

ES21869

Optimized 3D reinforcement in Complex Shapes Using Dynamo

Chris van der Ploeg
ABT bv.

Jelle Roks
ABT bv.

Learning Objectives

- Learn how to set up an integrated workflow between structural engineers, modelers/drafters, and fabricators
- Discover the possibilities to create and reinforce complex-shaped geometry in Revit using Dynamo
- Obtain and exchange rebar information for use in Finite Element Analysis (FEA)
- Learn how to optimize rebar configurations using Dynamo and FEA

Description

Deriving from 2D reinforcement drawings in the past, 3D reinforcement modeling is becoming common practice, and automatically generated reinforcement from calculations is becoming reality. The next challenges are optimization of reinforcement layout and dealing with complex-shaped geometry. This all starts with an integrated workflow, where information is preserved and exchanged along all disciplines—from modeling (Revit software), calculation, and documentation (SOFiSTiK software), to file-to-factory data exchange and on-site verification (BIM 360 Field software). We will use cases from practice to demonstrate this. Learn how to generate complex geometry and rebar parametrically using the Dynamo extension for Revit software. Dynamo software's parametric infrastructure in conjunction with the advanced 3D finite element method (FEM) analysis of DIANA FEA makes possible evaluation and optimization of structural performance. Hence, the realistic simulation of the structural behavior of complex-shaped structures in reinforced concrete is within reach.

Your AU Experts

Chris van der Ploeg is a structural engineer and computational design specialist for ABT Consulting Engineers, a multidisciplinary engineering firm based in the Netherlands. In 2013, he obtained his bachelor's and master's degrees from Delft University of Technology (TU Delft) in structural engineering. Van Der Ploeg is heavily devoted to helping advance the use of computational design in daily practice, not limited to the field of structural design only. He has a strong interest in applying parametric design and optimization strategies to increase insight into complex projects and achieve more efficient structures, and he won the international Oasys Project of the Year award in 2014 for his work. He has broad experience with parametric design software, including the Dynamo extension, and his programming and Revit software API knowledge enables him to extend functionality where necessary. He regularly shares his knowledge in meetings and conferences, including lectures about Building Information Modeling (BIM), Revit software, and parametric design at the TU Delft.



LinkedIn

<https://nl.linkedin.com/in/jcvanderploeg>

Jelle Roks is a Revit and BIM modeler specialized in 3D reinforcement and optimizations and programming in Dynamo. Since the start of his career at ABT Consulting Engineers Roks developed a lot of interest in optimizing workflows and processes. He obtained his bachelor degree in 2013 at the Hogeschool Arnhem and Nijmegen (University of Professional Education) in structural engineering and architecture.



LinkedIn

<https://nl.linkedin.com/in/jelle-roks-71344a29>



AUTODESK
UNIVERSITY 2016



ABT | Building ambitions

As an independent consultancy ABT adds value to the built environment - with regard to buildings as well as infrastructure, new building and re-use. We operate in the Netherlands as well as abroad. From our universal knowledge of technology and process we explore, together with our clients, the limits of the possible. This within the defined frameworks and while paying attention to sustainable and new developments.

ABT is a multidisciplinary consultancy which combines the knowledge of it's over 250 employees in a quick, flexible and goal-oriented way. This applies to projects of any scale and in any phase of the building process: from initiative to design, construction and exploitation. Our services are focused on the following disciplines: structural engineering, civil engineering, architectural engineering, building physics and installation technology. We also specialize in design, project and site management. In our opinion cooperation among these disciplines is self-evident. This way we reach high-quality and innovating solutions, coupling technique with knowledge of the process.

Our multidisciplinary approach helps us to be leader in the fields of sustainable building, renovation and transformation, BIM, parametric design and underground building. By means of cooperation and with passion for our profession we help the client to achieve his ambition. From vision to realization, from idea to building: ABT lives up to its clients' expectations.

ABT has offices in Velp, Delft, Haren, Düsseldorf and Antwerp.

More information:



Website

<https://www.abt.eu/EN/>



Twitter

<https://twitter.com/abtby>



Youtube

<https://www.youtube.com/user/ABTbv>

Content

Optimized 3D reinforcement in Complex Shapes Using Dynamo	1
Learning Objectives.....	1
Description	1
Your AU Experts.....	2
Content.....	4
Introduction in complexity	6
Optimisation of reinforcement	6
Complexity in design	7
A model for complexity.	7
Levels of complexity in the design of reinforced concrete	9
Dealing with complexity for optimization	10
Potential to use Dynamo for optimization.....	11
Use of dynamo in master and slave patterns.....	12
Integrated workflow of information exchange.....	13
Yesterday's workflow	13
Today's workflow	14
Tomorrow's workflow	16
Create and reinforce complex-shaped concrete elements in Revit using Dynamo	17
Creation of geometry	17
Unknown rules.....	17
Known rules	18
Creation of reinforcement.....	19
Case study: "The Couch" Roof of Tennis Paviljoen,	20
Case Study: Windmill foundation block	23
Obtain and exchange rebar information for use in finite element analysis.....	25
Differences in geometry for drawing and finite element models.....	25
Obtain rebar information from the Revit database	27
Obtain structural rebar data	27
Obtain structural fabric reinforcement data	30
Methods for exchange of reinforcement information.....	32
Standard IFC format or custom structured layout using XML.....	34
Intermediate file formats or direct link using APIs.	34
XML data exchange of reinforcement	34
Optimization of rebar configurations using Dynamo and FEA	38
Level of approximation in structural design	39

Calculation reinforcement using formula's	39
Evaluation of behaviour using finite element analysis (FEA) models	40
When to apply FEA	41
Workflow for optimization of reinforcement using FEA.	42
Modification of design variables.....	44
Feedback loop.....	44
Results from optimization	45
External control of Revit and Dynamo using journal files.....	46
Journal file definition.....	47
Use Dynamo to build a journal file.	47
External control of Revit	48
Optimization of rebar configurations using Dynamo for variation	50
Automation of reinforcement calculations using Dynamo	51
Simulation of multiple design variants using Dynamo.....	53
Optimisation of the number of unique reinforcement configurations using Dynamo.....	55

Introduction in complexity

Optimisation of reinforcement

Reinforcement is one of the main cost components of the reinforced concrete structures. Optimization of reinforcements, reducing the amount of material needed, contributes to realizing a cost effective structure. Not to speak of our common responsibility to reduce our material use to make more sustainable designs. In addition to these material savings, also reductions of man-hours for correct placement of rebar on-site should be considered.

By default, engineers try to determine design solutions that minimize the required amount of material (reinforcement) required given the design conditions using their knowledge, experience, and skills within the resources they are given. From the perspective of efficiency, simplified engineering methods are preferred over more advanced calculations methods as FEA to limit analysis time. Whereas the number of evaluated alternatives and differentiated elements checked is often limited to reduce the amount of work. Both compromising the optimal use of material.

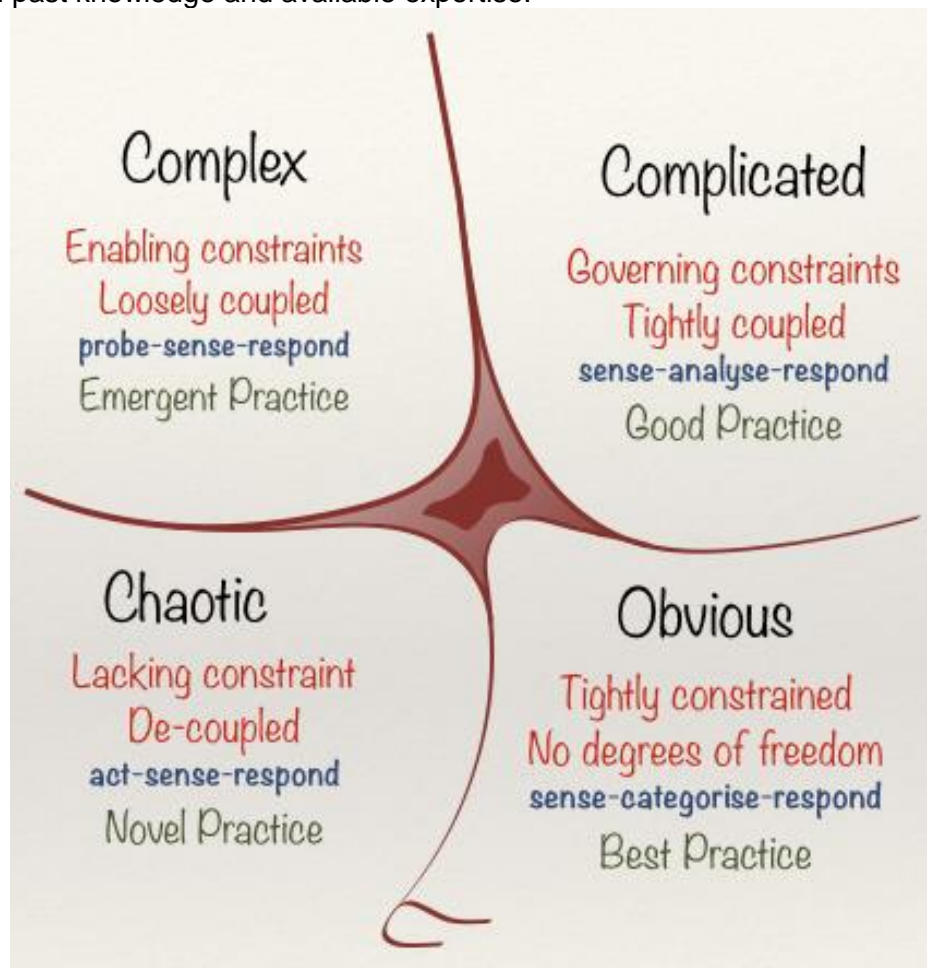
When the complexity of design increases, either due to the use of more complicated geometry, complex structural load bearing behavior or constraints from design or execution, the engineer's available knowledge may not be sufficient or efficient to provide optimal designs. Manually assessment these structures may require to significant effort or simplifications. And alternative approaches using computational simulation of design alternatives or the of more advanced analysis as finite element analysis (FEA) may be required.

Complexity in design

In our industry elements deviating from standard straight beams, columns or flat floors often easily perceived as complex shapes. The level of design complexity fully depends on the perception of the user or design participant. In reinforced concrete design the term complex shapes may either comprehend a complex geometry of the shape created by the architect, the complex structural behavior of the shape for the structural engineer, complex reinforcement layouts to draft and document for the draftsmen or a shape complex to make or understand by the fabricator.

A model for complexity.

A definition of the complexity of a system can be given using the Cynefin Framework which separates several domains of complexity. The Cynefin Framework (pronounced ki-neh-vin, Welsh, 'place of multiple belongings') by David Snowden is a decision framework that recognizes the causal differences between cause and effect that exist between different types of systems. It is divided into 5 domains: Simple(Obvious), Complicated, Complex, Chaotic and Disorder. In what domain a problem falls depends on past knowledge and available expertise.



CYNEFIN FRAMEWORK, SNOWDED 2014.

(https://commons.wikimedia.org/wiki/File:Cynefin_as_of_1st_June_2014.png)

Domains:

- Obvious (Simple): The relationship between cause and effect relations are well known and obvious to all. X always give Y. Problems are typically solved by categorisation and picking standard solutions. Best practice solutions are part of this. E.g. practical reinforcement for limited loaded beams, or placement of mesh reinforcement.
- Complicated: The relationship between cause and effect relation is direct but require analysis or some other form of investigation and/or the application of expert knowledge. E.g. calculation of required reinforcement.
- Complex: Cause and effect relation are so intertwined, that effects only be perceived in retrospect and cannot be predicted or quantified beforehand. It can be thought of as a connected network of relations with non-linear relations. These problems can be dealt with by means of testing or simulation. E.g. simulation of structural behavior by means of FEA models.
- Chaotic: In the Chaotic domain no perceivable relationship between cause and effect are available. Chaotic design or chaotic construction in the field of reinforced structures is should be avoided. No understanding of cause-effect relations compromises structural safety. And except for innovators and young children, no building takes without the intent of creation or without a predefined plan to learn from results. Disorder: state of the problem unknown

The last part of the model is the transition between the Obvious and Chaotic domain. This symbolizes the situation that a case is incorrectly categorized and dealt with as simple while more complex effects are ignored resulting in expensive recoveries (e.g. structural damage by neglecting assessment of design effects). For this reason, sufficient attention should be to giving problems in the complicated and complex domains if necessary.

Categorization of complexity is not considered to be rigid and may shift over time as result of the increase of knowledge or availability of new ways of approach. Alternative processes or new ways of dealing with problems are mostly dedicated to providing a transition in a clockwise direction to reduce complexity and making problems easier to deal with.

Refer for a more extend explanation to <https://m.youtube.com/watch?v=N7oz366X0-8>



Levels of complexity in the design of reinforced concrete

Complexity in reinforced concrete structures can arise from the fields of geometry, structural analysis, drawing and fabrication. All influence the overall complexity of the concrete structure. Below matrix provides an overview of examples:

	Design			Fabrication
	Geometry	Rebar modelling	Structural behaviour	
Obvious	Straight elements	Drawing of lines or use of default Revit rebar tools	Practical reinforcement	Use of standard bar configuration or rebar meshes
Complicated	Geometry derived discrete set of logical operations or rules	Reinforcement generated by rules	Reinforcement based on engineer's calculations	Interpretation of rebar drawings, insight in situation
Complex	Blob ("potato")	Trial and error way of modelling is non-effective.	Simulation or testing of structural behavior	Mockups or building with correcting measures
Chaotic	No definition or model means no geometry	No rebar modelling without a plan	No insight in cause-effect relations may compromise structural safety and is unacceptable	No application except innovator, art or children's playground

LEVELS OF COMPLEXITY ILLUSTRATED FOR ALL DISCIPLINES IN REINFORCED STRUCTURE DESIGN

Dealing with complexity for optimization

Optimal use of reinforcements means picking the best solution available. This means that either insight in cause and effect is required to determine the optimal solution by analysis (i.e. obvious or complicated problems) or sufficient design alternatives should be created by means of simulation (complex). For overall integral optimization for all disciplines, this means the insight the relation between cause and effect is required for each field.

Examples:

- For blob shapes without the definition of geometry, consequences of optimizing the shape cannot be evaluated since the influence of shape variations are hard to comprehend. Only by supplying new models for probe insight can be obtained.
- No rebar modeling can take place if the modeler does not understand how his actions may result in the desired rebar shape. Example: at the moment full free form rebar cannot be modeled using Revit standard tools (no obvious), using Dynamo with a rebar based on generated curve it can.
- The load bearing capacity of 3D structures where the reinforcement influences stiffness and the load distribution may hard to comprehend and additional capacity cannot be quantified without FEA simulation.
- Optimized reinforcement layouts can only be realized if the fabrication team understands from design documentation (cause) what they have to make (effect).

Dealing with complexity starts with recognizing of the level of complexity and picking a suitable method to evaluated.

Where the optimal reinforcement cannot be direct obtained from analysis or requires too much engineering effort, the design strategy from the complex domain should be applied: probe, sense, act. Or in other words: learn from simulation.



Potential to use Dynamo for optimization of reinforcement

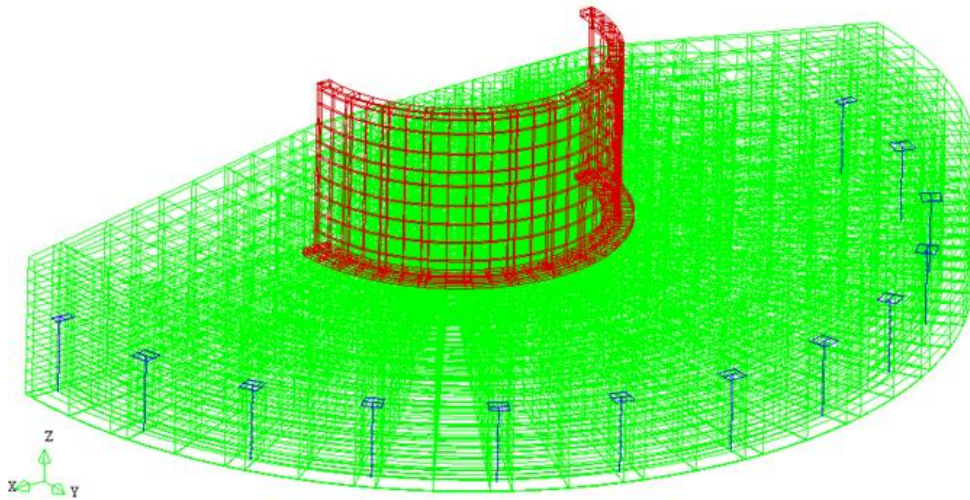
Two different approach for optimization of reinforcement can separated based on their level of complexity:

- Optimal solutions can be determined using analysis , e.g. the relation between cause and effect is known in advance to the engineer by formula's (complicated domain), but further optimisation comes at the cost of more engineering effort.
- Optimal solutions require the use of simulation using FEA to determine effect of modifications, so quantification of effects of variations (causes) only known after simulation (complex domain).

Application of more advanced analysis as finite element analysis (FEA) comes at the cost of engineering and analysis effort. Time spend to setup a full FEA model is significant longer than filling in a formula. A large part of the time required for advanced analysis is spent in the creation of the FEA models, in which in turn the majority is taken by the definition of rebar layout. Revit and Dynamo can be used to generate rebar definitions for reuse in the FEA package.

This stimulates the use of application of advanced FEA modeling in design the following ways:

- Less time required for setting up an FEA model.
- Model build up can be done by structural modelers instead of FEA specialists.
- Makes it possible to run multiple advanced analysis in cases where normally only a limited number of variants are evaluated



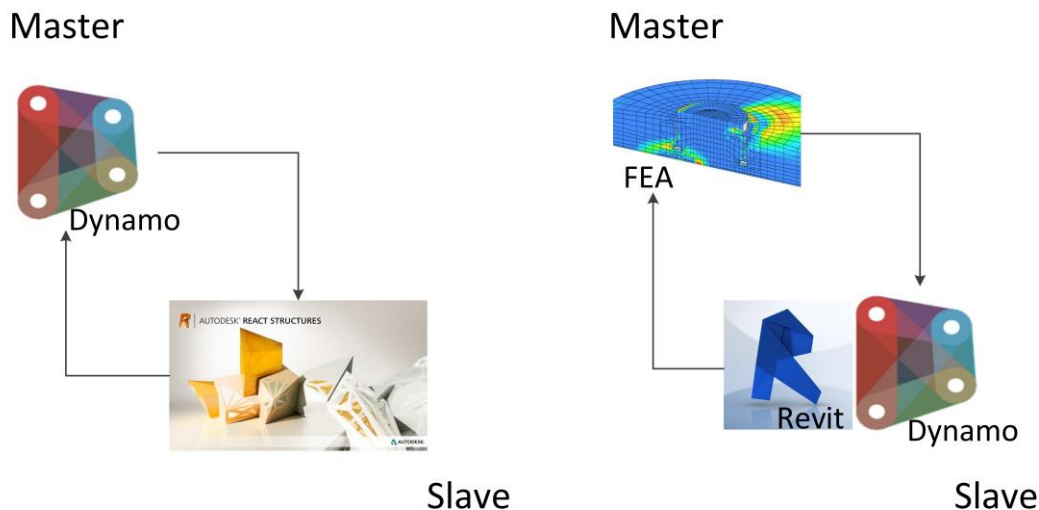
GENERATED REINFORCEMENT IN DIANA FEA

Summarized Dynamo can be used to:

- Automation design, allowing specialization in unique optimal solutions for individual elements without compromising engineering effort.
- Simulation of alternatives, providing insight in performance of multiple design variants
- Simplify setup of advanced calculations models in FEA.

Use of dynamo in master and slave patterns

Two distinctive patterns for the interaction of Dynamo with structural analysis software can be separated that depend on the perspective and purpose for which software will be used. Will Revit be used by the structural analysis software to simplify the generation design variants of reinforcement or will structural analysis software be used by Revit to assess design variants in order to document the best final solution? This can be referred to as a master and slave relationship. Which one serves as master or slave depends on the application.



*DYNAMO AS MASTER (RIGHT) IN CONTROL FOR SIMULATING DESIGN VARIANTS.
DYNAMO AS SLAVE (LEFT) TO PROVIDE INPUT FOR ADVANCED DESIGN SIMULATIONS AS FEA.*

Many structural analysis software packages, such as Robot Structural Analysis and React Structures have an API. This means also that this functionality can be included on the Revit side into for example Dynamo Node definitions, thereby making it possible to include this structural analysis software in your Dynamo Workflow (Revit master, Structural analysis software as slave) to assess the performance of alternatives. In the case of evaluation and interpretation of advanced FEA models, the expertise of a FEA specialist is required. His work can be simplified when FEA models are easier to setup and can be feed with reinforcement information from Revit. FEA software serves then as master, Revit as slave.

At this class, we will describe how Autodesk solutions such as Dynamo contribute to optimization and in particular the optimization of reinforcement in complex and complicated structures at our office for all four of these scenarios:

Create complicated geometry and rebar with dynamo

- Exchange reinforcement data for reuse in FEA packages using Dynamo
- Optimize reinforcement by FEA simulations making use of Revit and Dynamo as slave for geometry
- Optimize reinforcement by variation of design parameters using Dynamo

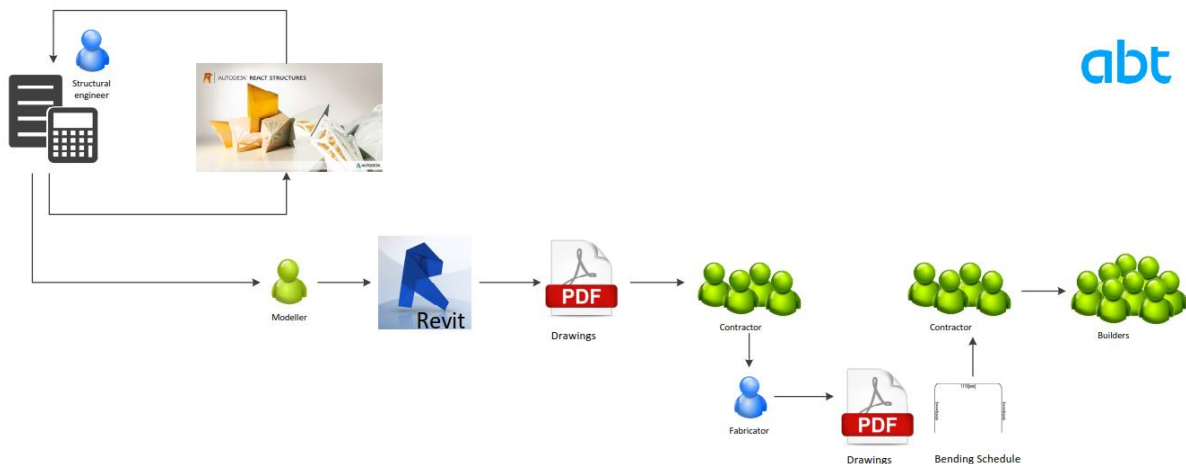
This all starts with an integrated workflow between all disciplines related to exchange of design data.

