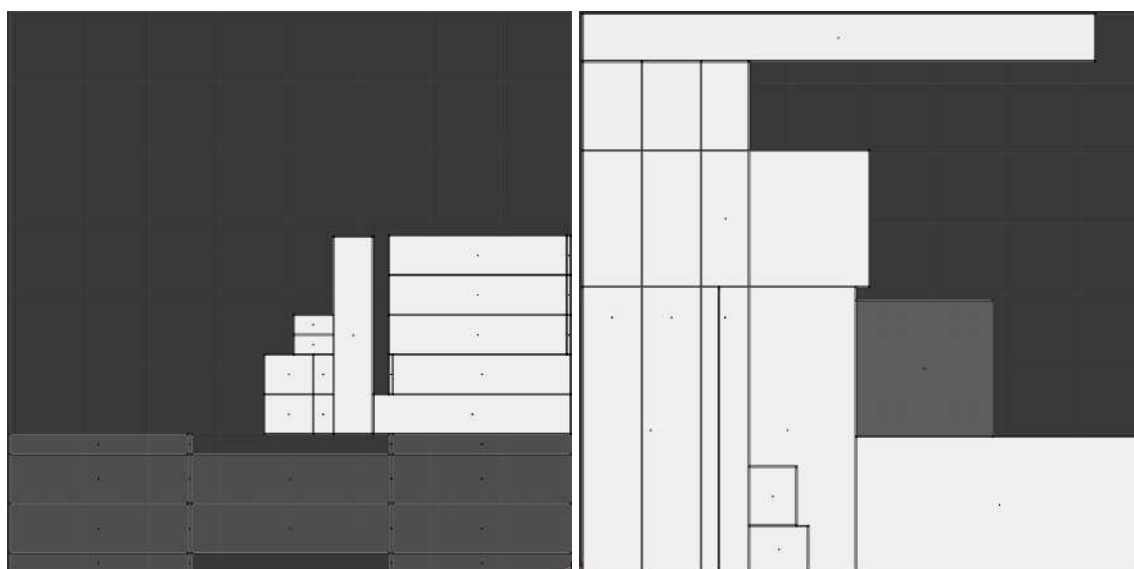
(a) Pohled z ptačí perspektivy na vygenerovanou mostovku

(b) UV mapa mostovky získaná původní metodou generování

Obrázek 4.20: Stará verze mostovky a její UV mapa, ve které jsou stěny silnice ohýbány do stran, což by ztěžovalo zarovnání textury vůči objektu, což by byl problém například u pruhů silnice, které musí procházet přímo středem silnice.

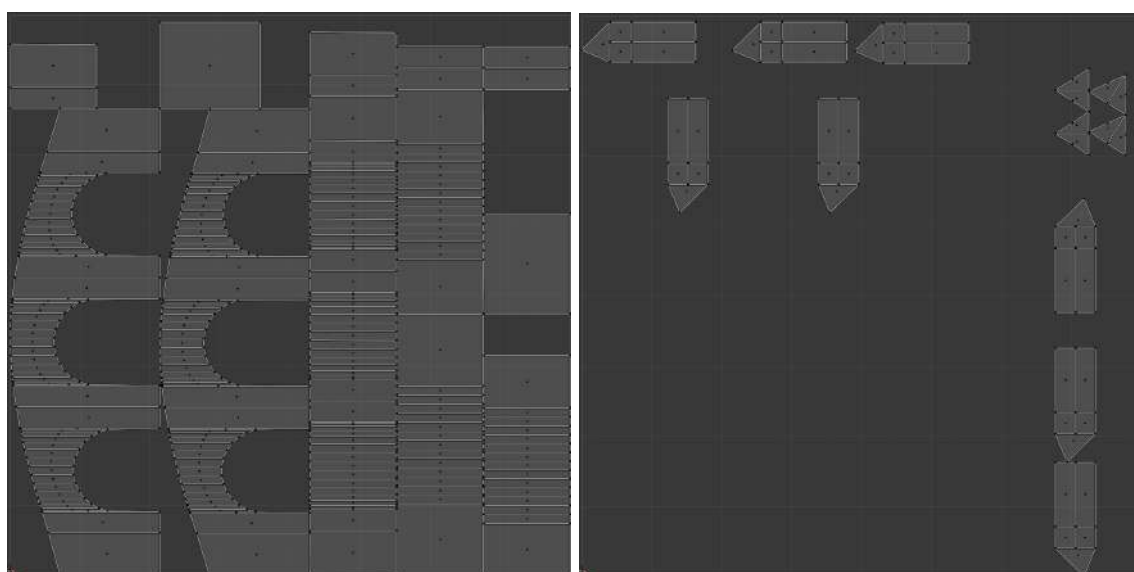
Mostovka je tvořena segmenty, které sdílí společnou UV mapu, což demonstruje obrázek 4.21a. Bílé části jsou překrývající se UV mapy jednotlivých segmentů, které by jinak

byly poskládány vedle sebe do jedné dlouhé nudle jako na obrázku 4.20b dřívější verze UV mapy vozovky. Obrázek 4.21b ukazuje prkenný most, u kterého se překrývají UV mapy jednotlivých prken. Současně jsou na obrázku vidět UV mapy trámů – součástí prkenných mostů jsou dva trámy vedoucí pod prkny od začátku do konce mostu. Nakonec obrázek 4.21c vyobrazuje UV mapu klenutých oblouků mostu a obrázek 4.21d ukazuje UV mapu ledolamů oblouků.



(a) UV mapa mostovky

(b) UV mapa prkenného mostu



(c) UV mapa oblouků

(d) UV mapa ledolamů

Obrázek 4.21: UV mapy hlavních generovaných částí mostu

UV mapy jsou generovány v Geometry Nodes uzlem UV *Unwrap*, který potřebuje znát, kde jsou na objektu umístěny švy. Základním kritériem pro identifikaci švu je hraniční úhel mezi hranami 66° (podobně jako je používáno u případě Smart UV Project). V případě

mostovky jsou navíc za švy označeny všechny hrany obrysu silnice. Výsledek UV rozbalení je uložen jakožto atribut UVMap, což je 2D vektor označující polohu rohu stěny. Při generování částí mostu dochází k slučování všech UV map (včetně těch pocházejících ze statických modelů) do jediného atributu. Nakonec Blender verze 3.5 při konverzi objektu na mesh automaticky rozpozná atributy rohů stěn typu 2D vektor jakožto UV mapu a provede převedení tohoto atributu na speciální atribut typu UV Map.

V rámci funkcionality je vyžadováno, aby bylo možné vybírat všechny strany náležící stejné části modelu pro ulehčení následného texturování. Z toho důvodu nesou všechny stěny výsledného modelu atribut Part, což je atribut datového typu integer. S tímto atributem je dále zacházeno jako s enum datovým typem – každá hodnota představuje určitou část mostu. Vysvětlení, která hodnota reprezentuje kterou část, nabízí tabulka 4.2.

0	Mostovka (solidní most) / prkna mostu (prkenný most)
1	Silnice
2	Chodník
3	Okraj chodníku
4	Strana chodníku
5	Trám (pouze prkenný most)
6	Věž
7	Kabel
8	Užší kabel
9	Oblouky
10	Ledolam (pouze obloukový most)
11	Zábradlí
12	Sloupek zábradlí
13	Svodidla
14	Upevnění na mostu
15	Upevnění na věži

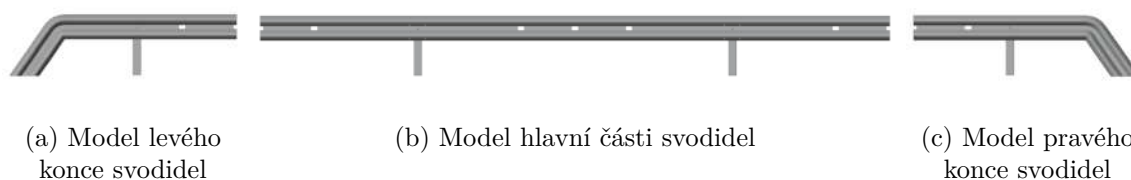
Tabulka 4.2: Tabulka vysvětlující, které části modelu odpovídají kterému klíči v datovém typu enum atributu **Part**

Schopnost vybírání všech stěn stejné části mostu je obzvláště vhodná pro přidělování materiálu vygenerovaným kabelům. Typicky (obzvláště u dlouhých zavěšených mostů) bývá vygenerováno velké množství kabelů, které by bylo velmi pracné manuálně vybírat. Zároveň jsou kabely oproti zbytku mostu velmi tenké a dlouhé, což se projevuje i na UV mapě. Ta zabírá minimální prostor, protože většinu prostoru mostní UV mapy zabírají stěny velkých částí mostu. Obrázek UV mapy kabelů zde proto není uveden, protože by byl nic neříkající. Možnost vybrání kabelů po stisknutí jednoho tlačítka je velké zjednodušení výběru všech kabelů.

