Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

Diplomová práce

Vývoj aplikací v otevřeném prostředí CAx systémů

Plzeň, 2015 Roman Polák

Originál zadání

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 14. května 2015

Roman Polák

Abstract

Theme: Development of application in open interface of CAx systems.

This master thesis deals with the possibility of making own application in the best known CAx systems. The first part is dedicated to the general basic principles of mechanical engineering, which are used in design practice. Further this thesis analyses CAx systems and their possibilities of programming, closer the possibilities of programming tools in CAx system Siemens NX, which is used in research on Regional Technological Institute at University of West Bohemia. On the basis of this knowledge is developed application, which is designed to make work easier and faster in repetitive work processes.

Abstrakt

Tato diplomová práce se zabývá možnostmi tvorby vlastních aplikací v nejznámějších CAx systémech využívaných ve strojním inženýrství. V práci jsou uvedeny základní principy práce v těchto systémech, které se běžně využívají v konstrukční praxi. Dále práce blíže analyzuje nástroje pro tvorbu vlastních aplikací, včetně přizpůsobení ve vybraném CAx systému Siemens NX využívaném ve výzkumu v rámci Regionálního technologického instituty Fakulty strojní Západočeské univerzity v Plzni. Na základě těchto principů, s využitím zmíněných nástrojů, je navržena aplikace zjednodušující konstrukční procesy z hlediska celkové časové náročnosti.

Obsah

1 Úvod 1

2 Systémy CAx 2

2.1 CAD, CAE, CAM 2

2.2 Nejznámější produkty 3

2.3 Otevřenost CAx systémů 4

2.4 Obecný formát JT 4

3 Základy práce v CAD 6

3.1 Modelování 8

3.2 Sestavy 9

3.3 Tvorba výkresů 11

3.4 Výpočty 13

3.4.1 Postup v CAx 13

4 Nástroje pro programování v systému NX 14

4.1.1 NX Open 14

4.1.2 Block UI Styler 14

4.1.3 Menuscript 16

4.1.4 Žurnál (Journaling) 16

4.1.5 Ostatní nástroje 17

4.1.6 Licenční podmínky 19

5 Analýza a návrh řešení zadaného problému 20

5.1 Vstupní data 20

5.2 Možná řešení 21

5.2.1 Uživatelské řešení 21

5.2.2 Automatizované řešení 24

5.3 Praktické využití 24

6 Realizace 26

6.1 Vývojové prostředí 26

6.2 Analýza potřebných funkcí NX 27

6.3 Architektura aplikace 29

6.4 Grafické rozhraní 30
6.5 Ošetření chybových vstupů 31
7 Testování 32
7.1 Ověření funkcionality v systému NX 32
7.2 Zhodnocení automatizace 34
8 Uživatelská dokumentace 35
8.1 Minimální požadavky 35
8.2 Instalace programu 35
8.3 Spuštění programu 36
9 Závěr 39
10 Přehled zkratek 40
11 Seznamy 41
11.1 Seznam obrázků 41
11.2 Seznam tabulek 41
Použitá literatura 42
Přílohy 43

1 Úvod

Tato práce se zabývá možnostmi vývoje vlastních aplikací v CAx systémech využívaných ve strojírenství. CAx jsou strojírenské systémy určené pro konstrukci a vývoj produktů. V konstrukční praxi může docházet k opakovaným procesům, neboli sledu určitých funkcí, které je potřeba provádět stále dokola a jsou časově náročné a tím i finančně nákladné. Uživatel, v tomto případě konstruktér, který tyto funkce provádí, může sice dosáhnout lepších výsledků (vyšší rychlosti) při opakovaném používání z hlediska časové náročnosti, ale v případě častějších a rozsáhlejších procesů se vyplatí zvážit zavedení automatizace.

K automatizaci procesů v rámci CAx systémů mohou sloužit oddělené aplikace třetích stran, které pouze externě využívají data například z navržených modelů a dále je zpracovávají. Avšak výstup takovéto aplikace nemusí být vždy v plně kompatibilním formátu pro zpětný import do daného CAx systému. Model pak například neobsahuje veškeré původní informace a vlastnosti a jeho využití v původním CAx systému je tak degradováno na nižší úroveň.

Výhodnějším řešením je využití nabízených programovacích nástrojů přímo výrobcem softwaru CAx. Pro uživatele mohou být dostupná například základní makra, pomocí kterých lze provést základní zjednodušení práce bez větších znalostí programovacích jazyků a principů programování. Pokročilým způsobem automatizace je vývoj vlastních aplikací s využitím dostupných knihoven daného softwaru. Podmínkou tohoto řešení je nejen znalost daného podporovaného programovacího jazyka, ale klíčová je i orientace v problematice konstrukce a vývoje.

Cílem této práce je seznámení se základními principy a postupy v oblasti konstruování, využití těchto znalostí a navržení jejich automatizace včetně reálné ukázky. Zároveň by práce měla poukazovat na výhody a dostupnost automatizovaných řešení, které si může koncový zákazník vyvinout sám dle svých potřeb a přizpůsobit tak celý systém svým požadavkům.

V první fázi se práce zabývá objasněním pojmů v oblasti CAx systémů, které se často opakují a používají ve strojním inženýrství. Dále jsou uvedeny základní principy konstruování, které platí obecně pro všechny typy těchto systémů. V práci jsou uvedeny nejznámější CAx systémy a porovnání jejich programovacích nástrojů. Bližší specifikace programovacích nástrojů jsou uvedeny pouze pro systém Siemens NX, který je následně používán po zbytek práce včetně ukázek a praktické části.

2 Systémy CAx

Pojem CAx[1] zahrnuje širokou oblast počítačových systémů na podporu návrhu, analýzy a výroby produktů. Všechny tyto systémy jsou klíčové při řízení životního cyklu výrobku (PLM[2]). Při využití PLM dochází ke sjednocení podnikových systémů správy procesů a dat včetně zmíněných CAx nástrojů pro vývoj a tvoří tak komplexní celek informací od požadavku až po výsledný produkt.

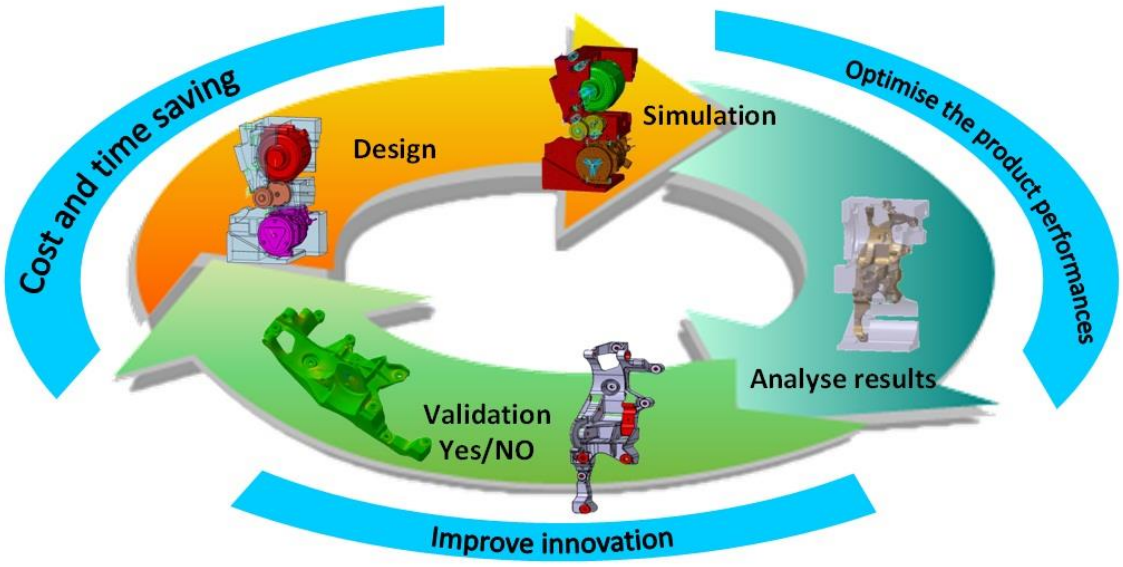
2.1 CAD, CAE, CAM

Mezi základní oblasti CAx systémů ve strojírenství patří CAD[3](Computer Aided Design), neboli počítačová podpora návrhu a kreslení. Software tohoto typu tak nahrazuje klasické rýsovací prkno a umožňuje navržené modely využít v dalších oblastech analytických výpočtů CAE[4](Computer Aided Engineering) nebo výroby CAM[5](Computer Aided Manufacturing).

V CAE nástrojích lze na základě CAD dat (modelu), v závislosti na daném softwaru, provádět pevnostní výpočty, analýzu tepelných vlastností, mechanické simulace a další. Tento typ nástrojů je tak nezbytnou součástí při vývoji produktů, jejich testováním s využití CAE simulací a následné optimalizaci. Proces vývoje produktu (viz obrázek 2-1) znázorňuje postup od návrhu konstrukce, přes simulaci s analýzou výsledků a následnou validaci, která případně pokračuje na výrobu CAM nebo se celý proces opakuje z důvodů optimalizace či inovace produktu.

Další fází konstruování je samotná výroba a její počítačová podpora obrábění. CAM software využívá CAD soubory a je schopen navrhnout dráhy pro obrábění včetně exportu instrukcí pro obráběcí stroj. Cílem této fáze je optimalizace času potřebného pro obrábění a také následná analýza povrchu po obrábění. Povrch součásti je závislý na použitém nástroji a jeho způsobu obrábění včetně navržených drah.

[1] CAx – Computer Aided - počítačová podpora
[2] PLM – Product Lifecycle Management – řízení životního cyklu
[3] CAD – Computer Aided Design – počítačová podpora návrhu
[4] CAE – Computer Aided Engineering – počítačová podpora inženýrských výpočtů
[5] CAM – Computer Aided Manufacturing – počítačová podpora výroby



Obrázek 2-1 – proces vývoje produktu CAD – CAE (Intech)

Velmi stručně lze proces konstrukce a závislost jednotlivých nástrojů shrnout jako návrh a tvorbu modelu - CAD, poté ověření požadovaných vlastností – CAE, případnou optimalizaci se zpětnou vazbou na CAD a následnou výrobu - CAM.

2.2 Nejznámější produkty

Z oblasti strojírenského 2D/3D softwaru patří mezi nejznámější CAx systémy následující produkty uvedené v Tabulce 2-1 včetně jejich knihoven neboli API[6] rozhraní s možností vývoje doplňků a aplikací. Při výběru produktů šlo zejména o porovnání podobných nástrojů označených jako CAx systémy používané v rámci Západočeské univerzity v Plzni nebo firemních prostředích a to od různých výrobců včetně dostupnosti rozhraní pro programátory.

Výrobce	Produkt	API rozhraní	API podpora
Autodesk[7]	AutoCAD	ObjectARX	VB.NET, C#, C++
Autodesk	Inventor	Inventor API	VB.NET, C#, C++
PTC	Creo	PTC Creo Toolkit	VB.NET, Java
Siemens PLM Software	SolidEdge	SolidEdge API	VB.NET, C#
Dassault Systemes	SolidWorks	SolidWorks API	VB.NET, C#, C++
Siemens PLM Software	NX	NX Open	VB.NET, C#, C/C++, Java, Python

[6] API – Application Programming Interface (rozhraní pro programování)
[7] Autodesk - (Autodesk company, 2015)

Výše uvedené systémy lze obecně nazvat CAx systémy protože obsahují kromě CAD i celou řadu dalších modulů (CAM, CAE, atd.) a aplikací třetích stran, které jejich nástroje a funkce doplňují.

Za zmínku stojí také aplikace poskytované jako freeware verze. Například prohlížeč JT (viz 2.4 Obecný formát JT) souborů pro jednoduché zobrazení geometrie na různých platformách (viz kapitola 2.4 Obecný formát JT) nebo parametrický 3D modelář FreeCAD (FreeCad software) vyvíjený pod Open Source licencí LGPL[8] , který obsahuje podobné funkce jako komerční software a je k dispozici pro platformy Windows, Mac OSX a Linux.

2.3 Otevřenost CAx systémů

Při porovnání jednotlivých programů a jejich možnosti implementace vlastních aplikací, případně dalších uživatelských zásahů je rozdíl následující. U všech zmíněných CAx produktů existují API rozhraní umožňující doprogramovat vlastní aplikace. Dostupnost těchto knihoven se liší zejména počtem podporovaných programovacích jazyků, licenčními podmínkami, za kterých je možné API využívat či distribuovat a rozsahem API knihoven ve smyslu jak prostřednictvím nich můžeme přistupovat k funkcím CAx systému.

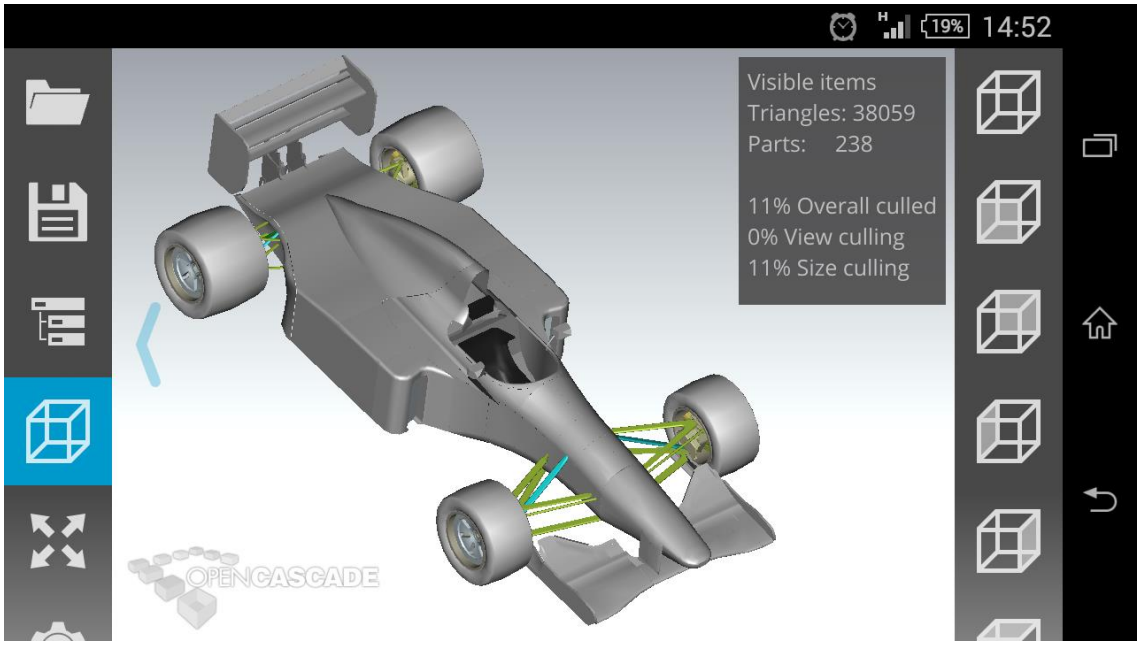
2.4 Obecný formát JT

Jedná se 3D datový formát vyvinutý společností UGS[9], nyní Siemens PLM Software, určený k vizualizaci dat, spolupráci a datovou výměnu s ostatními CAD softwary. Původně byl tento formát vytvořen jako alternativní velikostně zmenšená verze CAD souborů. Později společnost UGS uveřejnila zdrojové kódy tohoto formátu, což vedlo k dalšímu vývoji a také rozšíření JT jako standardu právě pro komunikace a další oblasti, kde je tento typ dat dostačující. JT soubor může být využitý například pro rychlé náhledy nebo sdílení, protože velikost souboru JT se pohybuje do 10% velikosti původních CAD dat. V případě, že je potřeba jen prohlížet, je JT soubor ideálním řešením, zejména pokud se jedná o rozsáhlá CAD data. Dalším způsobem využití je

[8] LGPL – Lesser General Public License (licence svobodného softwaru)
[9] UGS – Unigraphics Software – společnost, která je od roku 2007 součástí Siemens PLM

firemní prostředí, které využívá tzv. multiCAD (více CAD aplikací najednou) a je tak potřeba v některých případech zajistit kompatibilitu mezi těmito aplikacemi.