Integrated workflow of information exchange

To set up a properly integrated workflow of information exchange between all disciplines from tender stage to construction phase you first need to understand the old fashioned way of working and more important, why the building industry is still working this way. All participants have many (different) demands, most of the time depending on the entry point during the construction process.

Yesterday's workflow

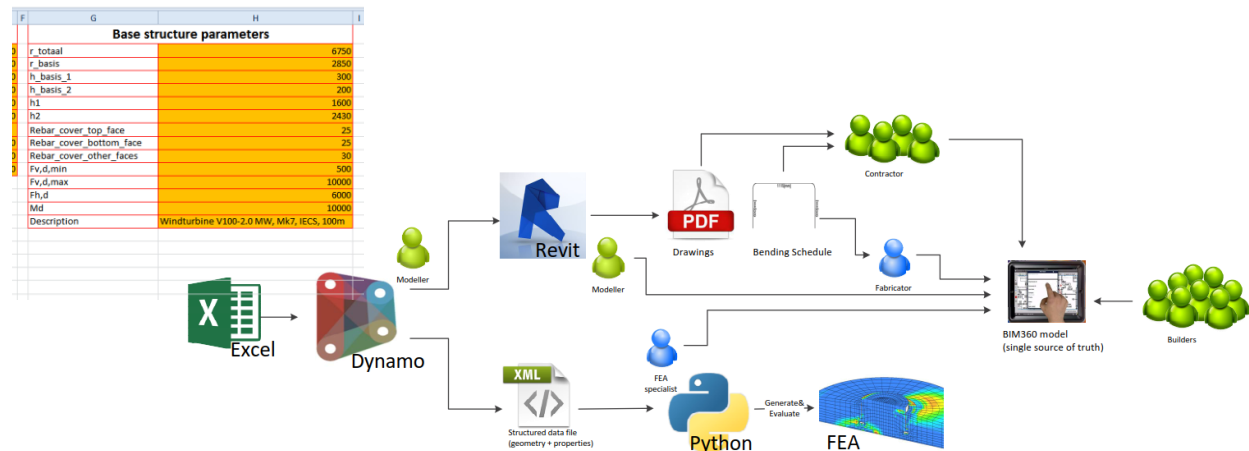
The current process looks like the below. During the tender stage, the structural engineer makes a structural design with some global calculations and sketches. Then the modeler starts building the BIM and during the different phases (primary design, definitive design etc.) the BIM becomes more and more detailed. Every change in dimension need to be adjusted by hand in the calculation model (FEM or similar) and in BIM. To start up the construction phase the modeler starts to produce shape and reinforcement drawings, all in 2D. When finished the drawings go to the contractor and the manufacturer who transforms it into a fabrication drawing. Meanwhile, the contractor starts to check all the incoming drawings before he plans his man to actual start building.



LINEAR DESIGN PROCESS

Today's workflow

From 2015 ABT bv. started to improve the workflow for their consultancy in windmill foundations between all the different parties, to get an integrated workflow from start to end. The main goal was to reduce to costs of failure due to loss of information. We introduced some new software in the workflow (BIM360 products, SOFiSTiK reinforcement detailing) and removed most of the manual adjustments and checks to get a workflow according to the Single Source Of Truth ideology. SOFiSTiK reinforcement detailing is needed to translate the 3D reinforcement into 2D drawings and to generate bending schedules. The plug-in generates several file extensions which can imported directly into the bending machine.

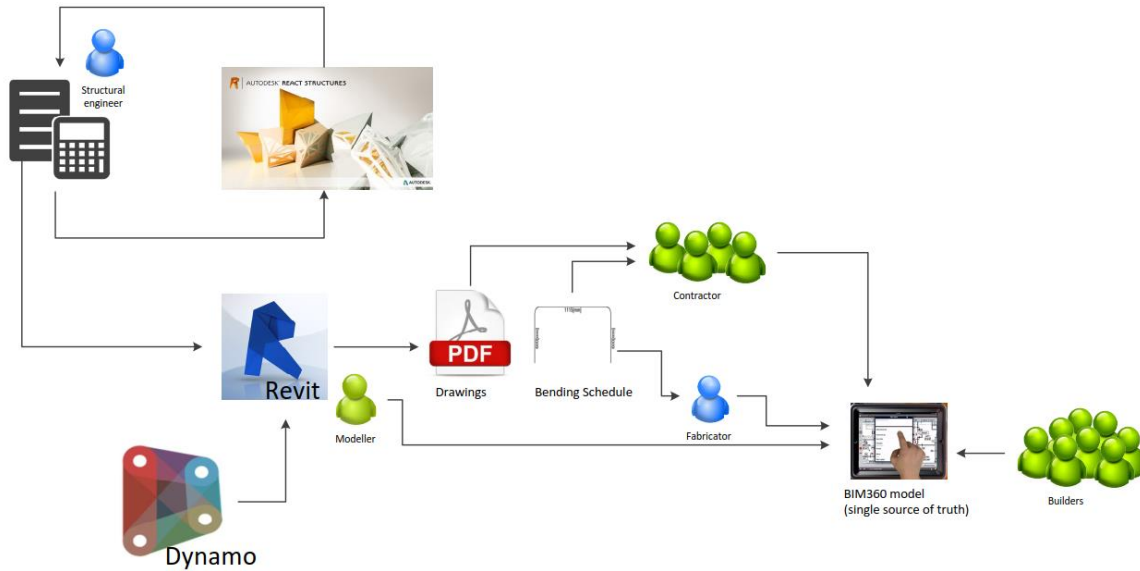


OPTIMIZED WORKFLOW, DYNAMO READS OUT A SIMPLE TO USE EXCEL FILE TO DRIVE THE PARAMETERS

Repetition is also a key-word for programming in general. If we take a look at the normal building process (like image 1), we can say that for most of the projects the single source of truth method is also applicable. For every building, bridge or dam we make, we start the design from scratch because none of the projects are the same. Every time a new prototype with new and different problems. This prototyping asks for a different approach than the repetition in the windmill foundations, but that doesn't say you can't use some of the programming. It is not all about creating geometry and reinforcement for FEA models. ABT used Dynamo to optimize individual parts of the process, to fill parameters, to check BIM models, to create sheets, to tag elements and even to get the correct quantities for environmental calculations (CO2 emission etc.).

clearance height above stairs_1.2.0.dyn	14-6-2016 10:36	DYN-bestand	329 kB
create_3Dview_per_level_1.0.0.dyn	1-6-2016 10:40	DYN-bestand	10 kB
create_sheets_1.0.0.dyn	18-3-2016 11:57	DYN-bestand	456 kB
organize viewports_0.9.0.dyn	17-3-2016 17:00	DYN-bestand	13 kB
place_piles_0.9.0.dyn	15-3-2016 16:32	DYN-bestand	2.391 kB
place_sonderings_1.0.0.dyn	8-4-2016 17:11	DYN-bestand	16 kB
Place3DRoomTextOnRoomInsertionPoint_1.0.0.dyn	11-8-2016 12:59	DYN-bestand	3.162 kB
rename_PDF_files_1.0.0.dyn	1-2-2016 14:52	DYN-bestand	22 kB
renumber_piles_0.9.0.dyn	1-4-2016 11:15	DYN-bestand	8 kB
set_viewname_by_string_1.0.0.dyn	17-3-2016 14:38	DYN-bestand	29 kB
write_durability_parameters_1.1.0.dyn	17-1-2016 21:48	DYN-bestand	109 kB
XYZ_piles_0.9.0.dyn	25-3-2016 15:38	DYN-bestand	17 kB

DYNAMO SCRIPTS USED BY ABT IN THEIR "NORMAL" PROJECTS



WORKFLOW "NORMAL" PROJECT WITH DYNAMO TO SPEED UP THE PROCESS

These scripts result in the flowchart stated above. Some links in the chain have disappeared and has been replaced by a more parallel process with one output source: BIM360. At the moment ABT uses the BIM360 product in projects as a pilot parallel with the 2D drawings.

Hand in hand with optimized concrete structures becomes construction according to specifications of greater importance since the used model is a better approximation of reality and additional overcapacity is reduced. It is thereby to the full design team to make decisions how to mitigate risks for the sensitive parts of the design. Whereas simulation of alternative rebar configurations may help to identify sensitive areas. Full 3D insight in rebar configurations contributes to on-site understanding of the structure to make. BIM Field brings that insight to the construction place. Thereby making it more likely that the structure will be created according to specifications.

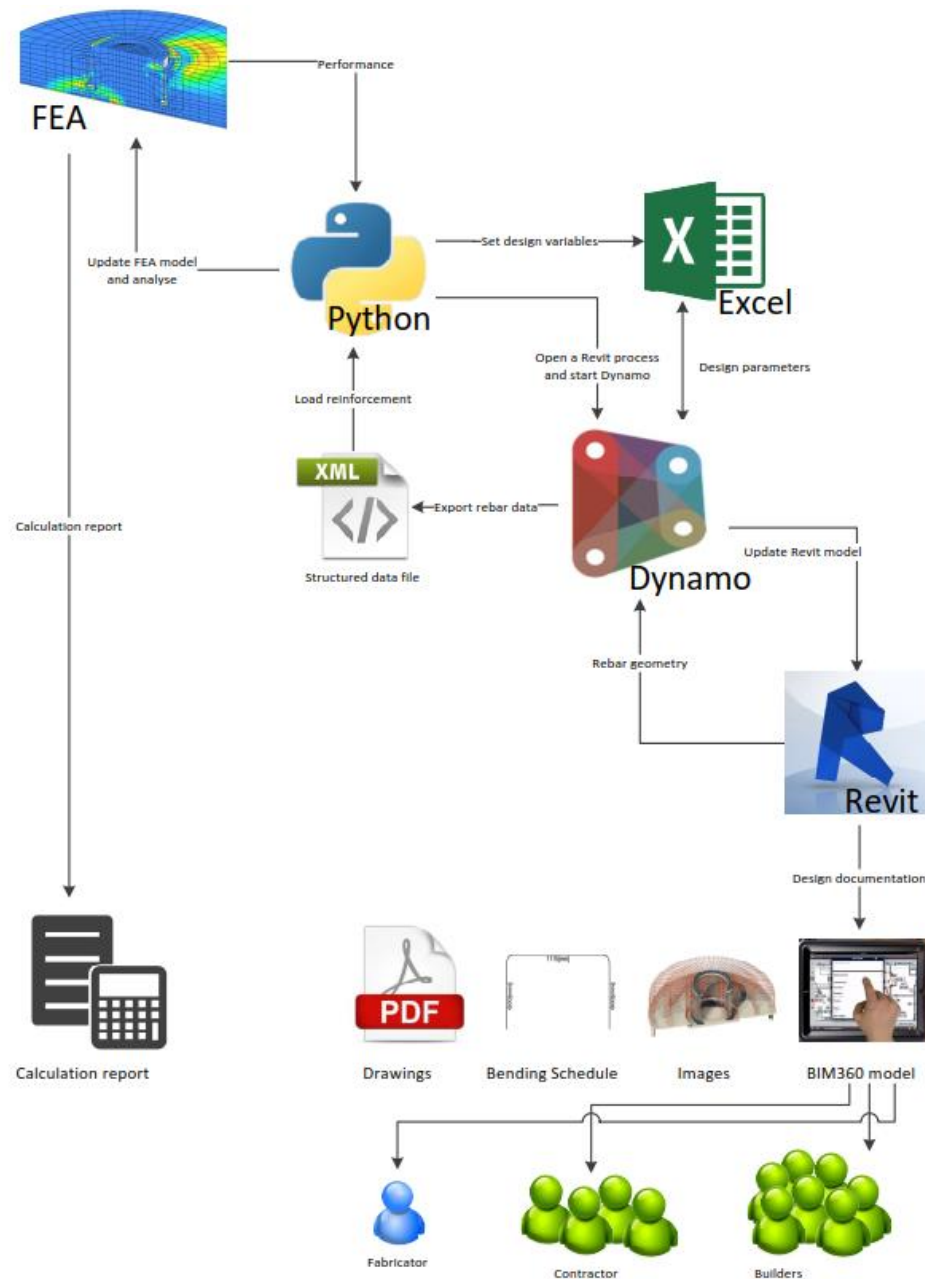


BIM360 MODEL FOR USE ON-SITE OF WINDMILL FOUNDATION REINFORCEMENT

Tomorrow's workflow

The optimization ABT started for the process of the windmill foundations (and similar projects) is an ongoing process. Currently we made a prove of concept for a bidirectional coupling between FEA, Revit an Dynamo.

The main advantage of this workflow is that there is no loss of information during the design phase and every change somewhere in the circle directly affects the other links in the chain.



LOOPEd WORKFLOW FOR OPTIMIZATION

Create and reinforce complex-shaped concrete elements in Revit using Dynamo

Creation of geometry

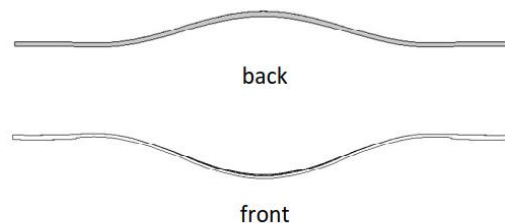
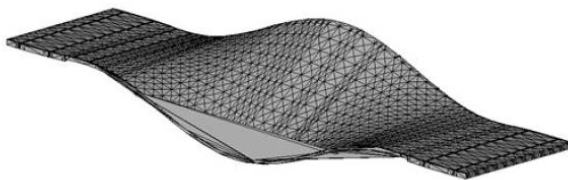
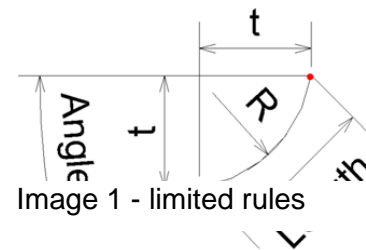
To create a shape in Revit you have several possibilities like families, in-place-models or free form import (from other software). You can also generate geometry using dynamo, but that is a bit devious for the standard geometry (beams, pile caps etc.) and misses the target of Dynamo.

The families we use in the BIM are (most of the time) parametric in the way that we can control the length, width, height, material and maybe some extra meta-data. In other words: "The rules of the geometry are known and limited".

When we take a look at complex shapes, those rules aren't that simple anymore and sometimes there are no rules to describe the shape (BLOB). We can make a difference between complex geometry with unknown and with known rules.

Unknown rules

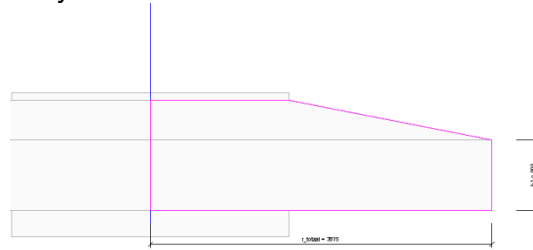
If we talk about geometry with unknown rules the best example is an import from third-party software, imported in a Free Form Mass family. If you are lucky you get a proper IFC with some intelligence which, such as structural material etc. Worst case is that you import geometry Revit can't convert into a mass and you're stuck with empty DWG. In that case, you have to make an overlay which matches only this version of the geometry, see image below.



For this form it is quite easy to create a rule based family (inner radius, outer radius, height, thickness, maybe driven by sine or cosine), but after all it will always be an approximation of the original form.

Known rules

For known rules, it is important to make a difference between simple and complex shapes. For simple geometry it is easy to use the standard family templates. This on the condition that geometry can be made a solid sketch (revolve, blend, extrusion etc.). When you want to create these simple shapes in Dynamo you need to describe them with a mathematical approach, which takes you much longer than let Revit do the maths for you. For the windmill foundations ABT used the structural foundations family to create the shape (simple revolve) and used Dynamo to drive these parameters. The advantage of this way of working is that Revit implants a lot of standard properties (like cover settings) into families which you, for example, can use in the next phase: reinforcing the geometry.



REVOLVED STRUCTURAL FOUNDATION FAMILY

Whereas a free form concrete the shape (blob) can be specified, rebar calculated, specified and fabricated, the complex level makes that optimization of geometry reduce reinforcement hard. The cause –effect relation of geometry modifications on structural performance, drawing complexity of reinforcement or fabrication are more difficult to predict and can often only be understood in hindsight. Thereby reducing the potential for optimization or requiring additional work of other participants for development of multiple design variants.

In our project we advise clients and architects to rationalize their complex shapes in basic mathematical operations or rule based generated geometry to have insight in how the form was created, rather than BLOB (“potato”) shapes without background. Using dynamo, geometry can only be created using straight forward logical operations. Any shape derived with dynamo can therefore be classified as complicated as rule based. Although, large parametric models may limit full understanding of cause and effect relation and thereby shapes may be perceived as complex. With insight in the shape derivation, effects on the complexity (and related costs) of individual building elements to be made can be derived by reasoning.

Creation of reinforcement

To model the reinforcement in a random geometry given by the conditions stated in the last paragraph, we can identify similar conditions. Reinforcement in Revit can be made within the functionality of Revit, with the Revit Extensions or with Dynamo. The Revit extensions are great for standard shapes like columns and pilecaps, but not for handmade families or in-place-models. The standard functionality of Revit for modelling 3D reinforcement let us choose between (single) bars, based on an area, according a path or fabric sheets. For flat or straight geometry these functionalities are enough. But when the shape differs in any way of that the manual operations starts growing, without even talking about changes in the design. To model reinforcement which has connection with the geometry and is not limited by the shape Dynamo is the perfect solution. Dynamo has can generate reinforcement in several ways:

- Point to point
- Copy and adjust manually placed bar (rebar container)
- Related to face (flat, curved)

For all these options it is important to relate the amount, shape and length of the reinforcement to the geometry so we can only use Dynamo to generate the bars if the rules of generation are known. It has to be said that the new nodes from DynamoRebar (<https://github.com/tt-acm/DynamoForRebar/tree/master/src/DynamoRebar>) made the world a lot brighter when it comes to generating reinforcement. The latest package includes the nodes for FollowingSurface, Perpendicular, Morphed and cut by plane. The image below gives an example of reinforcement placed by Dynamo on a double curved floor using the “Rebar.FollowingSurface” node. The other ways of generation are explained by a case study of ABT.

Case study: “The Couch” Roof of Tennis Paviljoen,

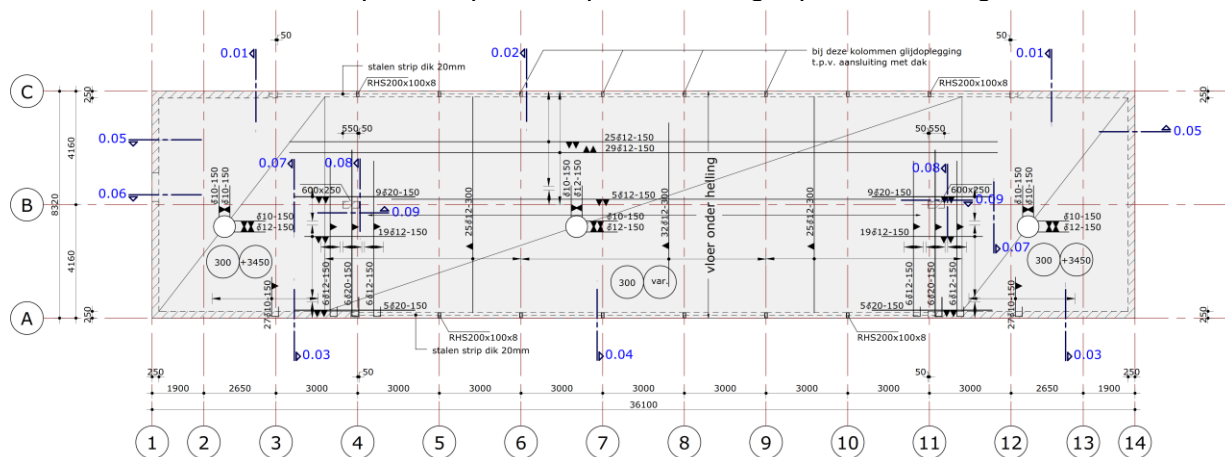
The pavilion at the Tennis Club IJburg, Amsterdam, realized in 2015, has as roof that functions both as the cover for the building below and as stand for all viewers to the sport match. The layout was designed specifically to improve viewing angles to the court. The building was designed by MVRDV and was constructed by Romijn Bouw and Vericon. ABT was involved as structural engineer.



RENDER OF THE COUCH (MVRDV)

In the design process the shape of the building was digitally produced by the architect and exchanged as “mass object” variation for further use in structural engineering. Reinforcement design was evaluated using structural analysis software and hand calculations.

Back in 2013 reinforcement design was mainly provided 2D drawings. As for this project where the reinforcement of the complex shape was specified using top view drawings.



If such a structure should be reinforced in 3D today by hand making use of standard Revit design tools, the modeler is up to challenge due to the varying shape of the building. Which would classify this shape as complex to reinforce.

Complex to model does however not mean complex to fabricate as illustrated below. Partly the simple reinforcement layout, layout out the rebar onsite was no problem.

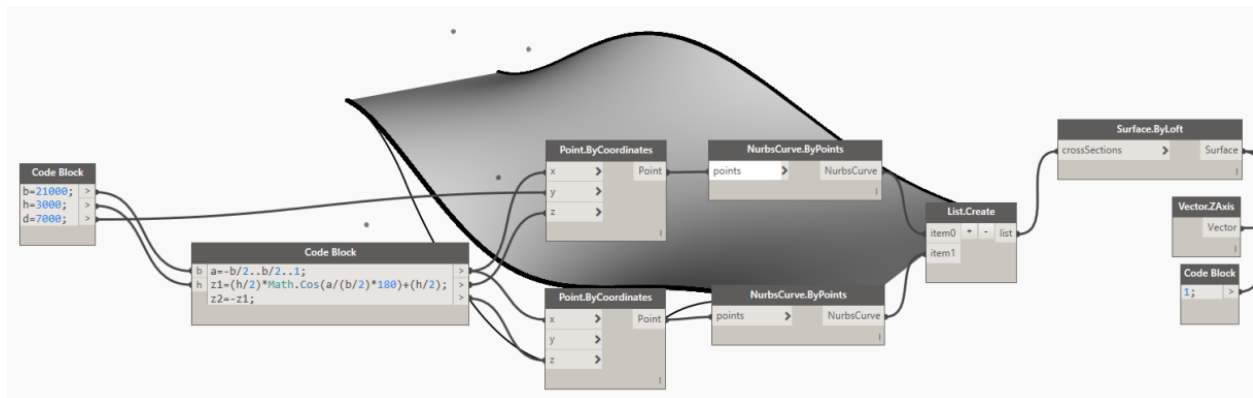


POURING THE CONCRETE ON THE COUCH. APPLYING A SIMPLE GRID OF REINFORCEMENT ON A COMPLEX SHAPE DOES NOT INCREASE COMPLEXITY OF CONSTRUCTION.

