11th World Congress on Structural and Multidisciplinary Optimization
7th - 12th, June 2015, Sydney Australia

Optimizing snap-through structures by using gradient-only algorithms

Schalk Kok[1], Daniel N Wilke[2]

[1] Department of Mechanical and Aeronautical Engineering, University of Pretoria, South Africa, schalk.kok@up.ac.za
[2] Department of Mechanical and Aeronautical Engineering, University of Pretoria, South Africa, nico.wilke@up.ac.za

1. Abstract
This paper presents a robust technique to design snap-through structures. The structural analysis of the snap-through structure makes use of an arc length control algorithm. To ensure robustness, the prescribed arc length per increment is halved whenever complex roots are encountered in the arc length control algorithm, or when the required number of Newton-Raphson iterations exceeds five. The resulting structural analysis is robust, but now different analyses makes use of different increment sizes. The resulting optimization problem, which minimizes the error between a target load-deflection curve and the simulated curve, now contains numerical discontinuities. We demonstrate how gradient-only optimization algorithms can robustly optimize such problems.
2. Keywords: Snap-through, arc length control, gradient-only optimization.

3. Introduction
Analysis of snap-through structures usually requires the use of the arc length control method [1]. In this method, the usual equilibrium equations are augmented with parametrizing the prescribed loads, and then solving this free parameter by setting the computed arc length increment equal to some prescribed value. Usually this prescribed arc length is selected as constant for each load increment during the analysis.

Sometimes, the arc length control algorithm fails to find a solution for a specific load increment. This can manifest in two ways. First, the quadratic control equation may indicate a complex root. Secondly, the Newton-Raphson scheme used to solve the global equilibrium equations may need an excessive number of iterations to solve. Even in cases where the number of iterations are reasonable ($<$20), our experience indicates that the algorithm may then find solutions on other branches of the equilibrium path. Therefore, we halve the prescribed arc length increment whenever complex roots are encountered, or whenever the number of Newton-Raphson iterations exceeds five.

The automatic adjustment of the prescribed arc length increment presents a difficulty when attempting to design a snap-through structure to provide a specific load-displacement curve, using classical gradient-based optimization algorithms. The adjusting of the prescribed arc length has the same consequence as allowing automatic time increments for transient problems: the cost function of the resulting optimization problem contains discontinuities since the same problem is analyzed repeatedly, each time using different prescribed arc length values.

We have significant experience in solving optimization problems that uses non-constant discretization algorithms [2]. Here we demonstrate how classical gradient based algorithms struggle to solve this snap-through design problem, since these algorithms can get stuck at the numerical (non-physical) discontinuities. We also demonstrate how gradient-only algorithms, a family of algorithms that do not use function values at all, succeed in designing the snap-through structures.

4. Numerical example
We demonstrate our algorithm using a simple 1D optimization problem. The structure of interest is similar to the well known Lee frame, but here we use a geometrically nonlinear truss code for the analysis. The structure is depicted in Figure 1. Note that two vertical loads are applied, as well as a single horizontal force. The structure is analysed for the case $\lambda = 0.5$, i.e. the horizontal force is one quarter of the total vertical load.

We use our robust arc-length control algorithm to first demonstrate that the problem can be solved for a large range of ideal prescribed arc length increments $\overline{L}$. For this problem a total arc length of 70 units is prescribed. The analyses are performed for ideal arc length increments of $\overline{L} = 0.21875$, 0.4375, 0.875, 1.75 and 3.5. If these ideal arc lengths can be used without modification throughout the analyses, this would translate to 320, 160, 80, 40 and 20 equal increments to analyse the problem. Figure 2 depicts the horizontal and vertical displacement of point C, using the two extreme choices for $\overline{L}$. Using $\overline{L} = 0.21875$ required 320 increments, while using $\overline{L} = 3.5$ required 26 increments (since some load steps required $\overline{L}$ to be reduced). Figure 3 depicts the equilibrium path of point C in space. Note that at about 70% of the total arc length, the equilibrium path undergoes a sharp turn. It is this corner that is difficult to negotiate if we use a large ideal arc length increment.