4.3 Předpřipravené modely

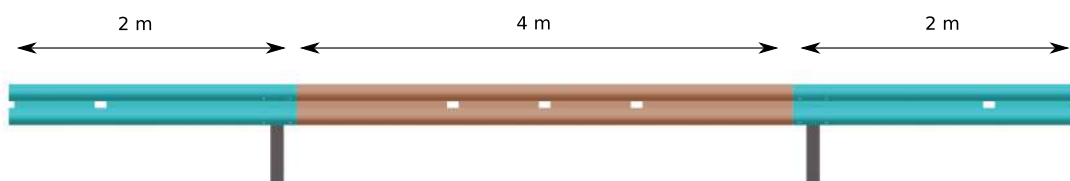
Doplněk nabízí několik modelů zábradlí, svodidel, věží, pilířů a upevňovacích mechanismů pro kabely, které jsou všechny součástí kolekce **BridgeAssets**. Tato kolekce je automaticky nalinkována do uživatelského `.blend` souboru při generování mostu.

Silniční svodidla byla vytvářena podle předlohy plánu jednostranných svodidel². Model byl modelován v několika krocích různými metodami v závislosti na tom, který nástroj byl na daný úkon nejvhodnější. Svodidla tvoří tři separátní modely zobrazené na obrázcích 4.22. Základní díl svodidla (ohýbaný plech) vznikl manuálním modelováním roviny pomocí operace extruze do hrubého tvaru a následným zaoblením lomů. Spodní a horní polovina dílu jsou symetrické, a proto byl použit modifikátor `Mirror`, přičemž pro následovné přidání tloušťky byl následovně použit modifikátor `Solidify`. Stejným způsobem je vymodelován sloupek. Nakonec bylo pomocí `Geometry Nodes` docíleno přidání děr do plechu svodidel a byly napozicovány sloupky. `Geometry Nodes` bylo také použito pro nasazení šroubků a matic (vymodelovány pomocí add-onu `Bolt Factory`³) na místa střetů komponent svodidel.



Obrázek 4.22: Modely jednotlivých částí svodidel

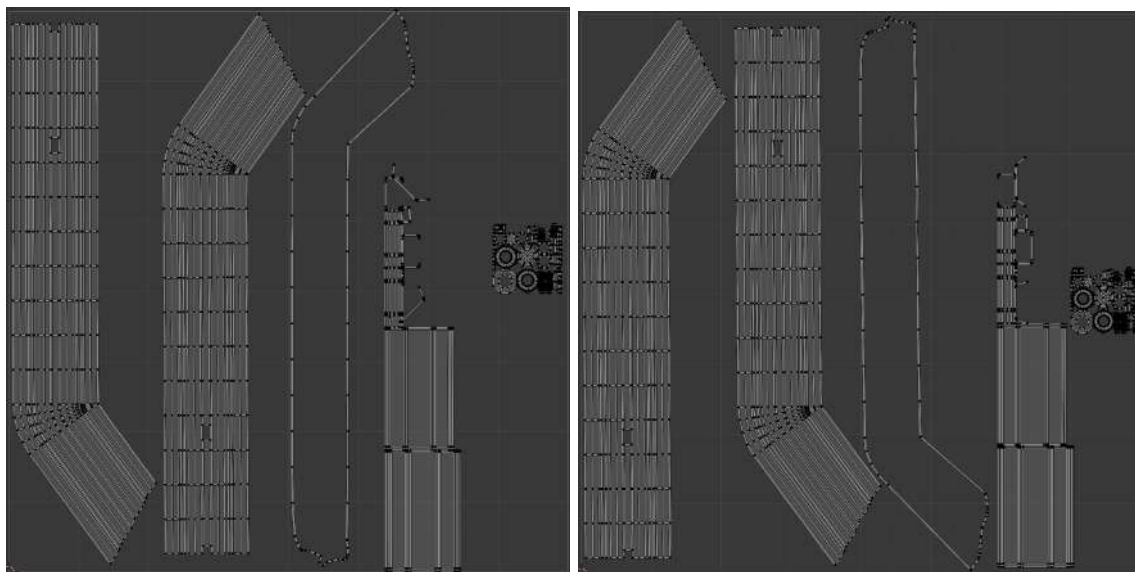
Jeden základní díl modelovaného svodidla má podle plánu délku čtyř metrů. Takovéto díly se skládají za sebe a spoj dvou překrývajících se dílů je proveden šrouby. Pravá strana svodidla musí navazovat na levou stranu, aby bylo možné je skládat vedle sebe. Střední část modelu tvoří celý základní díl svodidla dlouhý čtyři metry a po stranách jsou navázány dvoumetrové poloviny tohoto dílu, které jsou kvůli překryvu posunuty pár centimetrů dopředu, jak ukazuje obrázek 4.23.



Obrázek 4.23: Části svodidel; z důvodu překryvu plechů je červená část vůči modrým částem posunuta dále od kamery

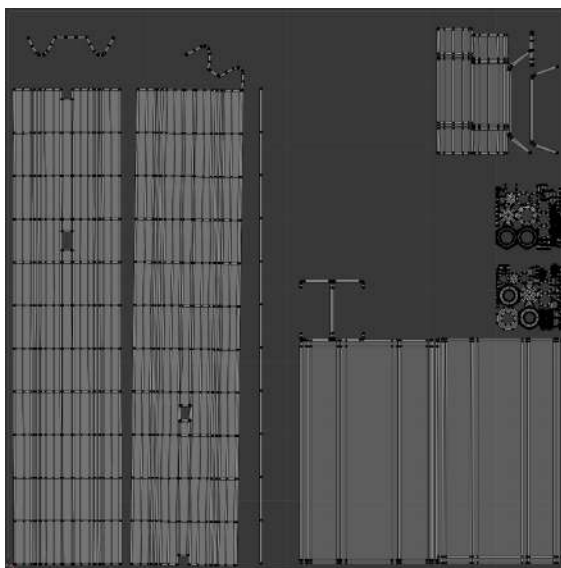
²https://svodidla.cz/wp-content/uploads/6_LIBERTY_prod_list_JSA-AM-4_H1.pdf

³https://docs.blender.org/manual/en/latest/addons/add_mesh/boltfactory.html



(a) UV mapa levého konce svodidel

(b) Model pravého konce svodidel



(c) Model hlavní části svodidel

Obrázek 4.24: UV mapy modelů svodidel

Obrázky 4.24 ukazují vygenerované UV mapy jednotlivých modelů svodidel. Byla snaha o eliminaci zbytečně velkého počtu vrcholů, které by zhoršily časovou náročnost výpočtu nasazování svodidel na most. Při nasazování svodidel dochází k jejich ohybu, proto muselo být ponecháno pravidelné rozmístění vrcholů pro délce svodidel, jinak by při ohýbání mohlo dojít k ostrým lomům.

První modely zábradlí a sloupku byly modelovány jakožto objekty pro testování funkcionality pokládání zábradlí, a proto nebyly modelovány podle žádných konkrétních existujících předloh. Oba byly generovány jednoduchými operacemi extruze a škálování v Geometry Nodes. Manuálně byly nastaveny polohy středů geometrie (*angl. origin of geometry*). Obrázky 4.25 ukazují tyto dva modely a jejich UV mapy lze vidět na obrázcích 4.26.