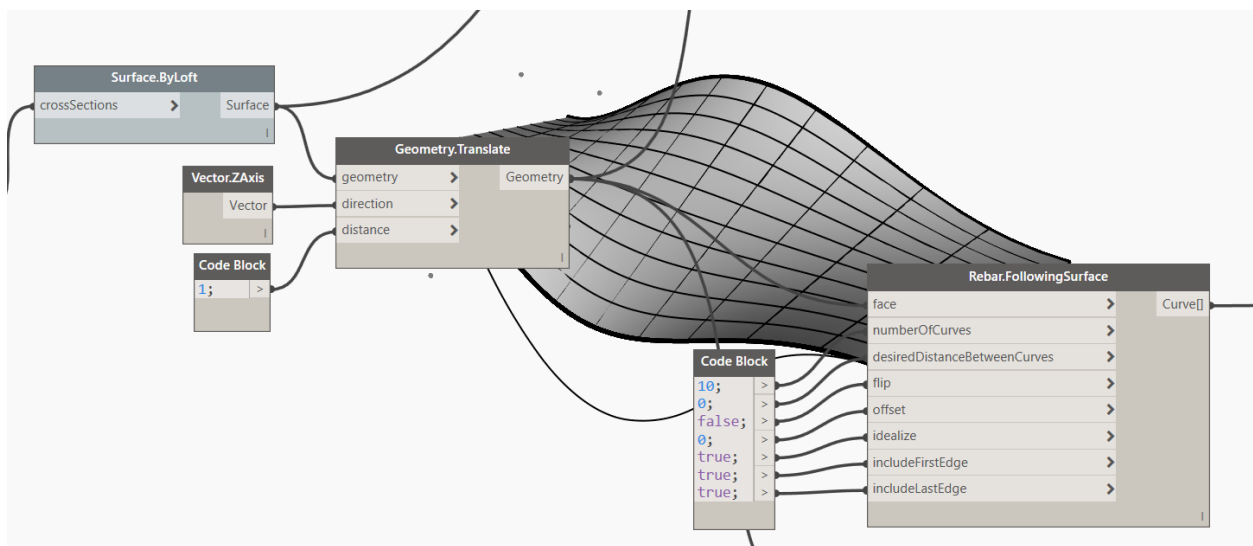
Providing the reinforcement design in 3D would however had simplified the quantity take-off of amounts of reinforcement required.



Given the straight forward shape of this building the geometry can nowadays easily be derived based on rules using Dynamo. The shape of the roof is based on two sinus shaped curved, one upwards (the back) and one downwards (the front). The surface in between is a ruled surface, with straight lines in one direction and curved lines in other. In the middle the roof is straight.



Starting this approach, rule based also allows us make use of rule based generation of reinforcement.



Another challenge which is easier solved in reality than 3D modelling is how to deal the varying number of bars required to limit bar spacing at 150mm. For the middle section that is under an angle almost $\sqrt{2}$ more bars are required than on the flat ends. This could simply mean that all bars from the middle section are extended till the end and effectively about $\sim 100\text{mm}$. Resulting in a waste material. In reality this was easily solved on site by just simply not do not elongate some of the bars at bar splices according to onsite insight. In 3D modelling this is time consuming to do manually and would require additional rules.

Case Study: Windmill foundation block

At ABT we have designed a large number of wind mill foundations over the last few years. This provides a continuous drive to improving the efficiency of the engineering process as the performance and optimality of our designed structures.



FOUNDATIONS FOR WINDTURBINES. LEFT: ALEXIA, THE NETHERLANDS. RIGHT: LAKE TURKANA, KENYA

Visualization windmill foundation process

<https://www.youtube.com/watch?v=aPiReCY1HZo>

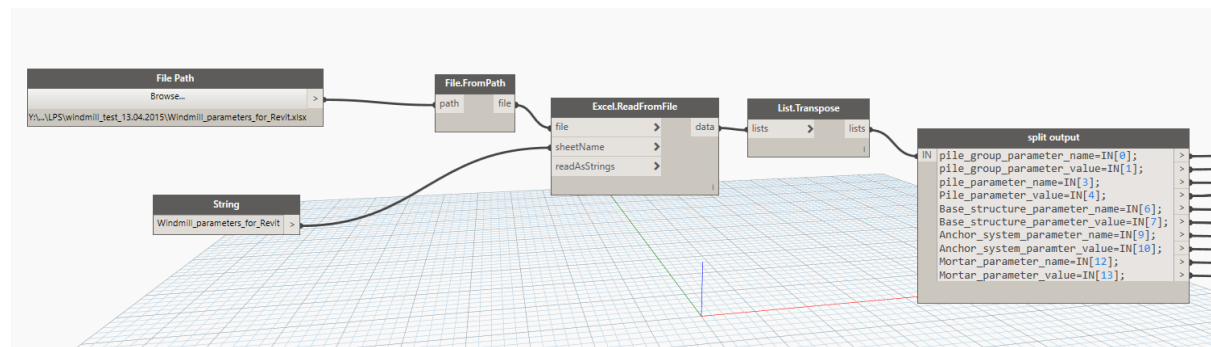
Generation reinforcement by Dynamo:

<https://youtu.be/G9DG7ENuaWg>

In this example we use a step-by-step explanation starting with the generation of the geometry. For the windmill foundation design we use a combination of the nodes from DynamoRebar and adjust method.

Step 1: Geometry

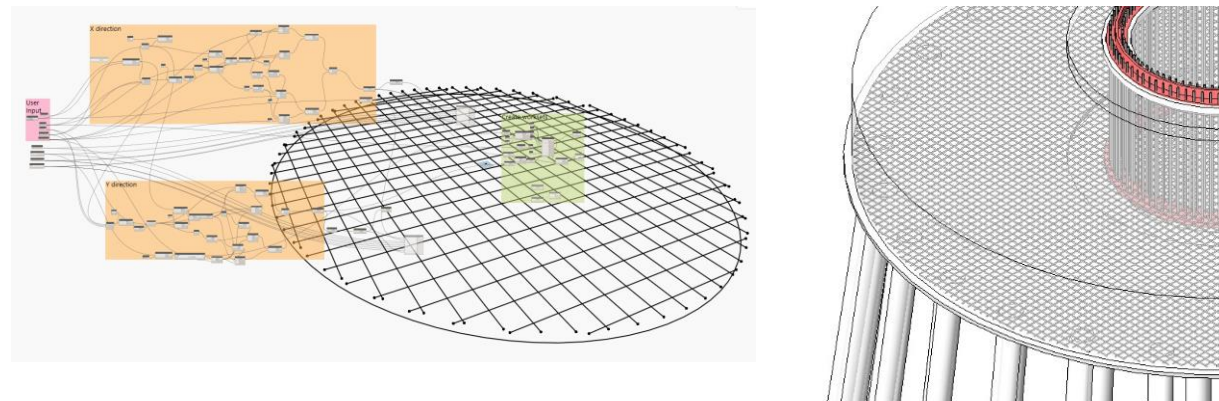
As a start we use a specially built template file for the windmill foundations. All the families required for common shapes, elements and rebar are all preloaded in this template. The basis of the geometry is an Excel file which is a database with information about previous projects. For the first run the structural engineer makes a global calculation based on experience and previous design data (for shape and reinforcement). An Excel input sheet with all this information be used by Dynamo to adjust the families.



Step 2: Bottom reinforcement

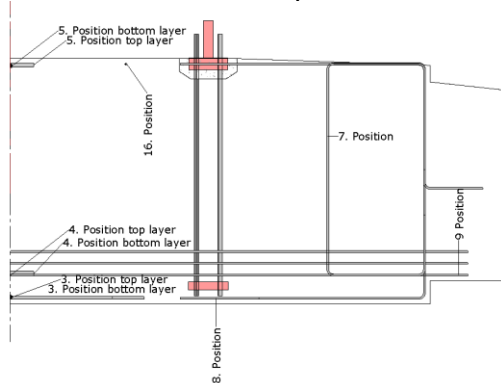
It is possible to generate all the reinforcement by using one Dynamascript, but it affects the speed and the stability of the problem in a bad way. That's why we use 3-4 scripts to generate the different layers in the model.

The bottom reinforcement is created by the point-to-point method where we define points on a circle with a specified cover. Creating net reinforcement with unique bars, taken into account the maximum stock length of the bars. This process needs to be repeated with to define all the single curves. The node "RebarByCurve" translates these lines into rebar.

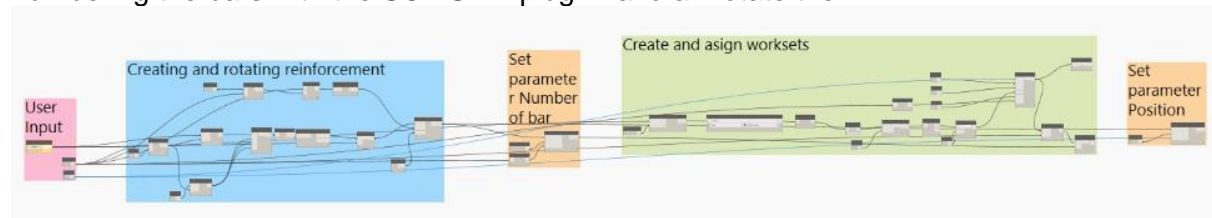


Step 3

For the stirrups and hairpins we use a different method. All the bars have a unique position in the foundation and all these positions have known rules related to the geometry. At first we generate the correct shape and then we rotate it from the center point.



We use worksets to visualize and filter the reinforcement on the drawing. That's why each script that generates reinforcement fills in the correct workset according to the position. The template has predefined viewports on the sheet so the only action that needs to be done manually is numbering the bars with the SOFiSTiK plug-in and annotate them.



Obtain and exchange rebar information for use in finite element analysis

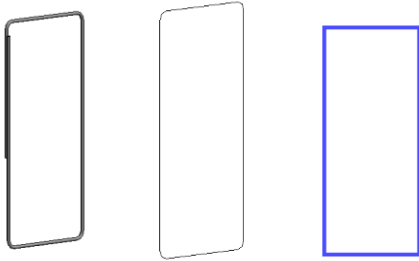
The relevant rebar definitions required for finite element analysis consists of two parts:

- Geometry of the rebar (curves)
- Properties of the rebar: bar diameter and material.

This information is required to define the reinforcement in the structural analysis software.

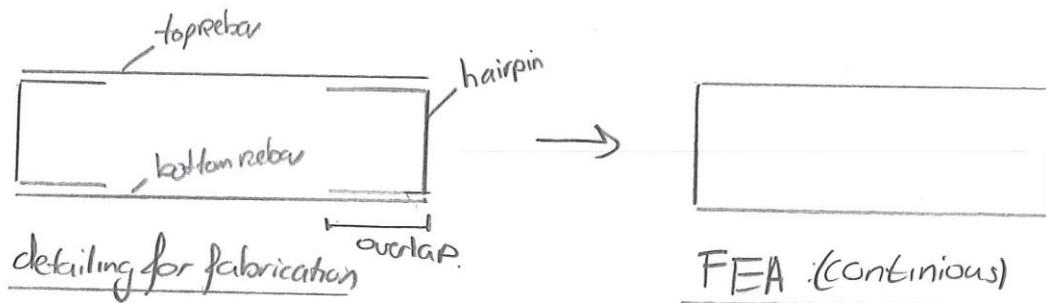
Differences in geometry for drawing and finite element models

The geometry of reinforcement drawings for fabrication and reinforcement geometry layout for analysis differ at the way how they deal with non-continuous rebar elements, overlaps and anchoring of rebar on the concrete.



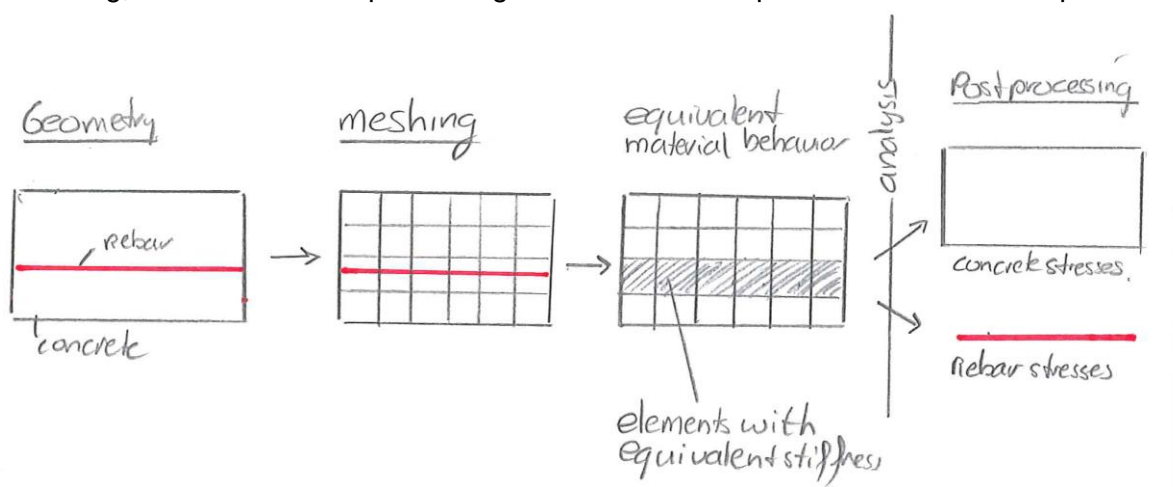
EXAMPLE 1 STIRRUP: OVERLAP WITHIN A REBAR SHAPE (FINE WITH OVERLAP, COARSE WITHOUT OVERLAP, HOW TO USE IN FEA)

Using the adjust intersection option in the export of geometry, the coarse bar layout can be exported instead of the fine bar layout. Thereby overlapping elements are eliminated. Suppressing of bends may eliminate the bending radius since this will have no effect on the overall behavior of the structural model. The overlap is then reduced and the rebar is continuous.



EXAMPLE 2: BEAM WITH HAIRPIN END ANCHORING. OVERLAP BETWEEN ELEMENTS.

In the calculation process inside the FEA software the reinforcement will be combined with the concrete material and converged to a finite element mesh with equivalent material behavior for calculation. So that both reinforcement and the concrete undergo the same strain. In the post processing, the behavior is separated again into a concrete part and reinforcement part.



The FEA package thereby assumes that the included reinforcement is fully effective over the full length. Effects of built up of anchoring lengths to build up friction for tensile stresses thereby not taken into account.

These problems can be dealt with in four ways:

- Neglect overlapping and use all as individual elements. This is sufficient the governing location is outside of the area affected by anchoring length reductions and therefore results are not affected. (i.e. capacity for max bending moment in middle of span of example 2, not affected by overlapping of rebar).
- A separate dynamo model or additional dynamo output can be made that generates both the geometry for FEM analysis as the layout for fabrication.
- Post processing of the reinforcement geometry to recognize overlaps and combine curves of elements. This could be solved using a Dynamo script that combines curves if the centerlines are closer to each other than the bar diameter.
- Geometry obtained from rebar elements can be shortened to exclude the required length for anchoring before becoming effective. This can either be done in a) Revit by drawing a reinforcement variant for structural analysis purposes only, b) shortening in Dynamo before export or c) shortening of the rebar elements on the FEA side.

For simplicity, the first alternative is preferred for export although requires expert insight if this is allowable for the specific situation. We use adjustment functionality on the FEA side (4c) to give the FEA export control over the behavior. Using geometry generation of reinforcement by Dynamo (2), additional continuous curve output can be easily created since all individual curves are available. Post processing (3) will be implemented in the future.

Obtain rebar information from the Revit database

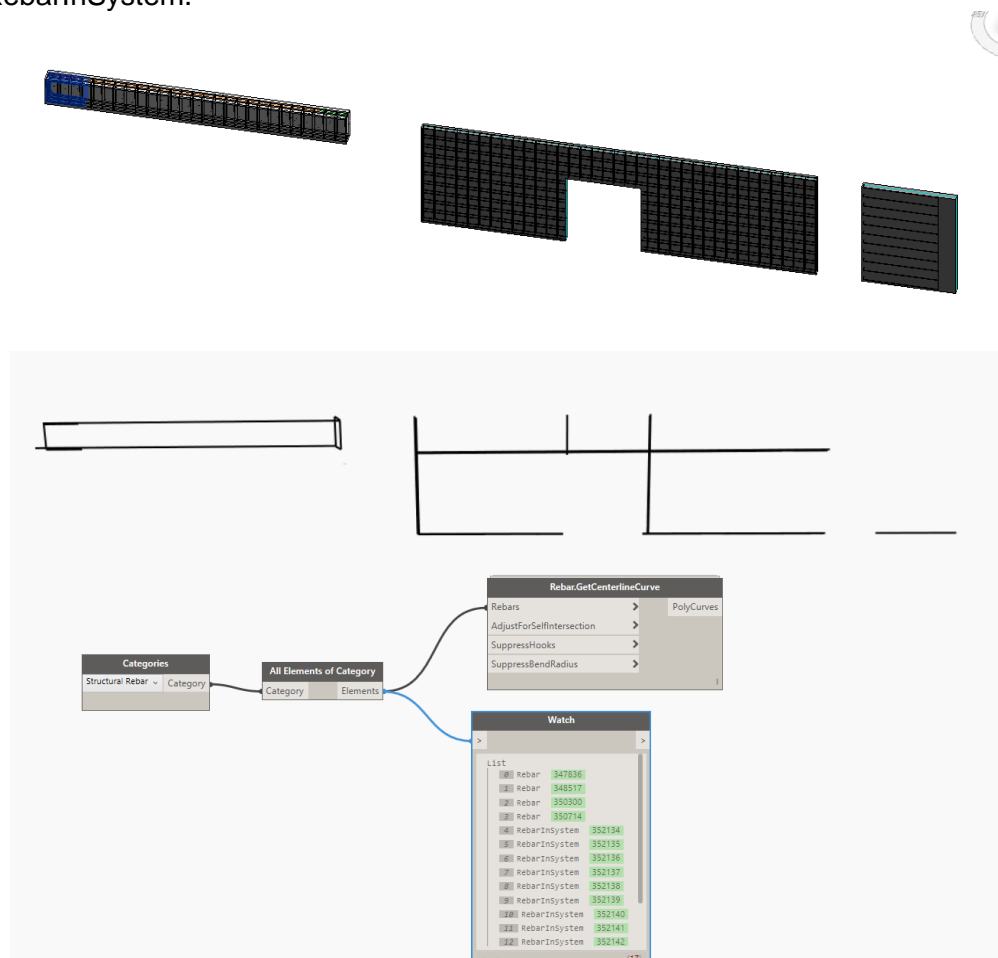
In Revit reinforcement can be generated using individual rebar, area reinforcement, path reinforcement or fabric areas and fabric sheets. In the data definitions of the underlying database of Revit individual rebar elements (“Structural Rebar”) and rebar sheets (“Structural Fabric Reinforcement”) are two defined different elements. Contrary as might be expected fabric sheets are not build up from individual Rebar but have their own definition. This means that in order to obtain rebar data both types should be accessed separately.

All required information can be obtained from the Revit database. Functions to provide access to this information are provided Revit API. For documentation of all available commands, take a look at the documentation at <http://www.revitapidocs.com/>

Obtain structural rebar data

Structural rebar geometry:

Individual bars, rebar by path or rebar by area created rebar elements are of the category “Structural Rebar”. Inside this category, two classes of elements are present: Rebar and RebarInSystem. RebarInSystem means that it is a Rebar Element which is part of a Structural Reinforcement Path or part of Structural Area Reinforcement. As long as the area or path system is not removed, these elements are classified as RebarInSystem.



FROM LEFT TO RIGHT: BEAM WITH INDIVIDUAL BARS, AREA REINFORCEMENT AND PATH REINFORCEMENT WITH CENTERLINE OUTPUT FROM DYNAMO USING THE BIM4STRUCT.REBAR PACKAGE

The Rebar geometry can be obtained by accessing the rebar centerlines. The node `Rebar.GetCenterLineCurve` is part of the `BIM4Struc.Rebar` package.

In extraction of the centerlines the follow options are provided:

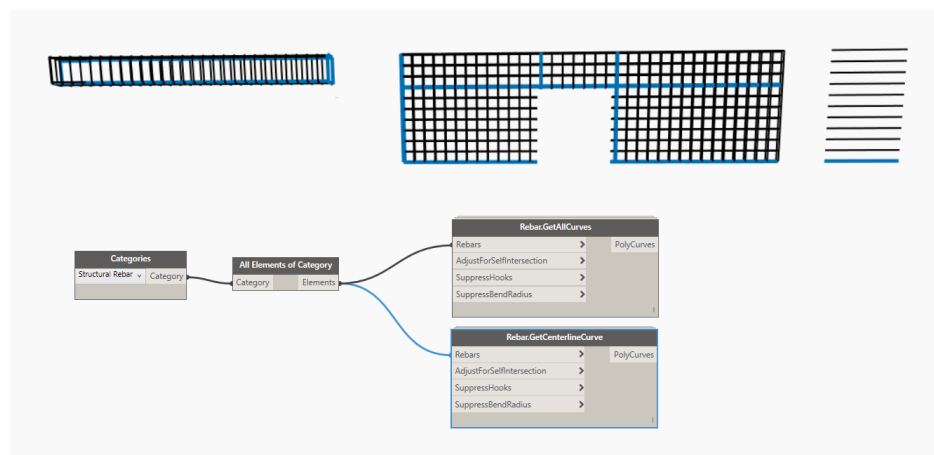
- Adjust self-intersection: If the curves overlap, as in a planar stirrup, this parameter controls whether they should be adjusted to avoid intersection (as in fine views), or kept in a single plane for simplicity (as in coarse views).
- Suppress the hooks: Identifies if the chain of curves will include hooks curves.
- Suppress the bend radius: Identifies if the connected chain will include fillet curves.

Depending on the detail level of FEA analysis bending radii may be ignored, which is the case in the evaluation of overall structural behavior. As addressed before, FEA analysis does require closed continuous bars (i.e. stirrups to be continuous) instead of overlaps, so self-intersection is adjusted, only the basic shapes are used.

As you can see in the dynamo output not all rebar geometry is obtained, but only each first element of the set. FEA analysis requires all rebar to be exported. In Revit 2016 the geometry of all other curves can be extracted by means of applying a transformation over the centerline curve geometry ("move the geometry of the first curve to the location of the second curve") for all rebar in the set. This reduces the amount of data to be stored in the Revit database but requires geometry operations to obtain all rebar geometry.