JT formát je vhodný k vizualizaci dat, kdy je například potřeba prezentovat ukázku na mobilním zařízení (telefon, tablet) viz Obrázek 2-2 v programu Open Cascade JT Assistant, který je jako jeden z mnoha dostupných zdarma pro zařízení s operačním systémem Android. V ukázce na Obrázku 2-2 je použita sestava (blíže popsána v kapitole 3.2 Sestavy), která byla převedena pomocí produktu Siemens NX do obecného formátu JT.



Obrázek 2-2 Sestava ve formátu JT v programu JT Assistant

3 Základy práce v CAD

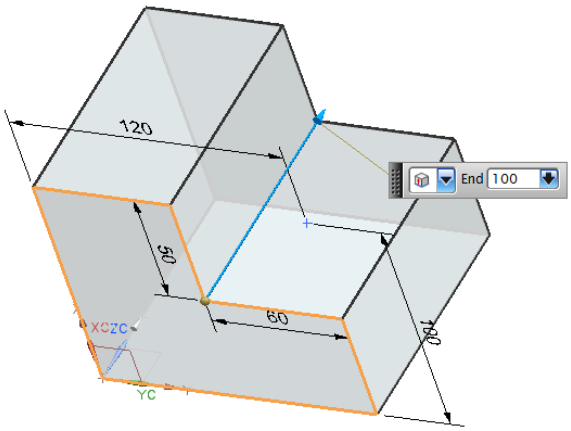
Obecně se CAD programy rozdělují na 2D a 3D s tím, že modelování ve většině dnes používaných programů probíhá ve 3D, z kterého se následně vytváří 2D výkresy. Základní metody pro práci v CAD je možné rozdělit dvou kategorií:

- tvorba pomocí základních 3D těles a operací,
- 2D skica v rovině + vytažení, rotace nebo tažení po křivce.

Mezi základní tělesa patří například kvádr, koule nebo válec, s jejichž využitím se pak provádí například tyto operace:

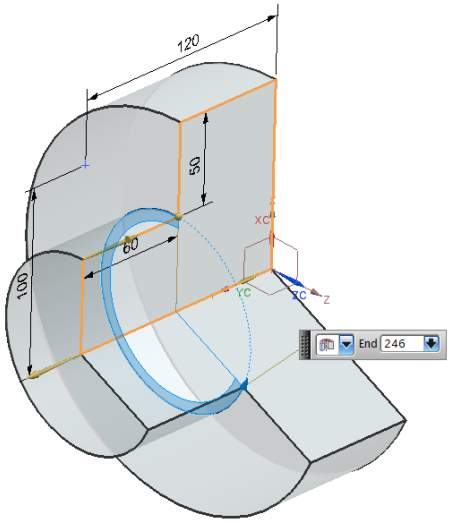
- s 3D objekty lze v případě jejich průniku provést tzv. sčítání a odečítání. Prakticky to znamená, že je možné jedním objektem vyříznout druhý objekt nebo naopak ponechat pouze jejich průnik a zbylé části odstranit.
- Další možností jsou úpravy a přizpůsobení základních těles, například sražení hran, jejich zaoblení, či vytvoření děr včetně závitů.

Druhým způsobem a metodou tvorby 3D modelů je vytvoření 2D profilu ve vybrané rovině. Tato rovina se může nacházet v počátku souřadnic, ale také na libovolném místě v již existujícím modelu. Ve vybrané rovině je možné vytvořit návrh neboli skicu pomocí 2D prvků (bod, čára, křivka, polygon atd.) a po návratu do 3D modeláře využít tohoto připraveného návrhu a pomocí funkce vytažení (viz obrázek 3-1) definovat body a křivky z připraveného návrhu, ale i existujících hran v modelu, určit vektor (na obrázku 3-1 označeno modrou šipkou) jako směr vytažení a nastavit omezení vzdálenosti vytažení případně ukončení k jinému objektu.

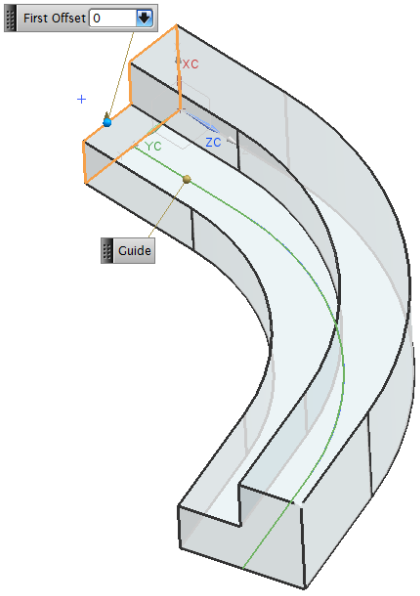


Obrázek 3-1 Vytažení

Ze stejného 2D náčrtu z předchozího příkladu je možné provést místo vytažení také rotaci (viz obrázek 3-2) nebo tažení po křivce (viz obrázek 3-3). U rotace je potřeba definovat osu rotace a úhel jako rozsah rotace, kterým vznikne 3D objekt. Pro tažení po křivce



Obrázek 3-2 Rotace



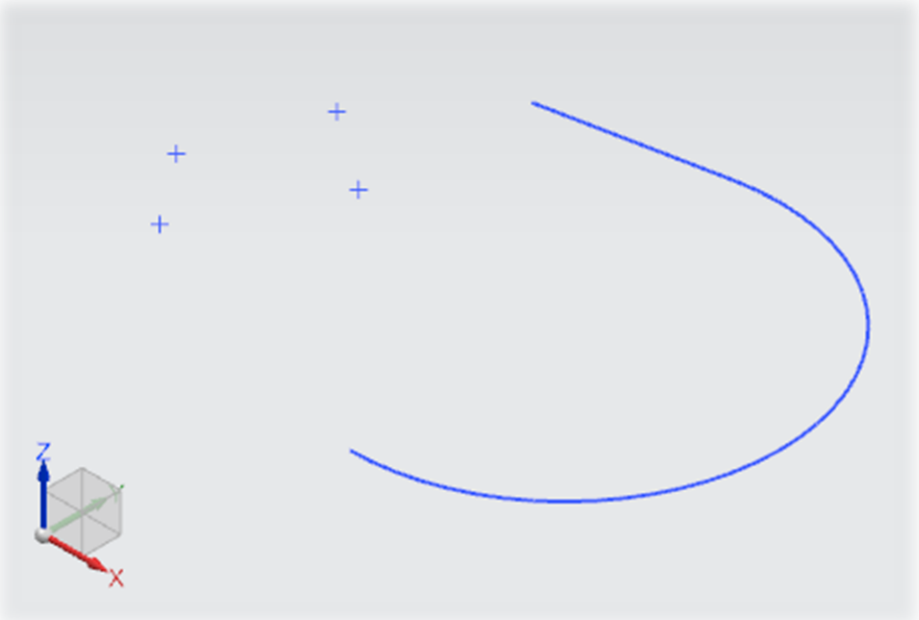
Obrázek 3-3 Tažení po křivce

Na základě jednoduchých 2D náčrtů se tímto způsobem vytváří 3D objekty. Například u obrázku 3-2 při rotaci 360° vznikne základ pro hřídel či přírubu. Dále je možné pokračovat ve 3D návrhu (zkosení hran, vrtání děr včetně závitů atd.) nebo provést úpravy ve 2D a po aktualizaci modelu se změny promítnou i do 3D.

3.1 Modelování

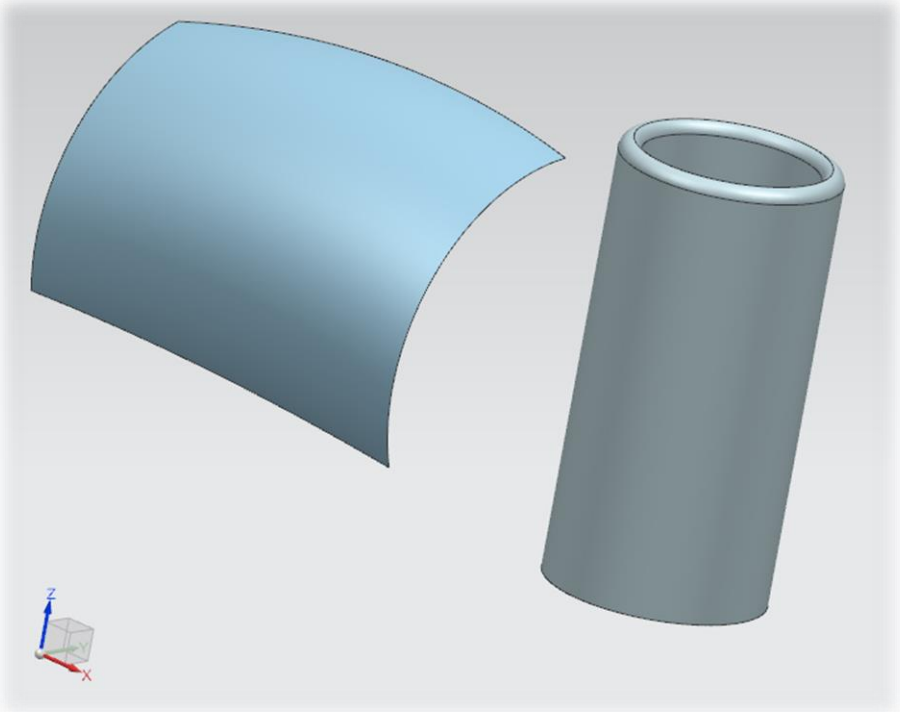
Samotné modelování se dělí na základní odlišné způsoby přístupu neparametrické a parametrické. Programy, které se označují neparametrickým modelováním, by se daly popsat statickým modelem s pevně danými vlastnostmi, kde změny v modelech znamenají úpravy s vazbou na okolí. Jinak řečeno je potřeba modelovat s přesně danými rozměry a není možné vytvářet automatické vazby mezi prvky v modelu. Typickým představitelem toho typu je například AutoCAD do verze 2010. Na rozdíl od toho parametrické modelování nabízí možnosti různých vazeb (kolmost, tečnost, rovnoběžnost, úhel atd.) a modelování v určitém měřítku (pouze geometrie) s pozdějším stanovením přesných rozměrů. Následné úpravy jedné části modelu se dle vazeb promítnou do zbylé části, kterou není potřeba ručně upravovat jako u neparametrického modelování. Zároveň existuje obousměrná vazba mezi výkresem a modelem. Změny tak lze provádět na obou místech v modelu či výkresu.

Třetím typem, využitelným s oběma typy modelování (parametrickým i neparametrickým), je modelování synchronní technologií. Obsahuje stejné funkce jako parametrické modelování, ale bez využití stromu s historií. Tímto způsobem je možné využít data z jiných CAD systémů, tzv. mrtvá data, právě bez historie modelování. Úpravy na modelu synchronní technologií probíhají pouze úpravou konkrétních prvků bez znalosti předchozí logiky.



Obrázek 3-4 Body a křivka

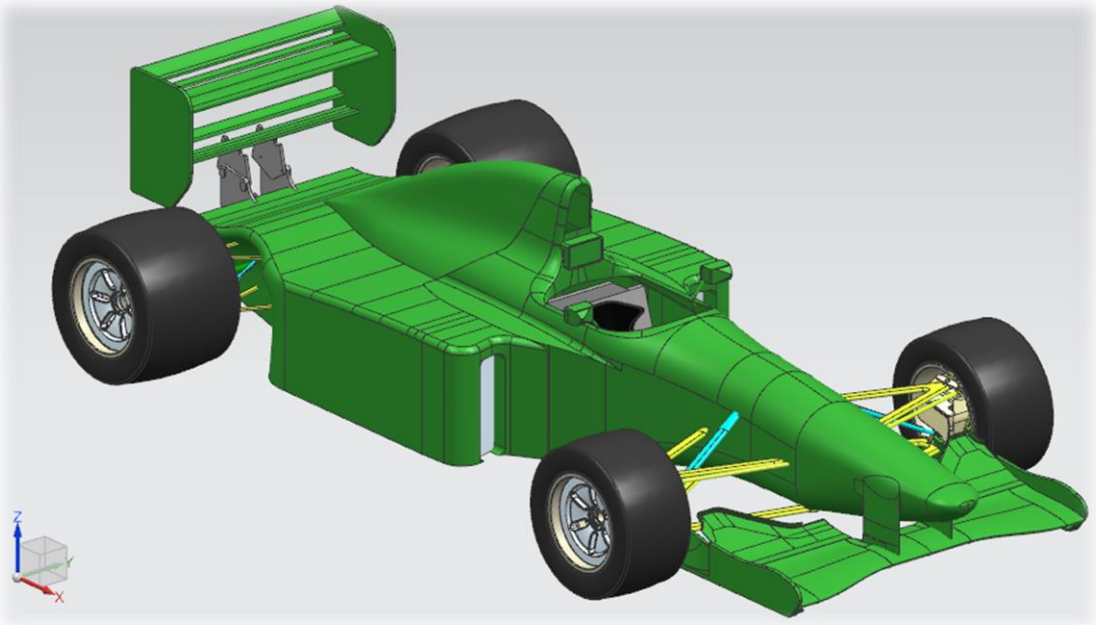
Níže jsou uvedeny základní stavební prvky modelování. Ukázky jsou vytvořeny v systému Siemens NX. Body a křivky je možné vytvářet přímo na daných souřadnicích nebo v závislosti na ostatní geometrii.



Obrázek 3-5 Plocha a těleso

3.2 Sestavy

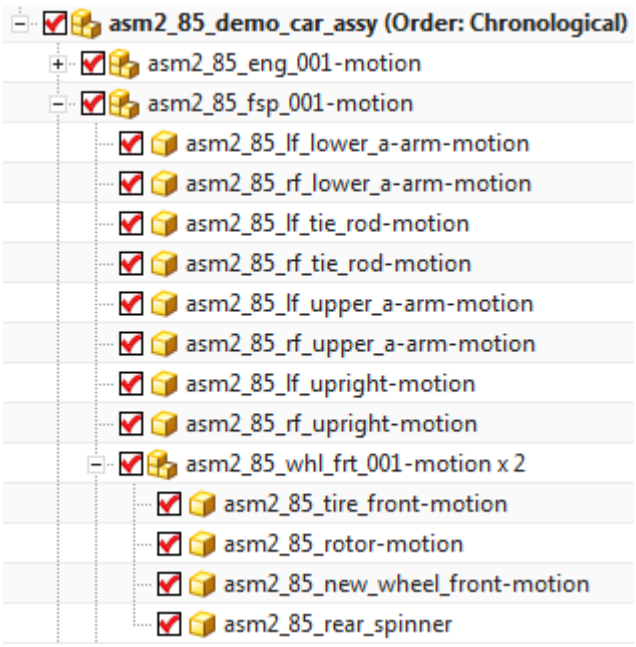
Předem zmíněné modely (kapitola 3.1 Modelování) se skládají do celků neboli sestav. Přidáním modelů do sestavy se zároveň definují různé vazby mezi díly v sestavě a vytváří se tak vztah vůči okolí. Skládáním dílů do sestav je možné graficky vizualizovat konečný výrobek, zjišťovat kolize, definovat sekvence pro zobrazení pohybů pro složení a rozložení sestav či vytvářet výkresy všech nebo jen vybraných komponent. V jedné sestavě může být více komponent, které mají stejnou geometrii (stejný zdrojový soubor) a liší se pozicí v sestavě. Pro představu například v sestavě, kde jsou využité šroubové spoje, a ve všech případech je spoj stejný, lze využít jedné geometrie šroubu a v sestavě se jen na tuto geometrii odkazovat. Využití komponent je viditelné na příkladu níže (Obrázek 3-6 Sestava), kde je jedna geometrie použita vícekrát, jako jsou přední a zadní kola včetně zavěšení, brzdových kotoučů a dalších prvků.