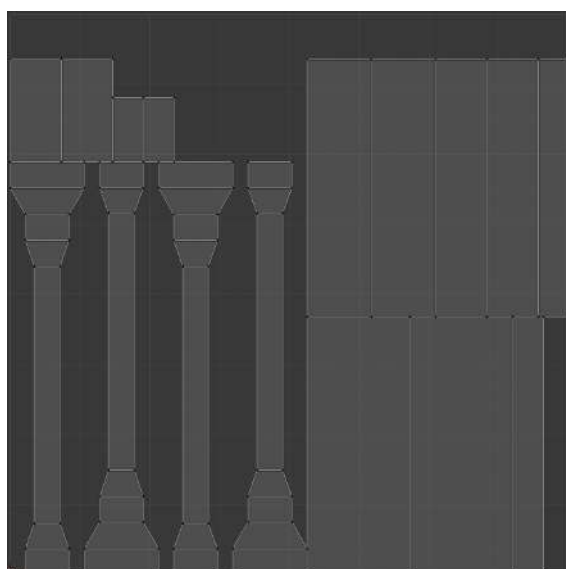


(a) Model hlavní části zábradlí

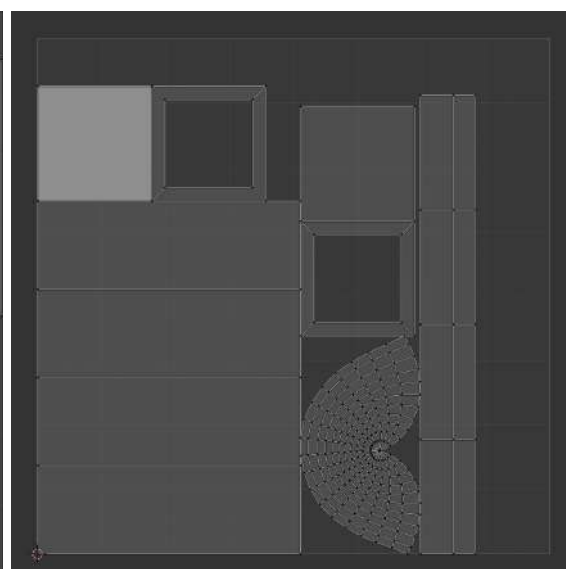


(b) Model sloupku zábradlí

Obrázek 4.25: Hlavní část zábradlí

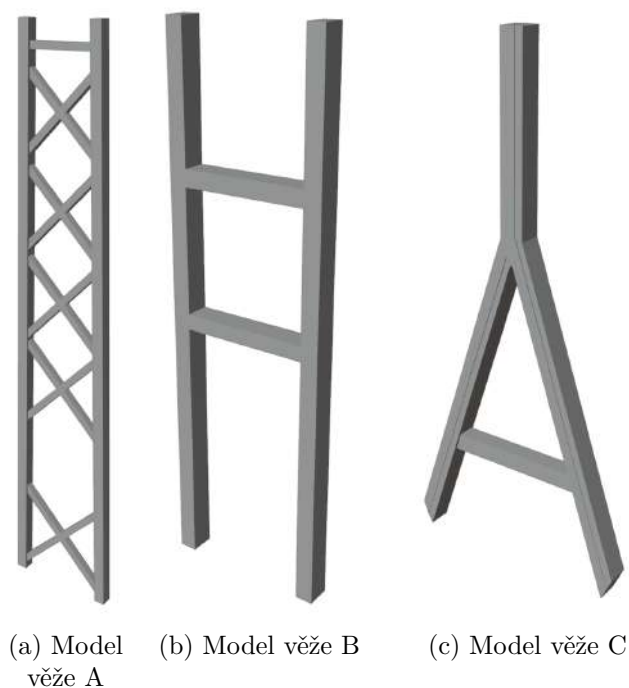


(a) UV mapa hlavní části zábradlí



(b) UV mapa sloupku

Obrázek 4.26: UV mapy modelů zábradlí



Obrázek 4.27: Modely věží

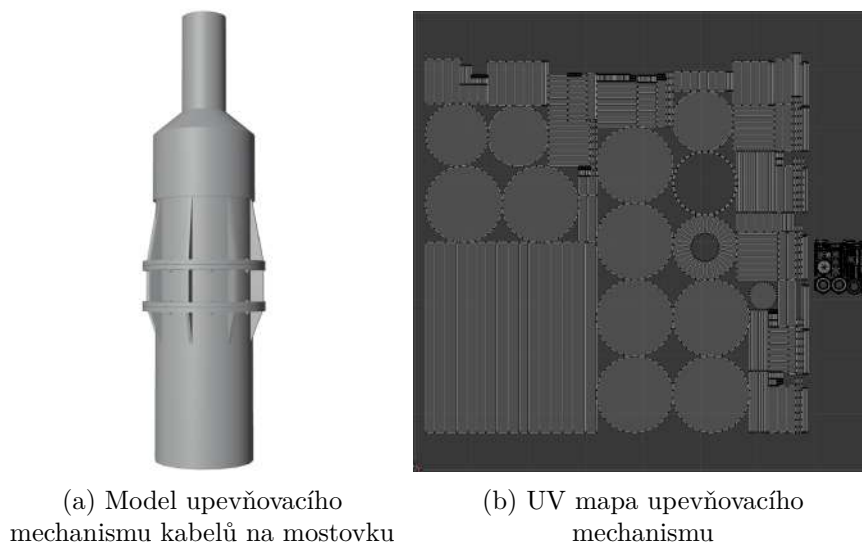
Model věže A (obrázek 4.27a) byl vymodelován podle věží mostu Akashi Bridge a předloha pro model C (obrázek 4.27c) byla věž mostu Flintshire Bridge použitá při finálním testování add-onu. Model B na obrázku 4.27b je model vytvořen v prvních fázích implementace pro účely testování funkcionality rozmístění věží a generování kabelů, a nebyl modelován podle žádného konkrétního reálného mostu. Všechny věže mají v nabídce doplněk dvě verze – jednu určenou pro zavěšené a druhou určenou pro visuté mosty. Tyto verze mají jinak rozmístěné upevňovací body kabelů, aby kabely byly umístěny rozumně s ohledem na zvolenou konstrukci mostu. Věže určené pro zavěšené jsou v nabídce panelu označeny písmeny *CS* (z anglického *Cable Stayed*) a visuté písmenem *S* (z anglického *Suspension*). Níže na obrázku 4.28 lze vidět UV mapy jednotlivých věží.



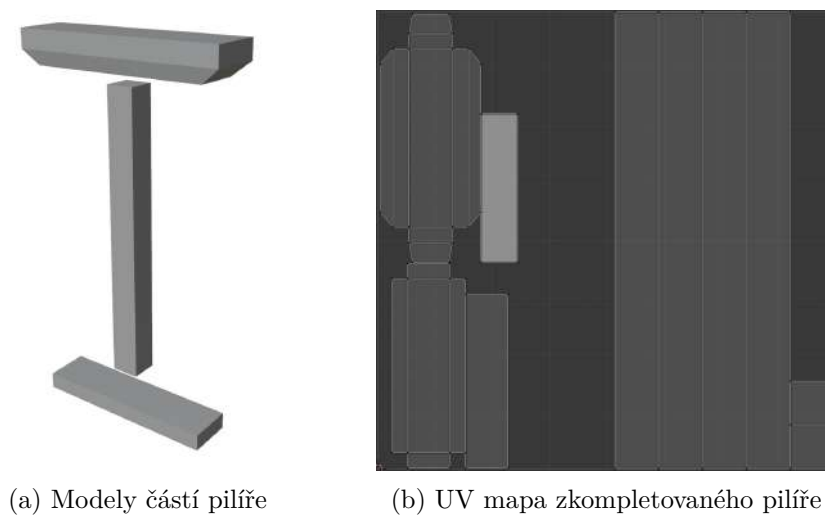
(a) UV mapa modelu věže A (b) UV mapa modelu věže B (c) UV mapa modelu věže C

Obrázek 4.28: UV mapy modelů věží

Upevňovací mechanismy (obrázek 4.29) byly modelovány podle těch vyskytujících se na reálném mostu Flintshire Bridge⁴. Modelování probíhalo v kombinaci procedurálního a manuálního přístupu. Model je tvořen ze dvou částí, z nichž každá část je zvlášť generovaná pomocí Geometry Nodes. Šrouby a matice jsou dodávány (stejně jako u dříve zmíněného modelu svodidel) s pomocí doplňku Bolt Factory. Následovně další modifikátor Geometry Nodes provede jejich spojení a manuálně je poté nastaveno Shade Smooth a Auto Smooth s parametrem Angle 30°.