This requires the following steps:

- Get the centerline curves of the first bar.
- Obtain the number of possible bars in the set by calling the number of bar positions. Up to Revit 2016 the number of rebar in the set is to the quantity parameter. From 2017 onwards, the number of positions is equal to the number of actual bars (the Quantity), plus one or two more positions since the first and the last position are optional (Revit 2017).
- For each bar position:
 - Check if there is a rebar at the bar position
 - Determine for each bar position in the set the geometry transformation
 - Applied the geometry transformation to the centerline curve.



ALL DERIVED BAR GEOMETRY

```

#Obtain all the curves of all rebar elements in the set.
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
# Import ToDSType(bool) extension method
clr.AddReference("RevitNodes")
import Revit
clr.ImportExtensions(Revit.Elements)
# Import Geometry conversion extension methods
clr.ImportExtensions(Revit.GeometryConversion)

#Inputs
IN[0]
bool1=IN[1] #adjust intersection
bool2=IN[2] #suppress hooks
bool3=IN[3] #surpress bendradius

crvset=[]
RebarCurves=[]
RebarCenterLineCurves=[]

#loop all rebar elements
for r in IN[0]:
    rebar=UnwrapElement(r)
    #get centerline curves from the selected rebar objects
    sketch=rebar.GetCenterLineCurves (bool1,bool2,bool3)

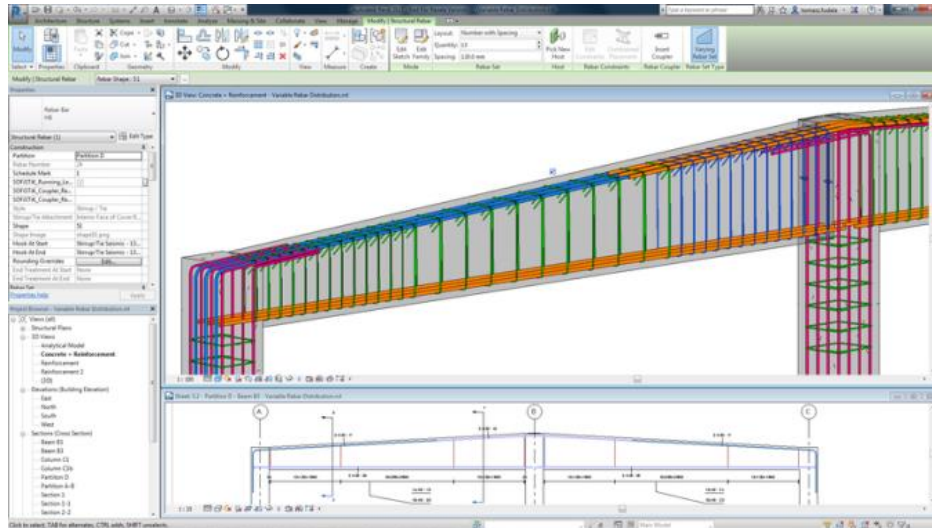
    #iterate all bar positions in set
    for iPos in range(0, rebar.NumberOfBarPositions):
        #check if there is a rebar at the position (first and endbar are optionally present)
        if(rebar.DoesBarExistAtPosition(iPos)):
            #get transformation to transform centerline to specific bar in set
            t=rebar.GetBarPositionTransform(iPos)
            #apply transformation to each curve segment
            for s in sketch:
                ss=s.CreateTransformed(t)
                #convert revit geometry to DS Geometry (lines, arc etc)
                c=Revit.GeometryConversion.RevitToProtoCurve.ToProtoType(ss, True)
                crv.append(c)
            crvset.append(crv)
            crv=[]
    RebarCurves.append(crvset)
    crvset=[]

OUT = RebarCurves

```

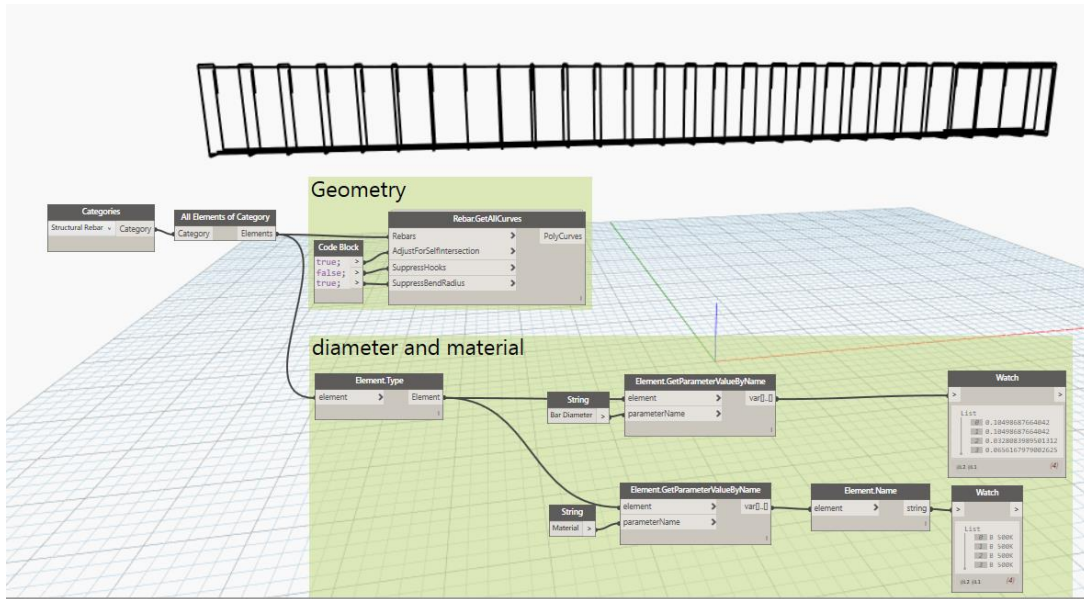
LOGIC IMPLEMENTED IN A CUSTOM PYTHON NODE

Note in Revit 2017 bars in a rebar set may have a varying shape automatically derived from constrained. This includes that centerline curves from the geometry of the first element will not be the same for all other elements. In the Revit 2017 API the command `GetCenterlineCurves()` does take a bar position as input to directly derive the correct shape. Refer to the Revit API documentation.



Structural rebar properties:

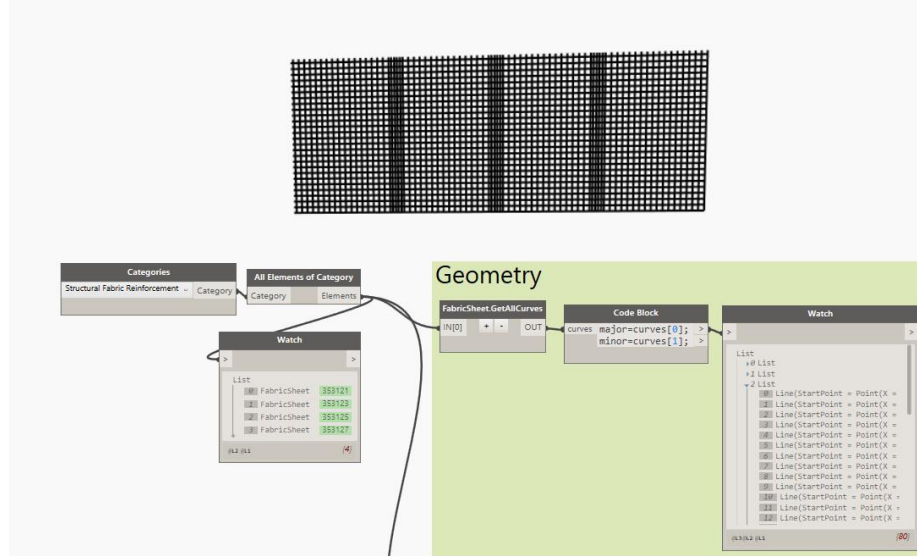
Bar diameter and material are defined in the family type of the Revit rebar element. Or this data can alternatively be access via the benddata command that provides the basic input for bending schedules.



DYNAMO WORKFLOW TO OBTAIN REBAR METADATA AS DIAMETER AND MATERIAL

Obtain structural fabric reinforcement data

Structural fabric reinforcement covers the fabric areas that are built up out of fabric sheets. These sheets are preassembled to the building site, and are in Revit also referred to as one object with its own properties. For FEA analysis the geometry from each individual bar should be obtained. For this purpose, a custom node is created that provides access to the WireCenterLines. In comparison with Revit rebar sheets in fabric sheets not just the first element is defined but directly all curve lines are obtained.



DYNAMO FILE TO OBTAIN CENTERLINE GEOMETRY


```

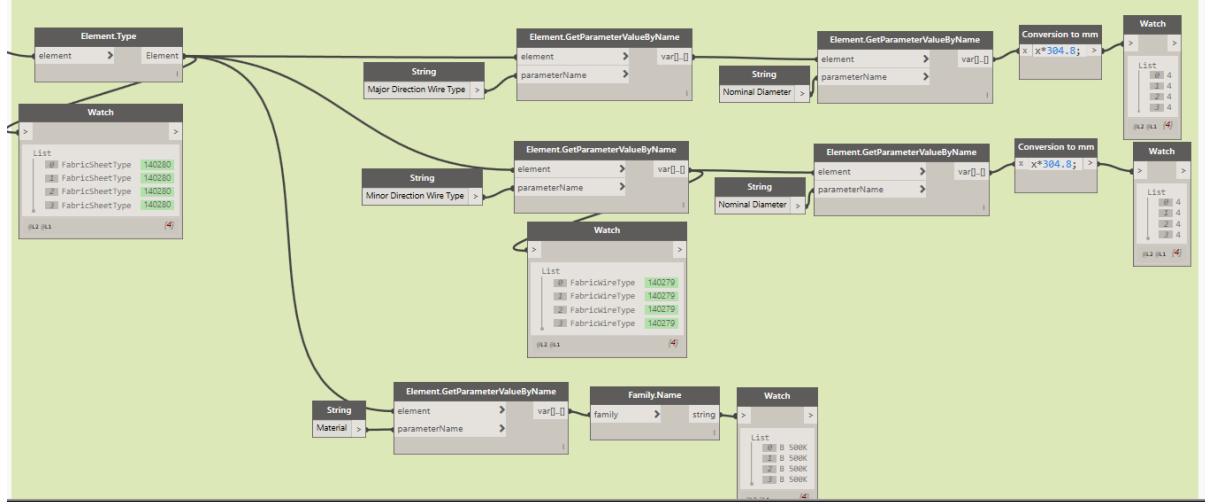
2 import clr
3 clr.AddReference('ProtoGeometry')
4 from Autodesk.DesignScript.Geometry import *
5 # Import rDType(bool) extension method
6 clr.AddReference("RevitNodes")
7 import Revit
8 clr.ImportExtensions(Revit.Elements)
9 # Import geometry conversion extension methods
10 clr.ImportExtensions(Revit.GeometryConversion)
11
12 #Access to revit api
13 clr.AddReference("RevitAPI")
14 from Autodesk.Revit.DB.Structure import *
15
16 #Inputs
17 IN[0]
18
19 crv1=[]
20 RebarCurves1=[]
21 crv2=[]
22 RebarCurves2=[]
23
24 #get centerline curves from the selected rebar objects
25 for rebar in IN[0]:
26     i=UnwrapElement(rebar)
27
28     #major direction
29     sketch1=i.GetWireCenterLines(WireDistributionDirection.Major)
30     #convert revit geometry to US Geometry (lines, arc etc)
31     for s in sketch1:
32         c=Revit.GeometryConversion.RevitToProtoCurve.ToProtoType(s, True)
33         crv1.append(c)
34     RebarCurves1.append(crv1)
35     crv1=[]
36
37     #minor direction
38     sketch2=i.GetWireCenterLines(WireDistributionDirection.Minor)
39     for s in sketch2:
40         c=Revit.GeometryConversion.RevitToProtoCurve.ToProtoType(s, True)
41         crv2.append(c)
42     RebarCurves2.append(crv2)
43     crv2=[]
44 OUT = [RebarCurves1, RebarCurves2]

```

CUSTOM PYTHON NODE TO OBTAIN WIRE CENTERLINES IN BOTH DIRECTIONS

Fabric sheet geometry must be obtained for major and minor directions individually in order to be able to match with possible variation in diameter or material in both directions. For fabric sheets, this metadata is not bar specific accessible. Instead, diameter and material are defined in the FabricSheetType (familyType) of the Fabric Sheet. Here the parameters Major Direction Wire Type provide access to the wire type that stores the nominal diameter.

Get fabric sheet diameter and material



Methods for exchange of reinforcement information

Stimulated by the BIM developments of the latest years, information is more and more preserved along the design process.

Exchange of reinforcement data between Revit and analysis software can be performed in several ways:

- No exchange, manually input by users.
- Exchange by intermediate file format
 - Exchange geometry only (e.g. .dwg format) + manual providing meta data as diameters and material
 - Exchange of all data in structured element based format (for example IFC or XML)
- Direct link between software (by means of API)

The use of a direct link or intermediate file format to communicate rebar geometry and rebar properties significantly reduces the amount of work to setup reinforcement in the FEA model and avoid possible errors. Structured file formats are able to export both geometry as additional meta data. Exchange of both geometry and rebar properties prevents mistakes in incorrect assignment of material or diameters.

The image on the next page illustrates this development over time..

Development of the exchange of reinforcement data to FEA modelling

Drawing/modelling process

Analysis process

Comments

No direct data exchange



Modelling reinforcement
(geometry + properties)

FEA

FEA software is specialised in calculations,
not in creation of geometry
Manual input of rebar geometry and properties
may result in errors

Export of geometry



export



diameter

FEA

Manual input of rebar properties may result in errors

Exchange of rebar information by means of export structured data



export



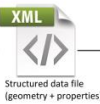
import

FEA

More support for standard file format for
BIM data exchange (i.e. IFC)



export



read



generate

FEA

In control of content of data file, to include all relevant information.
Interpretation of data file required in custom import function.
Ability to add modifications on input data
Easy file format to read and write

Parametric model generation

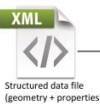


FEA

Programming rebar geometry, not as effective as
visually modelling in Revit
or visually programming rebar in Dynamo



export



read



generate

FEA

Use dynamo allows for quicker generation
of design variants



get design variables

set design variables

FEA

Control of design input from FEA perspective



AUTODESK
UNIVERSITY 2016



Standard IFC format or custom structured layout using XML

More and more software packages are supporting BIM formats as IFC nowadays. IFC are a good alternative for the exchange of all required rebar information and more. Custom file formats as XML allow to specifically control which information will be exchanged. XML-based formats have the advantage for programmers to they are easy to read and write compared to IFC files. With the con side that the structure and layout of the data is not commonly agreed on, so that files can only be used for specific situations. As the FEA package used in our company does not support the use of IFC files yet, functionality for reading the exchange had to be programmed. Secondly, custom import functionality was required to include possible modification of rebar lengths to correct for anchoring on the FEA side. Due to the simple file layout, XML data support was significantly easier to implement than reading functionality for the IFC format.

Intermediate file formats or direct link using APIs.

In computer programming, an application programming interface (API) is a set of subroutine definitions, protocols, and tools for building software and applications. An API provides an interface to the application allowing it to be accessed and controlled from outside. The subroutines definitions inside the API makes it easier to develop a program by providing building blocks, functions, which can be put together by a programmer. For example, the subroutines of the Revit API are the base of the Dynamo Nodes. These nodes contain instructions to communicate by Revit, all hidden inside the visual nodes to make them easier to use.

Considered in this case should be the presence of two disciplines in the process. FEA specialists to evaluate the models may be supported by structural modelers to setup Revit reinforcement models. Whereas for exchange by API, availability of both software packages (and licenses) on one workstation may be required. Exchange by file may split up this process to each individual discipline.

In our company, we apply the use of an intermediate data file (XML) to document input individual design variants and provide clear points of exchange of information in the process which can be used at a time when suited to each participant. In addition, it might be recommendable to save exchanged design data between disciplines for traceability. At the same time, use of an intermediate file doesn't prevent implementation in automated optimization procedures will be demonstrated in the next chapter.

This allows broader application than in optimization processes only, i.e. the reinforced layout for any case where an advanced FEA model is required can be prepared by structural modelers. Thereby improving flexibility and efficiency.

XML data exchange of reinforcement

XML is a file format to store and transport information in such a way that it is both human- and machine-readable. Reading and writing of XML files is common functionality that is available in almost all programming languages. The structure of data in elements and attributes can be set by the developer.

In Dynamo, using the custom python node functionality, a new node can be programmed that exports rebar information. Below example provides the background of such a node that is able to export all rebar from "structural rebar". Fabric sheet data is not included in this example. This node iterations over all rebar elements determines rebar geometry and exports it to XML format.

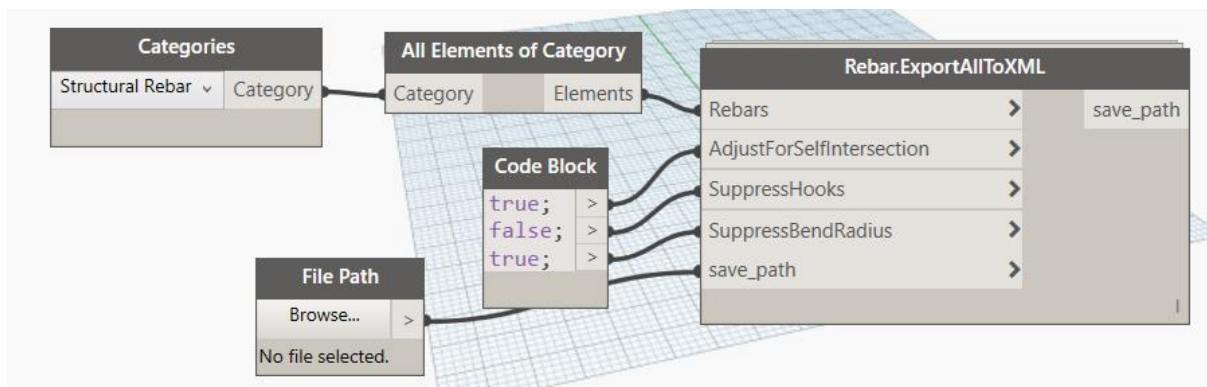


```

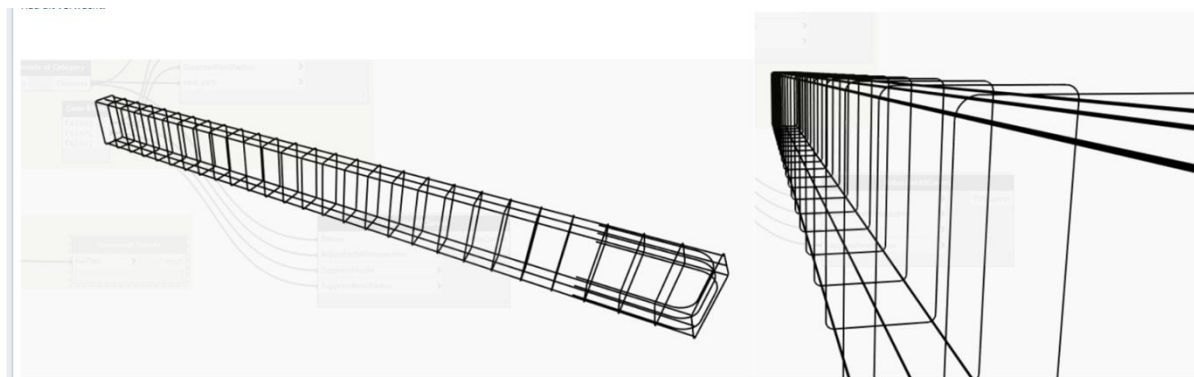
<?xml version="1.0" encoding="UTF-8"?>
- <Rebars>
  - <Rebar>
    <Data Length="7950.0" Id="347836" Host="346312"/>
    <Normal Z="0.0" Y="1.0" X="0.0"/>
    <benddata diameter="32" bendradius="240.0"/>
    - <line_data worksheet="Project Standards">
      - <Line>
        <SPoint Z="-52.172" Y="3225.887" X="-12714.73"/>
        <EPoint Z="-52.172" Y="3225.887" X="-4764.73"/>
      </Line>
    </line_data>
  </Rebar>

```

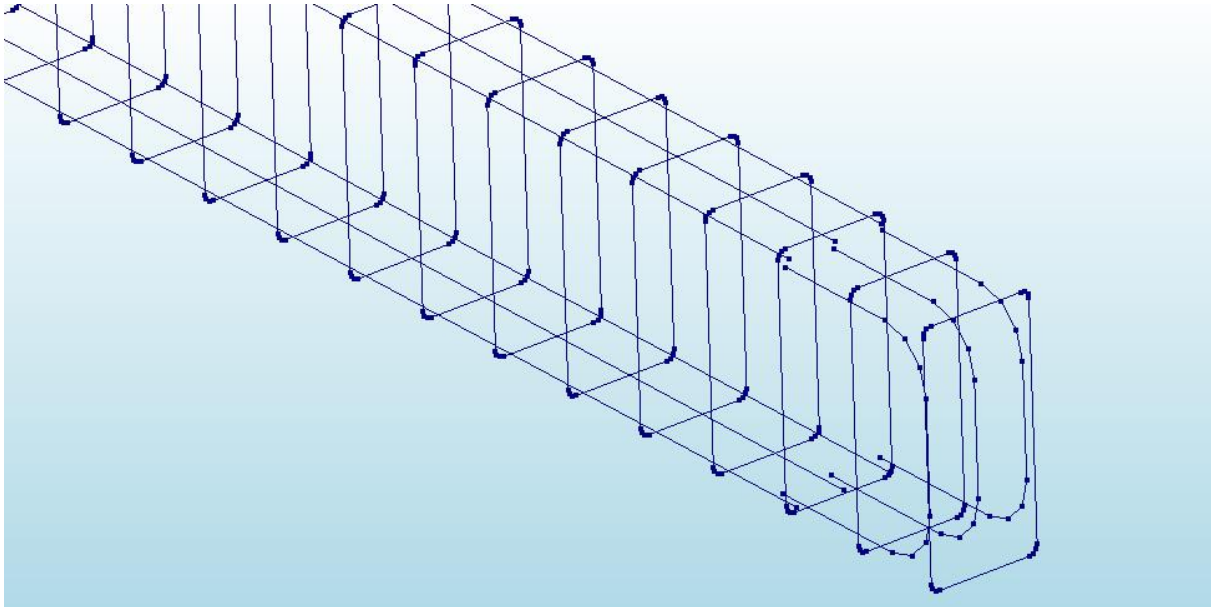
EXAMPLE OF XML OUTPUT FOR A STRAIGHT BAR:



DYNAMO WORKFLOW



REBAR GEOMETRY AS DERIVED IN DYNAMO.