Obrázek 3-6 Sestava (Siemens Product Lifecycle Management Software, 2014)

Tyto sestavy jsou součástí modeláře a tvoří tak celkovou geometrii. Na to pak dále navazují CAE výpočty, CAM obrábění nebo tvorba výkresů, která je blíže popsána v následující kapitole (3.8 Příklad výkresu).

Navigace v sestavě je možná pomocí stromové struktury (viz Obrázek 3-7 Částečný strom s prvky sestavy), jednotlivé prvky tak přeřazovat nebo jen vypnout jejich zobrazení ve 3D modelu pro lepší orientaci. Zároveň lze také vytvářet podsestavy zanořením tohoto stromu, jako je tomu třeba pro kolo formule (asm2_85_whl_frt_001-motion). Podsestava kola obsahuje pneumatiku, disk, brzdový kotouč se středem a ložiskem.



Obrázek 3-7 Částečný strom s prvky sestavy

Soubor se sestavou se označuje jako „Assembly“, který obsahuje „Part files“ s jednotlivými geometriemi dílů. V návrhu sestavy je možné přímo editovat komponenty se zachováním vazeb v sestavě.

3.3 Tvorba výkresů

Výkresy, které slouží jako dokumentace a zadání k výrobě je možné vytvářet přímo kreslením 2D výkresu. Dnes v naprosté většině začínají postupy konstrukce u 3D návrhu, ze kterého se následně generují výkresy s různými pohledy. Vzniká tak závislost mezi výkresem a modelem, kdy primárně vychází výkres z modelu, ale může tomu být i naopak. Tento opačný postup se nazývá reverzní inženýrství (RE[10]) neboli zpětná analýza. RE nemusí vycházet přímo z modelu, ale je také možné využít reálného objektu, u kterého se naskenují body a zpracují se do CAD programu.

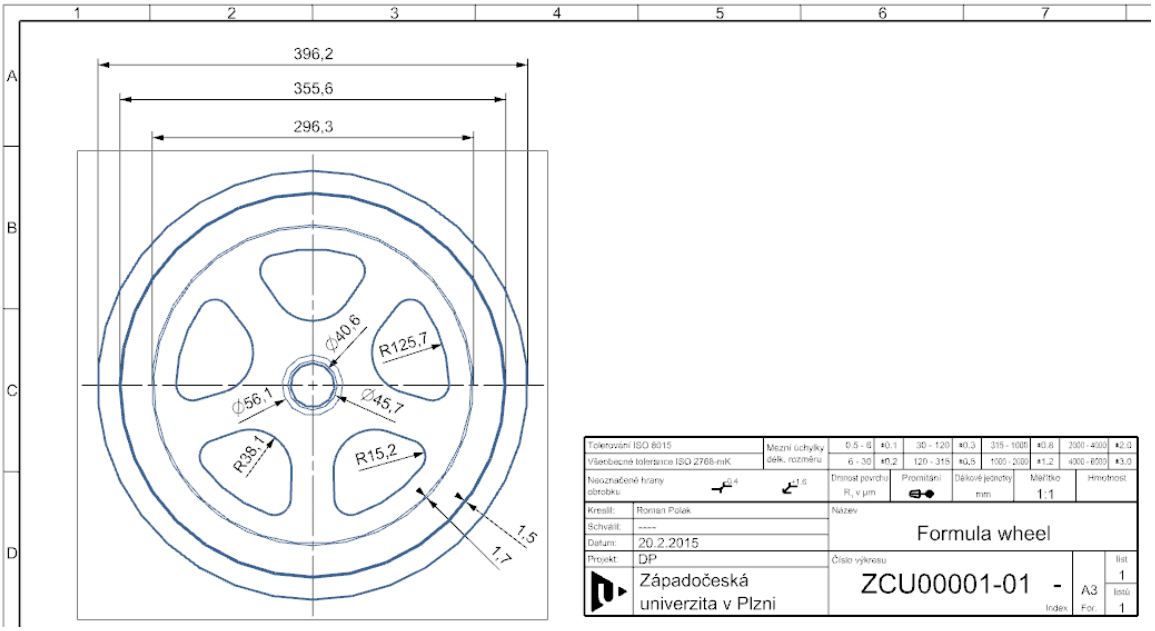
[10] RE – Reverse Engineering

Standardním postupem vytvoření výkresu je návrh 3D modelu, ze kterého se do výkresu vkládají různé pohledy vybraného dílu. Při tvorbě výkresů je možné vybrat rozložení pohledů a uspořádat tak celkový výstup. Ostatní části jako razítko včetně všech informací (název, datum, autor …), tloušťky čar atd. bývají přednastavené podle pěvně daných firemních předpisů.

Možnosti výkresových pohledů:

- Základní – dle evropského promítání normy ISO-E[11]
- Promítnutí – ze základních pohledů další pomocné promítnutí
- Detailní – z již existujících pohledů vytvoření detailu ve větším měřítku
- Řez součásti – umožňuje několik možností řezu (natočení, řez body, 3D řez atd.)
- Přerušený – horizontální či vertikální přerušení součásti

Na obrázku 3-8 Příklad výkresu je použitý předchozí model formule, konkrétně díl kola a jeho základní pohled včetně okótovaných rozměrů. Vzhledem k rozměrům byl použit pouze jeden pohled k ilustraci popisovaných možností při tvorbě výkresů.



Obrázek 3-8 Příklad výkresu

[11] ISO-E – evropská norma promítání obsahující základní pohledy a způsob jejich uspořádání

3.4 Výpočty

Obecně se modul výpočtů označuje jako CAE, který může existovat samostatně, ale ve většině případů je součástí komplexních CAx systémů. Hlavními prvky jsou metody pro různé analytické výpočty (pevnostní, tepelné …) a jejich simulace. Pomocí těchto výpočtů včetně simulací lze sledovat chování produktů v definovaném virtuálním prostředí, na základě výsledků dále optimalizovat produkt a v celkovém součtu tak snížit potřebné náklady na vývoj a výrobu.

Jako hlavní metodu pro výpočet inženýrských úloh je potřeba zmínit základní metodu FEM[12], v českém jazyce používaná také zkratka MKP[13]. Jedná se o metodu konečných prvků, která je numerickou metodou pro řešení teplotních, strukturálních, fluidních či elektromagnetických simulací. Principem metody je rozdělení počítaného objektu na konečný počet prvků, které společné tvoří síť a počítané parametry se určují právě v těchto prvcích. Výpočet probíhá na základě sestavených rovnic bodů a jejich posuvů v souřadném systému. Přesnost počítaného objektu je závislá na hustotě vytvořené sítě bodů. Ve většině případů je možné říci, že čím bude síť jemnější, tím bude výpočet časově náročnější, ale s přesnějším výsledkem. Nemusí to však platit u všech úloh, protože bude také záviset na použití typu sítě vzhledem k modelu a prováděné simulaci.

3.4.1 Postup v CAx

V komplexním CAx systému, který obsahuje modul CAD i CAE je možné řešit tyto výpočty s následujícím obecným postupem. Základem pro úlohu je vytvořený CAD model, který slouží jako vstupní data pro CAE. Na tomto modelu je potřeba provést tzv. „diskretizaci“, neboli nahrazení objemu konečným počtem prvků. K těmto účelům slouží v CAx systémech různé funkce dle typu modelu, pomocí kterých lze vytvořit síť podle modelu. Lze vytvářet 1D sítě pro maximálně zjednodušené modely nebo 2D a 3D sítě pro výpočtové modely s vyšší přesností. Dále se nastaví fyzikální vlastnosti modelu, jako je například použitý materiál konstrukce, či tloušťka a typ profilu, ze kterých je konstrukce složena.

Pokud je model včetně sítě a nastavenými vlastnostmi připravený, je možné vytvořit simulaci, kde je potřeba určit okrajové podmínky. Mezi tyto podmínky patří objekt k simulaci (model) s odebranými stupni volnosti a definicí působících sil. Po zadání těhto parametrů je možné spustit výpočet. Výsledky výpočtu jsou v textové podobě a následně jsou zobrazeny v grafické podobě a obsahují také animaci působících sil na definovaný objekt.

[12] FEM – Finite element method, FEM=MKP
[13] MKP – Metoda konečných prvků

4 Nástroje pro programování v systému NX

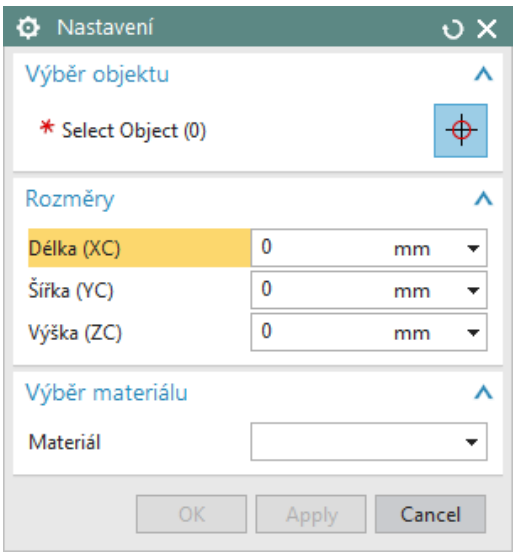
V systému NX existuje řada nástrojů pro programátory nebo také uživatele k vytváření vlastních aplikací a automatizaci prováděných operací včetně tvorby grafického uživatelského rozhraní. Níže jsou uvedeny všechny nástroje, které jsou k systému NX k dispozici.

4.1.1 NX Open

NX Open je sada knihoven (někdy také nazývaná jako NX API) s otevřenou architekturou pro vytváření vlastních aplikací s využitím známých programovacích jazyků C/C++, Visual Basic, C#, Java a nově Python od verze NX 10 vydané koncem roku 2014. Ke každému z těchto programovacích jazyků existuje programátorská příručka jako součást dokumentace NX. V závislosti na programovacím jazyce jsou k dispozici referenční programátorské příručky přímo ve webovém rozhraní základní dokumentace. Pro dokumentaci k javě existuje kolekce vlastních webových stránek a příručka pro .NET API je k dispozici ve formátu Microsoft Compiled Help (*.chm).

4.1.2 Block UI Styler

Nástroj Block UI Styler je součástí aplikace NX a slouží k vytváření dialogových oken pro vlastní aplikace se stejným vzhledem, který je použit v celém systému NX. Pro tvorbu vlastních uživatelských rozhraní slouží interaktivní editor se stromovou strukturou, pomocí kterého lze vytvářet dialogová okna (viz Obrázek 4-1 Block UI Styler - příklad). Příklad znázorňuje jednoduché uživatelské rozhraní s výběrem objektu v modelu, úpravou jeho rozměrů a změnou materiálu.

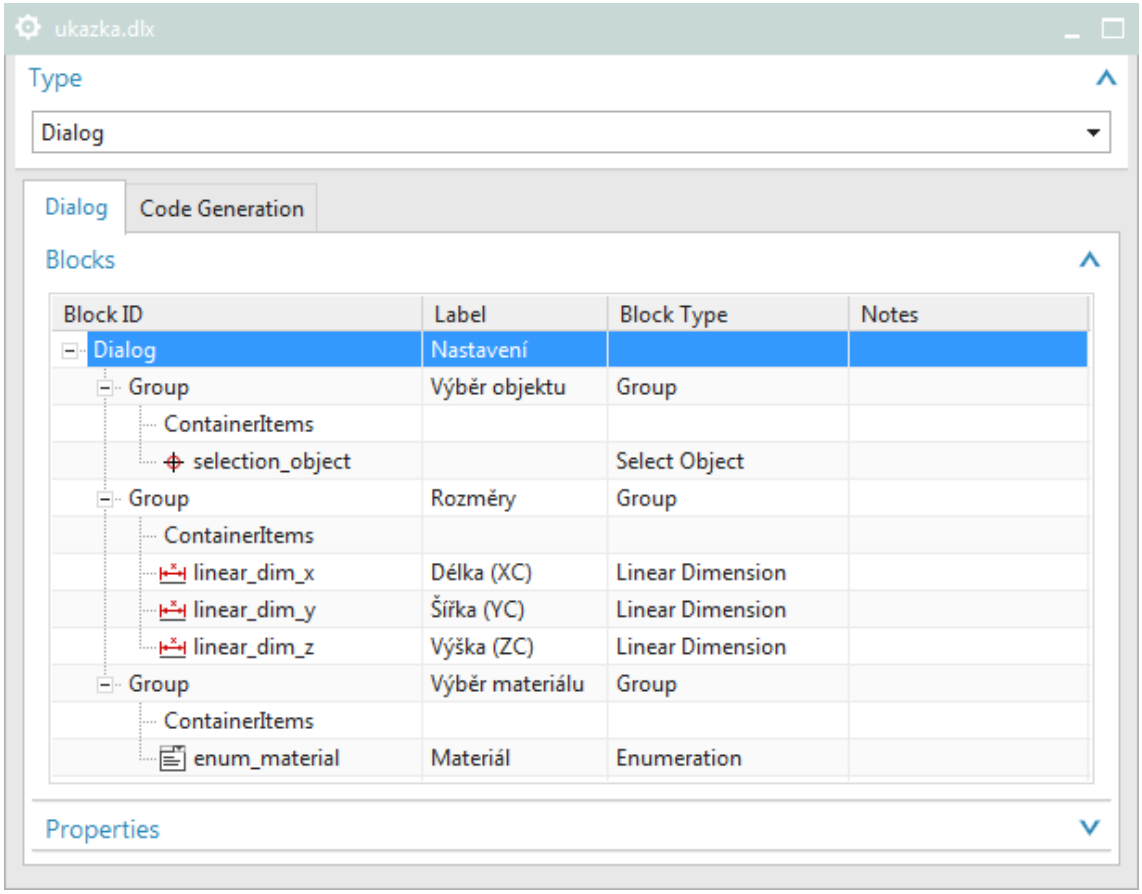


Obrázek 4-1 Block UI Styler - příklad

Tvorba uživatelského rozhraní probíhá výběrem ze seznamu bloků a skládáním do stromové struktury (viz Obrázek 4-2 Block UI Styler – struktura). Prvky jsou rozděleny do několika kategorií dle typu:

- Základní – popisky, textové vstupy, tlačítka atd.
- Číselné – vstupy v různých číselných formátech (Integer, Double, Radius, …)
- Výběrové – uživatelský výběr prvku přímo v modelu (bod, křivka, objekt, …)
- Organizační – zařazení bloků do skupin, tabulek, rolovacích oken atd.
- Speciální – otočení vektoru, vstup ze souboru, podmíněný výběr atd.