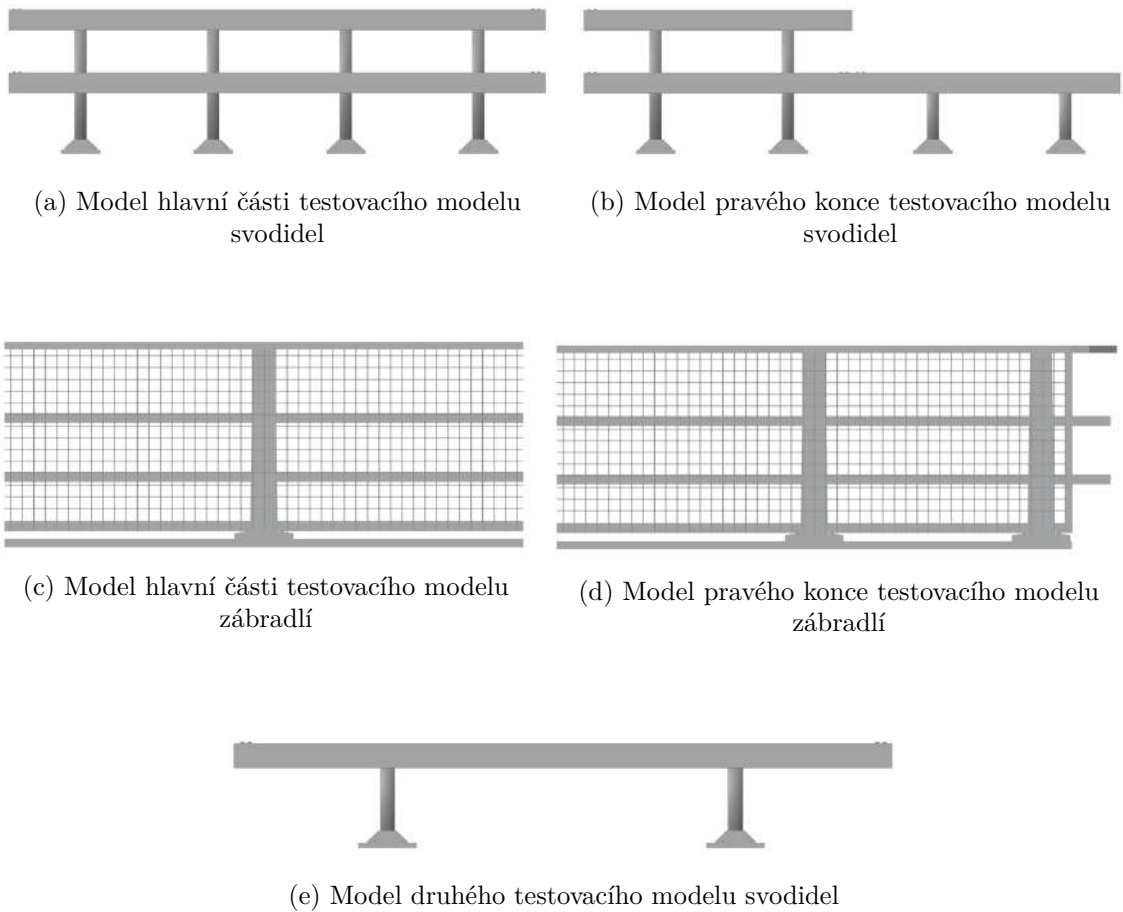
Obrázek 4.29: Upevňovací mechanismus kabelů na mostovku



Obrázek 4.30: Modely pilíře

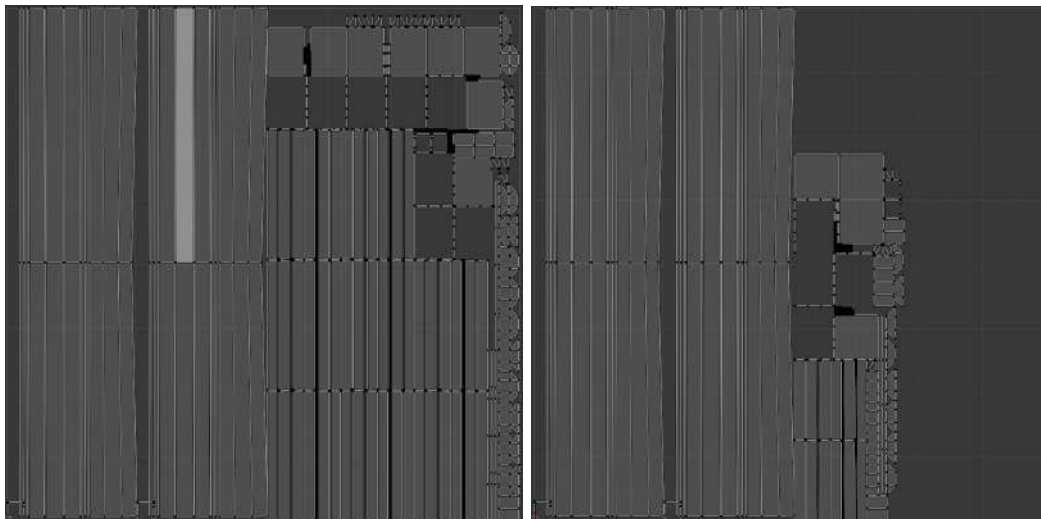
Nakonec obrázky 4.30 ukazují jednotlivé modely pilíře – základnu, hlavní část a svršek. Na obrázku 4.30a jsou ukázány s rozestupy pro odlišení jednotlivých částí. Obrázek 4.30b ukazuje celkovou UV mapu pilíře vzniklou spojením UV map všech tří částí.

⁴[https://maffin.co.uk/flintshire-bridge/cable-stayed-bridge/#iLightbox\[postimages\]/0](https://maffin.co.uk/flintshire-bridge/cable-stayed-bridge/#iLightbox[postimages]/0)



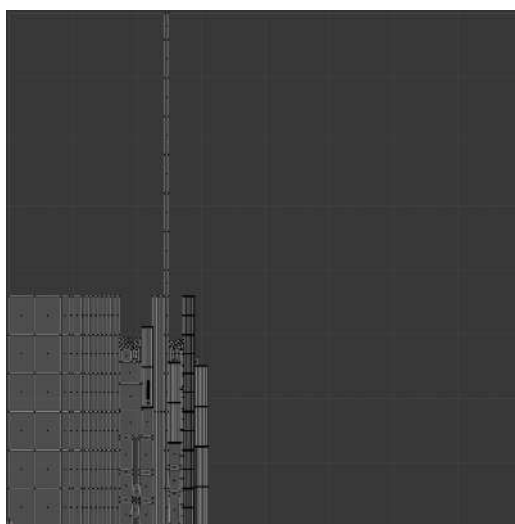
Obrázek 4.31: Modely svodidel a zábradlí vytvořených dle předlohy mostu Flintshire Bridge: Modely (a) a (c) jsou hlavní části, které jsou po bocích ukončeny koncovými modely. Z koncových modelů je vyobrazena pouze pravá varianta, jelikož levá verze je zcela stejná, pouze převrácená.

V rámci testování v kapitole 5 bylo potřeba vymodelovat několik modelů mostu Flintshire Bridge. Modely zábradlí a svodidel vytvořené pro testovací účely, které byly nakonec přidány do nabídky modelů doplňku, lze vidět na obrázcích 4.31. Jejich UV mapy jsou na obrázcích 4.32.



(a) UV mapa modelu svodidel 4.31a

(b) UV mapa modelu zábradlí 4.31c



(c) UV mapa modelu svodidel 4.31e

Obrázek 4.32: UV mapy testovacích hlavních částí modelů mostu Flintshire Bridge: UV mapy koncových modelů jsou velmi podobné mapám hlavních částí, proto zde nejsou zobrazeny.

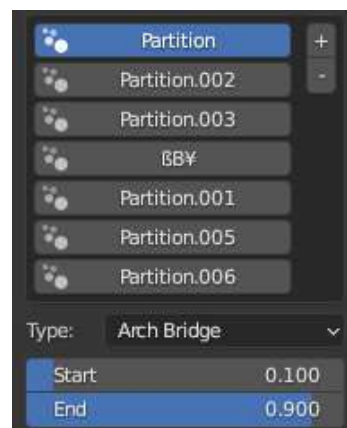
4.4 Zprovoznění uživatelského rozhraní

Panely pro řízení generátoru mostů jsou implementovány s pomocí Blender Python API⁵, které je k tomuto účelu určené. Jistý problém nastal v souvislosti s implementací dynamického dělení mostu na části (například pro mosty, jejichž část má stát na pilířích, střed má být zavěšen na lanech a koncová část má být také podpírána pilíři – tři odlišné části mostu s různými parametry). Pro tyto účely skript využívá element `DynamicEnum`, což je datový typ `enum`, který umožňuje dynamické přidávání a odebrání položek.