PART OF THE REBAR GEOMETRY AS GENERATED BASED ON THE XML FILE IN FEA PACKAGE

The logic inside custom python node for XML export, illustrating the simplified layout of setting up an XML file as a programmer.

```

#settings for rounding
RoundOffNumber=3
RoundOffVector=5
toFoot=304.8

#root element of XML tree
Rebars=ET.Element("Rebars")

#function to export rebar to xml
def Rebar2XML(rebar, curves):
    benddata = rebar.GetBendData();

    Xrebar=ET.SubElement(Rebars,"Rebar")
    ET.SubElement(Xrebar,"Data",
        id=rebar.Id.ToString(),
        Host=rebar.GetHostId().ToString(),
        Length=str(round(rebar.Get_Parameter(BuiltInParameter.REBAR_ELEM_LENGTH).AsDouble() * toFoot, RoundOffNumber)))
    ET.SubElement(Xrebar,"Normal",
        X=str(round(rebar.Normal.X, RoundOffVector)),
        Y=str(round(rebar.Normal.Y, RoundOffVector)),
        Z=str(round(rebar.Normal.Z, RoundOffVector)))
    ET.SubElement(Xrebar,"benddata",
        diameter=str(round(benddata.BarDiameter * toFoot, RoundOffNumber)),
        bendradius=str(round(benddata.BendRadius * toFoot, RoundOffNumber)))

    wname=doc.GetWorksetTable().GetWorkset(rebar.WorksetId).Name
    curveContainer = ET.SubElement(Xrebar,"line_data", wname)
    for curve in curves:
        if(type(curve)==Line):
            line=ET.SubElement(curveContainer,"Line")
            ET.SubElement(line,"SPoint",
                X=str(round(curve.GetEndPoint(0).X*toFoot, RoundOffNumber)),
                Y=str(round(curve.GetEndPoint(0).Y*toFoot, RoundOffNumber)),
                Z=str(round(curve.GetEndPoint(0).Z*toFoot, RoundOffNumber)))
            ET.SubElement(line,"EPPoint",
                X=str(round(curve.GetEndPoint(1).X*toFoot, RoundOffNumber)),
                Y=str(round(curve.GetEndPoint(1).Y*toFoot, RoundOffNumber)),
                Z=str(round(curve.GetEndPoint(1).Z*toFoot, RoundOffNumber)))
        else:
            #curve is an arc
            #get midpoint
            mp=curve.Evaluate(0.5, True)
            arc=ET.SubElement(curveContainer,"Arc")
            ET.SubElement(arc,"SPPoint",
                X=str(round(curve.GetEndPoint(0).X*toFoot, RoundOffNumber)),
                Y=str(round(curve.GetEndPoint(0).Y*toFoot, RoundOffNumber)),
                Z=str(round(curve.GetEndPoint(0).Z*toFoot, RoundOffNumber)))
            ET.SubElement(arc,"MPPoint",
                X=str(round(mp.X*toFoot, RoundOffNumber)),
                Y=str(round(mp.Y*toFoot, RoundOffNumber)),
                Z=str(round(mp.Z*toFoot, RoundOffNumber)))
            ET.SubElement(arc,"EPPoint",
                X=str(round(curve.GetEndPoint(1).X*toFoot, RoundOffNumber)),
                Y=str(round(curve.GetEndPoint(1).Y*toFoot, RoundOffNumber)),
                Z=str(round(curve.GetEndPoint(1).Z*toFoot, RoundOffNumber)))
            ET.SubElement(arc,"Center",
                X=str(round(curve.Center.X*toFoot, RoundOffNumber)),
                Y=str(round(curve.Center.Y*toFoot, RoundOffNumber)),
                Z=str(round(curve.Center.Z*toFoot, RoundOffNumber)))
            ET.SubElement(arc,"Data",
                Radius=str(round(curve.Radius*toFoot, RoundOffNumber)),
                Length=str(round(curve.Length*toFoot, RoundOffNumber)))

```

THE FUNCTION TO WRITE THE XML FILE.

```

#Saves an XML file with rebar geometry information.
#Authored by Chris van der Ploeg, ABT, 2016
#https://www.abt.eu/

import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
#Import ToDSType(bool) extension method
clr.AddReference("RevitNodes")

import Revit
clr.ImportExtensions(Revit)
clr.ImportExtensions(Revit.Elements)

# Import DocumentManager
clr.AddReference('RevitServices')
import RevitServices
from RevitServices.Persistence import DocumentManager

# Import geometry conversion extension methods
clr.ImportExtensions(Revit.GeometryConversion)

#access to revit api
clr.AddReference('RevitAPI')
clr.AddReference('RevitAPIUI')
from Autodesk.Revit.DB import *

#package for XML creation
import sys
sys.path.append("C:\Program Files (x86)\IronPython 2.7\Lib")
import xml.etree.ElementTree as ET

#Pointers to Revit application
doc = DocumentManager.Instance.CurrentDBDocument
uiapp = DocumentManager.Instance.CurrentUIApplication
app = uiapp.Application

```

REFERENCES TO STANDARD LIBRARIES TO IMPORT FUNCTIONALITY.

```

#inputs
IN[0]
bool1=IN[1]
bool2=IN[2]
bool3=IN[3]
save_path=IN[4]

crv=[]
crvset=[]
RebarCurves=[]
test=[]

#loop all rebar elements
for r in IN[0]:
    rebar=UnwrapElement(r)
    test.append(rebar.Id)
    #get centerline curves from the selected rebar objects
    sketch=rebar.GetCenterlineCurves (bool1,bool2,bool3)

    #iterate all bar positions in set
    for iPos in range(0, rebar.NumberOfBarPositions):
        #check if there is a rebar at the position (first and endbar are optionally present)
        if(rebar.DoesBarExistAtPosition(iPos)):
            #get transformation to transform centerline to specific bar in set
            t=rebar.GetBarPositionTransform(iPos)
            for s in sketch:
                ss=s.CreateTransformed(t)
                #convert revit geometry to DS Geometry (lines, arc etc)
                #c=Revit.GeometryConversion.RevitToProtoCurve.ToProtoType(ss, True)
                crv.append(s)
            #export to xml
            Rebar2XML(rebar,crv)
            crv=[]

#create xml tree and write to file
tree = ET.ElementTree(Rebars)
tree.write(save_path, encoding="utf-8", xml_declaration=True)

#output xml aswell
OUT=save_path

```

PYTHON CODE TO LOOP OVER ALL REBAR ELEMENTS AND CREATE TRANSFORMED GEOMETRY (AS DESCRIBED BEFORE) AND EXPORT THIS TO XML:

Optimization of rebar configurations using Dynamo and FEA

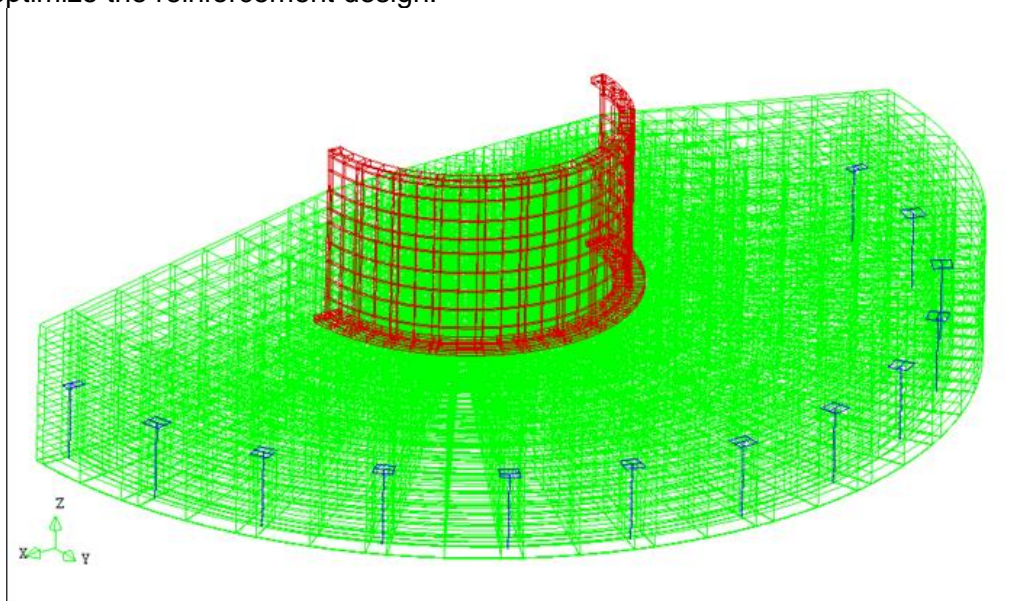
When the complexity of design increases, either due to the use of more complicated geometry, complex structural load bearing behavior or constraints from design or execution, the engineer's available knowledge may not be sufficient or efficient to provide optimal designs. Manually assessment these structures may require to significant engineering effort or require simplifications that in turn compromise the optimal use of resources. FEA software can be used to approximate the structural behavior with increased accuracy.

Application of FEA comes at the cost of engineering and analysis effort. Time spend to setup a full FEA model is significant longer than filling in a formula. A large part of the time required for advanced analysis is in the creation of the FEA models, in which in turn the majority is taken by the definition of rebar layout. With the ability of using Revit and Dynamo to generate rebar in the FEA package, analysis times can be reduced.

This stimulates the use of application of advanced FEA modeling in design the following ways:

- Less time required for setting up a FEA model.
- Model build up can be done by structural modelers instead of FEA specialists.
- Makes it possible to run multiple advanced analysis in cases where normally only a limited number of variants are evaluated.

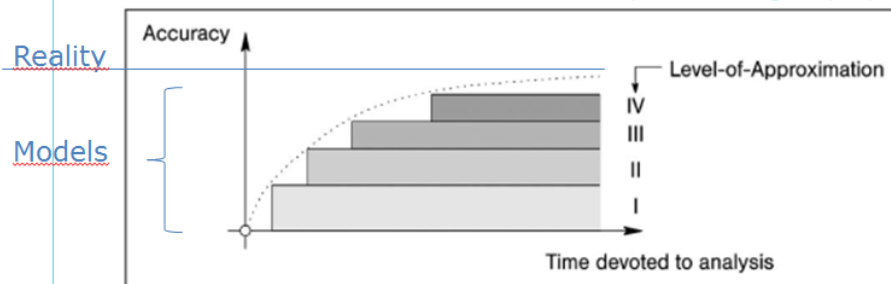
Evaluation of multiple models increases insight in the design behavior, which can be used to further optimize the reinforcement design.



GENERATED REINFORCEMENT IN DIANA FEA FOR A (HALF) WINDMILL FOUNDATION

Level of approximation in structural design

Structural design is all about proving that the structure has sufficient resistance to withstand loads during lifetime with a certain probability. In order to prove that, we need experience, formulas and analysis models. Several levels of approximation can be used. The graph below shows the relationship between time devoted to analysis and accuracy. We can say that when we can achieve 100% accuracy, we have exactly reproduced reality. Level I to IV stand for the different ways of modeling, ranging from low (formula's) to high accuracy (FEA). Engineering effort required increases with the complexity of the level of approximation applied, but better solutions may be achieved.

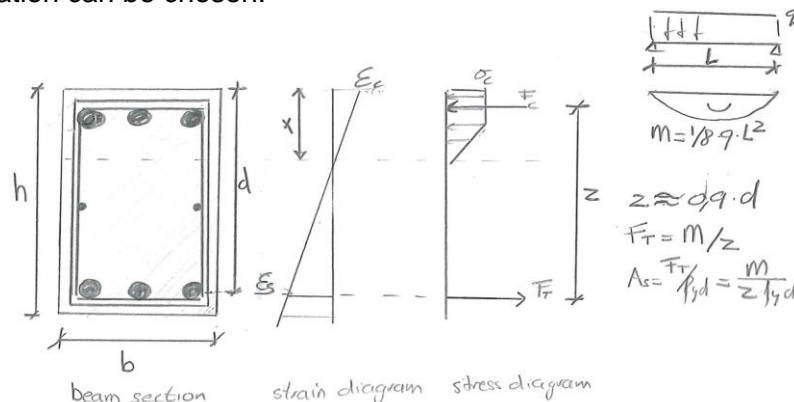


Source.: CEP FIB model code 2010

Calculation reinforcement using formula's

The most conventional concrete structure designs apply a level I or level II approach. This is an analytical design method where formulas are used to describe overall behavior. In order to describe behavior in a single formula, assumptions and simplifications are needed.

The most straightforward example of this is the design of the design of reinforcement in a simple beam on bending. Given a situation with span L and load q , the maximum bending moment M can be derived. With a cross section shape often derived from rule of thumbs combined with aesthetical or functional constraints, the height (h) and width (b) of the beam are known. The location of the reinforcement is known as (d) and the leverage arm between compressive force and tensile force (z) can be well approximated by $z=0.9d$. Now the required amount of reinforcement can directly be obtained and a bar configuration can be chosen.



With formula's known the optimal solution for this level of approximation is either directly obtained or can be simply derived by reordering of the formula.

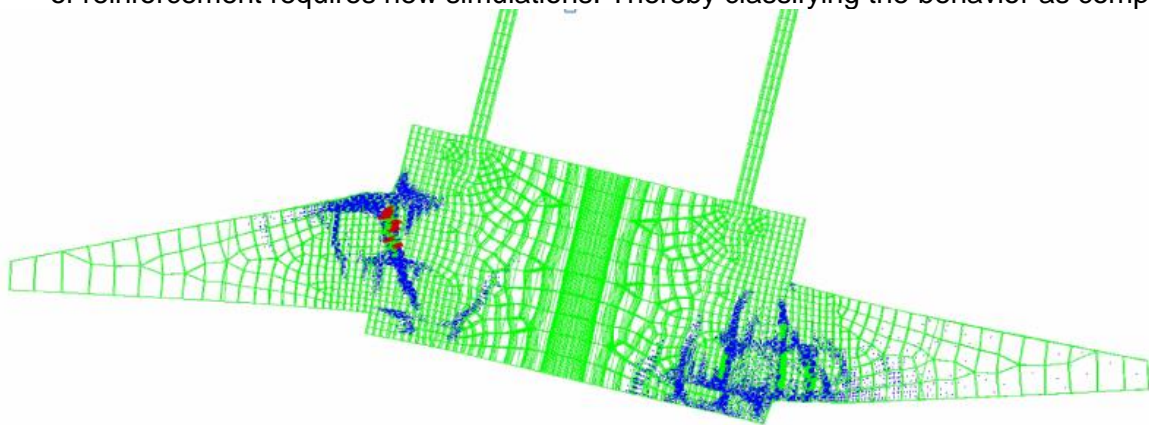
In order to describe more complicated behavior in a single formula, assumptions and simplifications are needed. It seems that a level I design always lead to over-dimensioned structures and therefore is “to save”. Although failure mechanisms may be overlooked by just simply applying the formulas without expertise. Especially in structures with more complex ways of load bearing behavior alternative failure mechanism may exist.

Similarly, more complex load resisting mechanisms may exist that provide additional load bearing. These mechanisms are often harder to comprehend and capacity cannot be quantified without advanced analysis.

Evaluation of behaviour using finite element analysis (FEA) models

Opposite to level I modeling a level IV model uses the real geometry of the foundation including all internal components (like reinforcement and tower anchorage). Also, the real, non-linear material behavior is modeled. Evaluation requires the use of non-linear 3D Finite Element Analysis software (FEA) as i.e. Ansys, Abaqus or as used in our company Diana FEA by TNO (<http://dianafea.com/SolutionsReinforcedConcrete>) .

These FEA models actually show the behavior of the structure under increased loading and the interaction of all components. For concrete, this means that realistic crack patterns develop and you can actually see the stress distribution in the reinforcement steel. The real failure mechanism becomes apparent. So instead of guessing what failure mechanism will occur, all failure mechanisms are implicitly verified and the real weak spot can be found. This may lead to both more insight in the governing failure mechanisms and more optimal solutions. This advanced behavior can often not be quantified by designers without simulation of FEA model. This includes that optimization of reinforcement requires new simulations. Thereby classifying the behavior as complex.



FEA RESULTS OF CRACK WIDTH SIMULATION FOR WIND MILL FOUNDATION

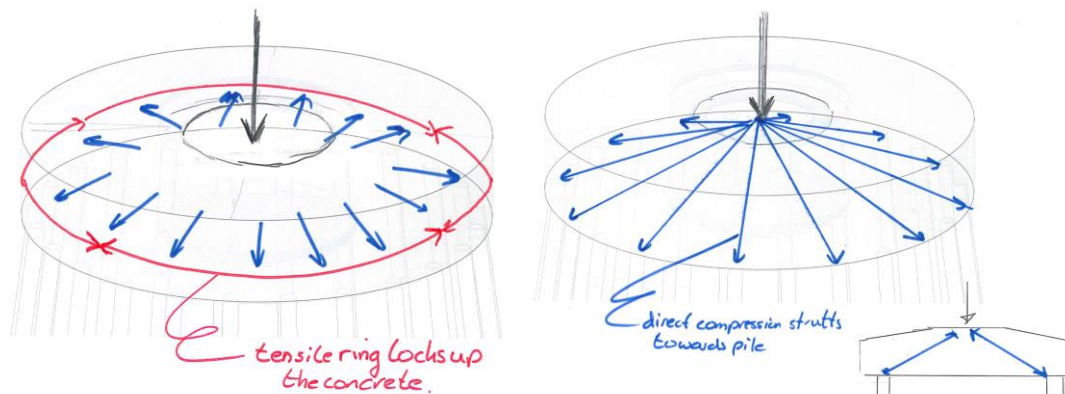
As we compare our now-a-day design performance using advanced FEA analysis (level IV analysis) with previous design data using conventional design methods (level I, II analysis) the following data is found up to 50% total saved reinforcement steel, 30% total saved concrete and 20% total cost benefit on foundations (engineering and construction). Even further improvement can be expected with this new workflow that allow more FEA models to be generated based on Revit data.

When to apply FEA

The main area where advanced FEA modeling may provide improved results over simplified calculations is situations where the distribution of stiffness influences force distribution. For simple 1D elements with simple spans where a change of stiffness due to cracking of the concrete does not alter the bending moment distribution, similar results will be found as using simplified formula's.

Some situations where FEA may lead to better results than manual formula based analysis:

- Statically undetermined systems as continuous beams, moment frames with stiff joints between columns and beams, vierendeel girders. As simplified assumptions to use in manual calculations regulations allow up to 20% redistribution of hogging moments to the field.
- Introduction of concentrated loads on 2 or 3-dimensional structures. The width of a structure that will contribute to concentrated load bearing can be determined exactly instead of conservatively approximated following regulations.
- Structures with large height over span ratios (i.e. high beams, walls, concrete foundation blocks) a mix between bending and more favorable direct load bearing by means of direct compressions struts may occur depending on stiffness.
- 3 Dimensional structures in which multiple internal loadbearing mechanisms may be present which will contribute according their stiffness ratio 3D load bearing behavior may be complicated to predict. For example in the windmill foundation design, the side reinforcement provides an additional tensile ring that lock's up the concrete and load bearing through direct compression struts is more efficient than bending.

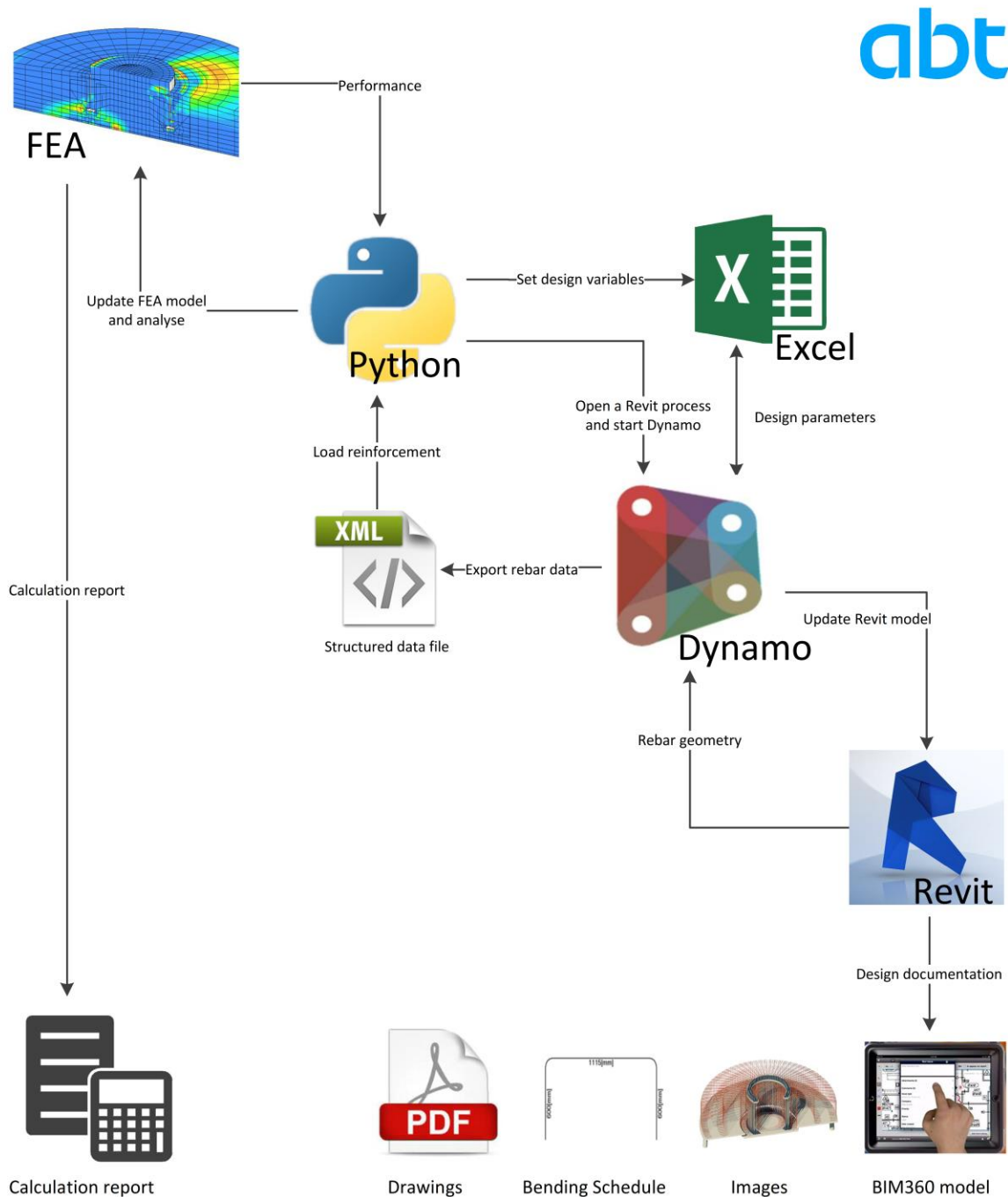


ADVANCED LOAD BEARING MECHANISMS

Workflow for optimization of reinforcement using FEA.