V příkladu jsou uvedeny pouze základní prvky s popisem pro grafickou ukázku bez dalších nastavení.



Obrázek 4-2 Block UI Styler - struktura

Vlastnosti jednotlivých bloků umožňují další nastavení, jako je například omezení rozsahu vstupních dat, a lze tak ve většině případů ošetřit vstupy na úrovni tvorby uživatelského rozhraní. Výstupem nástroje Block UI Styler je automaticky generovaný skript ve zvoleném jazyce (C#, C++, VB.NET, Java), který obsahuje již připravené funkce k blokům uživatelského rozhraní. Druhým výstupem je datový soubor *.dlx ve formátu

XML. Předchozí skript využívá tohoto XML souboru a načítá z něj bloky včetně všech nastavení.

4.1.3 Menuscript

Celým názvem NX Open Menuscript umožňuje úpravu menu v systému NX, vytváření menu pro vlastní aplikace, případně skrytí nepotřebných položek. Základní úpravy menu zvládne běžný uživatel dle dokumentace, která je součástí systému.

Také přímo v prostředí NX existuje možnost úpravy menu, ikon a dalších příkazů. Z uživatelského pohledu je tak možné přizpůsobit prostředí daným potřebám. Nicméně tímto způsobem nelze editovat všechny nabídky, jako to umožňuje menuscript. V základní instalaci jsou k dispozici tzv. role předdefinované pro různé oblasti CAx. Z těchto základních rolí lze zdědit všechny vlastnosti a vytvořit si tak kopii s vlastním nastavením. Pro potřeby vlastních aplikací je v tomto nástroji možné vytvoření vlastních menu, záložek a ikon. K samotným příkazům se přidá akce ke spuštění existující funkce nebo spuštění vlastního programu např. pomocí NX Open.

Konfigurační soubory jsou rozdělené podle obsahu do několika kategorií, které lze odlišit podle přípony souborů.

Přípona	Typ UI[14] objektu
.tbr	Toolbar
.grb	Group
.gly	Gallery
.ddb	Drop-down
.rtb	Ribbon tab
.csb	Cascade
.abr	Attachment

Tabulka 4-1 Typy konfiguračních souborů menuscript

4.1.4 Žurnál (Journaling)

NX Journal je nástroj umožňující zaznamenávat prováděné operace a zpětně tyto kroky znovu provést na základě uloženého skriptu. Funkce Journaling jsou umístěné přímo v prostředí NX jako volitelné menu. Tímto způsobem si tak může uživatel bez znalosti programovacího jazyka nahrát prováděné operace a v případě opakovaných postupů spouštět tento záznam.

Journaling využívá k záznamu jazyka Visual Basic .NET s napojením knihovny NX Open a je možné ho editovat přímo v základním textovém editoru systému NX.

[14] UI – User Interface (uživatelské rozhraní)

Při zahájení záznamu je u každého příkazu zobrazena informace o způsobu jakým bude použitá funkce vytvořena přímo ve zdrojovém kódu VB.NET. Příkazy mají ve většině případů plnou podporu žurnálu (viz Obrázek 4-3 Full Journal Support), kde jsou zaznamenány všechny vlastnosti příkazu. U některých příkazů je zobrazené označení částečné podpory (viz Obrázek 4-4 Partial Journal Support). V tomto případě je potřeba počítat s omezeným záznamem a jeho funkcionalitu lze ověřit spuštěním nahraného záznamu, případně ověřit uložené funkce včetně atributů přímo v editoru žurnálu.

Obrázek 4-3 Full Journal Support

Obrázek 4-4 Partial Journal Support

4.1.5 Ostatní nástroje

Dále jsou uvedeny nástroje méně využitelné pro vytváření vlastních komplexních aplikací zejména z důvodů, že neumožňují využít veškeré funkce, které nabízí NX Open nebo jsou zastaralé a neexistuje k nim adekvátní podpora.

4.1.5.1 SNAP

SNAP[15] patří k dalším nástrojům, které se vyznačují jako snadné programovací nástroje a jednoduché k nastudování pro běžné uživatele NX. Slouží k automatizování jednoduchých operací, kde je například potřeba dokola opakovat jednoduché operace pro více prvků v modelu a nekladou se větší požadavky na rozsah a větší funkcionalitu vytvářeného programu. Nástroj je založen na programovacím jazyce Visual Basic (VB.Net). V kombinaci se SNAPem je možné využít záznamu pomocí žurnálu (viz 4.1.4 Journaling) a editace ve standardním IDE Visual Studio.

[15] SNAP – Simple NX Application Programming

Obrázek 4-5 Fotografie klíče

Obrázek 4-6 Převod fotografie na body

Příklad vytvořený na základě dokumentace k SNAP ukazuje jednoduchý převod bitmapy na model. Zdrojovým souborem je Obrázek 4-5 Fotografie klíče, který se podle odstínu jednotlivých pixelů převede na body 3D modelu (viz Obrázek 4-6 Převod fotografie na body) v jedné rovině. V detailním náhledu konce klíče jsou viditelné jednotlivé body. Algoritmus prochází všechny pixely a vyhodnocuje jejich světlost na stupnici 0 až 1, kde 0 znamená nejtmavší pixel. V případě, že světlost pixelu je menší než zadaná hranice, tak proběhne vykreslení bodu. V opačném případě se pokračuje na další pixel a stejně tak se posouvá ukazatel v kartézské soustavě souřadnic systému NX. Pro převod fotografie byla použita hranice 0,6. Tento konkrétní příklad by bylo možné použít jako převod z výkresu nebo fyzické součástky na digitální 3D model. Fotografií a převodem se zachytí základní geometrie a následně je možné využít koncové body, definovat jejich vzdálenosti a pokračovat dále v úpravě modelu.

4.1.5.2 GRIP

Programovací jazyk GRIP[16], který se v mnoha směrech podobá jazykům BASIC a FORTRAN, je k dispozici jako další nástroj pro automatizaci v systému NX. K programování využívá anglická slova jako klíčové příkazy. Vzhledem k tomu, že již několik let neprobíhá vývoj a není k dispozici podpora, je tento nástroj zastaralý, ale stále využitelný a to i v nové verzi NX 10.

4.1.6 Licenční podmínky

Licencování produktu NX je rozdělené do stovek modulů, které například umožňují spuštění aplikace, vytvoření modelu, výkresu nebo jen využití nějaké funkce případně skupiny funkcí. Aby licenci nebylo potřeba sestavovat z takto malých částí, tak existují licenční balíky, které se skládají z těchto modulů. Obecně platí, že využití modulů ve vlastní aplikaci je stejné, které umožňuje daná licence. Nelze tak například se základní licencí pro CAD doprogramovat aplikaci využívající knihovní funkce CAM.

Moduly pro licencování vývojářských nástrojů NX:

- NX Journal – součástí základní licence
- SNAP (nx_snap_author)
- NX Open (dotnet_author)
- Knowledge Fusion (ug_kf_author)
- GRIP (grip_development)

Výše uvedené moduly, kromě NX Journal, jsou přídavné placené licence určené k vývoji vlastních aplikací. Umožňují spuštění těchto aplikací využívající knihovní funkce NX. Dále je možné vlastní zkompilované aplikace podepsat touto licencí a poskytnout tak aplikaci i pro uživatele bez vývojářské licence, kteří již jako koncoví uživatelé nemusí mít licenci ke spuštění. Nicméně pro šíření aplikací a doplňků je potřeba mít vůči Siemensu podepsanou vývojářskou smlouvu. S touto licenční politikou dochází vždy k zaplacení vývojářské licence a to buď přímo od zákazníka, který si aplikace vyvíjí sám nebo od firmy, která vývoj pro koncového zákazníka zajišťuje.

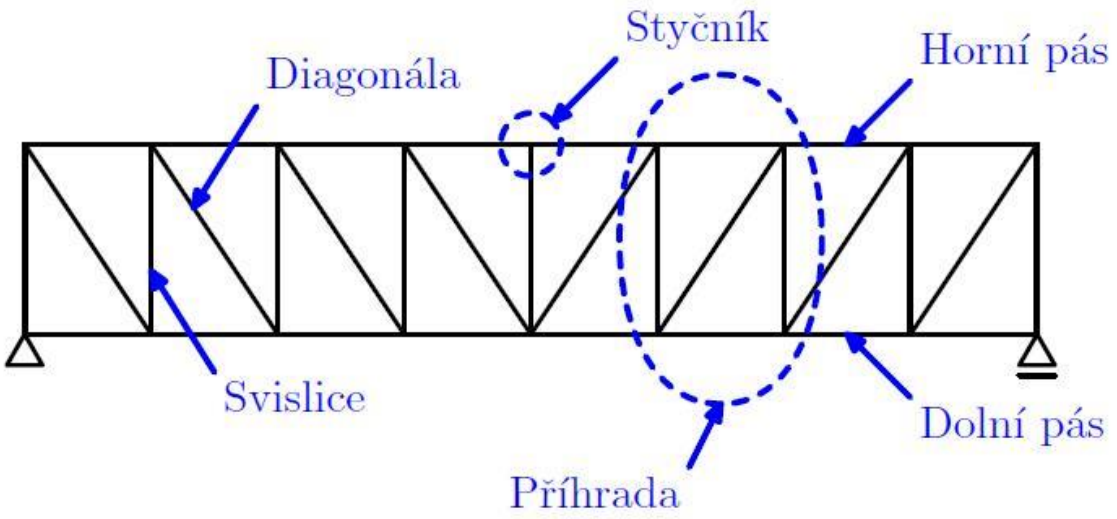
[16] GRIP – Graphics Interactive Programming

5 Analýza a návrh řešení zadaného problému

Zadaným problémem k řešení je vytvoření příhradové konstrukce, kterou je potřeba potáhnout určitým profilem a určit zda je zvolený profil vhodným řešením z hlediska pevnosti celé konstrukce. Celý proces tvorby bude předmětem analýzy a navržením vhodného řešení včetně automatizace, která povede k menší časové náročnosti.

5.1 Vstupní data

Základem je vytvoření příhradové konstrukce uživatelem (viz Obrázek 5-1 Příklad příhradové konstrukce včetně pojmů), která se využívá při různých konstrukčních řešeních. Tyto konstrukce se dají označit jako nosné či podpěrné, kde jsou využity tenké elementy (pruty) jako diagonály a svislice místo plných desek nebo stěn. Jednotlivé pruty jsou spojeny v bodě zvaném styčník. Smyslem takového konstrukčního řešení je přenesení a rozložení zátěže mezi nosnými prvky tak, aby vznikla taková konstrukce, kde budou jednotlivé nosné oddíly namáhány tlakem a tahem s ohledem na co nejmenší váhu a rozměry celého návrhu soustavy (Trávníček, 2013).



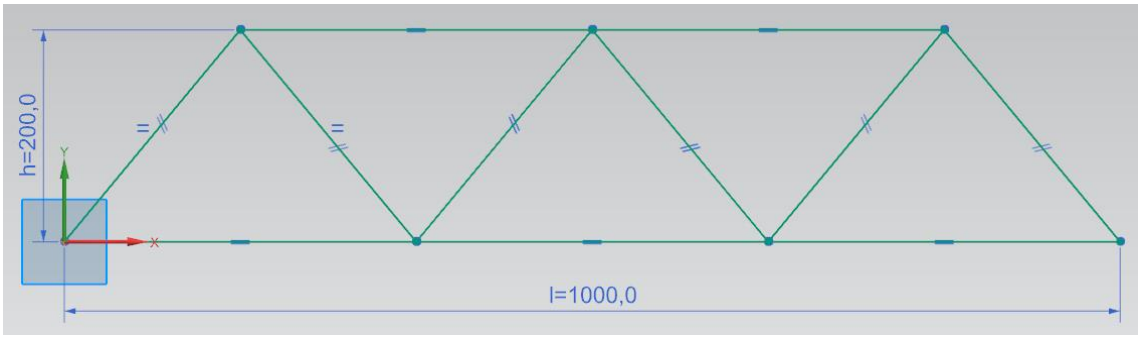
Obrázek 5-1 Příklad příhradové konstrukce včetně pojmů (Trávníček, 2013)

5.2 Možná řešení

Zadaný problém je možné provést ručně, ale při větší konstrukci je způsob realizace zdlouhavý a neefektivní. Stručný postup v uživatelském prostředí bez přídavných funkcí a vylepšení je popsaný v následující kapitole 5.3 Uživatelské řešení. Druhá část řešení se zabývá tímto postupem popsaným v uživatelském řešení a za použití programovacích nástrojů NX jej vylepšuje a zjednodušuje tak konstrukční práci.

5.2.1 Uživatelské řešení

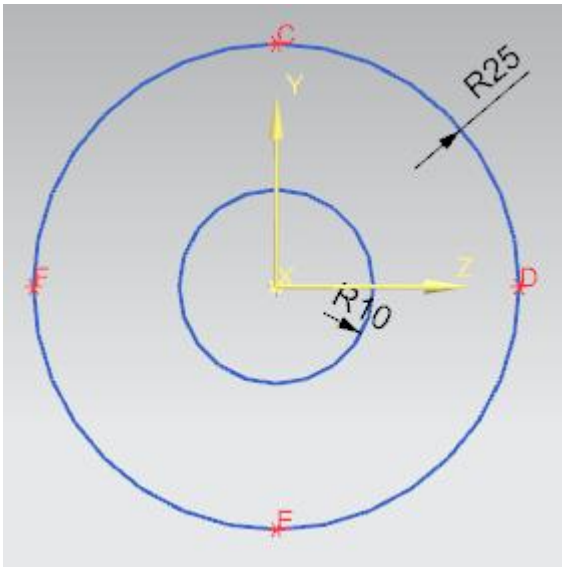
Zadanou úlohu lze vyřešit přímo z uživatelského prostředí NX. Jedná se o tzv. ruční řešení, kdy neexistují funkce ke zjednodušení a zrychlení celého procesu. Jsou tedy využité pouze základní funkce systému NX. V první fázi je potřeba vytvořit model konstrukce s danými vlastnostmi. Způsobů existuje několik a záleží na uživateli, jaký postup použije. V principu jde o základní vytvoření modelu popsané v kapitole 3.1 Modelování, kde je možné vytvořit 2D skicu ve vybrané rovině nebo přímo vytvářet model ve 3D. V obou případech je potřeba definovat body jako uzly konstrukce a na ně pak navázat přímky. Uzly lze zadávat přímo v souřadnicích a zajistit tak rovnou požadované rozměry i vlastnosti nebo zvolit druhou variantu, kdy se všechny prvky nadefinují do přibližných rozměrů a až následně se upraví parametry konstrukce. Upravit je možné rozměry celé konstrukce ale i vlastnosti jako vzdálenosti bodů, přímek nebo jejich rovnoběžnosti, úhly a jiné. Na obrázku 5-2 je ukázka skicáře NX, kde byla nejprve vytvořena přibližná konstrukce a poté definovány vlastnosti (Geometric constraints[17]) jako rovnoběžnosti přímek a celková velikost.



