



m o s t

martin klicpera

grasshopper

cad IV scripting
6. semestr 2008/09



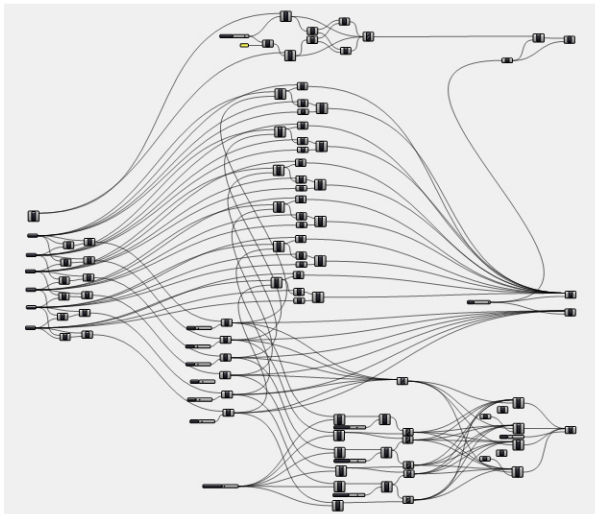
cíl práce

Minulý semestr jsem se naučil v předmětu CAD III pracovat s programem **Rhinceros**, rozhodl jsem se navázat a pokračovat dál a vybral jsem si zásuvný **modul Grasshopper**. Jelikož jsem s ním, ani se skriptováním, neměl dosud žádné zkušenosti, zadání mé semestrální práce není zasazené do kontextu a slouží k ovládnutí základních funkcí. Projekt měl **využívat podstatu parametrického modelování**, tzn. neměl být snadno vytvořitelný běžným modelováním v Rhinu, aby mělo použití Grasshopperu smysl.

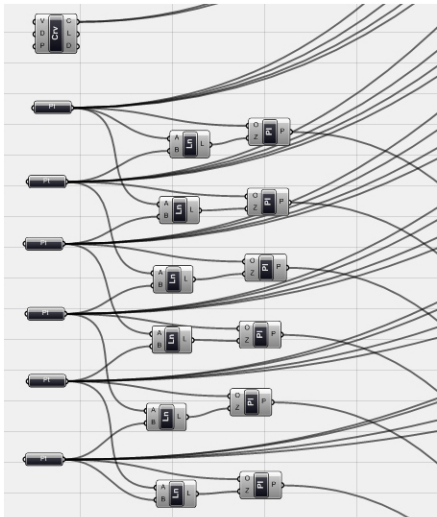
Obecně jsem si dal **za cíl vytvořit most** v podobě proděravěného tunelu, jehož plášť by byl abstraktní, tvar otvorů v podstatě **náhodný, ale říditelný a upravitelný** dle mých záměrů, a aby se dalo s tvarem objektu i jednotlivými prvky stále **hýbat a měnit je** z estetického i technického hlediska.

Po prvních pokusech jsem si stanovil konkrétní zadání - zdánlivý tvar mostu jako **tubus měnící průměr** s osou v podobě křivky, nosné obruče s přičlemy nesoucími mostovku a určujícími místa změny průměru tubusu, nepravidelně pospojované ohýbanými pruty opisujícími oblý tubusovitý tvar a zároveň nesoucí plachtové zastřešení konstrukce.

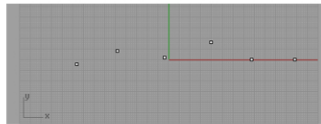
vytvořený skript



postup



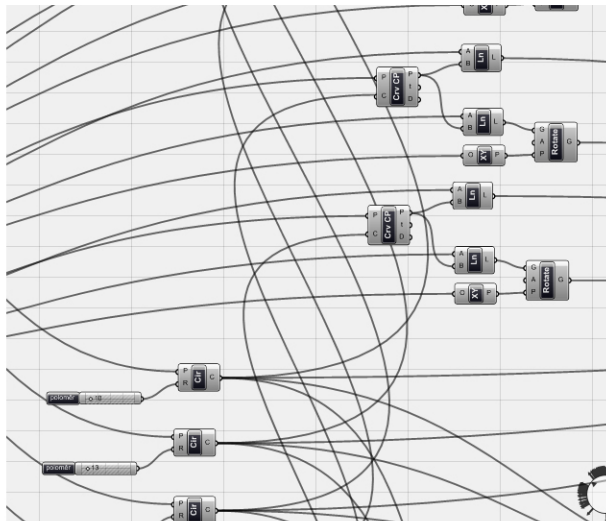
základem pro vytvoření **hrubého tvaru** mostu, tedy nepravidelného tubusu, a dalších prvků bylo 6 bodů (Pt), což byl jediný vstupní parametr vytvořený „ručně“ v Rhinu