The workflow for optimization of reinforcement layout using FEA with Revit and Dynamo is illustrated in the following graph.

Optimisation of rebar configurations using Dynamo and FEA



The point of origin is from the perspective of the FEA modeler. From inside the FEA software the master script (python) will be started, that will perform the following functionality:

- Iterate over design variants
 - Create design alternative by writing design parameters to Excel
 - Start Revit with the journal file
 - Revit starts
 - Open Revit model
 - Open dynamo model
 - Executes Dynamo model
 - Dynamo model reads design data
 - Makes modifications to the Revit model
 - Exports XML file with reinforcement data
 - Saves the model and exports other design data
 - Closes Dynamo and Revit
 - Reads the generated XML file and convert rebar information to reinforcement for use in FEA analysis.
 - Saves the calculation model
 - Analyse
 - Save the analysis results.
 - Export predefined results and analysis images for review.

This obtained design data can be used by the FEA specialist to obtain a quick insight into the influence and sensitivity of design parameters. This insight can be used to pick the best solution.

This leads to the selection of a given rebar configuration of which the Revit model was already created during the process. Or alternatively, new design data can be specified in the Excel sheet which the structural modeler in combination with the Dynamo script can use to create the structural model. A Revit model with correct geometry and layout is now created. Ready to send to production directly (bending schedules, 3d access BIM field) or as a basis to derive drawings from.

Steps to take in preparation:

- Setup a Revit model including all basic geometry and reinforcement that will not be changed.
- Setup a Dynamo model to vary all design parameters. This Dynamo file always reads in the excel file from a predefined location to load the design variables and exports an XML file.
- Create a journal file control Revit
- Setup an FEA model including materials, geometry, loading and analysis settings.
- Setup a python script to do the optimization

For now, only the reinforcement is seen as variable in this workflow. Future workflows may also include the variation of geometry or loading.

Modification of design variables

In optimization processes, alternative variants are created by variation of design parameters.

In automated evaluation, the dynamo variables cannot be changed in Dynamo by means of user interaction during the process. Alternative input parameters should be specified in the master graph and passed to the slave models. One way of doing this is by generating a data file that the slave Dynamo model can read as input. This file contains all the design variables and the location where to save additional Dynamo output and generated Revit model.

The commonly known file format of Excel is chosen for specification of input parameters. Using a common file format ensures that all design participants can specify parameter values for alternatives to be generated, even those that do not have any hands-on experience with Dynamo their self. Besides that, it provides a clear overview for documentation of the applied design parameters. This eliminates errors in communication since the same file is used for both specification of design variables, communication between structural engineer and draftsmen, and reinforcement generation.

Feedback loop

The current use of this workflow is based on a parametric run: a predefined run of variants are evaluated.

Calculation feedback is currently not used by the optimization algorithm to generate specific variants of new models. Future applications may use known relations as exceedance of crack width or high stresses in rebar to increase reinforcement at specific areas. This feedback may lead to faster convergence of models towards optimum situations.

At the moment large FEA models still require significant analysis time (~several hours). Using a parametric run, multiple design variants can be evaluated at nights, weekend, in one run without having the FEA specialist to interact. This allows the specialist to work on other projects and prevents the scenario of step by step adjustment since there are already alternatives available if one design fails. With the ever improving computational power or cloud computing techniques the number of models to be evaluated for each design is expected to increase further in the future.



Below an overview of some of the results of the first applications of variation of rebar geometry to generate multiple FEA models.

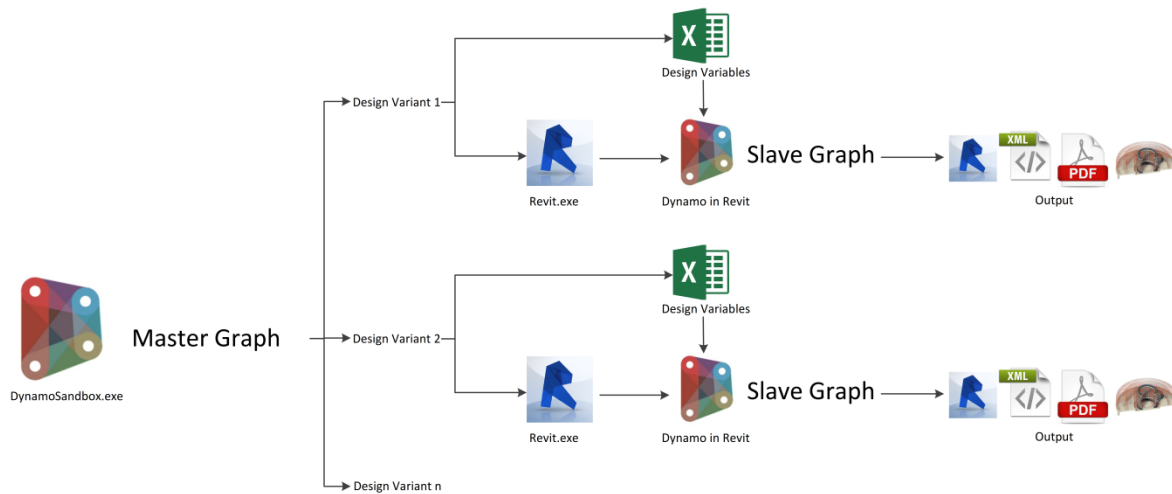
The parametric run of alternative reinforcement calculations provides the FEA specialist with several design variants to choose from, without traditional trial and error based assessment of alternatives requiring manual intervention.



External control of Revit and Dynamo using journal files

In this process Revit and dynamo work as “Slave” for another master program. This means they will execute some work for the master and then close. In this optimization, Revit and Dynamo should autonomously be external controlled by the master without user intervention. This can be done using the Revit journal files.

For the purpose of reinforcement optimization, this means that we can include Revit and Dynamo as a source to obtain new geometry information for design variations in the optimization workflow.



OVERVIEW OF A DYNAMO MASTER GRAPH CREATING MULTIPLE DESIGN VARIANTS

Journal file definition

Journal files capture the actions taken by the software during a session of Revit, from the time the software starts to the time it stops. Besides recording, the journal file can also be used to repeat the recorded actions in Revit. In the same way, a set of journal instructions can be setup to execute specific actions in Revit.

The following journal file used in this case contains the following instructions:

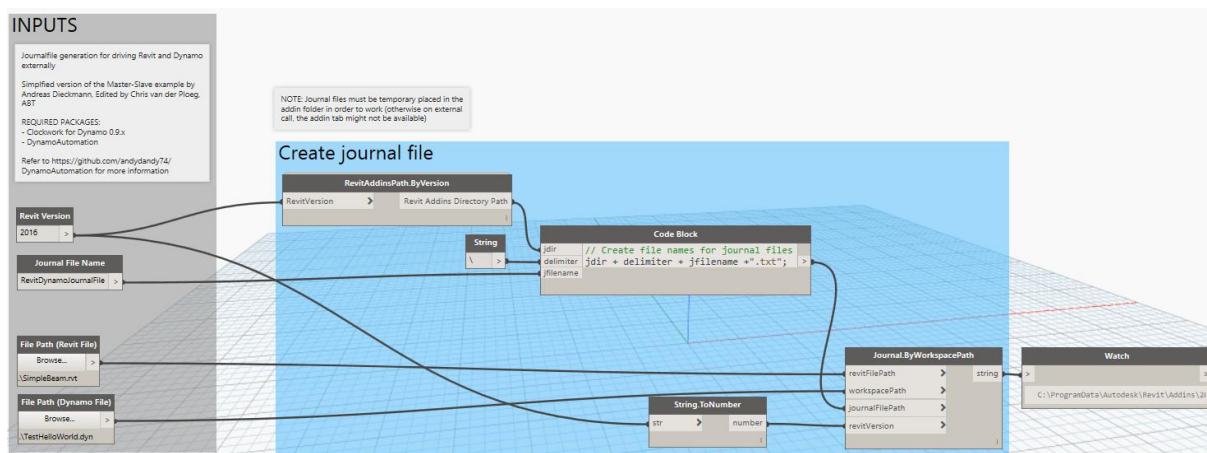
- opens a specific Revit project,
- starts Dynamo,
- opens a Dynamo file in automatic evaluation mode to run it directly,
- closes the active project,
- quits the application

In the journal file the paths to the Revit file and Dynamo file can be clearly recognized:

```
'Dim Jrn Set Jrn = CrsJournalScript
Jrn.Command "StartupPage", "Open this project , ID_FILE_MRU_FIRST"
Jrn.Data "MRUFileName", "C:\Users\PGC\Desktop\Rebar\SimpleBeam.rvt"
Jrn.RibbonEvent "Execute external command:CustomCtrl_%CustomCtrl_%Add-Ins\Visual Programming\Dynamo 1.2:Dynamo.Applications.DynamoRevit"
Jrn.Data "APIStringStringMapJournalData", 3, "dynPath", "C:\Users\PGC\Desktop\Rebar\TestHelloWorld.dyn", "dynShowUI", "false", "dynAutomation", "true"
Jrn.Command "Internal", "Flush undo and redo stacks , ID_FLUSH_UNDO"
Jrn.Command "Internal", "Close the active project , ID_REVIT_FILE_CLOSE"
Jrn.Command "SystemMenu", "Quit the application; prompts to save projects , ID_APP_EXIT"
```

Use Dynamo to build a journal file.


Using the nodes from package DynamoAutomation this journal file can be built up using Dynamo itself. DynamoAutomation is a dynamo package by Andreas Dieckmann that allows Dynamo users to batch process Revit models by driving Revit (and the Dynamo adding) from outside. This workflow opens up the possibility of including Revit in automated workflows for i.e. time consuming repeating tasks as model exports or model checking. Refer to <https://github.com/andydandy74/DynamoAutomation> for more in-depth information and examples about the DynamoAutomation package. The package itself can be loaded from the package manager.



DYNAMO WORKFLOW TO CREATE A JOURNAL FILE USING THE DYNAMOAUTOMATION PACKAGE

External control of Revit

The journal file can be executed by starting a new Revit process with the journal file specified as additional argument. A simple way of doing this is by using a cmd batch file.

 StartRevit2016withJournalFile.cmd

Containing the following code:

```
"C:\Program Files\Autodesk\Revit 2016\Revit.exe" /language ENU  
C:\ProgramData\Autodesk\Revit\Addins\2016\DynamoJournalFile.txt
```

In which:

C:\Program Files\Autodesk\Revit 2016\Revit.exe the program to start

/language ENU argument to set the language of Revit to English (required for proper execution of journal files).

C:\ProgramData\Autodesk\Revit\Addins\2016\DynamoJournalFile.txt The path to the journal file to execute. Note that this must be located in the Revit addin folder to ensure correct loading of Revit. When placed at other locations, Revit will startup without addins and Dynamo is not available.

A similar functionality can be obtained using Dynamo in a stand-alone version Dynamo Studio or the DynamoSandbox installed alongside Dynamo for Revit (see C:\Program Files\Dynamo\Dynamo Revit\1.2\DynamoSandbox.exe). DynamoSandbox.exe is the free Dynamo core technology (scripting language, node diagramming, execution engine) that can use some of the geometry tools available in Revit, and doesn't have the ability to sign into cloud services. This stand-alone tool can be used to drive Revit externally.

Optimization of rebar configurations using Dynamo for variation

In most situations the required reinforcement is be directly calculated using analysis and formula's (complicated domain), e.g. the relation between cause (loads) and effect (reinforcement). By default, engineers try to determine design solutions that minimize the required amount of material (reinforcement) required given de design conditions using their knowledge, experience, and skills within the resources they are given. For example, for any column given its dimensions, material loads, and support conditions, the design rules are known to calculate the optimal minimal required amount of reinforcement

Where things become more complicated when we introduce multiple design variables as the column diameter, concrete strength and reinforcement. Normally these problems are solved step by step during the process by manual evaluation of a limited number of alternatives combined with engineering experience from earlier projects. A full overview of all design alternatives or combinations may provide a more optimal solution in terms of material use, costs or fabrication effort. However, engineering time for evaluation of more alternatives is limited. Thereby can this situation be classified as complex as no complete overview can be given using traditional means.

Dynamo can be used to overcome this limitation by:

- Automation design, allowing specialization in unique optimal solutions for individual elements without compromising engineering effort.
- Simulation of alternatives, providing insight in performance of multiple design variants

Simulation results provide insight in the whole evaluation design. This obtained insight can be used to select better reinforcement configurations and further optimize the reinforcement design.

This will be illustrated in the following sections with Dynamo workflows for:

- Automated calculation of reinforcement
- Simulation of the performance of multiple design variants with varying reinforcement configurations.
- Optimize the number of alternative configurations relating the material use to the number of unique elements

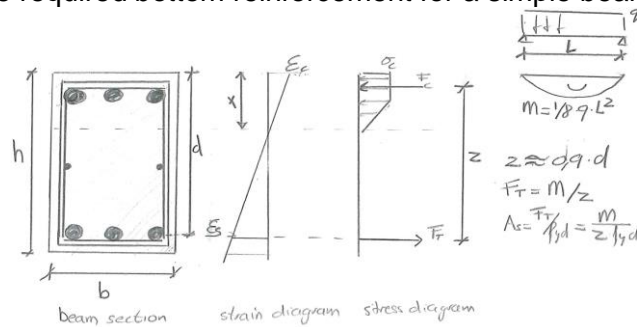
Automation of reinforcement calculations using Dynamo

In these models Dynamo will make use of available rules to determine the solution for each variant. This can be done in a number of ways:

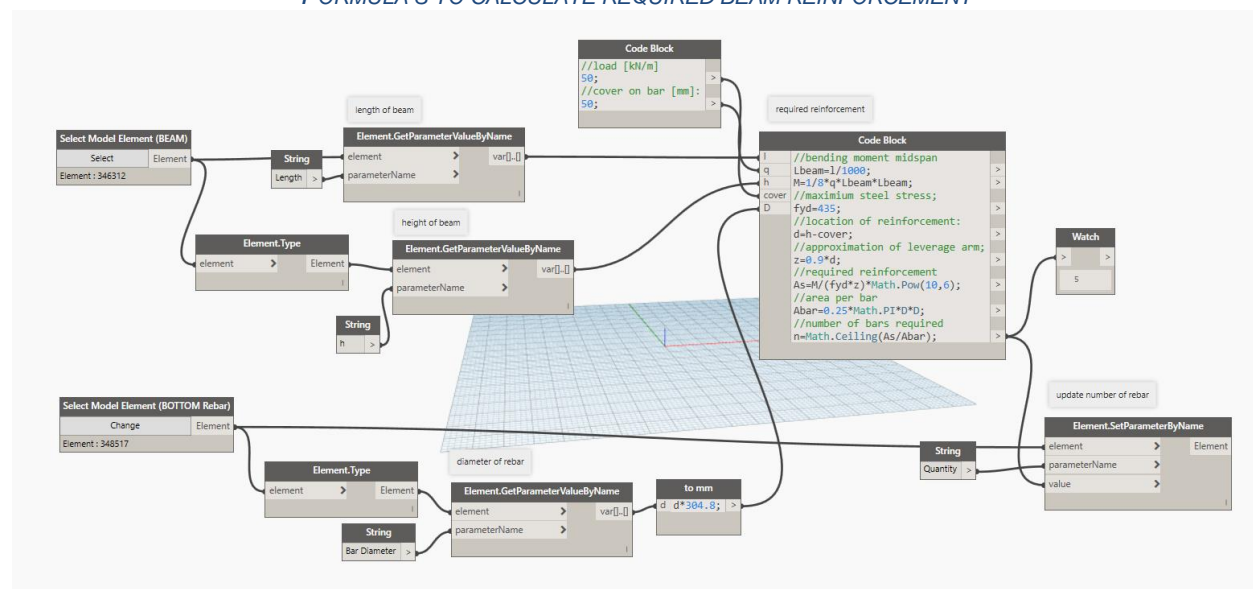
- Determine required reinforcement by implementing formula's in Dynamo:
- Design by comparison with design data
- Use external software can be used to calculate performance.

Determine required reinforcement by implementing formula's in Dynamo.

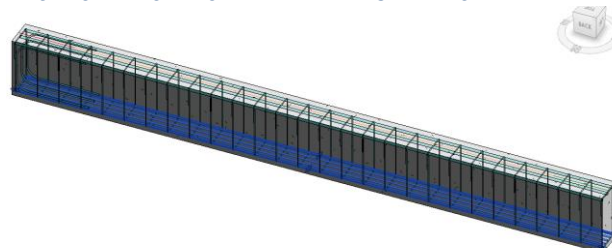
As example the required bottom reinforcement for a simple beam will be used.



FORMULA'S TO CALCULATE REQUIRED BEAM REINFORCEMENT



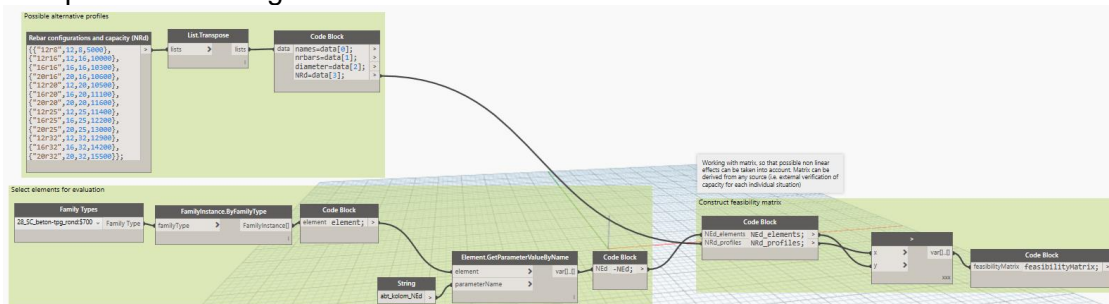
DYNAMO WORKFLOW TO DETERMINE NUMBER OF REBAR REQUIRED



UPDATED REINFORCEMENT IN BEAM

Design by comparison with design data:

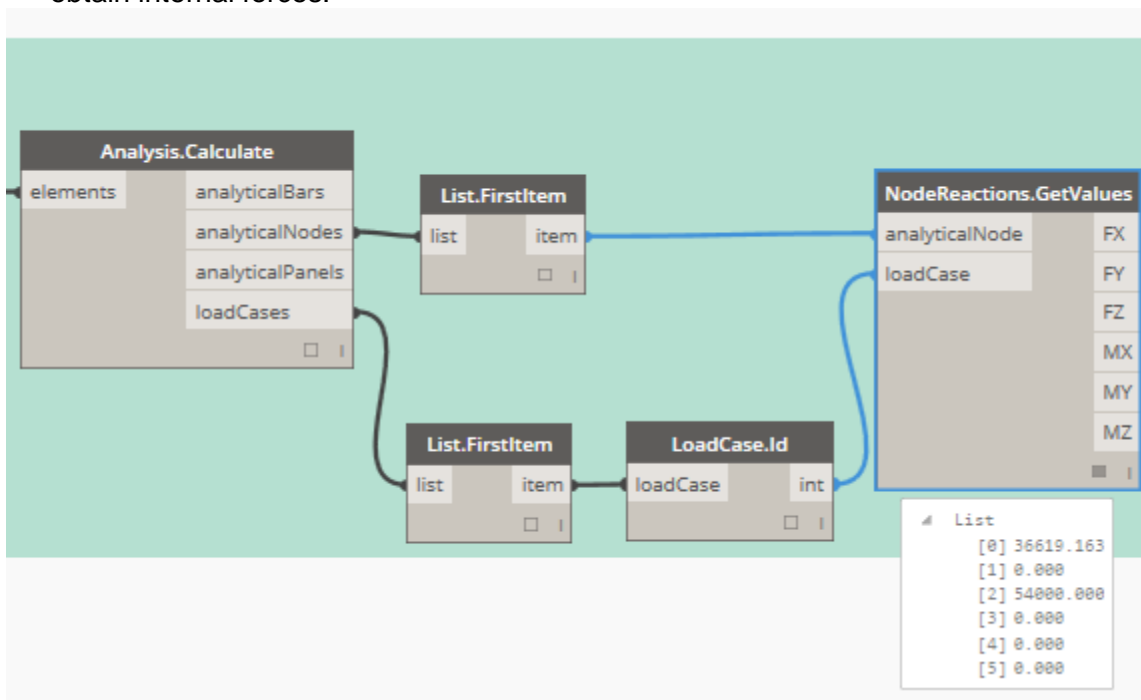
Reinforcement capacities can be selected by comparing internal forces with a the capacities of configurations.



EXAMPLE OF A DYNAMO WORKFLOW TO COMPARE THE RESISTANCE OF REINFORCEMENT CONFIGURATIONS WITH THE INTERNAL FORCES OF THE ELEMENTS (NRD (CAPACITY) > NED (LOAD)). MINIMUM COST SOLUTION CAN BE SELECTED AFTERWARDS.

Use external software can be used to calculate performance.

The package Dynamo for Structural Analysis includes Dynamo node definitions for interaction with Robot Structural analysis and React structures. At the moment, definitions are available to create and adjust structural models, analyze the structure and obtain internal forces.



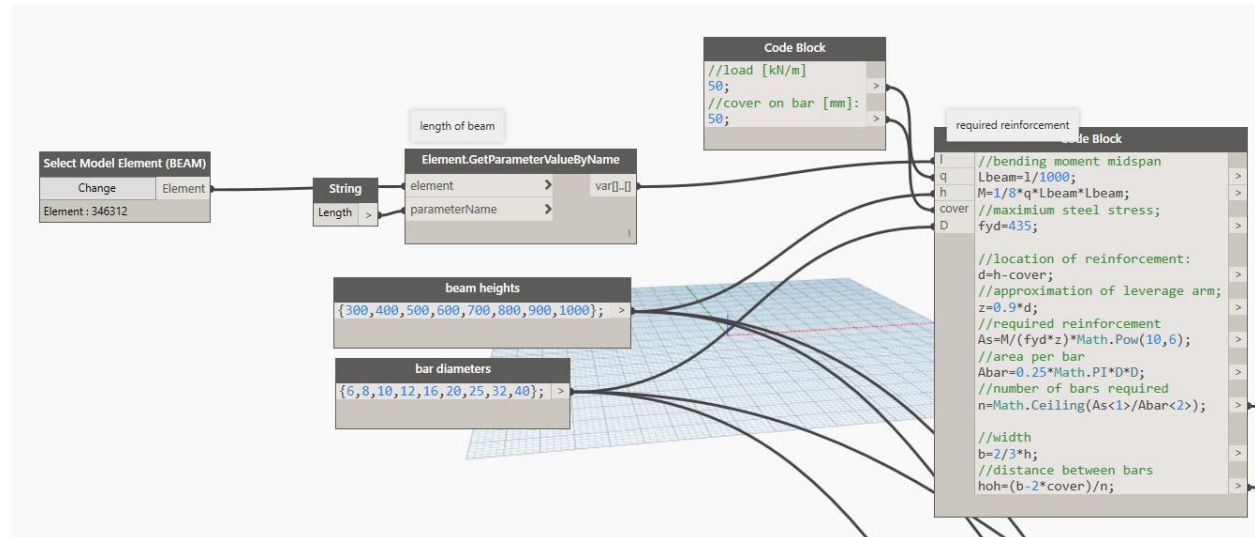
DYNAMO EXTRACT FROM THE PARAMETRIC TRUSS EXAMPLE USING DYNAMO AND REACT STRUCTURES

Refer to <http://react.autodesk.com/blog/2016/01/21/example-parametric-truss-using-dynamo-react-structures/> for more information of this workflow.