Obrázek 5-2 Uživatelsky vytvořená skica konstrukce

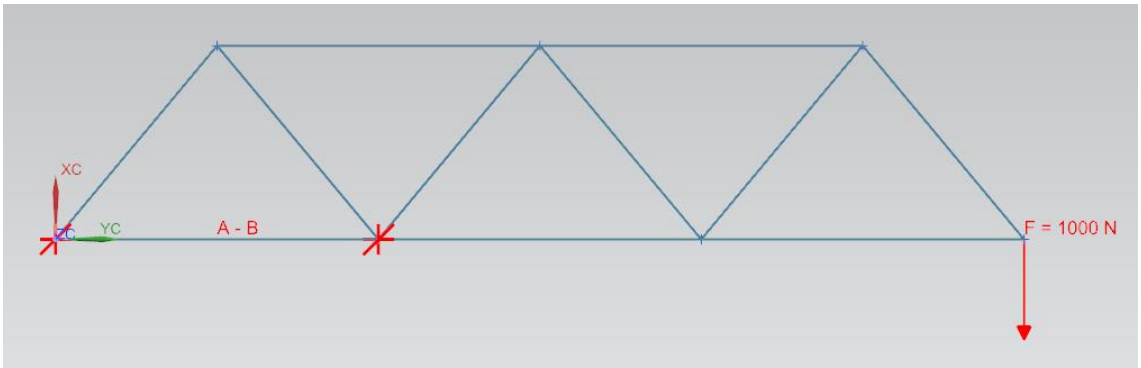
[17] Geometric constraints – funkce v systému NX, která umožňuje nastavit geometrické vlastnosti a vztahy objektů (rovnoběžnost, tečnost …)

V dané fázi je vytvořený základní model konstrukce, který slouží jako vstup pro modul CAE (NX Advanced simulation). Nad modelem (soubor *.prt) se tak vytváří nový soubor (*.fem), který zahrnuje nastavení okrajových podmínek a slouží dále k vytvoření simulace (*.sim). Tento postup je blíže popsán v kapitole 3.4.1. Jak bylo zmíněno je nyní potřeba vytvořit na tomto modelu síť a vybrat odpovídající fyzikální vlastnosti. Pro představu se může jednat o část ramene jeřábu, které je uchyceno na dvou spodních bodech a na konci tohoto ramene je zavěšený náklad, který působí silou F svisle dolů. Na této konstrukci postačuje vytvoření 1D sítě pro každý prvek (přímku) v konstrukci. Je potřeba vybrat postupně všechny prvky, které se nechají sjednotit pod jednu kategorii (collector), pro kterou se nastaví fyzikální vlastnosti najednou pro všechny prvky v kategorii. U této konstrukce byl použit profil, viz obrázek 5-3 a jako materiál byla zvolena ocel s hustotou 7829 kg/m^3.



Obrázek 5-3 Rozměry profilu konstrukce

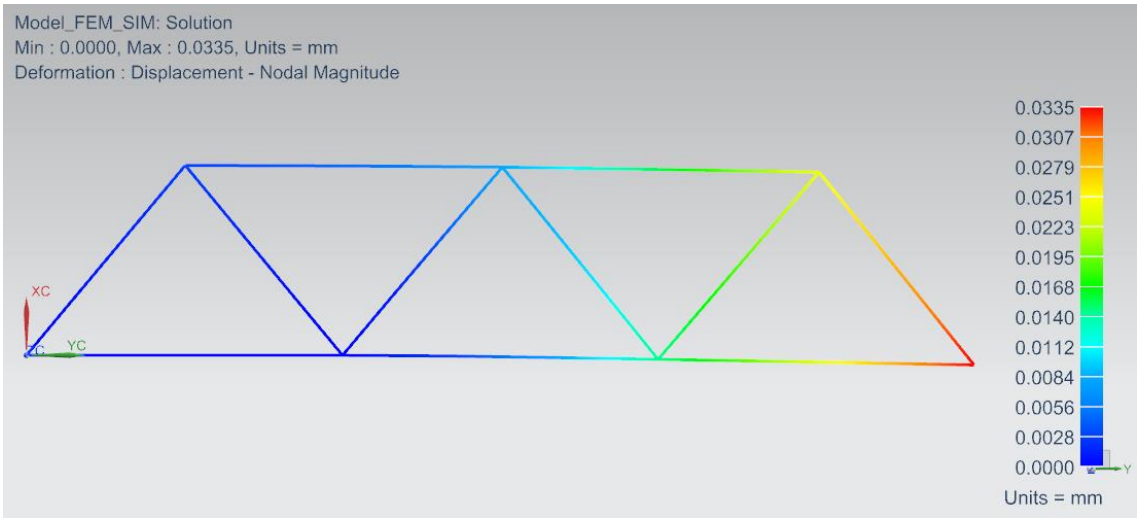
V současné chvíli je možné vytvořit simulaci (soubor *.sim), s nastavením okrajových podmínek. Výsledky výpočtu budou uloženy v souboru *.op2. Nastavení okrajových podmínek spočívá v tomto případě v nastavení bodů kde je konstrukce uchycena (body A – B označené červeně na obrázku 5-4) a vytvoření působící síly F.



Obrázek 5-4 Okrajové podmínky pro simulaci

Parametry konstrukce, které je potřeba brát v úvahu při vyhodnocení výsledku simulace jsou následující.

- Rozměry: 1000 x 200 mm
- Profil: viz obrázek 5-2 s vnitřním poloměrem 10 mm a vnějším 25 mm
- Materiál: ocel (7829kg/m^3)
- Uchycení konstrukce: body A,B dle obrázku 5-3
- Působící síla: F = 1000 N v koncovém bodě dle obrázku 5-4



Obrázek 5-5 Výsledné řešení – ohyb konstrukce

Výše je graficky znázorněný výsledek výpočtu (obrázek 5-5), na kterém je možné vyhodnotit ohyb celé konstrukce. Oproti normálnímu stavu, bez působení okolních sil, se konstrukce posune maximálně o 0,0335 mm a to v koncovém bodě působení síly. V okolních bodech je reakce na sílu menší, viz legenda grafického znázornění.

5.2.2 Automatizované řešení

Na základě předchozího postupu se jeví jako časově nejnáročnější část modelování samotné konstrukce a její nastavení podle požadovaných parametrů. Časová náročnost pak roste podle velikosti konstrukce, kde je potřeba dokola opakovat stejné postupy jako je nastavení rozměrů jednotlivých částí a vzájemných vztahů (vazby, rovnoběžnost, atd.). Z tohoto důvodu je první fází automatizovaného řešení právě první část vytvoření modelu. Od uživatele jsou vyžadované pouze vstupní parametry pro vytvoření konstrukce (délka, výška, počet příhrad) a případně další volitelné parametry. Vzhledem k pravidelné konstrukci lze vytvořit jednoduchý algoritmus k vytvoření celé konstrukce.

Takto vytvořený model je plně parametrický a je možné ho ihned upravit dle aktuálních požadavků. Automatizovaný postup je blíže popsán v následující kapitole (6. Řešení). Druhou časově nejnáročnější částí je příprava pro strukturální výpočet. V tomto bodě je potřeba na celé konstrukci vytvořit 1D síť, neboli rozdělení prutů konstrukce na malé prvky. Pokud je tato síť v jedné skupině, pak už může uživatel jednoduše změnit fyzikální vlastnosti jako typ použitého profilu a jeho materiál. Poslední krok závisí na požadavcích na konstrukci. Zde je potřeba jen vybrat body uchycení a okolní působící síly jako je gravitace nebo jiné zatížení.

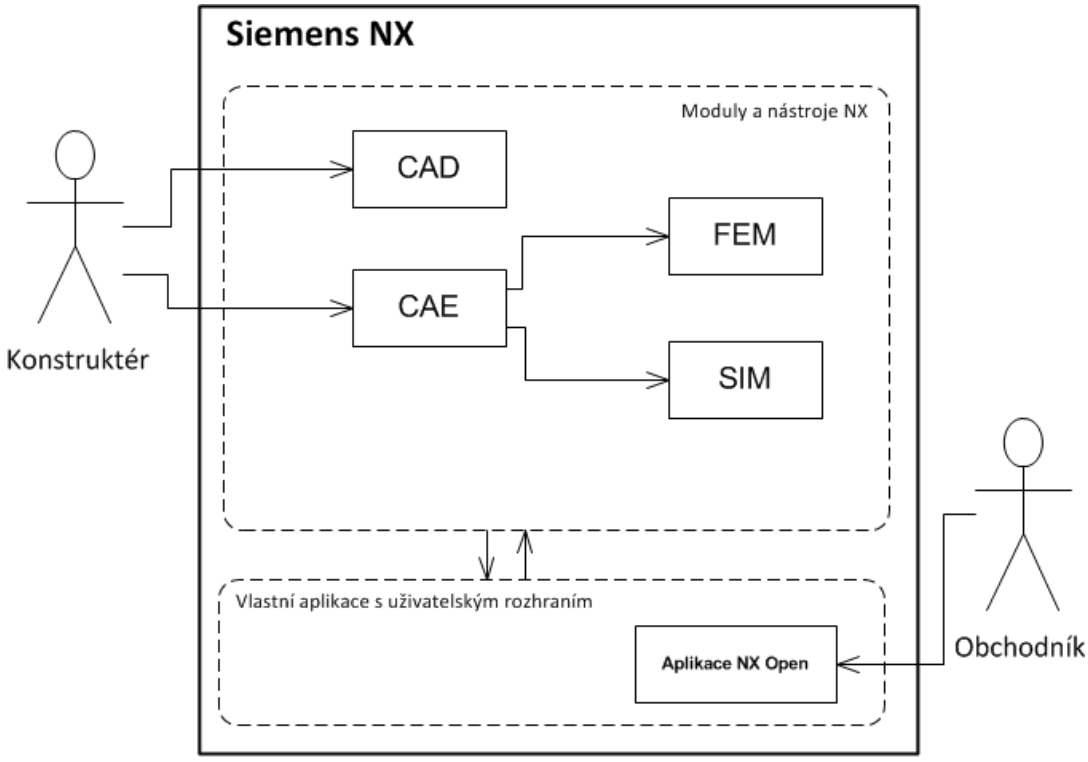
5.3 Praktické využití

V předchozích částech byly uvedeny různé příklady využití příhradové konstrukce. Zde je přehled těch nejběžnějších konstrukcí, na které lze aplikovat stejný postup tvorby.

- Rameno jeřábu
- Mostní konstrukce
- Výplň dutin z hlediska pevnosti
- Samonosné konstrukce

U samonosných konstrukcí je možné zmínit pevné (nepohyblivé) části jako jsou běžné oplocení pozemků, u kterých však mohou být použité jednodušší prvky vzhledem k tomu, že nepodléhají nárokům na působení okolních sil (například zatížení v případě mostní konstrukce). Větší potenciál využití se nachází v konstrukci samonosných vjezdových bran, které jsou součástí oplocení. V tomto případě se jedná přesně o zmiňovaný typ příhradové samonosné konstrukce včetně kladených nároků na pevnost.

Na obrázku 5-6 je uveden případ užití takovéto aplikace v komerčním prostředí firmy zabývající se výrobou samonosných pojízdných vrat. V případě zákazníkovy poptávky jsou možné dva postupy vedoucí k řešení. V prvním případě jsou předány informace od obchodníka konstruktérovi, který navrhne řešení včetně ověření konstrukce a výsledky předá zpět obchodníkovi k sestavení nabídky. Tento postup není časově příliš efektivní a tak je možné vytvořit zjednodušující aplikaci pomocí NX Open, která poskytne obchodníkovi takové uživatelské prostředí, aby se základními znalostmi dokázal udělat návrh jako konstruktér a na počkání sestavit zákazníkovi nabídku.



Obrázek 5-6 Obecný případ užití v komerční praxi

Zmíněný model užití je v rámci testování aplikace (viz kapitola 7. Testování) vysvětlený na reálném případu a fungování firemního procesu od poptávky přes nabídku až po samotnou výrobu a dodání zákazníkovi.

6 Realizace

V této části je podrobně popsán realizovaný program včetně ukázek z prostředí systému NX. Základní osnovou pro automatizované řešení bude předchozí postup ručního vytvoření. Časově nejnáročnější se jeví vytvoření modelu příhradové konstrukce s požadovanými vlastnostmi a vytvoření sítě na všech prvcích konstrukce pro následný výpočet. Cílem je stejný výsledek jako u ručního řešení s tím rozdílem, že za použití dostupných nástrojů budou vytvořeny automatizované části spolu s interakcí uživatele k dosažení lepších výsledků z hlediska časové náročnosti.

6.1 Vývojové prostředí

Součástí knihovny NX Open jsou také další zdrojové soubory přímo do vývojového prostředí MS Visual Studio, které slouží jako šablona pro vytváření nových projektů s využitím NX Open. Samotný průvodce vytvořením nového projektu nabízí celou řadu nastavení a předdefinovaných funkcí. V úvodním výběru je potřeba určit, zda se bude jednat o externí aplikaci (*.exe) s využitím instance systému NX bez grafického prostředí nebo naopak půjde o tvorbu doplňku (*.dll), který bude založen na běžící instanci s využitím grafického rozhraní. Knihovna API NX Open je rozdělena na dvě základní části NXOpen.UFAPI a NXOpen.UIAPI a jejich použití je možné vybrat také v tomto průvodci. V principu jde o import knihoven s uživatelskými funkcemi (UFAPI[18]) a knihoven pro tvorbu grafického rozhraní (UIAPI[19]). V následujícím programu budou potřeba obě tyto knihovny. V poslední řadě průvodce nabízí možnost, jakým bude program spuštěn a ukončen, aby mohly být připraveny funkce ve vytvořené šabloně programu. Možnosti spuštění a ukončení jsou následující.

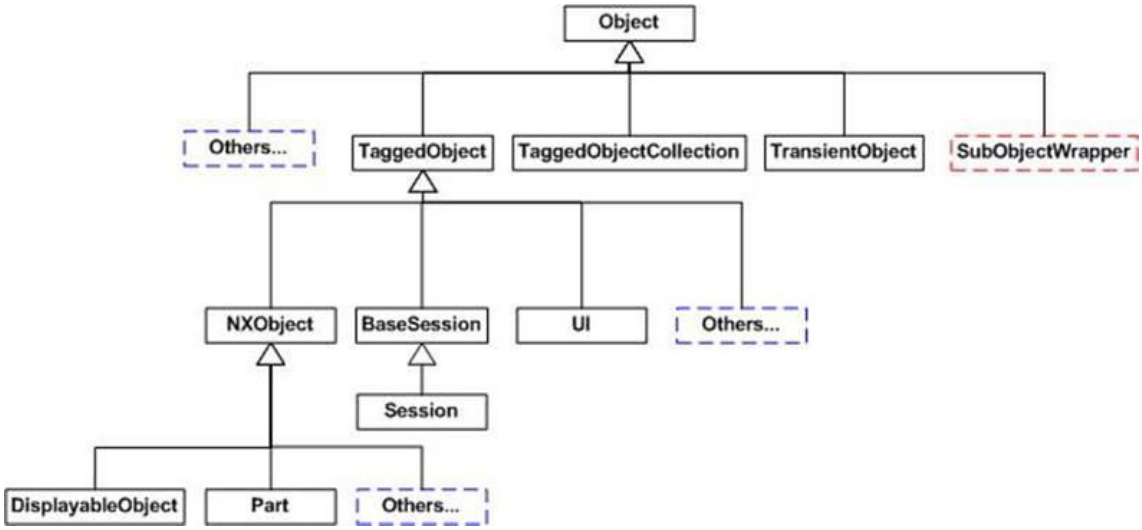
- Spuštění
    - Explicitní (Main) - spuštění například přes ikonu na panelu nástrojů
    - Automatické (při spuštění instance)
    - Při uživatelské akci (User Exit) – obsahuje seznam předdefinovaných akcí, např. otevření dílu, uložení dílu, přidání komponenty, změna atributů, spuštění modulu (CAE, CAM) a další.
- Ukončení
    - Automaticky při ukončení instance NX
    - Automaticky při ukončení vlastní aplikace
    - Explicitně pomocí ukončovacího dialogu

[18] UFAPI – User Function Application Programming Interface
[19] UIAPI – User Interface Application Programming Interface

Pro následující program je využitelné explicitní spuštění, kdy uživatel vyvolá samotný program z nabídky a automatické ukončení ve chvíli skončení programu.