body jsem použil jako **řídící body** pro vytvoření **křivky** (Crv), která bude později potřeba pro vytvoření mostovky

dalším krokem byla **konstrukce rovin** (Pl) zadaných pomocí bodu a normálového vektoru; v každém bodě byla tedy vytvořena rovina a jako normálový vektor sloužila **úsečka** (Ln): pro počáteční a koncový bod (+ rovinu) byla řídicí úsečka tvořena tímto bodem a jeho sousedním, pro vnitřní body byla tvořena nejbližším bodem před a za, čímž jsme získali vektory orientované rovnoběžně s tečnami křivky a roviny jsou tím pádem v bodech vždy kolmé na křivku

postup



spodní část obrázku:

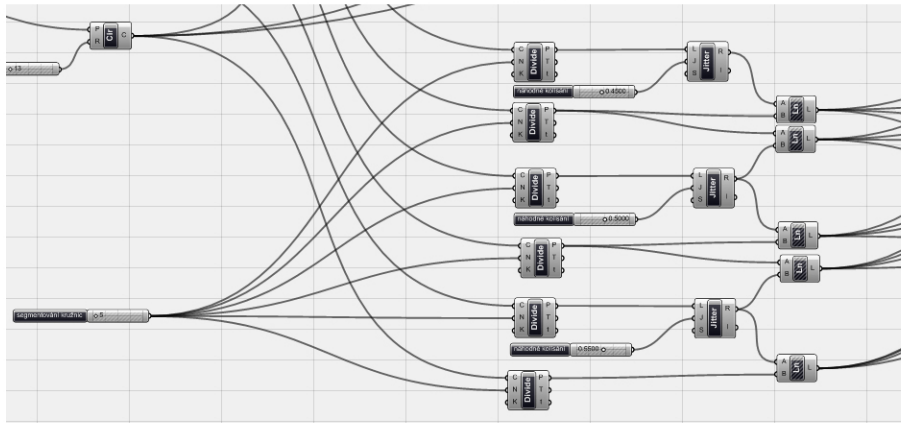
roviny kolmé na příčce v řídicích bodech jsem použil pro tvorbu **kružnic** (Civ), které jsou zadané rovinou, ve které leží, a poloměrem; pro zadání poloměru jsem použil Slider, díky němuž lze okamžitě **měnit průjezdný profil mostu** na každém místě zvlášť, tedy v místech řídicích bodů a tím pádem i nosných obručí, na kterých leží mostovka a konstrukce vozovky

horní část obrázku:

pomocí funkce Crv CP, jež slouží pro nalezení **nejbližšího bodu na křivce** od bodu zadaného, jsem vytvořil nosné příčce pro mostovku - vstupním parametrem je původní řídicí bod a příslušná kružnice mající střed v tomto bodě; funkci je třeba zadat pro každý bod (celkem 6x)

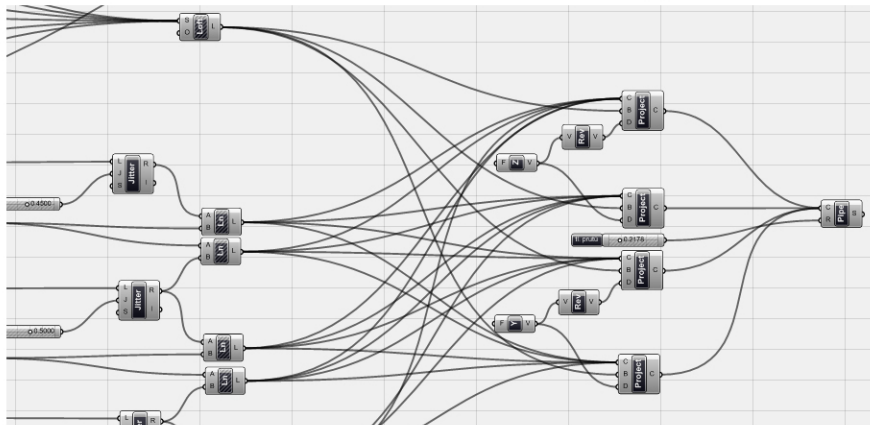
následovalo **vytvoření 1/2 příčce** pomocí funkce úsečka (Ln), která je zadána řídicím bodem a nalezeným nejbližším bodem, čímž v podstatě zobrazíme poloměr kružnice; 2. polovina příčce je vytvořena příkazem Rotate, jehož vstupním parametrem je první půlka příčce, úhel v radiánech ($=\pi$ pro 180°) a vodorovná rovina otáčení, pro kterou použijeme napojení funkce XY Plane a jako počátek roviny připojíme opět původní řídicí bod