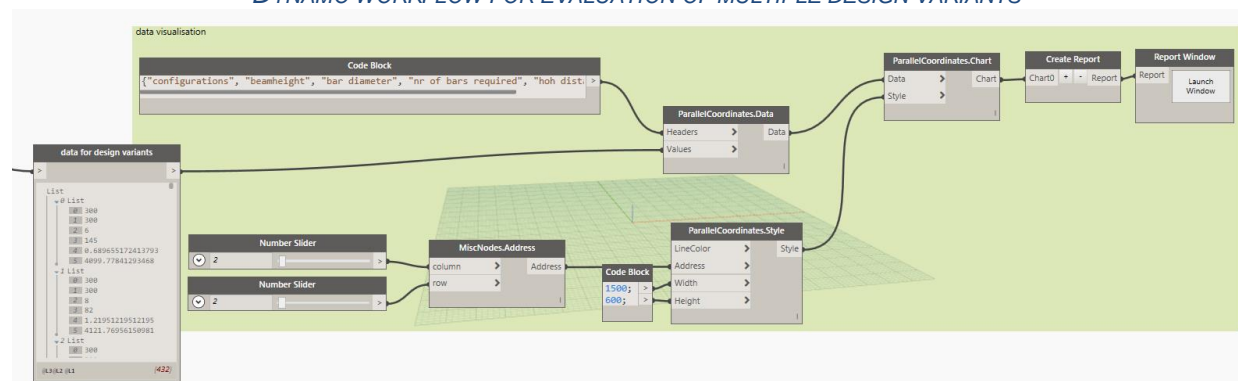
The ability to interact with a structural analysis software allows to obtain internal forces or required reinforcement area from a structural calculation directly from the structural analysis software. Additional custom Dynamo nodes could be developed that directly obtain calculated required reinforcement area results to drive reinforcement generation.

Simulation of multiple design variants using Dynamo

Availability of automated evaluation opens the way for optimizing of reinforcement as the performance of multiple design variants can be simulated. As for example alternative cross section dimensions, bar configurations and bar diameters.

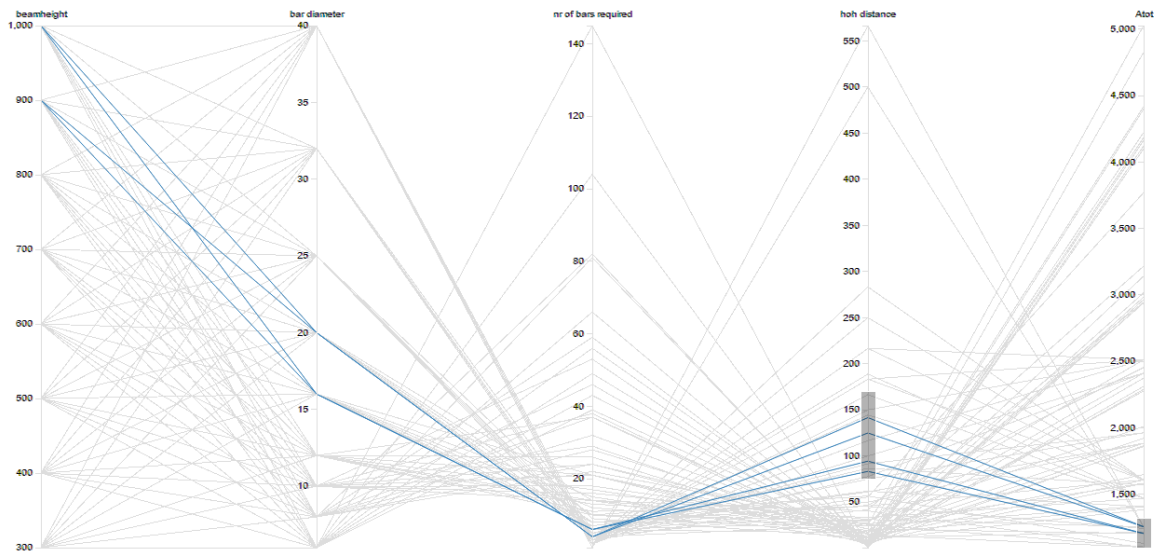


DYNAMO WORKFLOW FOR EVALUATION OF MULTIPLE DESIGN VARIANTS



DYNAMO WORKFLOW FOR VISUALIZATION OF DESIGN DATA USING PARALLEL COORDINATE PLOT

The design variants can be visualized using parallel coordinate plots that can be generated using the Mandrill package



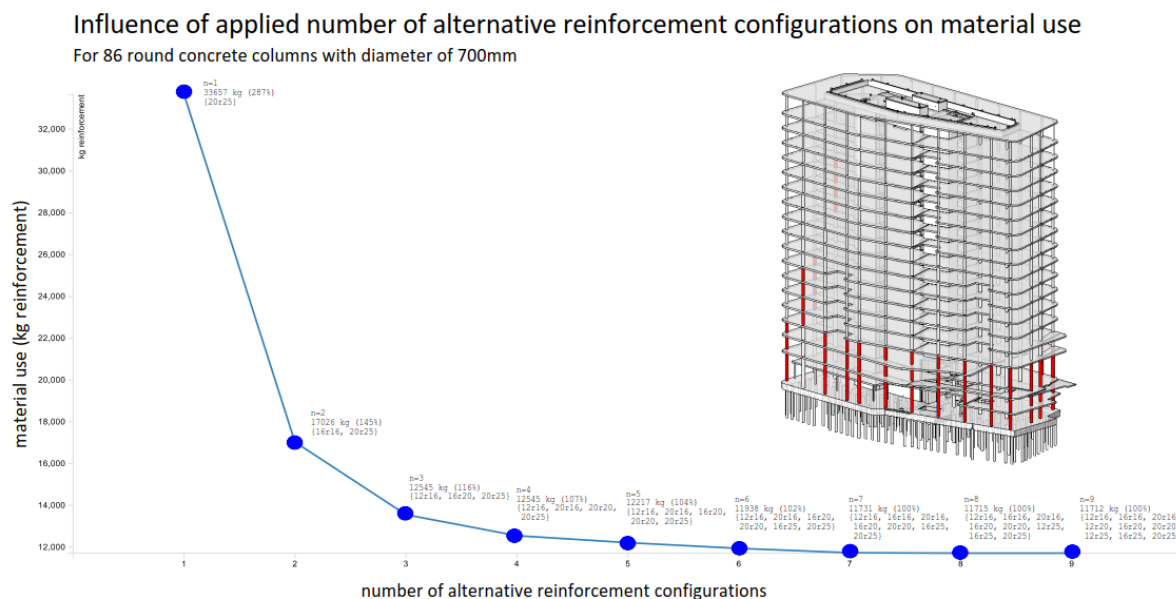
ALTERNATIVE SOLUTIONS FOR SIMPLE BEAM REINFORCEMENT. EACH LINE IS AN ALTERNATIVE DESIGN VARIANT. FROM THE WHOLE SET SELECTIONS CAN BE MADE TO SELECT DESIRED SOLUTIONS. AS BARS NOT TO CLOSE OR TO FAR AWAY FROM EACH OTHER AND THEN THE MINIMUM WEIGHT (ATOT) SOLUTIONS. AS EXPECTED MINIMUM REINFORCEMENT DESIGN IS FOUND FOR BEAMS WITH LARGER HEIGHTS. WHEN ALSO THE TOTAL AMOUNT OF CONCRETE IS INCLUDED IN THIS EVALUATION OPTIMIZATION CAN TAKE PLACE ON OVERALL TOTAL COSTS.

Optimisation of the number of unique reinforcement configurations using Dynamo

While from fabrication perspective, reduction of material use is always stimulated, large numbers of unique optimal elements / reinforcement configurations are not preferred. Partly this is related to costs of preparation for each unique solution (i.e. documentation, calculation, handling). On the other side this is related to the human participation in construction, where more unique solutions may increase the risk of errors made. Until robotized manufacturing will take over, more standardized solutions are preferable over unique elements with minimal material use, from a fabrication perspective.

This creates the design problem of how many alternative rebar configurations should be used, which configurations that should be and how this affects optimal use of resources. Allowing more alternative configurations results in a more optimal solution. Normally this is solved by engineering judgement that they pick just a few configurations purely based on experience. While exactly this point, an informed decision should be made by both the contractor and the engineer together. Again Dynamo may help to simulate this problem.

We will demonstrate how Dynamo can be used simulation and optimization of the number of column reinforcement configurations to be used in the design of a large office building.



RESULTS FROM SIMULATION WITH DYNAMO, THE NUMBER OF SELECTED UNIQUE REINFORCEMENT COMBINATIONS VS THE MATERIAL USE. (PICKING 3 REBAR CONFIGURATIONS INSTEAD OF 2 SAVES 4500 KG (29%) OF REINFORCEMENT. PICKING MORE THAN 5 DOES NOT MAKE SENSE (500KG (4%)). IN MANUAL DESIGN NO QUANTIFICATION OF THE INFLUENCE NUMBER OF ALTERNATIVE CONFIGURATIONS IS AVAILABLE AND THE OPTIMAL COMBINATION OF REINFORCEMENT CONFIGURATIONS MAY NOT BE SELECTED, BOTH RESULTING IN LESS OPTIMIZED USE OF REINFORCEMENT.

This information can help to bring final reinforcement design from the engineer's back office to the discussion table where alternatives can be discussed and decided on with all participants in the project. The same method can be applied in the preliminary design stage to evaluate which and how many alternative column diameters should be included.

The problem of how many alternative profiles or rebar configurations to apply in a project to avoid wasting material, while at the same time limiting the number of unique elements to reduce errors and costs is a common problem in all engineering projects.

This is a classical combination problem: Given a set of m rebar configurations, from which we will pick n alternatives, which configurations should we choose and what is the influence of the number of alternatives. Dynamo can be used to simulate this problem, find the best configurations for any number of $n=1..m$.

The total number of possible combinations is given by:

$$nr\ of\ combinations = \binom{m}{n} = \frac{m!}{n!(m-n)!}$$

So for a set of 10 rebar configurations of which we pick 4 alternatives we have 210 possible combinations. To simulate $n=1,2,3, \dots 10$. A total sum of 1023 unique combinations are possible.

Up to 1000 combinations can be addressed in seconds. And dynamo is able to deal with thousands of alternative configurations within minutes calculation time. However set sizes should be limited as with a set of 13 already 8191 combinations are required to address for simulation of all $n=1..13$. And picking 5 out of 20 requires already 15504 combinations on its own.

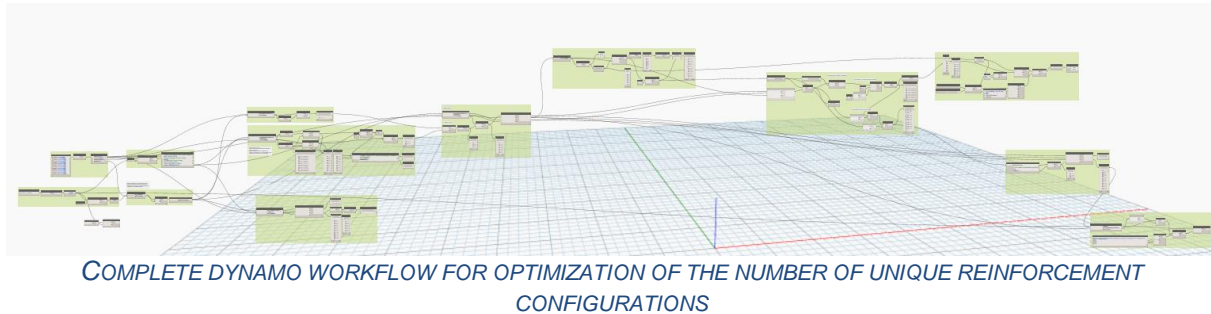
Ranges up to 10 are advisable for stability. For larger sets, smarter algorithms should be used. For most engineering problems this number of alternative profiles is often already sufficient. Using traditional methods infeasible solution can be eliminated first before assessment of unique combinations.

The workflow below makes use of a cost and feasibility matrix to select the best but feasible combination problem independent. These matrixes will be constructed first. Where after some simplifications are introduced to remove unnecessary profile types that will never be part of the best combinations. (too heavy, too weak). Thereafter all possible combinations of profiles will be tested using a brute force tactic. More advanced methods could be used for example when capacity increases with costs for all elements. However, this suggested approach works for all situations.

Feasability matrix							Cost matrix						
Element	Rebar Configuration						Element	Rebar Configuration					
	12r16	16r16	20r16	12r20	16r20	20r20		12r16	16r16	20r16	12r20	16r20	20r20
1	0	0	1	1	1	1	1	57	76	95	89	118	148
2	0	0	0	1	1	1	2	76	101	126	118	158	197
3	0	0	1	1	1	1	3	57	76	95	89	118	148
4	0	0	1	1	1	1	4	66	88	110	104	138	173
5	0	1	1	1	1	1	5	76	101	126	118	158	197
6	0	1	1	1	1	1	6	95	126	158	148	197	247
7	0	0	0	0	1	1	7	133	177	221	207	276	345

FEASIBILITY MATRIX (TESTING FEASIBILITY FOR POSSIBLE CONFIGURATION FOR EVERY ELEMENT) AND COST MATRIX (COSTS OF EACH CONFIGURATION FOR EACH ELEMENT)

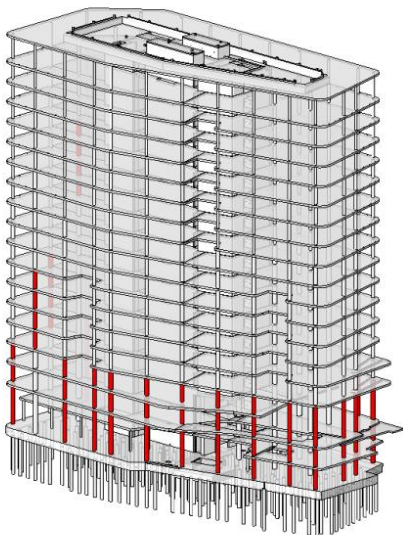
Dynamo workflow for optimization of the number of unique reinforcement configurations in concrete columns:



This workflow consist of the following steps that will be addressed below.

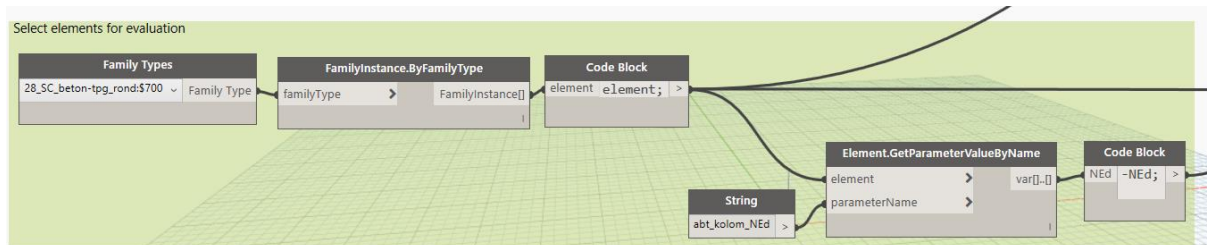
- Select elements for evaluation
- Set available rebar configurations to evaluate
- Create feasibility matrix
- Create costs matrix
- Reduce problem dimensions by elimination of not to be used alternatives
- Determine the best solution for each individual solution (all unique)
- Determine the best solution for a given combination
- Create all possible unique combinations
- Evaluate all combinations
- Visualize simulation results
- Make an informed design decision and specify the number of unique solutions
- Match the optimal rebar configurations to elements.

As example we will use the design of column reinforcement of the 86 columns with a diameter of 700mm for a large office building.

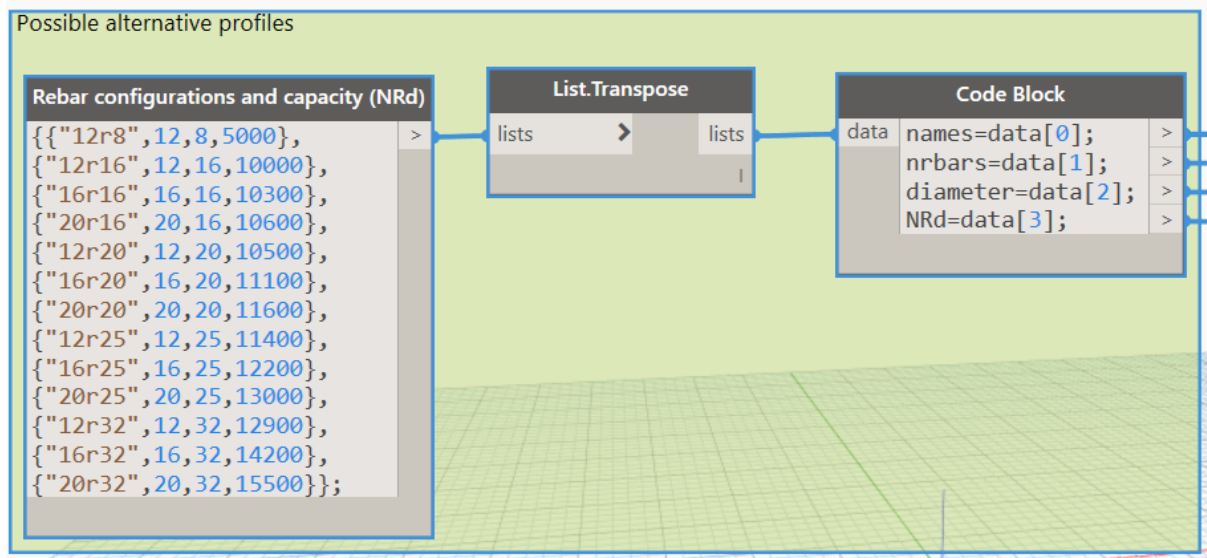


Step 1: Selection elements for evaluation and get structural analysis data.
Select 700mm columns (86 columns) for evaluation.

In this BIM model the structural analysis result was added as meta data to the structural element in an earlier stage. Alternatively in constructing the feasibility matrix, normal forces could have been directly obtained from structural analysis software.

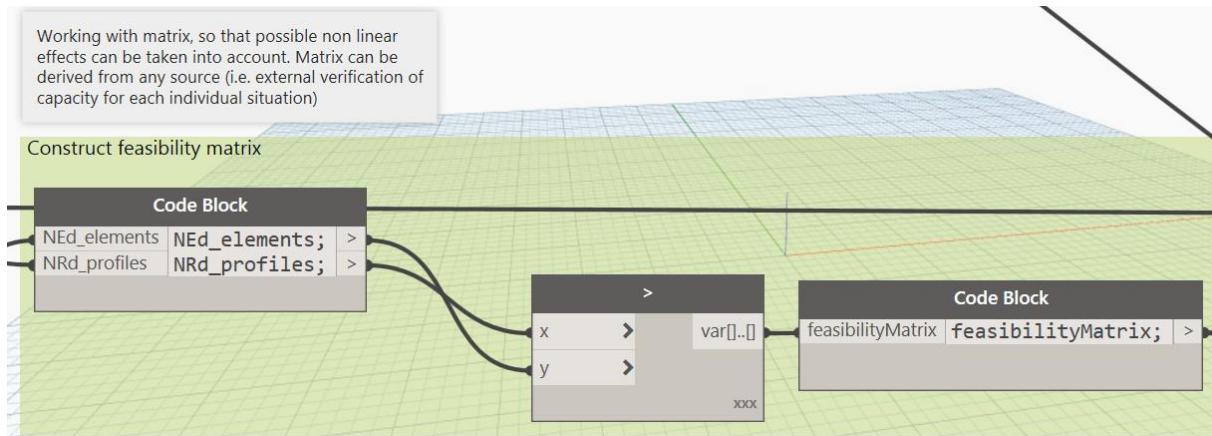


Step 2: Rebar configurations and load bearing capacity
For simplicity load bearing capacity is assumed constant for each configuration (varying bending moments or non-linear effects as buckling are ignored for simplicity).
Alternatively load bearing capacity can be verified for each individual element and situation while constructing the feasibility matrix.
The below example gives 13 alternative column configurations with name of the cross section, number of bars, bar diameter and load bearing capacity.



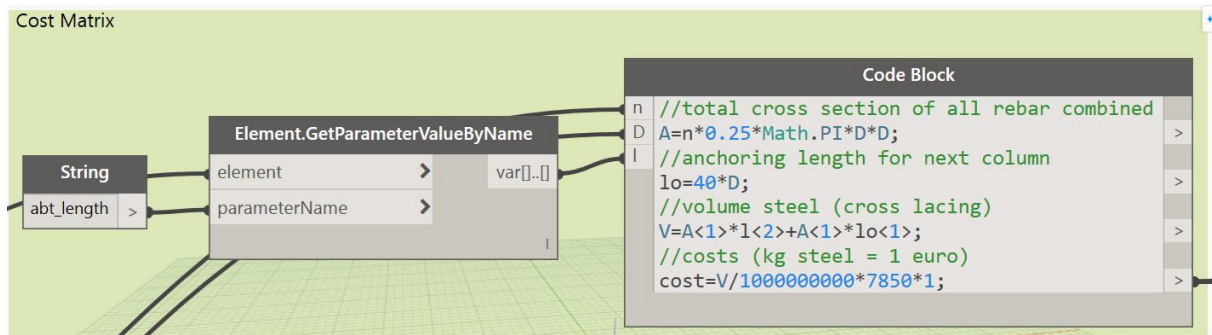
Step 3: Create feasibility matrix

In this step a matrix will be constructed in which each configuration will be tested on every element. This matrix can also be obtained making use of external evaluation tools or be imported from any other source. For simplicity of this example a simple check is used ($N_{Ed} < N_{Rd}$), resulting in a true/false matrix.