`DynamicEnum` je implementován v aplikaci Blender tak, že své položky pravidelně získává návratem listu z callback funkce. Problém nastává v tom, že Blender po návratu neuchovává referenci na navrácené položky a dochází ke ztrátě dat. Jedná se o známý a zdokumentovaný bug. Dokumentace Blender Python API se zmiňuje, že je nutné tuto referenci uchovat v kódu, ale k tomuto problému dochází dokonce i v případě, že jsou položky ukládány do globální proměnné. Na obrázku 4.33 lze pozorovat, jak se bug projevil v tomto případě. Názvy jednotlivých položek jsou přeházené (v samotném seznamu položek jsou seřazené) a jedna položka má namísto názvu zobrazeny náhodné znaky z ASCII tabulky.

Dokumentace Blender Python API, která bug zmiňuje, postrádá popis funkčního řešení problému (pouze se zmiňuje o tom, že je potřeba ukládat referenci na proměnnou `items`). Nestačí ukládat celý seznam položek v globální proměnné, ale problém je řešitelný ukládáním jednotlivých string prvků každé položky seznamu, které provádí funkce `intern_enum_items`⁶. Tato funkce ukládá jednotlivé string elementy do globálního listu ještě před tím, než callback funkce položky vrátí `EnumProperty`, čímž zabrání ztrátě reference.

Druhý problém, který se při implementaci uživatelského rozhraní vyskytnul, se projevoval v případě, že uživatel uložil a uzavřel Blender před dokončením modelování mostu. Doplňek si k objektu ukládá data o stavu modelování a aktuálním nastavení panelu. Při zavření aplikace taková data Blender zahazuje, tudíž při dalším otevření nemohl uživatel pokračovat v práci. Z toho důvodu bylo implementováno tlačítko s názvem `Recover Bridge`. Operátor tohoto tlačítka detekuje, když objekt obsahuje modifikátory spjaté s doplňkem, o kterých chybí informace. Když uživatel otevře svůj soubor, aby mohl pokračovat v modelování, pak v ten moment je v panelu `add-onu` aktivní pouze toto tlačítko. Po jeho stisknutí si doplňek projde všechny modifikátory objektu a obnoví si ztracené informace o stavu modelování.



Obrázek 4.33: Projev známého bugu souvisejícího s `DynamicEnum` v panelu doplňku

⁵<https://docs.blender.org/api/3.5/>

⁶<https://blender.stackexchange.com/a/245145>

Kapitola 5

Testování

Testování je důležitá součást práce, která probíhala jak v průběhu samotného vývoje, tak při konečném zhodnocování kvality produktu. První sekce kapitoly 5.1 je věnována průběžnému testování funkcionality a kvality jednotlivých elementů doplňku. Konečné měření kvality add-onu, které probíhalo formou srovnání manuálního modelování konkrétního mostu Flintshire Bridge s modelováním s pomocí doplňku, je popsáno v sekci 5.2.

5.1 Průběžné testování funkcionality

V průběhu implementace docházelo k pravidelnému testování rozpracované i již dokončené funkcionality. Bylo zapotřebí pravidelně kontrolovat a spravovat kvalitu následujících položek:

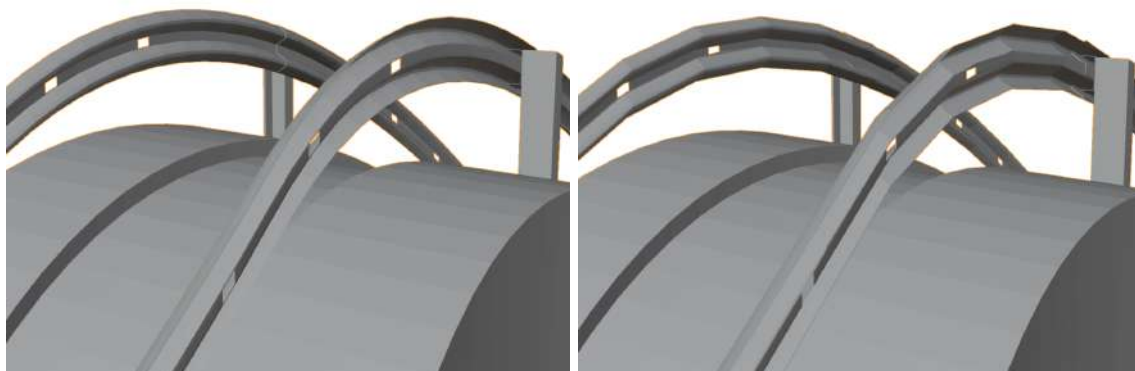
- generování geometrie
- UV mapy
- modely
- uživatelské rozhraní

Testování probíhalo na různých vstupních parametrech, ze kterých byla nejzásadnější vodící křivka uživatele. Při implementaci nové funkcionality byl modifikátor Geometry Nodes (obsahující danou vyvíjenou funkcionalitu) aplikován na několik různých křivek: rovných či v různé míře zakřivených do stran a do výšky. Při práci bylo díky povaze systému Geometry Nodes možné v reálném čase pozorovat, jak se změny v implementaci projevují na jednotlivých křivkách, a bylo tak možné rychle odchytnout případné chyby závislé na typu křivky.

Sledováním efektu modifikátorů na vstupní křivku bylo možné kontrolovat všeobecný tvar generované geometrie. Pro sledování topologie (kontrolování uspořádání a množství vrcholů, hran a stěn geometrie) bylo pravidelně prováděno převedení objektu na mesh. Tento převod umožnil realizaci vrcholů, hran a stěn, které následně bylo možné sledovat v editačním módu. Po převodu na mesh byly následně pozorovány UV mapy – zda byly vygenerovány správně.

V případě modelů docházelo ke kontrolám jejich topologie, UV map a skupin vrcholů potřebných ke generování, které je na nich postavené. Modely, u nichž dochází při generování mostu k ohybu (zábradlí a svodidla ohýbající se podle povrchu mostu), musely být v tomto ohledu testovány. Pokud při jejich ohýbání došlo k ostrým lomům namísto hladkého ohnutí,

musely být po délce modelu umístěny vrcholy navíc (tzv. *loop cuts*). Po přidání vrcholů je efekt lomu rozprostřen rovnoměrně napříč vrcholy po délce modelu.



(a) Svodidla s efektivním rozmístěním vrcholů jsou ohnuta rovnoměrně (b) Svodidla bez vrcholů po délce modelu jsou ohnuta lánaným způsobem

Obrázek 5.1: Extrémní příklad pro porovnání ohybů modelů svodidel: Model (a) byl vytvořen přidáním vrcholů pomocí *loop cuts*, zatímco model (b) ne

Nakonec uživatelské rozhraní bylo testováno po propojení se zbylou funkcionalitou. Testování probíhalo simulováním práce uživatele s doplňkem. Tím docházelo k postupnému testování všech částí funkcionality – generování mostu, UV rozbalení a součástí panelu jako dynamické přidávání a odebrání částí mostu. Také byly testovány zvláštní případy, jak se chová doplněk po uložení, zavření a zpětném otevření souboru před dokončením modelování mostu.

5.2 Konečné testování využití v praxi

Hlavním cílem testování je ověření využitelnosti dokončeného produktu – zda je add-on pro uživatele přínosný a do jaké míry. To lze ověřit prakticky na modelování konkrétního mostu z reálného světa ve dvou bězích. První běh zahrnuje klasický přístup k modelování využívající pouze nástroje, které poskytuje Blender verze 3.5, bez využití žádných externích rozšíření. Následně v druhém běhu je k modelování použit pouze doplněk pro generování mostů. Tyto dva procesy modelování je poté možné porovnat a z jejich průběhů a výsledků vyvodit patřičné závěry.

Míra ušetřeného času využitím doplňku namísto modelování mostu ručně závisí na tom, zda uživateli stačí modely poskytované doplňkem. Pokud uživatel vyžaduje konkrétní vzhled věže, zábradlí nebo jiné části mostu, pak je nucen si jej sám vymodelovat nebo získat jiným způsobem. Vytváření vlastních modelů a následné UV rozbalení představuje pro uživatele nezanedbatelnou časovou ztrátu. S ohledem na schopnosti běžného uživatele a na náročnost modelu může tento proces trvat v řádu jednotek až desítek hodin. Pokud je ale modelovaný objekt velmi jednoduchý nebo pokud je uživatel velmi zkušený, může se jednat o časovou náročnost několika minut. V případě modelování konkrétního mostu dle reálné předlohy nebudou uživateli nabízené modely stačit. Čas modelování vlastních modelů je v takovém případě z pohledu testu zanedbatelný, protože se tato časová ztráta projeví nehledě na tom, jestli je most následně modelován ručně nebo procedurálně. Čas vytváření modelů pro části mostu proto není v testu sledován.