6.2 Analýza potřebných funkcí NX

Ke zjištění potřebných funkcí systému NX byl využit nástroj NX Journal (Journaling, 2015), který je schopen většinu kroků v uživatelském rozhraní zaznamenat do souboru ve vybraném programovacím jazyce .NET. Pokud záznam obsahuje pouze plně podporované funkce, je možné jej zpětně spustit v prostředí NX. Takto vytvořený skript je možné editovat a vytvářet tak potřebné úpravy přímo v tomto kódu. Nicméně orientace ve vygenerovaném skriptu není optimální vzhledem k tomu, že i záznam jediné funkce na vytvoření bodu ve 3D obsahuje 120 řádek kódu a to zejména s definicemi dále nevyužitých vlastností a objektů. Záznam není nijak strukturovaný ani blíže komentovaný. Pro potřeby prvotního zjištění umístěných funkcí v knihovně NX byl tento nástroj vhodnější místo hledání v programátorské dokumentaci (Siemens PLM, 2014). Respektive se osvědčil postup nalezení základních funkcí pomocí NX Journal s následným dohledáním ve zmíněné programátorské dokumentaci s konkrétní specifikací.



Obrázek 6-1 NX Open object model (Siemens Product Lifecycle Management Software, 2014)

Na obrázku 6-1 je uvedený obecný objektový model knihovny NX Open API, který je společný pro všechny podporované jazyky v rámci NX Open. Pro potřeby této aplikace jsou nejpoužívanější Session, která představuje aktuální instanci spuštěného programu NX na kterou se aplikace odkazuje a dále zpracovávaný model reprezentující třída Part. Níže jsou uvedeny funkce a stručné postupy pro vytvoření vlastní aplikace v prostředí NX. Jedná se pouze o ukázky zdrojových kódů hlavních částí nezbytných pro práci s NX

Open. Na obrázku 6-2 jsou použité třídy NX Open v hlavním programu. Zmínka o Session je uvedena výše, dále jsou k dispozici uživatelské funkce *UFSession* (UserFunction) a rozhraní pro uživatele UI[20]. Poslední z těchto hlavní tříd je *Part*, který reprezentuje vytvořenou instanci modeláře (CAD) a v tomto modeláři se dále vytváří všechny prvky návrhu.

```
// class members
private  Session theSession;
private  UI theUI;
private  UFSession theUfSession;
public static bool isDisposeCalled;
private Part workPart;
```

Obrázek 6-2 Třídy hlavního programu z NX Open

Ve zmíněném návrhu, modeláři geometrie, vytvořeném jako *workPart* se dále volají funkce pro tvorbu bodu, křivky, zaznamenání historie do stromu atd. Samostatné vytvoření bodu znamená přípravu souřadnic do připravené struktury *Point3d* a její využití pro tvorbu bodu *Point.* V této fázi jsou připravené souřadnice bodů a automaticky vytvořený seznam (*pointList*). Není tedy potřeba zakládat vlastní seznamy, či jiné struktury k uchování těchto informací. Toto pravidlo platí ve většině případů, že součástí konstruktorů jednotlivých objektů jsou zároveň i automaticky vyvářené seznamy a další informace dále použitelné například pro tvorbu stromu s historií. Pro historii existují dva stavy, které jsou přepínatelné v GUI NX. Jedná se o „History Free Mode[21]“, kdy historie provedených operací není zaznamenána a „History mode[22]“ naopak tuto aktivitu zaznamenává do stromové struktury pro případnou pozdější editaci. Historii je však v tomto případě nutné tvořit ručně přes tzv. *pointFeatureBuilder* a je tak možné do stromové struktury umístit jen vybrané položky.

```
Point3d pointCoordinates = new Point3d(y, x, z); //vytvoreni souradnic
Point point = workPart.Points.CreatePoint(pointCoordinates); // vytvoreni bodu
```

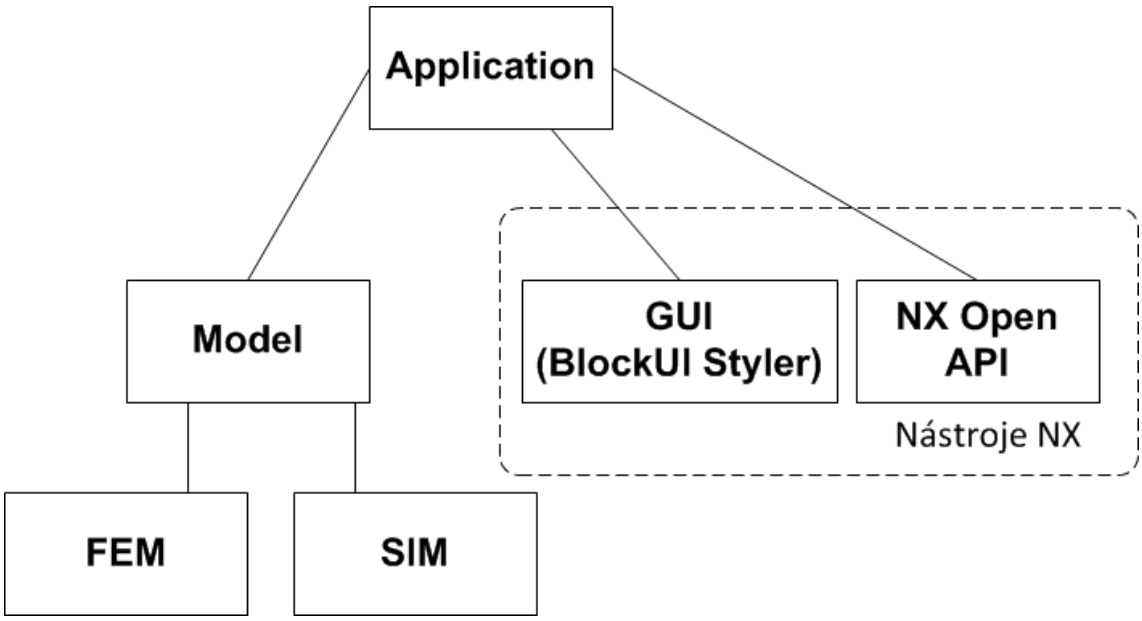
Obrázek 6-3 Vyvoření bodu

Veškeré funkce v rámci modeláře jsou dostupné právě přes instanci modeláře *workPart*. Stejným způsobem je pak vytvořena křivka, respektive *Line* a jako vstupní parametry jsou uvedeny předem vytvořené body.

[20] UI – User Interface
[21] History Free Mode – bez záznamu historie modelování
[22] History Mode – zapnutý záznam historie do stromové struktury

6.3 Architektura aplikace

Z pohledu tvorby vlastní aplikace je potřeba zohlednit stávající systém a jeho strukturu, respektive moduly, do kterých bude tato aplikace zasahovat. V tomto případě se jedná využití knihovny NX Open API a BlockUI Styler jako nástrojů NX. Dále je nutné vytvořit vlastní objekty v hlavních využívaných modulech modelování pro vytvoření návrhu a přípravě pro metodu konečných prvků FEM a dále pro samotný výpočet včetně simulace výsledků.

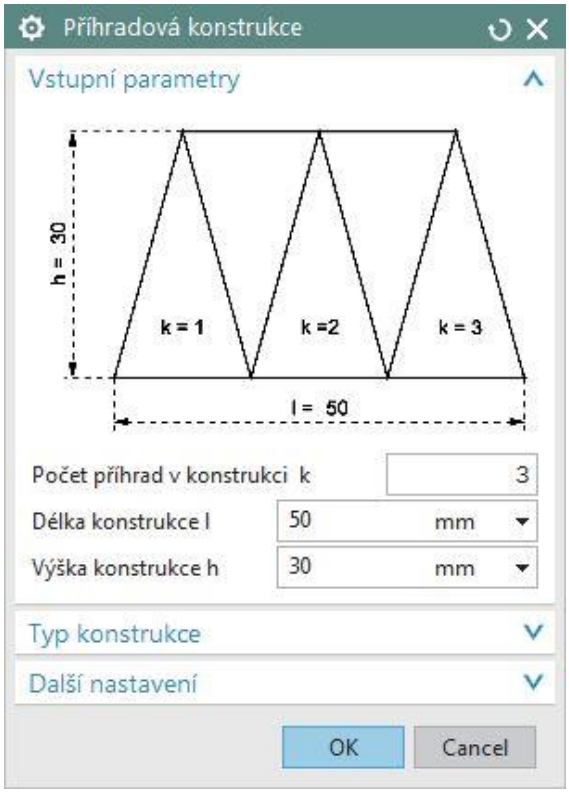


Obrázek 6-4 Návrh architektury systému

S využitím této architektury je pak navrhovaný systém velmi podobný reálné situaci z pohledu uživatele, který se orientuje právě pomocí vnitřních aplikací systému NX. Tyto aplikace se také někdy nazývají jako moduly modeláře, výpočtů, simulací atd.

6.4 Grafické rozhraní

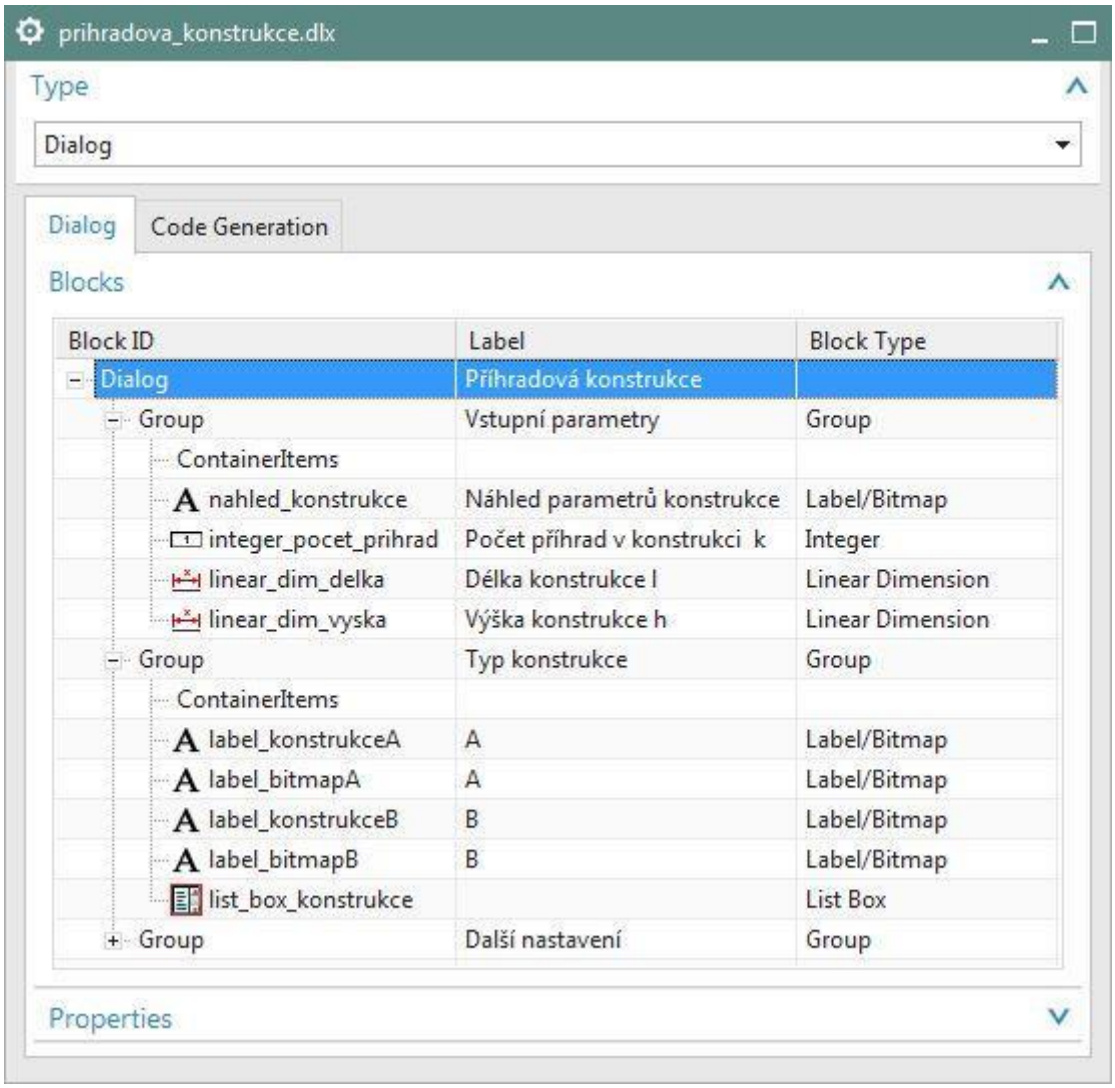
Na tvorbu grafického uživatelského rozhraní (GUI[23]) slouží Block UI Styler blíže popsaný v kapitole 4.1.2. Pomocí tohoto nástroje je možné vytvářet dialogová okna se stejným grafickým stylem, který je použitý v systému NX. Na obrázku 6-1 je uvedený počáteční uživatelský vstup při spuštění programu pro tvorbu příhradové konstrukce.



Obrázek 6-5 Vstupní parametry

Vytvoření dialogového okna probíhá výběrem dostupných bloků z katalogového seznamu a přidáním do stromové struktury (viz obrázek 6-2). Každý z těchto bloků má kromě typu a jména také svá vlastní nastavení, která je možné upravit ve vlastnostech objektu. Patří mezi ně například nastavení mezních hodnot v případě zadávání rozměrů, přidání popisků při najetí kurzoru myši, nebo nastavení počátečních hodnot. Veškeré údaje o struktuře GUI a dalších nastavení vlastností jsou uloženy do XML souboru generovaného při uložení. Druhým výstupním souborem je zdrojový kód ve zvoleném programovacím jazyce obsahující třídu včetně obslužných funkcí k incializaci a zpracování dat po potvrzení uživatelského vstupu.