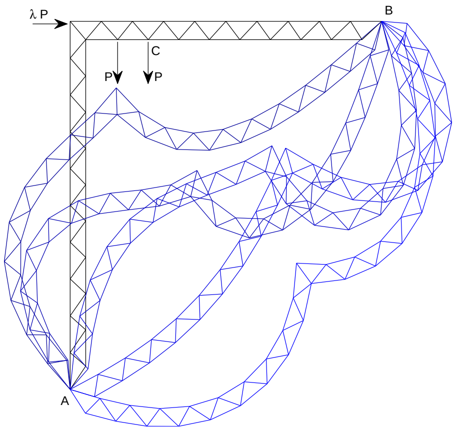

Figure 1: Undeformed and deformed truss, depicting 5 equal arc length increments

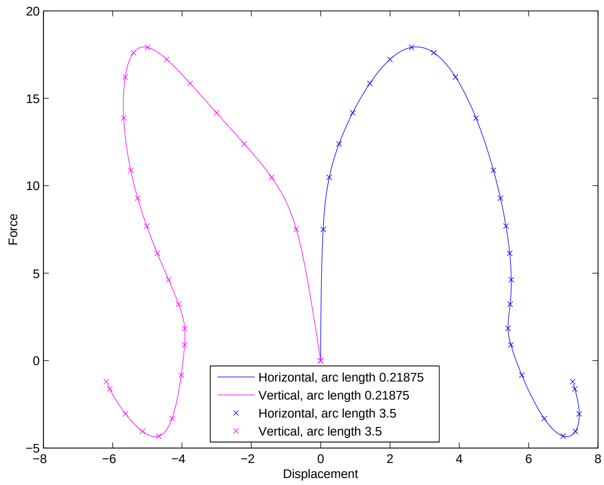

Figure 2: Undeformed and deformed truss, depicting 5 equal arc length increments

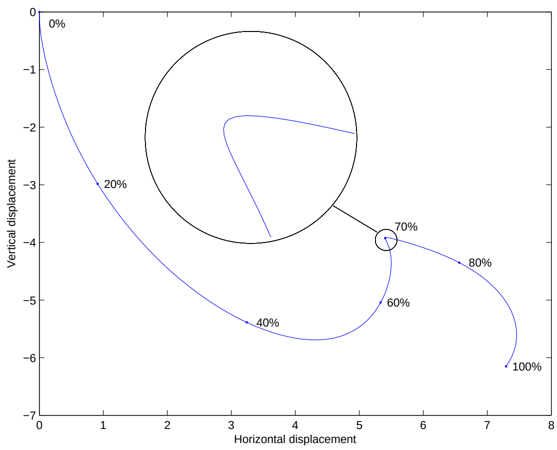

Figure 3: Displacement of Point C for a total arc length of 70 units.

Now that the analysis has been demonstrated to be robust, we can proceed to investigate an optimisation problem. We pose a 1D optimisation problem, using the horizontal force component $F_x = \lambda P$ as the single design variable. We use the horizontal and vertical force-deflection curves as target curves, for the case $\lambda = 0.5$ and $\overline{L} = 0.21875$ (visible as the solid lines in Figure 2). In cases where we use a larger value for $\overline{L}$, the analysis requires fewer increments. In order to compute a cost function (sum of square error for 320 increments) for these cases, we construct an interpolating spline between the total arc length and the quantity of interest (horizontal displacement, vertical displacement and load). This allows the computation of the cost function for any number of load steps available. Note however that this step does introduce discontinuities in the cost function as soon as one analysis requires a different number of increments as another. To demonstrate the nature of these discontinuities, we compute the cost function for various choices of $\overline{L}$. The cost function is computed for $\lambda \in [0.47; 0.53]$. Since the target curves were computed for $\overline{L} = 0.21875$ and $\lambda = 0.5$, we expect the cost function minimum to be at $\lambda = 0.5$.