postup



kružnice, ze kterých budou později vytvořeny nosné obruče, jsem nadělil pomocí funkce Divide - vstupním parametrem je kružnice a počet **segmentů**, který je zadán pomocí Slideru, díky němuž lze počet segmentů plynně regulovat; každou druhou segmentaci kružnic jsem ještě nechal projít funkcí Jitter, která zajistí **nestejně délky segmentů** - vstupním parametrem je kromě výstupu segmentů z funkce Divide ještě „intenzita promíchání hodnot“ (J), pro kterou jsem opět vytvořil zvláštní Slidery pro interaktivní změnu intenzity body mezi segmenty kružnic byly spojeny funkcí úsečka (Ln), čímž jsme vytvořili přímé **spojující pruhy** mezi obručemi, které následně ponесou krycí plachtu

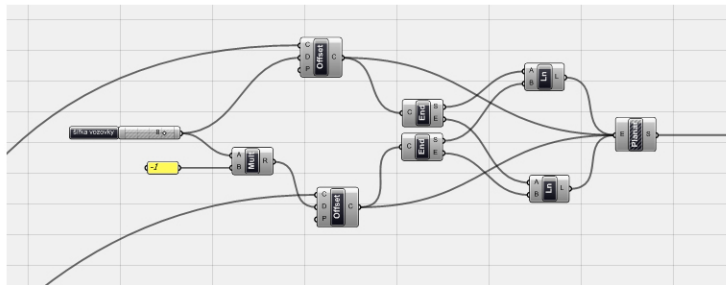
postup



pomocí všech kružnic zadaných do funkce Loft byl vytvořen pomocný **tubus**, tedy jakýsi hrubý objem a tvar konstrukce

všechny úsečky (spojovací pruty) byly, abychom dosáhli **zakřivení** do tvaru zdeformovaného tubusu, **promítnuty na plochu** tubusu (Loft) pomocí funkce Project, jejímiž vstupními parametry jsou promítané prvky, promítací plocha a směr promítání; funkce byla použita celkem 4x vzhledem k počtu směrů promítání (pro docílení žádoucího rozmístění prutů), které byly vloženy funkcí kladně orientovaného **vektoru** Z, záporně orientovaného vekt. Z (pomocí funkce Reverse) a taktéž obou orientací vektoru Y pomocí stejného principu promítnuté úsečky/pruty dostaly **tloušťku** funkcí Pipe, jejíž vstupní parametr pro poloměr trubice byl opět zadán Sliderem

postup



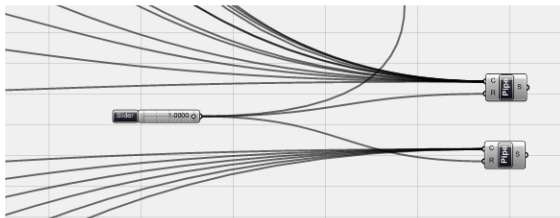
jedním z posledních prvků mostu je **vozovka**, jejíž středovou linii máme již zadanou od začátku křivkou s řídicími body

hranice vozovky jsem vytvořil pomocí funkce Offset použité dvakrát, jednou se vstupními parametry křivka a polovina šířky komunikace (zadáno Sliderem) a podruhé s křivkou a polovinou šířky kom. v záporné hodnotě (je potřeba propojit Slider s funkcí Mult, kde zadáme hodnotu B = -1)

