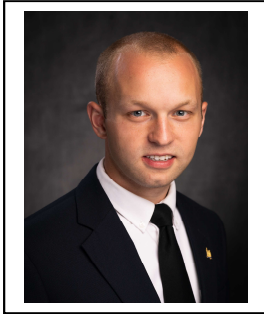


# Comparison of Computer Programs for Analysis of Braced and Anchored Excavations



**Student:** Roger Knittle

**Faculty Advisors:** George M. Filz (filz@vt.edu)

**Sponsor:** CGPR

**Start/Completion Dates:** August 2018 / August 2019

## Project Background

Design and analysis of braced and anchored excavations are tasks that can be expedited through the use of computer software. While some in-house software packages have been developed and used, more often companies are turning to commercially available software. Despite the ease with which these commercial software packages can be used, they can introduce an element of uncertainty because their underlying assumptions and methodology are sometimes hard to identify.

## Project Objectives

This project seeks to address the following questions for several readily available software packages for analysis of excavation support systems:

1. What methodology is being used? What assumptions accompany each methodology?
2. What are the capabilities of the software? Can multi-level as well as single-level systems be considered?
3. What degree of support for the software is offered by the vendor?
4. How do the results from multiple software packages compare for the same problem?

By addressing the above questions, the report will offer a single reference source for considering different software options for braced and anchored excavation systems. The scope of the project excludes general purpose finite element programs that are not specifically tailored to excavation support design.

## Research Plan and Progress

A list of research tasks is provided below:

1. Compiled a list of commercially available software packages for anchored and braced excavations. These include CWALSHT, PYWALL, GEO 5, DEEPEX, CT-SHORING, GEOSTUDIO, TBWALL, WALLAP, and SPW911.
2. Compiled supporting program materials. This included reading through the vendors' websites and any available user's manuals.
3. Obtained access to free or demo copies of some programs and ran sample problems to gain a working-knowledge of the software.
4. Created structure for draft of the final report.
5. Compile a list of high-quality and relevant example problems to run through the software packages to permit comparisons of capabilities and results.
6. Create a matrix comparing the features of the software packages.
7. Prepare the findings in a report format.