Výsledek tohoto testování je závislý na tom, jaký konkrétní most je předmětem modelování. Add-on je schopný generovat mosty podle měnících se parametrů velmi rychle, ale jeho funkcionalita je omezená – nedokáže vygenerovat cokoliv. Na druhou stranu uživatel modelující celý objekt od nuly je omezen pouze svými vlastními schopnostmi. Je tedy zapotřebí vybrat most, který doplněk dokáže napodobit. Pro provedení tohoto testu byl vybrán britský most jménem Flintshire Bridge, což je představitel zavěšených mostů. Lze si jej prohlédnout níže na obrázku 5.2. Je to most přes řeku, jehož střední část je zavěšená na věži a jehož konce stojí na sloupech a postupně přechází v obyčejnou silnici. Most je mírně zakřivený. Pro jeho modelování jsou použity obrázky z internetových zdrojů. Hlavní předlohou pro modelování jsou letecká mapa a pohledy z ulice internetových map.



Obrázek 5.2: Fotografie věže mostu Flintshire Bridge¹

5.2.1 Postup manuálního modelování mostu

Manuálně lze cílený objekt vymodelovat nekonečně mnoha způsoby s různými časovými náročnostmi vzhledem ke zvolené metodě. Při tomto testu jsou použity modifikátory Mirror, Curve, Array a Geometry Nodes. Za jejich použití lze provést modelování analogicky ke způsobu, kterým generuje most doplněk. Následující seznam je postup manuálního modelování použitý při tomto testu:

1. Nakreslení vodící křivky

¹<https://www.geograph.org.uk/photo/411436>

2. Vytvoření segmentu mostovky
3. Sestavení a ohnutí mostovky
4. Texturování a položení věží, zábradlí, svodidel
5. Upevnění kabelů věže
6. Aplikování materiálů

Vymodelování segmentu mostovky lze provést jednoduchými úpravami v editačním módu s použitím modifikátoru `Mirror`, aby byly pravá a levá strana mostu symetrické. Operací `extrude` je provedeno na stranách segmentu vytvoření chodníku. Zároveň je touto metodou uprostřed segmentu vyvýšena platforma, na které mají být umístěna svodidla mezi jízdnicemi pruhy. Následně je nutné následně provést UV rozbalení – manuálně vytvořit švy a použít `UV Unwrap` či přímo použít `Smart UV Unwrap`. V dalším kroku je nad segmentem použit modifikátor `Array` pro sestavení rovné verze mostovky. Před jeho aplikací je vhodné u modelu segmentu odstranit přední a zadní stěny. Segmenty jsou skládány těsně za sebe a bez odstranění těchto stěn vzniknou po aplikaci modifikátoru `Array` stěny uvnitř geometrie. Nakonec je potřeba modelu vrátit stěny na jeho koncích a otexturovat. Dále je model ohnut podle vodící křivky modifikátorem `Curve`, čímž je mostovka hotová.

Flintshire Bridge má pouze jednu věž, tudíž není příliš náročné umístit ji na most ručně. Naopak pracné pro uživatele je následovně vytváření kabelů, které může představovat tu nejnáročnější část ručního modelování tohoto mostu. Na věž se upíná celkem třicet osm párů kabelů, což je celkem sedmdesát šest kabelů, které musí být ručně upevněny nejprve na věž a poté na mostovku. Uživatel tedy bude chtít umístit sedmdesát šest tenkých válců mezi věží a mostovkou, které navíc musí být rozmístěny rovnoměrně. Pro ulehčení práce bude nutné nejen označit body na věži, kde mají být upevněny konce lan, ale budou muset být označeny i na mostovce. Pomocí `Geometry Nodes` lze během pár minut umístit na body věže úsečky, jejichž druhé konce bude následně nutno manuálně umístit na mostovku. Umístění kabelu na mostovku lze s pomocí 3D kurzoru. Stačí vybrat na mostovce bod, kde má být kabel umístěn a položit do něj kurzor (pomocí `Cursor to Selected`). Následně je vybrán konec křivky kabelu a ten je umístěn na pozici kurzoru (`Selection to Cursor`). Tento proces je nutné opakovat pro každý kabel. Při větším množství věží a kabelů by se vyplatilo tento proces automatizovat. Nakonec jsou všechny křivky kabelů vybrány a v `Geometry Nodes` převedeny na mesh s kruhovým profilem, čímž dostanou určitou tloušťku.

Dále je potřeba na povrchu mostu umístit svodidla a zábradlí. Modely je potřeba si s pomocí modifikátoru `Array` připravit do rovné linie a ohnout pomocí modifikátoru `Curve`. V tomto případě jsou nasazovány dvě řady svodidel, dvě řady zábradlí a jedna řada svodidel na platformě uprostřed mostu. Kromě svodidel uprostřed mostu jsou zábradlí a svodidla zakončena koncovými modely. Po aplikování modifikátoru `Array` je potřeba na každý konec tyto koncové modely ručně umístit. Tento poměrně pracný proces spočívá v umístění středu geometrie (`origin`) v místa, kde se mají modely dotýkat, následném umístění modelů do středu souřadnic (`world origin`) a spojení modelů do jednoho. Správné nastavení středů geometrie zajistí zarovnání modelů před jejich sjednocením. Po sestavení zábradlí (svodidel) je potřeba z mostovky separovat kopii hrany, na které má být daná svodidla či zábradlí umístěna. Tyto hrany je poté nutné převést na křivku, aby bylo možné ji použít pro ohyb pomocí modifikátoru `Curve`.

Nakonec je potřeba umístit sloupy po obou koncích mostu. To lze udělat jednoduše dalším použitím modifikátoru `Array` a `Curve`, čímž je model mostu hotový. Pokud uživatel

zapomněl nastavit materiál pilířům, zábradlím, svodidlům a dalším, než je začal duplikovat a sestavovat z nich most, pak v této fázi může narazit na problém takový, že musí ručně vybrat jednotlivé stěny každé části a postupně jim přidělovat materiál. Pokud již uživatel všechny modely sjednotil do jednoho, tak tento proces vybírání jednotlivých částí modelu může znamenat velkou časovou ztrátu.

Tabulka 5.1 shrnuje časové náročnosti jednotlivých úkonů potřebných pro vytvoření mostu manuálním modelováním. Jeden sloupec udává odhadovaný všeobecný čas trvání dílčího úkonu modelování závislého na počtu modelovaných objektů. Poslední sloupec ukazuje časy modelování mostu Flintshire Bridge v tomto konkrétním testu.

Segment procesu modelování	Odhadovaný čas	Čas u zvoleného modelu
kompletní vytvoření mostovky	20 min	30 min
položení n svodidel a zábradlí	$10n$ min	1 h
umístění n věží	$5n$ min	5 min
upevnění kabelů na věže	10 min	10 min
nasazení n konců kabelů na mostovku	$10n$ s	10 min
položení upevňovacích mechanismů	10 min	10 min

Tabulka 5.1: Časová náročnost jednotlivých částí ručního modelování: odhadované časy a časy při testu modelování mostu Flintshire Bridge, které jsou horší než odhadované, protože zahrnují časy navíc způsobené chybami.

Vytvoření mostovky je poměrně jednoduchý proces, který zahrnuje větší množství kroků. Je snadné na nějakou součást tohoto kroku zapomenout, což může způsobit, že uživatel mostovku musí začít modelovat od začátku. To stejné platí i o pokládání svodidel či zábradlí. Umístění věží nezahrnuje žádné ohýbání či skládání více modelů do jednoho, takže jeho časová náročnost je při nízkém počtu věží malá. Následovně umístění křivek na věže pro vytvoření kabelů je sám o sobě jednoduchý proces, který lze provést s pomocí Geometry Nodes velmi snadno. Časové náročné může být následně nasazování konců kabelů na mostovku při velkém množství kabelů. Nakonec umístění modelů upevňovacích mechanismů je díky Geometry Nodes rychlý proces.