Figure 4 depicts the cost function curves for various choices of $\overline{L}$. Note that as the ideal arc length is increased, the optimum of the cost function curve drifts from the correct solution ($\lambda = 0.5$), and some discontinuities occur. This behaviour is identical to the variable time-stepping problem in [2]. Classical gradient based algorithms may become trapped in the highlighted local minimizers, if a line search process happens to locate these local minimizers. These step discontinuities manifest as ridges in higher dimensional optimization problems, and in our experience are more likely to affect classical gradient based algorithms.

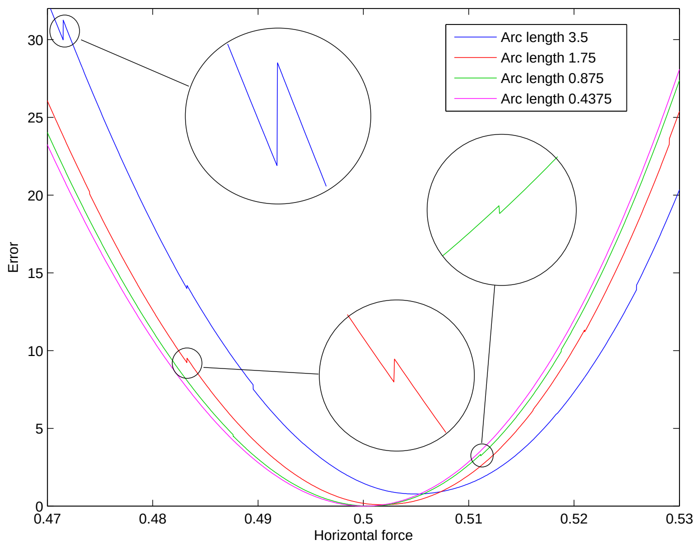

Figure 4: Cost function as the horizontal force varies

Figure 5 depicts the gradient of the cost function, using analytical derivatives. Although the cost function suggest discontinuities, any analysis is either to the left or the right of the discontinuity. For such an analysis, an analytical sensitivity analysis can be performed. Therefore, we have the gradient of the cost function available everywhere.

The suggested algorithm to solve this discontinuous optimization problem, is a gradient-only algorithm. For algorithmic details, refer to [3]. In essence, function values are ignored and we search for a sign change in the cost function gradient. Notice from Figure 5 that the function gradient also contains small discontinuities, but the information remains consistent (i.e. there is no sign change in the gradient over the discontinuity).

Note the drift in the optimum for the largest choice of $\overline{L} = 3.5$, from the actual solution $\lambda = 0.5$ to $\lambda = 0.5049$. The quality of the solution can only be judged by comparing the target curves to the solved load-displacement curves, as done in Figure 6. Notice that all the features of the target curves are captured.

Finally, to further motivate the use of varying arc-length increments, Table 1 summarizes the analyses times for various choices of $\overline{L}$. There is clear benefit in using large ideal arc-lengths, but this benefit diminishes gradually. If the ideal arc-length is chosen too large, almost every increment has to be re-run with a smaller arc-length, and this reduces the benefit of using large arc-lengths. Nevertheless, for this problem we can expect a speed increase of a factor 6 when we solve the optimization problem using $\overline{L} = 3.5$ rather than $\overline{L} = 0.21875$.

