

Generation of efficient command sequences for the creation of 3D models in Plaxis 3D

Background

Plaxis 3D is a system for creating, calculating and evaluating 3D models of soil and structures, such as tunnels, dikes and skyscraper foundations. The model creation is command-driven, but there is a rich, interactive GUI that generates and executes commands based on the user's manipulation of the model. The application records all executed commands into a file, which can be imported in Plaxis 3D again.

The recorded command history is, however, most likely not the most efficient way to arrive at a particular end result: users make mistakes, change their minds, refine the model or include superfluous features. There are numerous commands that either do not contribute anything to the end result (e.g. a command that is subsequently undone, or a calculation whose results are thrown away), or could be consolidated into a single command (e.g. numerous move operations that can be replaced with a single move).

It is also possible that the geometry in a model can be specified in different ways, where one way is more effective than another. For example, two cylinders with equal radius could be positioned in such a way that they together constitute one larger cylinder. Obviously, it would, in this case, be more effective to create the geometry as one cylinder. More general cases of the above example can be thought of.

Problem definition

Redundant commands make it difficult to understand a model simply by reading its command history, which in turn means that it is not easy to create variations on existing models by modifying the command history. An additional disadvantage is that simply playing back an existing command sequence will result in a significant amount of time being wasted performing superfluous commands.

Project goals

This project comprises the implementation of a module that either analyses the model directly or through the command history and subsequently generates a readable, optimised set of commands that can be used to create the same model. The implementation must be extensible, and as independent from the current modelling facilities as possible, in order to support future extensions to the modelling facilities.

How to apply

Plaxis bv • Attn: Andrei Chesaru • Delftechpark 53 • P.O. Box 572 • 2600 AN Delft • The Netherlands

Tel +31 (0)15 2517 720 • Fax +31 (0)15 2573 107 • E-mail a.chesaru@plaxis.nl • www.plaxis.nl