5.2.2 Postup modelování za použití add-onu

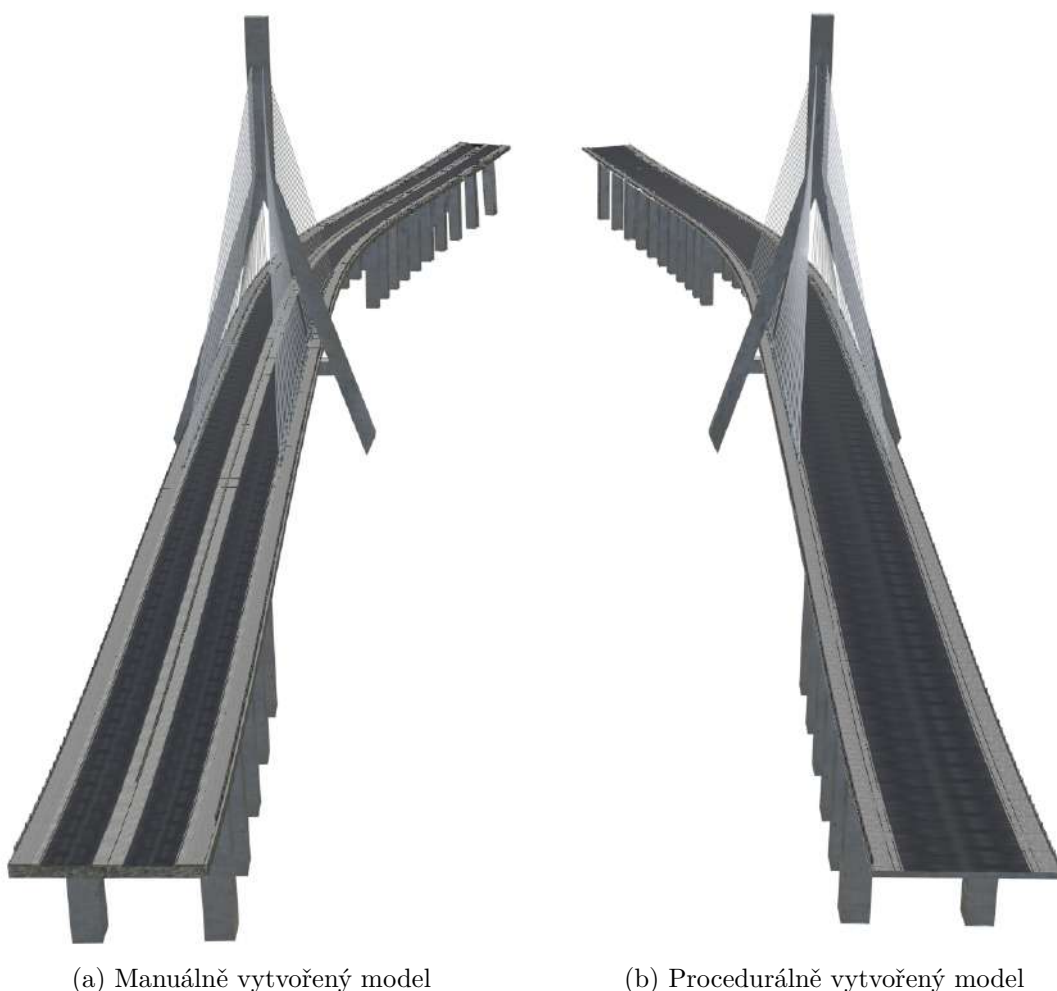
Vytváření mostu s použitím add-onu sdílí pár společných kroků s metodou manuálního modelování: jmenovitě nakreslení vodící křivky a vyhledání / vytvoření modelů a textur. Následovně zbytek práce na modelování může uživatel provést v ovládacím panelu doplňku, který most dokáže vygenerovat ve zlomku vteřiny. Cenou za rychlost a jednoduchost tohoto přístupu modelování je omezení možností generovaného mostu.

Celkový čas modelování s pomocí add-onu byl u tohoto testu zhruba 15 minut včetně texturování výsledného mostu, což je 12 % času stráveného manuálním modelováním stejného modelu. Vygenerovaný model měl kvůli velkému počtu opakujících se modelů svodidel, zábradlí a zbylých částí mostu velké množství stěn. V procesu texturování potřebuje operátor pro výběr všech stěn stejné části mostu iterovat přes všechny stěny. Kvůli velkému množství stěn trval každý průchod pro výběr části mostu několik vteřin a z toho důvodu trvalo texturování déle, než by trvalo u menšího modelu.

5.2.3 Závěr finálního testování

Vyjma vytváření vlastních modelů a hledání materiálů vykonává add-on většinu práce za uživatele, přestože je omezený. Po poskytnutí doplňku na vstupy všechny modely částí mostu (věž, zábradlí, svodidla), pak doplněk velmi rychle vygeneroval most, který bylo možné během následujících pár minut upravit. Nevýhodou bylo omezení doplňku v tom smyslu, že nedokázal vygenerovat svodidla mezi pruhy silnice a ve fázi texturování byl pomalý při výběru stěn částí mostu. Nicméně v případě manuálního přístupu byl celý proces časově náročný.

Jak lze vidět na obrázku 5.3, kvalita výsledků je srovnatelná vyjma svodidel mezi jízdnicími pruhy, které u vygenerovaného mostu chybí (5.3b). Oba modely byly texturovány použitím volně dostupných textur dostupných online na stránce Poliigon² a nástroje BlenderKit³.



Obrázek 5.3: Modely mostu Flintshire Bridge vytvořené různými způsoby modelování: Manuálně vytvořený model (a) zahrnuje svodidla uprostřed silnice, což add-on nepodporuje (b)

¹<https://www.poliigon.com/>

²<https://www.blenderkit.com/get-blenderkit/>

Obrázky 5.4 nabízí bližší pohled na modely vytvořené při finálním testování. Lze pozorovat, že kabely bylo možné rozmístit rovnoměrně i při ručním způsobu. Body pro upevnění kabelů na mostovku musely být vytvořeny už při modelování samotné mostovky. V případě manuálně vytvořeného modelu nelze snadno zpětně měnit rozmístění kabelů. To platí pro většinu parametrů mostu. Výhoda add-onu oproti manuálnímu přístupu je v tom, že před aplikováním modifikátorů lze velmi snadno a rychle takové parametry měnit.



(a) Přiblížený pohled na manuálně vytvořený model

(b) Přiblížený pohled na procedurálně vytvořený model

Obrázek 5.4: Srovnání manuálně a procedurálně vytvořených modelů zblízka

Kapitola 6

Závěr

Cílem této práce byl návrh následovaný implementací a testováním add-onu pro generování modelů mostů na platformě Blender. Jedním z pilířů funkcionality doplňku, kterým se odlišuje od již existujících řešení, je možnost zakomponování modelů uživatele do tvorby výsledného mostu. Dílčím cílem bylo poskytnout širokou nabídku typů konstrukcí a již připravených modelů pro větší flexibilitu při tvorbě mostů.

Součástí návrhu bylo zvážení kladů a záporů generátorů mostů, které již byly implementovány, a vybrání aspektů, na kterých bylo možné stavět a případně je vylepšit. Od existujících řešení bylo převzato definování základního tvaru mostu pomocí vstupní křivky, kterou lze v průběhu modelování upravovat bez omezení a v reálném čase sledovat generování mostu. Následně byl kladen důraz na zajištění velkého přizpůsobení mostu potřebám uživatele.

Pro implementaci byl zvolen nástroj Geometry Nodes. Generování mostu je prováděno sadou modifikátorů typu Geometry Nodes, z nichž každý se věnuje vytváření právě jedné dílčí části mostu. Toto rozdělení zajišťuje jistou míru optimalizace, které by nebylo možné dosáhnout při generování pouze jedním modifikátorem. Zároveň tak bylo dosaženo větší přehlednosti zdrojového souboru. Následně bylo uživatelské rozhraní implementováno pomocí Blender Python API.

Geometry Nodes spadá do kategorie vizuálního programování, což umožnilo provádět průběžné testování funkcionality zároveň s implementací. V závěru práce bylo nakonec provedeno testování spočívající v porovnání náročnosti modelování konkrétního mostu s výsledným add-onem a poté bez něj. Za pomoci doplňku bylo možné model dokončit v daleko kratším čase než při manuálním způsobu modelování. Experiment také ukázal, že mezi mosty reálného světa existuje široká diverzita, kterou není add-on schopný pokrýt, a tudíž je finální produkt vhodný pro volné modelování spíše než pro modelování konkrétních mostů.

Práci by bylo do budoucna vhodné vylepšit o generování většího množství typů mostů, jmenovitě obloukových mostů s obloukovou konstrukcí a provazových mostů. Dalším námětem na rozšíření je větší přizpůsobení mostovky – různé tvary profilu, umožnění rozdělení silnice na jízdní pruhy a přidání svodidel doprostřed mostu. Značné vylepšení využití doplňku by také představoval větší výběr modelů částí mostu a přidání nabídky základních materiálů, aby je uživatel nemusel hledat či sám vytvářet.