Step 4: Create cost matrix

For each element and every rebar configuration the total costs are calculated. Including longitudinal reinforcement only. Total costs are determined by multiplying the cross section area of the individual bar, times the numbers of bar times the bar length including anchoring length. Costs are related to kg only for simplicity. More advanced costs functions can be implemented or other goal functions instead of costs as carbon footprint can be used.



Step 5: Reduce matrix sizes by eliminating not to be used alternatives

The set size of number of rebar configurations can be reduced by a first initial analysis of these matrixes. This reduces the number of unique combinations and thereby calculation time.

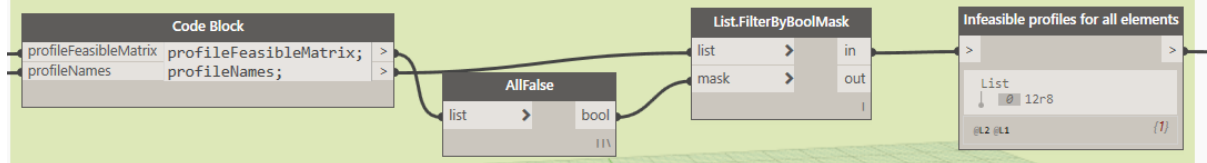
Feasibility matrix							Cost matrix						
Element	Rebar Configuration						Element	Rebar Configuration					
	12r16	16r16	20r16	12r20	16r20	20r20		12r16	16r16	20r16	12r20	16r20	20r20
1	0	0	1	1	1	1	1	57	76	95	89	118	148
2	0	0	0	1	1	1	2	76	101	126	118	158	197
3	0	0	1	1	1	1	3	57	76	95	89	118	148
4	0	0	1	1	1	1	4	66	88	110	104	138	173
5	0	1	1	1	1	1	5	76	101	126	118	158	197
6	0	1	1	1	1	1	6	95	126	158	148	197	247
7	0	0	0	0	1	1	7	133	177	221	207	276	345

infeasible for all elements

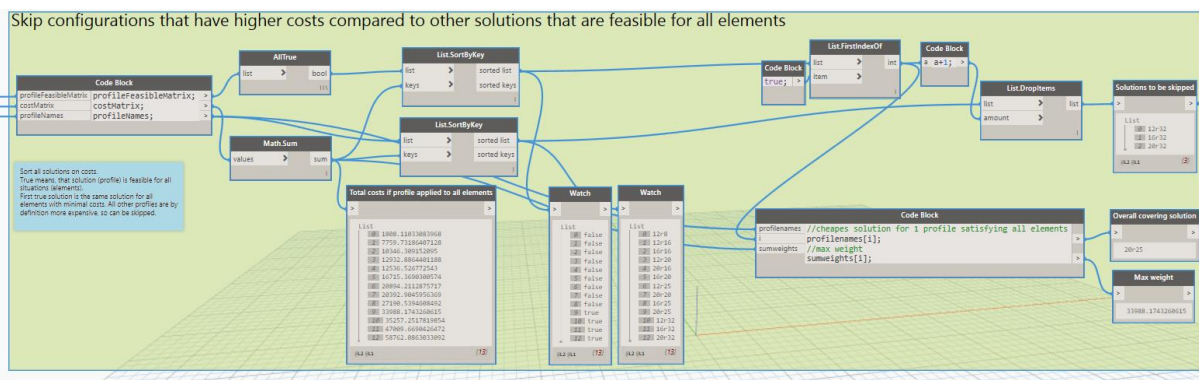
solution is more expensive than other solution that is feasible for all elements aswell (16r20)

Remove the configurations that are infeasible (to weak) for all elements. These will never be used they always fail. In terms of reinforcement this simply means that the capacity is too low. In this example the capacity of the configuration 12r8 is too low for all elements (Ned > NRd of 12 r8)

Profiles infeasible for all elements

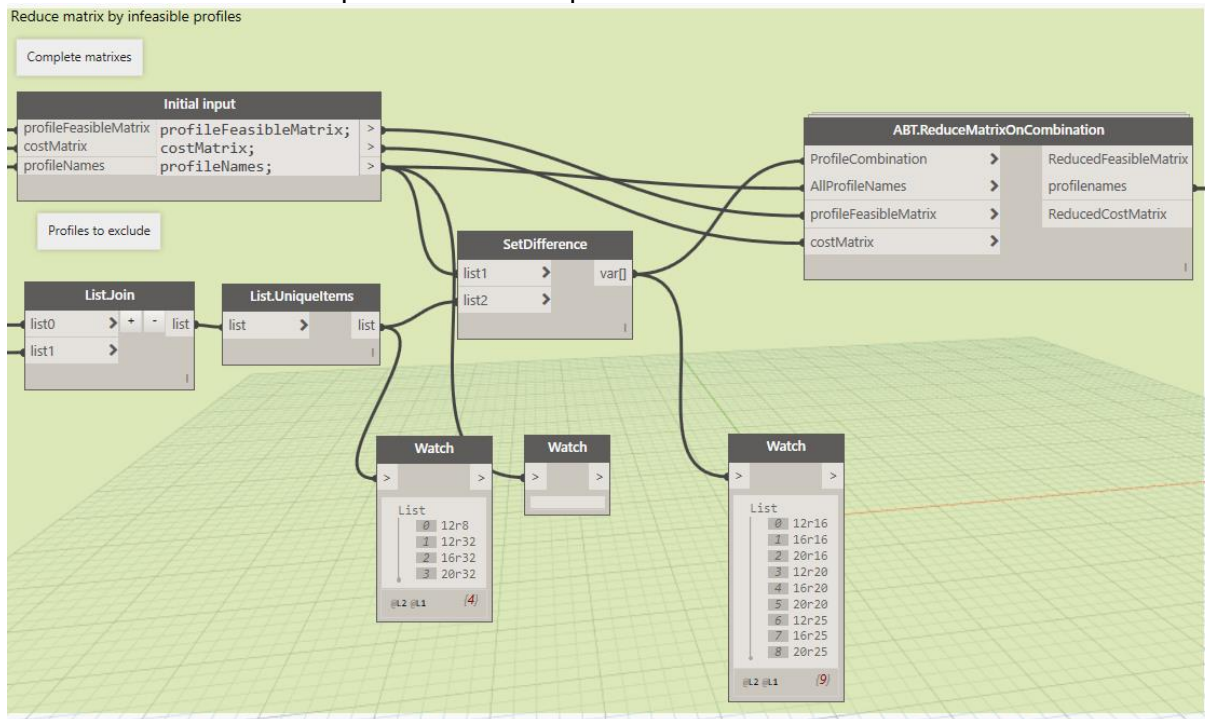


Remove the configurations that have higher costs than others feasible for all elements. In reinforcement design this means that there is more reinforcement available than required and there is another configuration that will be feasible for all elements. (20r25 sufficient for all designs, so the more expensive configurations of 12r32, 16r32 and 20r32 can be excluded).



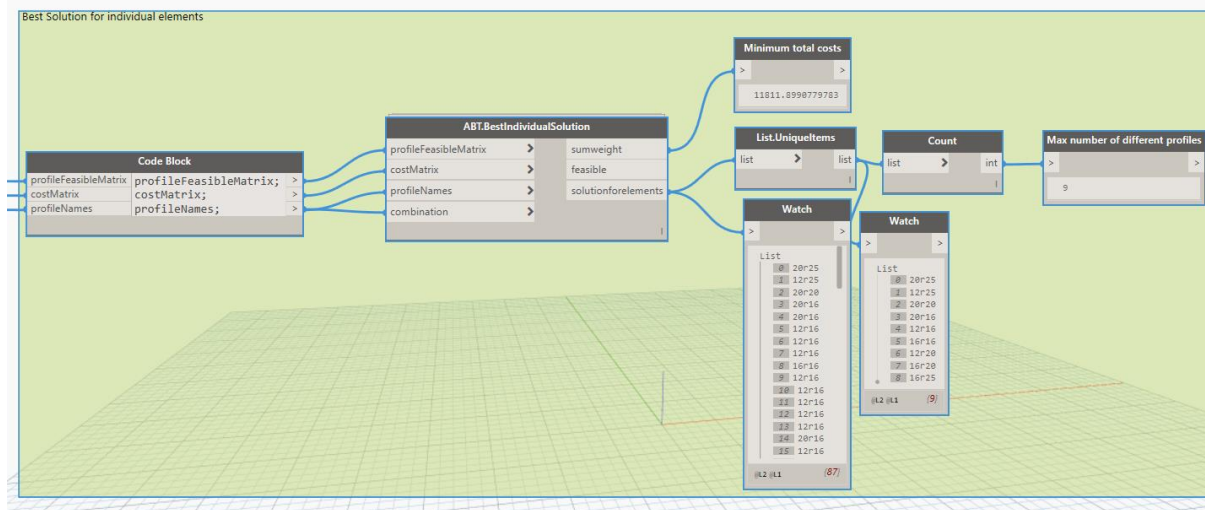
Step 6: Reduce matrixes

Reduce the costs and feasibility matrix with configurations to exclude in order to reduce the number of unique combinations possible.



Step 7: Determine the best solution for individual elements

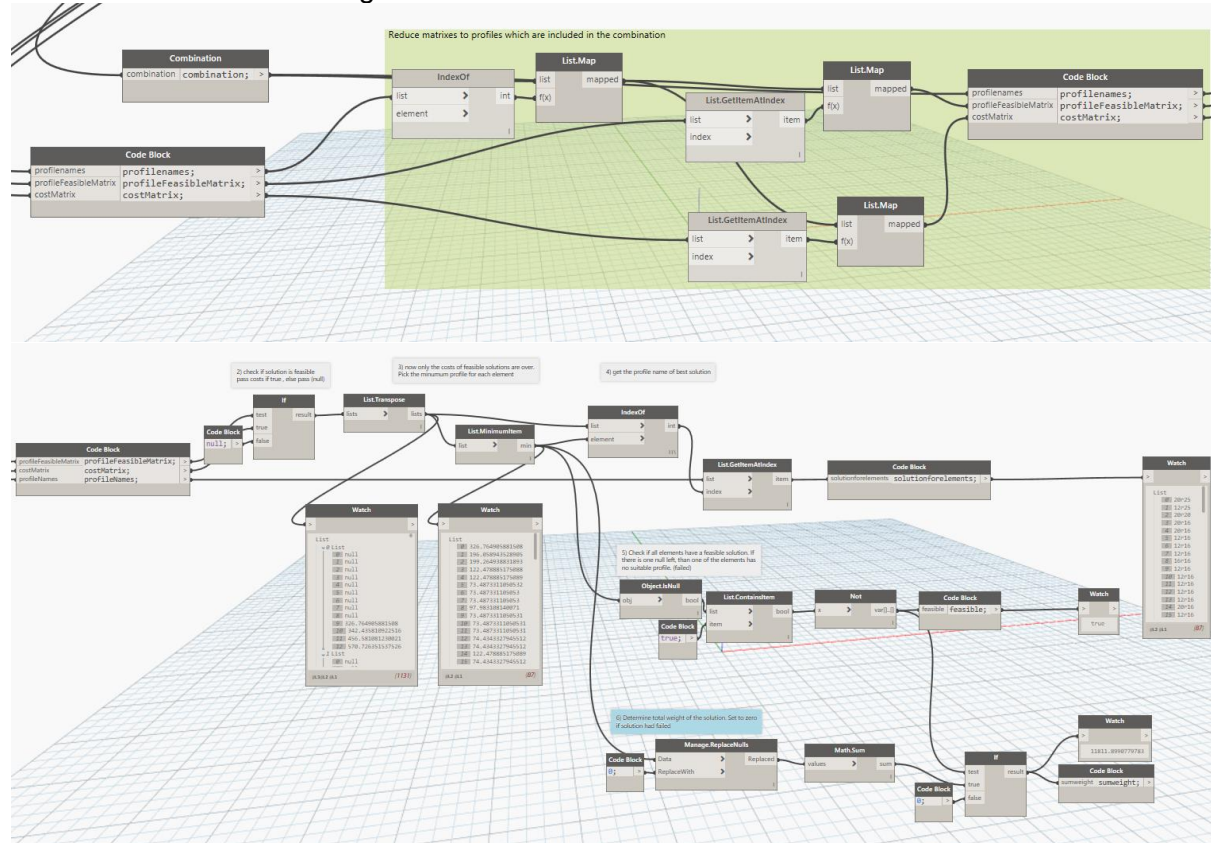
Before starting to think about combinations, first for each element the best solution can be selected. We are finding the best solution for the combination that all possible reinforcement configurations are applied. For this case the best solution is 11812 kg and uses 9 alternative rebar configurations.



Best individual solution for given combination:

This determines the optimal solution for each individual element by the following steps:

- Reduce the matrixes to only the profiles included in the combination (image above)
- Filter the cost matrix on only feasible solutions (if statement using the feasibility matrix as input). If not feasible the null value is passed.
- Pick for each element the item with minimum costs
- Match the minimum cost item to the profile name to get the profile of the best solution
- Check if all elements have a found a solution. If not, this combination is not feasible (fail).
- Determine the total weight of this combination.



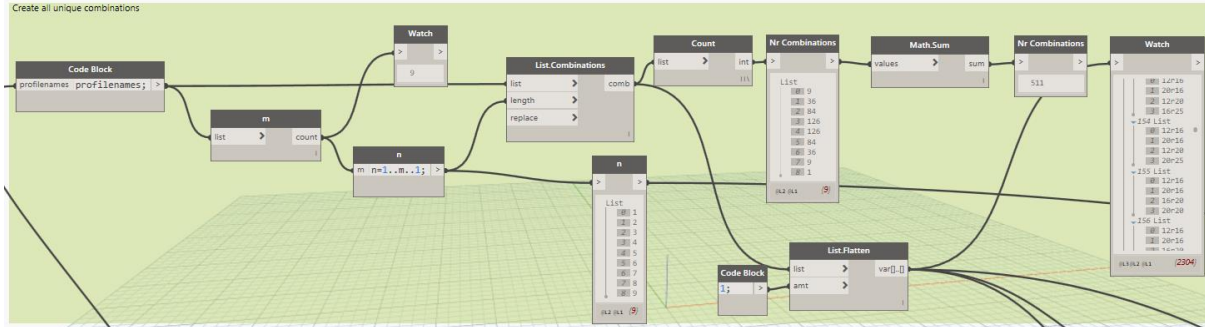
Cost matrix

	Rebar Configuration					
Element	12r16	16r16	20r16	12r20	16r20	20r20
1			95	89	118	148
2				118	158	197
3			95	89	118	148
4			110	104	138	173
5		101	126	118	158	197
6		126	158	148	197	247
7					276	345

VISUALISATION OF COST MATRIX, WITH ELIMINATED INFEASIBLE SOLUTIONS AND SELECTED BEST SOLUTIONS

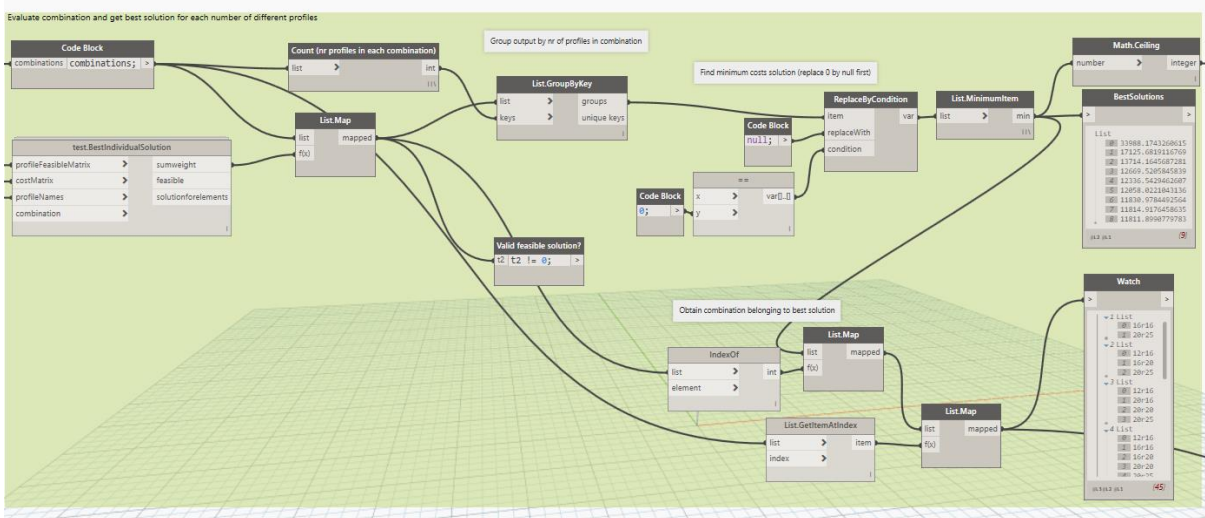
Step 8: Create all unique combinations:

First determine the count of available rebar configurations m and make a list of number of elements to choose $n=1..m$. All unique combination can be generated by dynamo. Total number of combinations for 9 possible configurations is 511 in total.



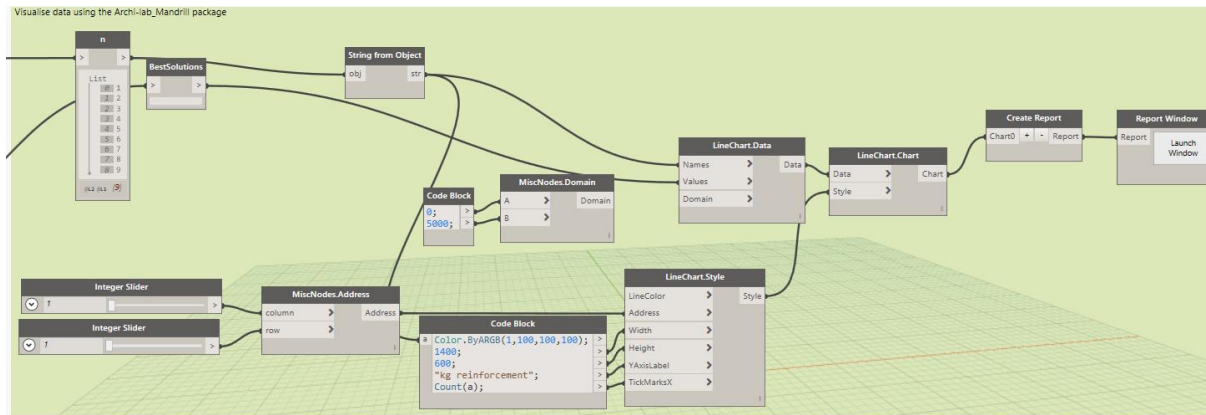
Step 9: Evaluate all combinations:

Evaluate the performance of each combinations of reinforcement configuration. The performance is than grouped depending on the number of configurations in the combination. The minimum value for each count of n is determined, giving the best pick of n profiles from the whole set (minimum weight solution for picking n out of m profiles)

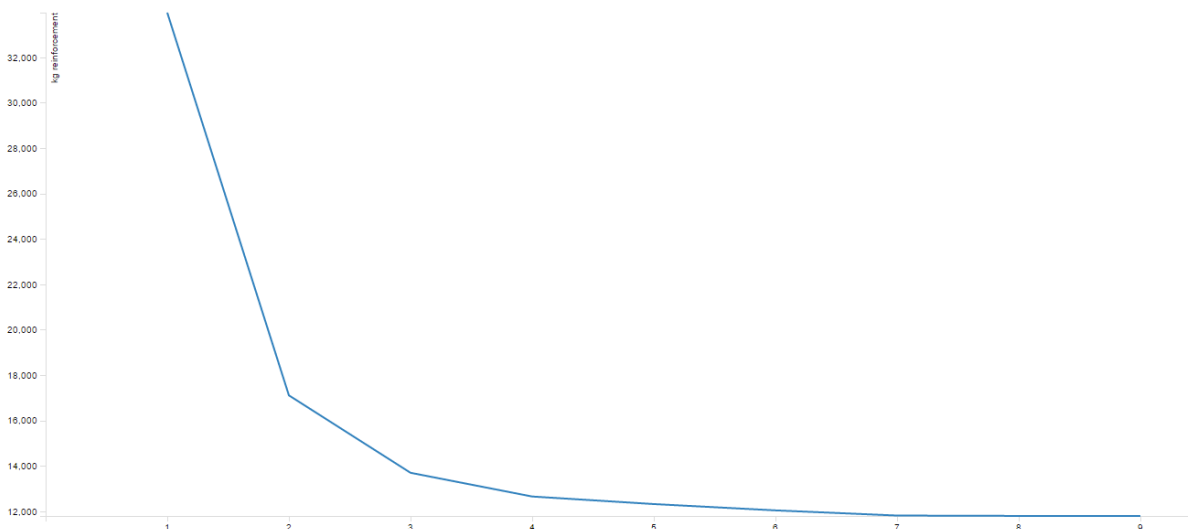


Step 10: Visualize data results

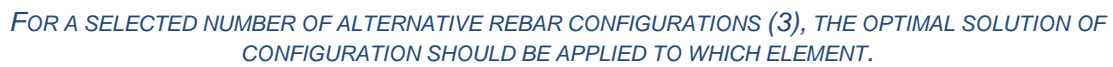
Simulation results from dynamo can be visualized using the Mandrill package which generates charts from data built on D3.js (<https://d3js.org/>) . More information can be found on <http://archi-lab.net/mandrill-a-data-visualization-for-dynamo/>



The obtained graph plotting the optimal minimum weight solution for each quantity of picked unique reinforcement configurations.



The last step is up to the design team which can now make an informed decision of how many alternative rebar configurations will be applied. After which 3D reinforcement can be assigned to the columns either using standard Revit functionality or a Dynamo based workflow.



The screenshot shows a Jupyter Notebook interface with a workflow for matching profiles to elements. The workflow consists of the following components:

- Code Block:** Contains a loop that iterates over `bestsolutions` and calls `List.GetItemAtIndex` to retrieve an item from a list.
- List.GetItemAtIndex:** A block that takes a list and an index as input and returns the item at that index.
- test.BestIndividualSolution:** A block that takes a list of items and returns a solution for each item.
- Watch:** Two blocks that display the results of the matching process. The first watch displays a list of profiles, and the second watch displays a specific value.

The workflow is as follows:

```

bestsolutions: bestsolutions;
//pick number of alternative profiles
n=3;
index=n-1;
List.GetItemAtIndex(list, index)
test.BestIndividualSolution(profileFeasibleMatrix, costMatrix, profileNames, combination)
Watch
Watch

```

The **Watch** blocks display the following results:

- Watch 1:** A list of profiles:

Index	Profile
0	20r25
1	20r25
2	20r25
3	16r20
4	16r20
5	12r16
6	12r16
7	12r16
8	16r20
9	12r16
10	12r16
11	12r16
12	12r16
13	16r20
14	12r16
- Watch 2:** A specific value: 13714.1645607201

[illegible]