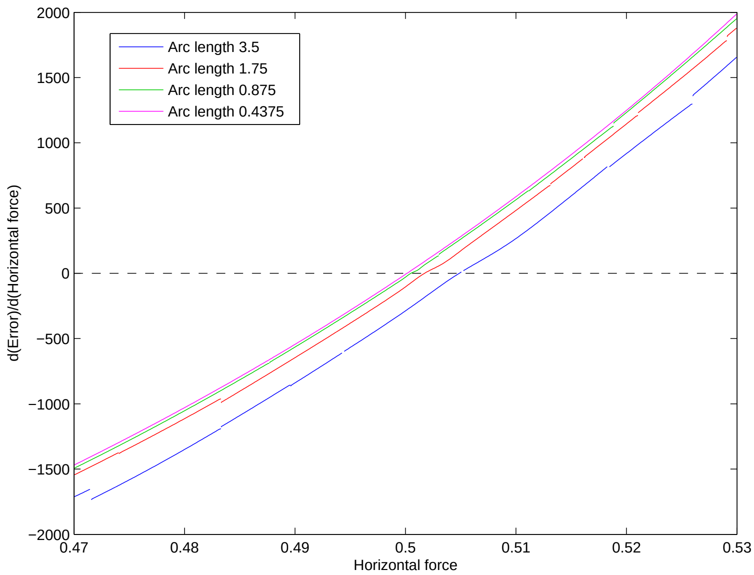

Figure 5: Cost function derivative as the horizontal force varies

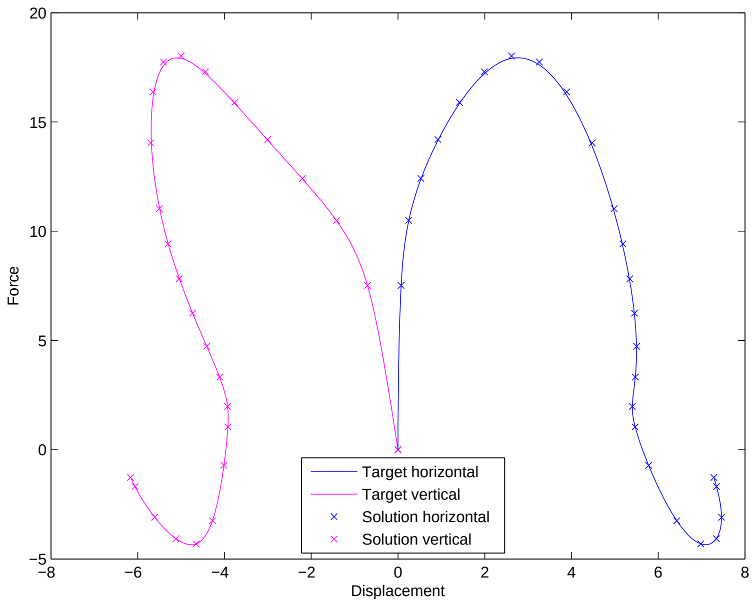

Figure 6: Optimal solution using ideal arc length $\overline{L} = 3.5$ versus the target load-displacement curves

Table 1: Analysis times for different choices of $\overline{L}$

$\overline{L}$	Time(s)
0.21875	8.83
0.43750	5.17
0.87500	2.76
1.75000	1.68
3.50000	1.46

5. Conclusions

We demonstrated that gradient-only algorithms can be used to solve optimization problems that use variable arc-length control methods in the analysis step. Although the analysis algorithm introduces numerical discontinuities into the cost function, gradient-only methods are insensitive to these discontinuities. We are currently working on higher dimensional problems, including shape design variables.

6. References

[1] M. Ritto-Corrêa and D. Camotin, On the arc-length and other quadratic control methods: Established, less known and new implementation procedures, *Computers & Structures*, 86, 1353-1368, 2008.

[2] D.N. Wilke, S. Kok and A.A. Groenwold, The application of gradient-only optimization methods for problems discretized using non-constant methods, *Structural and Multidisciplinary Optimization*, 40, 433-451, 2010.

[3] D.N. Wilke, S. Kok, J.A. Snyman and A.A. Groenwold, Gradient-only approaches to avoid spurious local minima in unconstrained optimization, *Optimization and Engineering*, 14(2), 275-304, 2013.