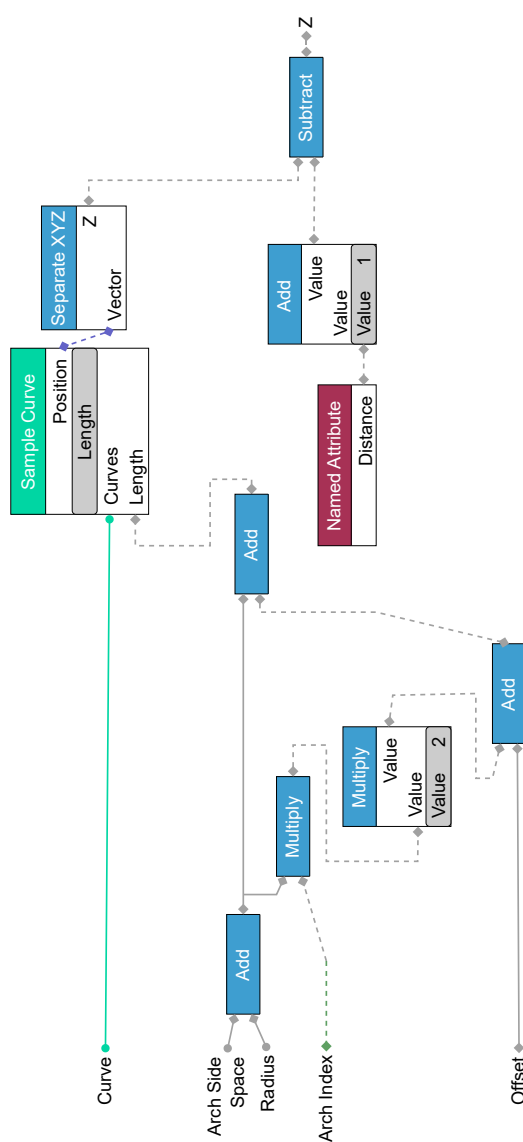
Literatura

- [1] BAILYLDESIGN. *What Version Of Blender Has Geometry Nodes?* [online]. Blender Base Camp, 16. června 2022 [cit. 2022-11-12]. Dostupné z: <https://www.blenderbasecamp.com/home/what-version-of-blender-has-geometry-nodes/>.
- [2] BENEŠ, B., ŠŤAVA, O., MĚCH, R. a MILLER, G. Guided Procedural Modeling. *Computer Graphics Forum*. 2011, sv. 30, č. 2, s. 325–334. DOI: <https://doi.org/10.1111/j.1467-8659.2011.01886.x>. Dostupné z: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2011.01886.x>.
- [3] BILLINGTON, D. P., BILLINGTON, P. N. a SHIRLEY SMITH, H. *Bridge* [online]. Encyclopedia Britannica, 28. července 1999. revidováno 3. 3. 2022 [cit. 2022-09-23]. Dostupné z: <https://www.britannica.com/technology/bridge-engineering>.
- [4] BLENDER BASH. *Curve Deform Geometry Node in Blender* [online]. [cit. 2022-04-07]. Dostupné z: <https://blenderbash.gumroad.com/l/rllsl?layout=profile>.
- [5] CHEN, W.-F. a DUAN, L. *Bridge Engineering Handbook: Fundamentals*. 2. vyd. CRC Press, Taylor & Francis Group, září 2013. ISBN 978-1-4398-5234-7.
- [6] FREIKNECHT, J. a EFFELSBERG, W. A Survey on the Procedural Generation of Virtual Worlds. *Multimodal Technologies and Interaction*. 2017, sv. 1, č. 4. ISSN 2414-4088. Dostupné z: <https://www.mdpi.com/2414-4088/1/4/27>.
- [7] HEWITT, P. G. The Delightful Catenary Curve. *The Science Teacher*. Apr 2017, sv. 84, č. 4, s. 14–16. Copyright - Copyright National Science Teachers Association Apr/May 2017; Document feature - Photographs; Illustrations; Last updated - 2017-04- 17; CODEN - SCTEAF. Dostupné z: <https://www.proquest.com/scholarly-journals/delightful-catenary-curve/docview/1888642340/se-2>.
- [8] MYERS, B. A. Visual Programming, Programming by Example, and Program Visualization: A Taxonomy. New York, NY, USA: Association for Computing Machinery. Duben 1986, sv. 17, č. 4, s. 59–66. DOI: 10.1145/22339.22349. ISSN 0736-6906. Dostupné z: <https://doi.org/10.1145/22339.22349>.
- [9] REIS, A. J. a PEDRO, J. J. O. *Bridge Design: Concepts and Analysis*. 1. vyd. Wiley, červen 2019. ISBN 978-0-470-84363-5.
- [10] STROBL, D. *Návrh betonového obloukového mostu s uvážením technologie výstavby*. Praha, CZ, 2018. Diplomová práce. České vysoké učení technické v Praze, Fakulta stavební. Dostupné z: <https://dspace.cvut.cz/handle/10467/74287>.

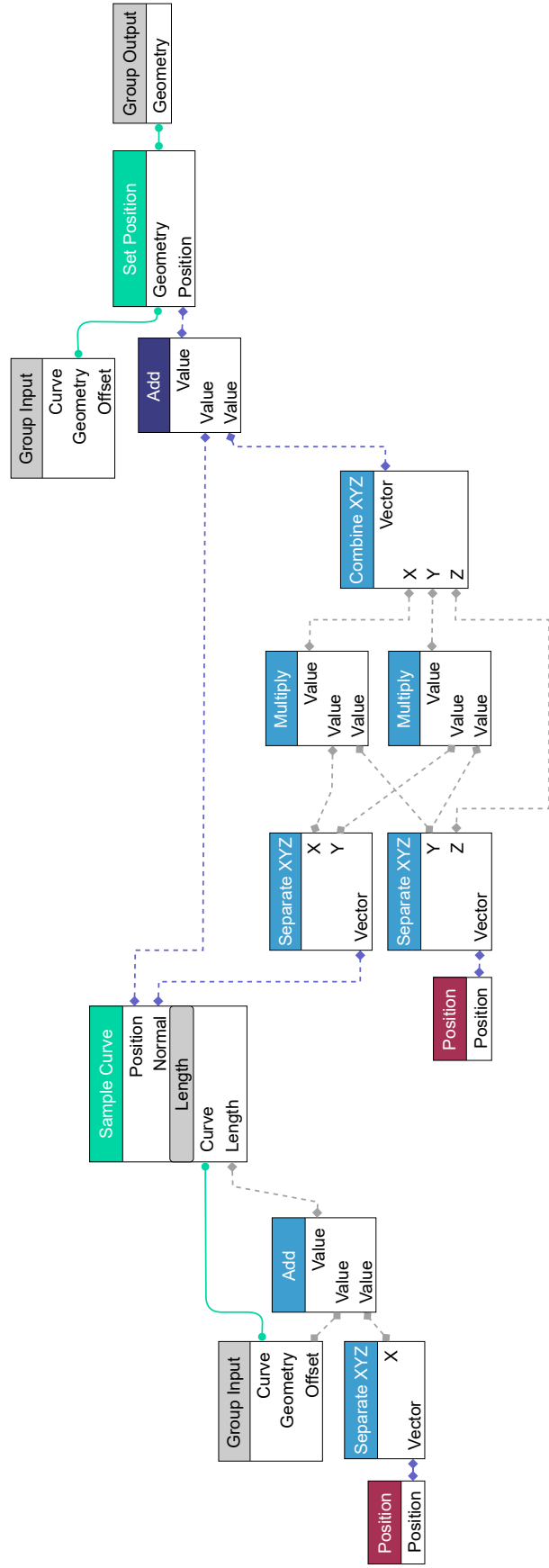
- [11] THE BLENDER FOUNDATION. *Glossary* [online]. [cit. 2023-04-16]. Dostupné z: <https://docs.blender.org/manual/en/latest/glossary/index.html>.
- [12] TROYANO, L. F. *Bridge Engineering: A Global Perspective*. 1. vyd. Thomas Telford, září 2003. ISBN 978-0727732156.
- [13] UDIIT, S. A., NAGARANI, S. a HARIHARAN, A. Catenary Curves – A case study. *International Journal of Management, Technology And Engineering*. 2018, s. 644–649. ISSN 2249-7455.

Příloha A

Implementační schémata



Obrázek A.1: Segment schématu výpočtu souřadnice Z při ohybu oblouků



Obrázek A.2: Schéma algoritmu ohybu objektu podle křivky