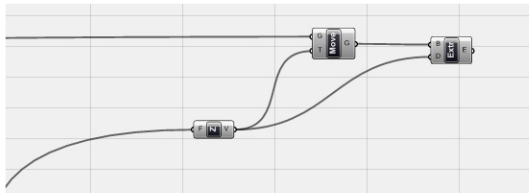
dalšími hraničními křivkami jsou **spojnice konců a počátků** obou okrajových křivek; konce i počátky nalezneme pomocí funkce End

vytvoření **plochy vozovky** docílíme funkcí Planar, kde zadáme jako vstupní parametr vytvořené hraniční úsečky počátků a konců (Ln) spolu s hraničními křivkami

postup



nosné **obruče** i středové **příčle** nemají tloušťku, proto ji vytvoříme opět funkcí Pipe stejně jako u spojovacích prutů, hodnota tloušťky zadána Sliderem



tloušťka vozovky a konstrukce pod ní bude stejná jako poloviční tl. příčlí, vytvoříme proto nejdříve vektor Z o velikosti poloměru obručí a příčlí (spojíme s příslušným Sliderem), poté vytvoříme funkci Move a přesuneme povrch vozovky po tomto vektoru, čímž ho dostaneme na příčle a dále tuto rovinnou plochu extrudujeme o poloměr obručí a příčlí - tedy použijeme funkci Extr a jako druhý parametr napojíme vektor Z

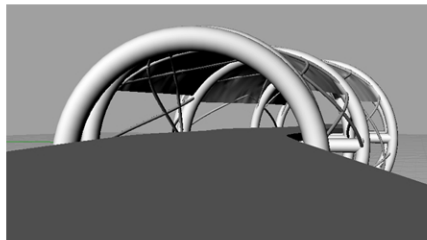
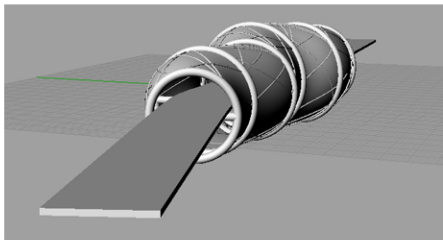
z á v ě r

Nutné opakování funkcí lze zjednodušit a zautomatizovat pomocí souboru funkcí Tree, bohužel na seznámení se s touto funkcí jsem neměl dost času. V projektu jsem nepoužil skript, protože s ním nemám zkušenosti.

Parametričnost návrhu je výhodná pro libovolnou změnu dimenzování a rozmístění prvků mostu a pro docílení požadovaného estetického vzhledu. Změn lze dosáhnout úpravou rozmístění základních 6 řídicích bodů (které lze v Rhinu snadno posouvat) a změn různých hodnot, které byly zadávány Slidery, tedy rozpětí, tloušťky apod.

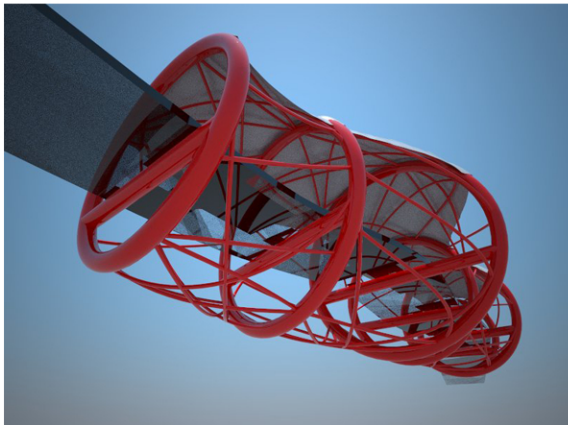
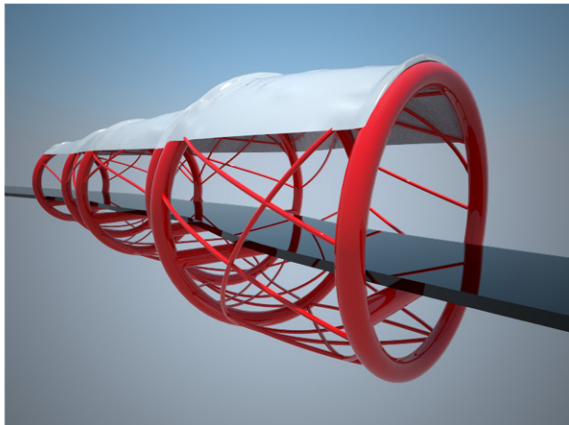
poznámka:

Pro prezentaci renderů jsem v Rhinu upravil umístění příčli a vozovky do požadovaného/zamýšleného stavu a přidal zastřešovací plachtu.





r e n d e r i n g



rendery