[23] GUI – Graphical User Interface



Obrázek 6-6 Nastavení vstupního GUI

6.5 Ošetření chybových vstupů

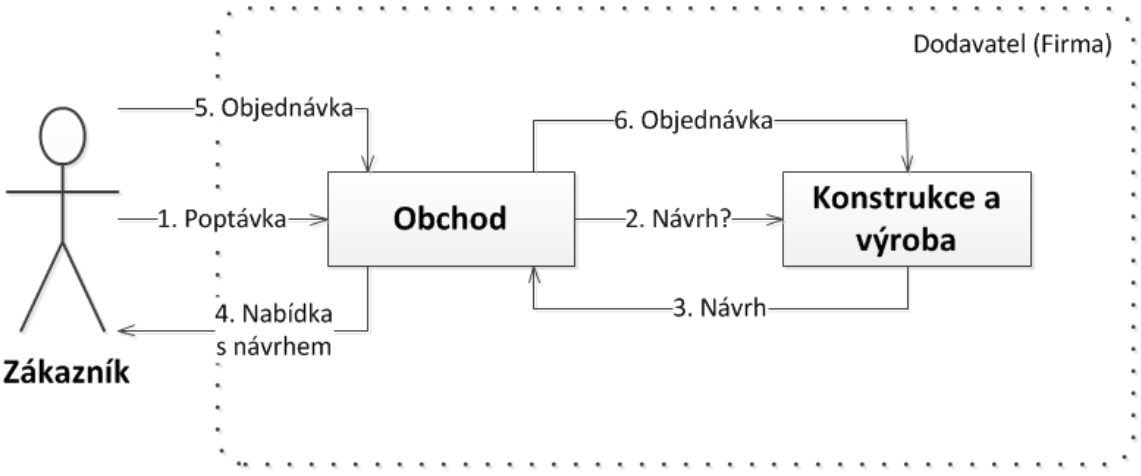
Aby nedocházelo k chybovým hláškám z prostředí NX, bylo nutné ošetřit uživatelské vstupy a možnosti spuštění programu. Zkompilovaný program je pouze doplňkem ve formátu *.dll a nelze ho tedy spouštět bez využití instance NX ať už s nebo bez grafického prostředí. Nicméně nelze vyloučit ruční uživatelské spuštění přímo v prostředí NX tehdy, kdy to není možné, ale uživatel program takto z nějakého důvodu spustí. V tomto případě záleží, zda jsou splněné podmínky pro spuštění programu a to zejména vytvořená instance modeláře, aby bylo možné vytvořit model konstrukce. Pokud by nastala neošetřená výjimka, tak systém NX neukončí rozdělanou práci, ale pouze zobrazí chybové hlášení, které není z uživatelského pohledu srozumitelné. Pro tyto případy je tedy uživatele potřeba informovat, co v této konkrétní situaci udělal špatně a vypsat k tomu příslušnou chybovou informaci jak daný problém napravit.

7 Testování

Ověření implementovaného programu proběhlo v systému Siemens NX 10.0.0.24. Jedná se o základní instalaci verze NX 10 bez dalších aktualizací či doplňků. Kromě menších testů ověření funkčnosti jednotlivých částí programu, proběhl také komplexní test demonstrující jednu z možností nasazení do praxe. Tento test je blíže popsán v následující kapitole 7.1.

7.1 Ověření funkcionality v systému NX

Testování se odvíjí od předem daného ukázkového scénáře z kapitoly 5.3 Praktické využití, kde je navržen firemní obchodní proces pro komunikaci mezi zákazníkem a konstrukční firmou. Na jedné straně se jedná o zákazníka, který poptává pojízdná vrata na míru, na straně druhé je samotná firma zastoupená obchodníkem a konstruktérem, který dodává návrhy dle požadavků zákazníka. Na obrázku 7-1 je znázorněný původní proces komunikace bez využití automatizovaného postupu. Jednotlivé činnosti jsou číslované, tak jak jdou v procesu za sebou, aby bylo zřejmé jejich pořadí a návaznosti.

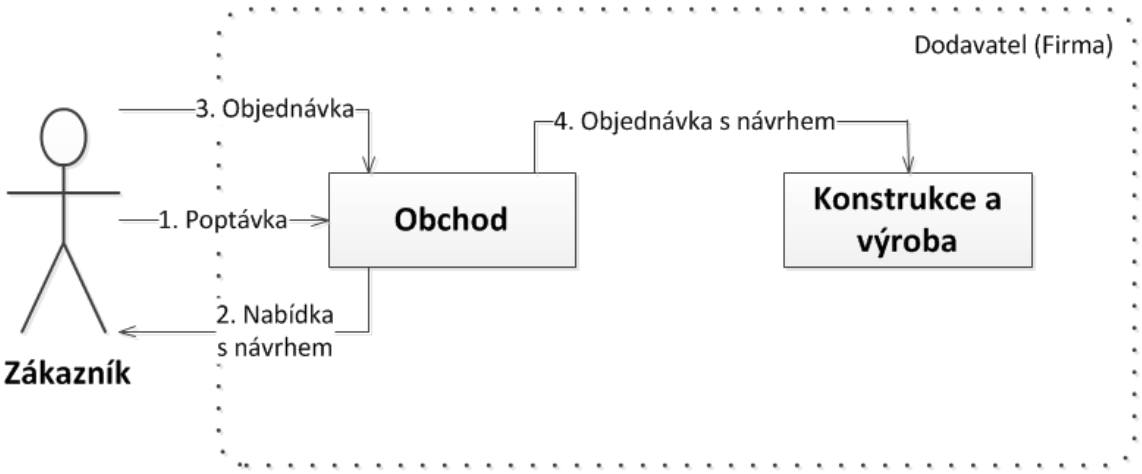


Obrázek 7-1 Původní obchodní proces

Dle procesu na obrázku 7-1 nejprve zákazník kontaktuje obchodního zástupce s poptávkou (1) na řešení konstrukce na míru. Obchodník doplní se zákazníkem požadované parametry a předá (2) zakázku dále na konstrukci. Konstruktér na základě požadovaných parametrů navrhne model a předává zpět (3) jako podklady k nabídce. Obchodník pak kontaktuje zákazníka s nabídkou včetně návrhu na řešení (4). Posledním krokem v tomto procesu, který stojí za zmínku z důvodu porovnání

s následujícím řešením, je zákazníkova objednávka, kterou zpracuje obchodní část a předá k výrobě a dodávce zboží.

Automatizovaný proces obsahuje stejné subjekty zákazníka na jedné straně a obchodníka s konstruktérem na straně druhé. S využitím aplikace pro uživatelskou tvorbu příhradové konstrukce se ale mění jednotlivé akce v průběhu procesu (viz obrázek 7-2).



Obrázek 7-2 Automatizovaný obchodní proces

Na tomto inovovaném procesu (viz obrázek 7-2) je na první pohled vidět menší počet akcí, které spočívají v komunikaci mezi subjekty. Po zákazníkově poptávce (1) následuje nabídka s návrhem řešení, ale s tím rozdílem, že není potřeba kvůli návrhu kontaktovat konstruktéra. Obchodník, na základě obecných znalostí a možností programu, vytvoří podle zadaných parametrů návrh řešení včetně ověření konstrukce a může tak rovnou zákazníkovi vytvořit nabídku včetně podkladů z výstupu programu (2). Konstruktér se tak do procesu zapojuje až v případě objednávky (3-4) kvůli dalším úpravám a výrobě.

7.2 Zhodnocení automatizace

Testovaný obchodní případ v předchozí kapitole 7.1 znázorňuje zjednodušení procesu ve firmě a popisuje tak navržený případ užití z kapitoly 5.3 s využitím automatizace. Níže je uvedené porovnání těchto procesů (stávající vs. navrhovaný) z pohledu časové náročnosti dodavatele.

Činnost v procesu	Časová náročnost v hodinách
Poptávka	Jednorázová akce
Zpracování poptávky a žádost o návrh	0,5
Návrh konstrukce a ověření pevnosti	0,5 – 1
Odeslání zpět obchodníkovi	Jednorázová akce
Zpracování nabídky	1
Zpracování objednávky s předáním podkladů na konstrukci a výrobu	0,5
Čistý čas celkem	**2,5 - 3**

Tabulka 7-1 Časová náročnoststávajícího procesu

Časové nároky na uvedený proces jsou ve stávajícím stavu cca 2,5 – 3 hodiny na jednu objednávku. Jedná se pouze o čistý čas hlavních činností a prodlevy navíc, které vznikají při předávání informací mezi subjekty jsou značné.

Činnost v procesu	Časová náročnost v hodinách
Poptávka	Jednorázová akce
Zpracování poptávky a návrh konstrukce včetně ověření pevnosti	0,5
Zpracování nabídky	1
Zpracování objednávky s předáním podkladů na konstrukci	0,5
Čistý čas celkem	**2**

Tabulka 7-2 Časová náročnost navrženého procesu

Zlepšení časových nároků je patrné na základních činnostech procesu a zároveň se výrazně minimalizovaly časové prodlevy mezi jednotlivými kroky.

Reálně pak může být úspora času mnohem větší, než se zdá být na součtu čistého času prováděných akcí. Například při porovnání časové prodlevy mezi poptávkou a nabídkou jsou mezi předchozím procesem a navrženým procesem velmi rozdílné. V první případě musí zákazník čekat v intervalu od pár hodin po několik dní v závislosti na vytíženosti konstrukce ve firmě. V navrženém případě lze poptávku a nabídku sloučit do jedné akce, kterou je možné provádět přímo se zákazníkem a tím tak výrazně ušetřit čas obou stran.

8 Uživatelská dokumentace

Níže je popsán stručný uživatelský postup ke spuštění dodávaného programu.

8.1 Minimální požadavky

Ke spuštění programu příhradové konstrukce je potřeba mít nainstalovaný systém Siemens NX ve verzi 10.0 a novější. Licenční požadavky jsou spojené s využitím modulů pro modelování, výpočty s následnou simulaci a případně výkresy. Jedná se tedy o tyto licenční moduly, které musí vaše licence obsahovat.

Kategorie	Popis	Modul
CAD	Vytvoření modelu	gateway solid_modeling
CAE - FEM	Vytvoření sítě Nastavení fyzikálních vlastností	nx_masterfem nx_ftk
CAE - SIM	Okrajové podmínky, simulace a výpočet	nx_nastran_export ug_nas_bn nx_nas_bn_basic_dsk
CAD	Vytvoření výkresu z CAD modelu	drafting

Tabulka 8-1 Požadované licenční moduly

Tyto moduly je možné dohledat a ověřit přímo v licenčním souboru, případně u dodavatele licence k softwaru Siemens. Uvedené moduly jsou elementárními prvky licence a v případě nákupu licence budou zahrnuty v komplexních licenčních balících. Obecně tento program a další popisované části pokrývá licence pro Modeling, Drafting a Advanced Simulation.

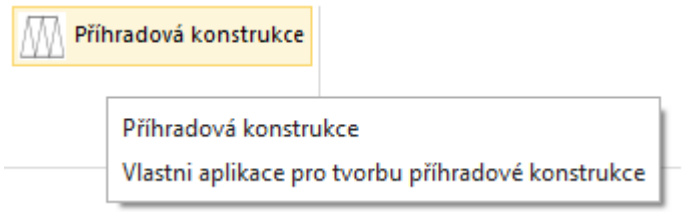
8.2 Instalace programu

Instalace do prostředí NX probíhá pomocí přiloženého skriptu, který se nachází v umístění „*bin\prihradova_konstrukce_install.bat*“. V případě, že se při instalaci nevytvořili základní systémové proměnné nebo v době instalace tohoto programu z nějakého důvodu neexistují, nastavte následující.

UGII_BASE_DIR = „cesta k NX, např. C:\PLM\NX10“
UGII_ROOT_DIR = „např. C:\PLM\NX10\UGII“
SPLM_LICENSE_SERVER = „adresa serveru s portem, např. 28000@muj-server“

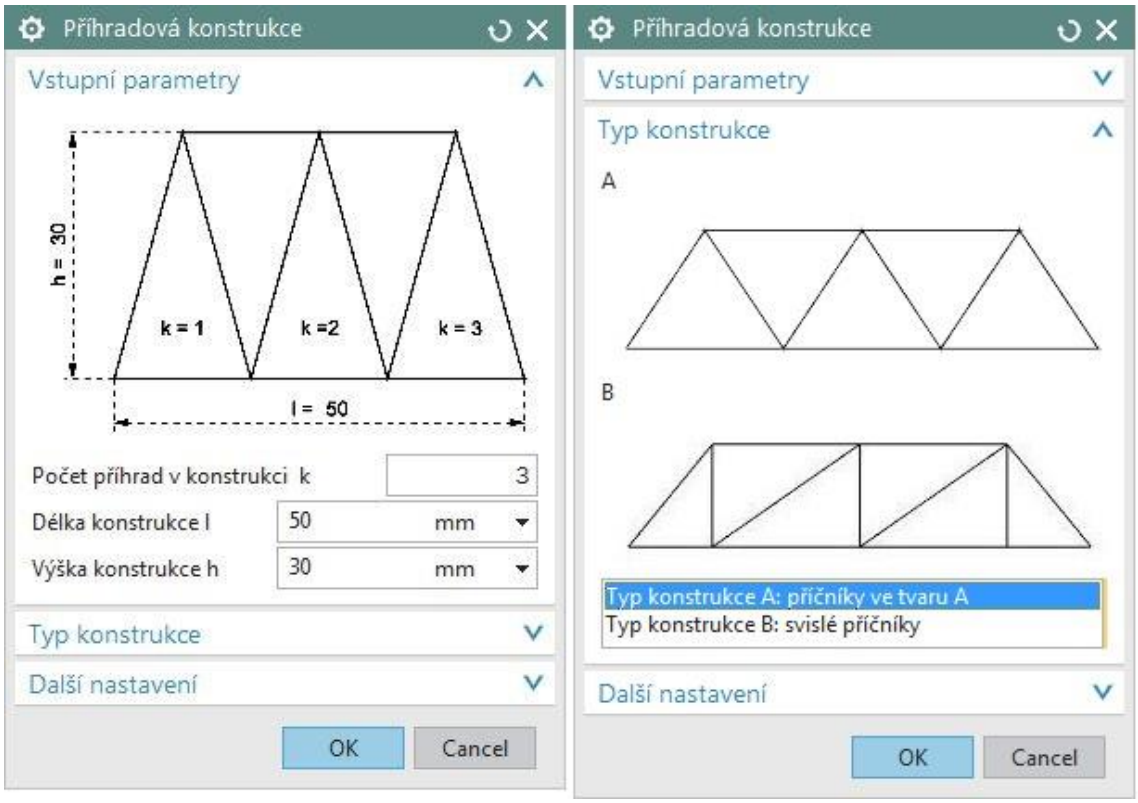
8.3 Spuštění programu

Program je možné spustit ihned po otevření NX nebo přímo v prostředí modeláře, kde se také nachází ikona (viz obrázek 8-1) ke spuštění programu k tvorbě příhradové konstrukce. Tuto funkci najdete přímo na hlavním panelu „*Home*“.



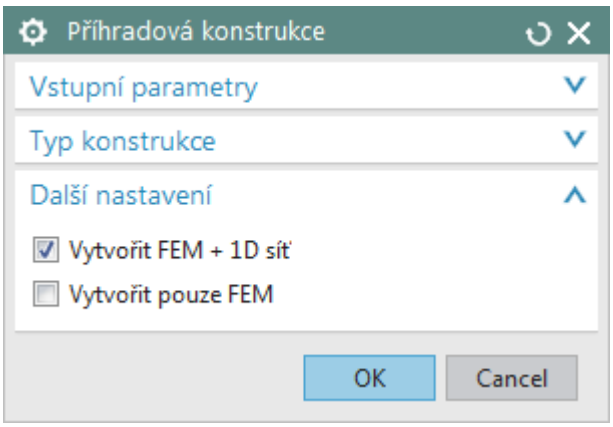
Obrázek 8-1 Spuštění programu z prostředí NX

V následujícím dialogovém okně (viz obrázek 8-2) zadejte vstupní parametry. Použití jednotlivých parametrů je znázorněno v ukázce přímo v dialogovém okně. V druhé části dialogového okna je možné vybrat typ konstrukce, aby jej pak nebylo nutné příliš upravovat. Výchozí konstrukce je typ A, pokud není vybrána ani jedna možnost. Pokud chcete tuto konstrukci, pak stačí zadat pouze 3 úvodní parametry (počet příhrad, délk a výška), které jsou povinné.



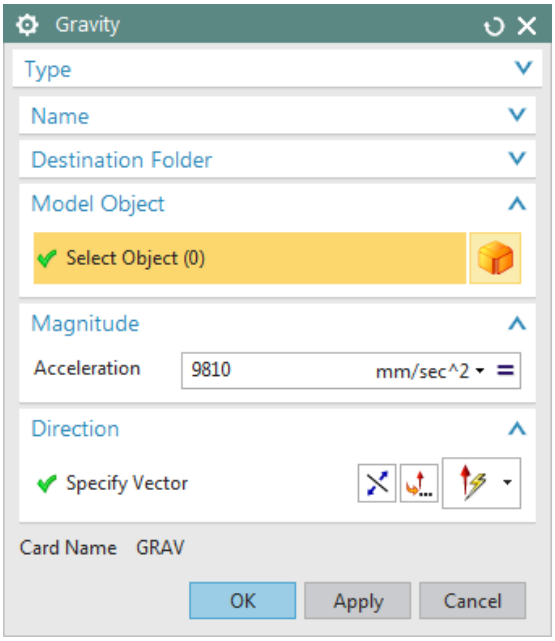
Obrázek 8-2 GUI s úvodními parametry

V dalších nastaveních jsou volitelné možnosti, pokud budete chtít připravit FEM soubor. Po vytvoření konstrukce se automaticky prostředí NX přepne do modulu Advanced Simulation (FEM) a na zadané konstrukci se vytvoří 1D síť pro každý element (příčník) zvlášť. Všechny tyto sítě jsou sjednocené pod „Beam Collector“ u kterého je pak možné hromadně měnit fyzikální vlastnosti jako je typ profilu či materiál. Na výběr jsou možnosti vytvoření prázdného FEM, pokud chcete síť definovat ručně nebo FEM včetně automatického vytvoření 1D sítě (viz obrázek 8-3).



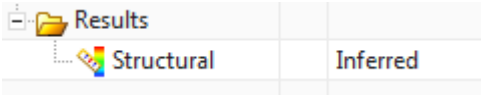
Obrázek 8-3 Volitelné vstupní parametry

Při výběru vstupních parametrů, včetně dalšího nastavení s FEM a 1D síti, dostanete připravený model, u kterého je možné ze základních vlastností materiálu a profilu změnit na požadované z knihovny NX. V případě, že není potřeba dělat další úpravy lze rovnou vytvořit samotnou simulaci pro ověření konstrukce. Pokračujte tedy v hlavní nabídce na *File -> New -> Simulation*. Poté nastavte body uchycení včetně působící síly. Pro samonosnou konstrukci postačuje vložení obecné gravitační síly (viz obrázek 8-4), protože se nepředpokládá další zatížení.



Obrázek 8-4 Vložení gravitační síly na objekt

Po dokončení těchto nastvení je možné spustit výpočet a simulaci pod pravým tlačítkem myši vyberte z nabdíky Solution -> Solve. Grafické výsledky budou ve složce *Reuslts*.



Obrázek 8-5 Výsledky výpočtu

9 Závěr

Tato práce měla za cíl prozkoumat možnosti programování vlastních aplikací v CAx systémech a blíže se zaměřit na konkrétní systém Siemens NX, který je využíván v Regionálním technologickém institutu Fakulty strojní Západočeské univerzity v Plzni (dále jen RTI). V počáteční fázi bylo potřeba prostudovat základní principy a pojmy z oblasti strojního inženýrství, které byly nezbytné pro vytvoření této práce. Co se týče otevřenosti nejznámějších dostupných systémů, bylo určitě pozitivním zjištěním, že veškeré zmíněné aplikace nabízí možnosti tvorby vlastních aplikací a liší se pouze v počtu podporovaných programovacích jazyků a licenčních podmínkách.

Software Siemens NX, využívaný při realizaci výzkumných úkolů RTI, nabízí širokou škálu nástrojů na přizpůsobení prostředí nebo zjednodušení práce. Dostupné nástroje jsou popsané v teoretické části včetně praktických ukázek. Pro realizaci zadaného problému na tvorbu příhradové konstrukce byly využity nástroje NX Journal pro záznam prováděných operací, knihovna NX Open API, nástroj pro tvorbu uživatelských rozhraní BlockUI Styler a v neposlední řadě nástroj pro tvorbu a úpravu uživatelských nabídek.

Vzhledem k tomu, že vývoj vlastních aplikací pro NX je podmíněn zakoupením licence, je na zvážení konkrétního subjektu, zda se investice do licence a vývoje vyplatí. V případě vlastního vývoje ve firmě je nezbytná spolupráce mezi IT oddělením, které se zabývá samotným vývojem a implementací, s ostatními odděleními konstrukce, technologie a dalších podle oblasti vytvářené aplikace.

Vytvořená aplikace pro tvorbu příhradové konstrukce demonstruje využití nástrojů NX a ukazuje tak na možnosti k usnadnění práce, o kterých většina uživatelů ani neví, že existují. Výsledná aplikace nemusí sloužit pouze pro demonstraci nástrojů NX a jako ukázka k popisovaným možnostem v této práci. Využití příhradové konstrukce je z hlediska tvorby a vývoje konstruktéra velmi časté, stejně jako počet oblastí, ve kterých se příhradové konstrukce vyskytují.

Výsledná aplikace splňuje zadané požadavky na tvorbu celé konstrukce a výrazně zkracuje celkovou časovou náročnost řešeného problému. Obecně lze tyto procesy a vylepšení aplikovat v rámci výzkumného centra RTI ke zjednodušení často prováděných operací.

10 Přehled zkratek

Použité zkratky jsou uvedeny vždy v poznámce pod čarou na stránce, kde je zkratka použita. Pro přehled je zde seznam všech použitých zkratek se stručným komentářem. Zkratky jsou uvedeny v pořadí, v jakém se vyskytují v textu práce.

Zkratka	Popis zkratky
CAx	Computer Aided technology – počítačem podporované technologie návrhu, analýzy a výroby produktů
PLM	Product Lifecycle Management – řízení životní cyklu výrobku
CAD	Computer Aided Design – počítačová podpora návrhu
CAE	Computer Aided Engineering – počítačová podpora inženýrských výpočtů
CAM	Computer Aided Manufacturing – počítačová podpora výroby
API	Application Programming Interface – rozhraní pro programování
LGPL	Lesser General Public License – licence svobodného softwaru
UGS	Unigraphics software – původní společnost, která je od roku 2007 součástí Siemens PLM software
RE	Reverse Engineering – opačný postup k zavedenému procesu, zpětná analýza
ISO-E	Evropská norma promítání obsahující základní pohledy a způsob jejich uspořádání
FEM	Finite Element Method = MKP
MKP	Metoda konečných prvků
UI	User Interface – uživatelské rozhraní
SNAP	Simple NX Application Programming
GRIP	Graphics Interactive Programming
UFAPI	User Function Application Programming Interface
UIAPI	User Interface Application Programming Interface
GUI	Graphical User Interface

11 Seznamy

11.1 Seznam obrázků

Obrázek 2-1 – proces vývoje produktu CAD – CAE (Intech) 3
Obrázek 2-2 Sestava ve formátu JT v programu JT Assistant 5
Obrázek 3-1 Vytažení 6
Obrázek 3-2 Rotace 7
Obrázek 3-3 Tažení po křivce 7
Obrázek 3-4 Body a křivka 8
Obrázek 3-5 Plocha a těleso 9
Obrázek 3-6 Sestava (Siemens Product Lifecycle Management Software, 2014) 10
Obrázek 3-7 Částečný strom s prvky sestavy 11
Obrázek 3-8 Příklad výkresu 12
Obrázek 4-1 Block UI Styler - příklad 14
Obrázek 4-2 Block UI Styler - struktura 15
Obrázek 4-3 Full Journal Support 17
Obrázek 4-4 Partial Journal Support 17
Obrázek 4-5 Fotografie klíče 18
Obrázek 6-1 NX Open object model (Siemens Product Lifecycle Management Software, 2014) 27
Obrázek 6-2 Třídy hlavního programu z NX Open 28
Obrázek 6-3 Vyvoření bodu 28
Obrázek 6-4 Návrh architektury systému 29
Obrázek 6-5 Vstupní parametry 30
Obrázek 6-6 Nastavení vstupního GUI 31
Obrázek 7-1 Původní obchodní proces 32
Obrázek 7-2 Automatizovaný obchodní proces 33
Obrázek 8-1 Spuštění programu z prostředí NX 36
Obrázek 8-2 GUI s úvodními parametry 37
Obrázek 8-3 Volitelné vstupní parametry 37
Obrázek 8-4 Vložení gravitační síly na objekt 38

11.2 Seznam tabulek

Tabulka 2-1 Nejznámější produkty CAx 4
Tabulka 4-1 Typy konfiguračních souborů menuscript 16
Tabulka 7-1 Časová náročnoststávajícího procesu 34
Tabulka 7-2 Časová náročnost navrženého procesu 34
Tabulka 8-1 Požadované licenční moduly 35

Použitá literatura

Autodesk. 2015. *Autodesk Developer Network - ObjectARX.* [Online] 2015. [Citace: 5. Leden 2015.] http://usa.autodesk.com.

FreeCAD. An Open Source parametric 3D CAD modeler. [Online] [Citace: 5. Leden 2015.] http://www.freecadweb.org/.

Journaling, NX. 2015. NX Journal Tutorials. [Online] 2015. http://www.nxjournaling.com/.

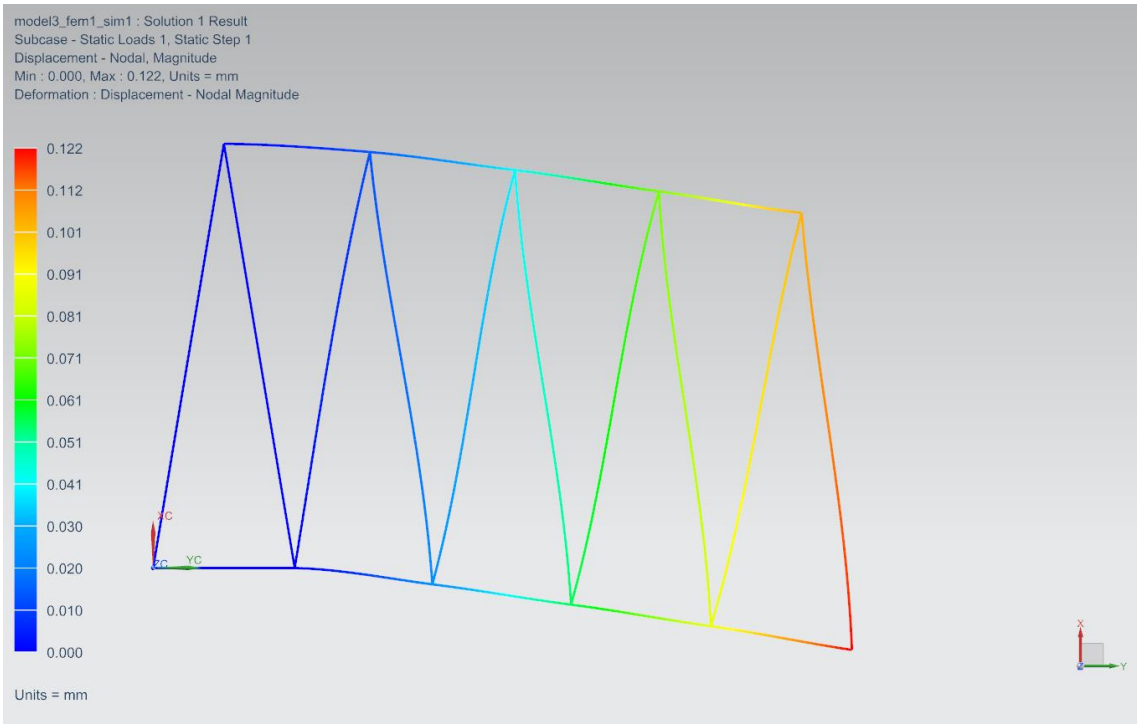
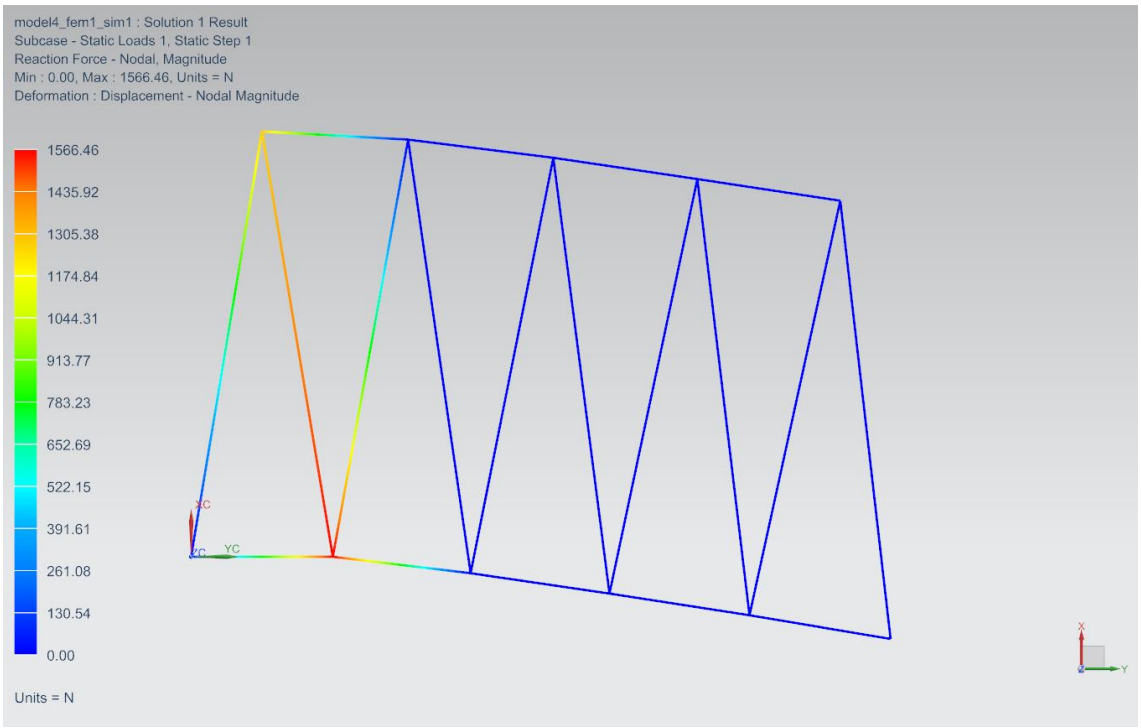
Siemens PLM. 2014. *NX Open for .NET Reference Guide.* 2014.

Siemens Product Lifecycle Management Software. 2014. *Siemens Documentation.* [NX Documentation] 2014.

Trávníček, Karel. 2013. Příhradová konstrukce. *Stavitelův blog.* [Online] 26. 7 2013. [Citace: 2015. 2 5.] http://stavitel.bloger.cz/Clanky/Co-je-prihradova-konstrukce-a-k-cemu-slouzi.

Přílohy

Výsledek simulace při rozměrech 50mm x 30mm a působící síle 1000N. Pro tento případ bylo využito více elementů na výpočet a v grafickém výstupu nastaveno 10% zvýraznění ohybu konstrukce. Reálný posuv lze odečíst z legendy.



Jiný typ výsledku, který je také součástí výpočtu NX. Jedná se o reakční síly v tom samém případě jako na obrázku výše (parametry konstrukce jsou tedy stejné). Znázorněné je, jak budou v konstrukci působit deformační síly.