DAIMLER

White Paper

Common Pitfalls
in
Parametric CAD Systems

Version:	1.0
Erstelldatum:	August 2008

Verantwortlich: Dr. Edgar A. Gerteisen
GR/EPP
☎: +49 731 / 505 2423
🖹: +49 711 / 3052119019
💻: edgar.gerteisen@daimlerchrysler.com

Verteiler:

Name	Organisation	Kontakt
info@ledas.com	Ledas Ltd	**avenue Lavrent'ev, 6, Novosibirsk, 630 090, Russia** phone: **+7 383 335 65 04** fax: **+7 383 335 62 56** e-mail: info@ledas.com

Änderungshistorie:

	Version	Datum	Bearbeiter	Änderungsgrund
1	1.0	2008-08-14	Gerteisen	First Version
2				
3				
4				

Content

1 Introduction

Parametric CAD Systems promise great potential with regard to product optimization. A given parametric set may be varied to produce variant instances of the geometry that further are going to be functionally assessed by means of computational methods, so-called CAE methods. If such a methodology is being embedded within a numerical optimization procedure, e.g. evolutionary strategies, the parametric set that rules the geometric and thereafter the functional behavior may converge to an optimum or at least may lead to an improvement of the corresponding product.

Of utmost importance in such an optimization layout is the strong causality between the parametric set and the geometric entity, i.e. a small disturbance of the parametric set should lead to a small disturbance of the geometric entity. Although this causality may not be strictly accomplishable in any situation, it should be strived as much as possible. Several effects can be figured which influence the procedure from the parametric input set to the resulting geometric entity. Beside the design concept applied by the CAD designer, the robustness of an update upon parameter variation depends on the sequence of geometric operations and the stability of the operations themselves. Robustness is much less restrictive compared to strong causality, though suboptimal designed geometries exhibit severe update issues. While those parametric sets that are not able to be updated can simply be skipped and constrain the parameter's corridor within the automatic optimization procedure, missing minimal causality may mislead the overall optimization strategy.

An update is reevaluating the geometric design from the very beginning of the input parameter set according to the design history known by the design system and stored implicitly in the CAD model file. The typical design of a 3D CAD surface model starts with sketches that afterwards are being extruded or sweeped to three-dimensional surfaces, which thereafter are being treated with operations such as intersection, trimming, merge, fillet, offset, etc. Disregarding the effects of the 3D operations, at least the aforementioned causality depends on the sketching system. The characteristics of such a sketching system are investigated in the following sections.

2 CAD Sketch definition

2.1 General considerations

The definition of a sketch is usually carried out interactively in a two-dimensional (2D) planar setup. Thereby one-dimensional (1D) geometric elements such as points and curves are positioned in the plane and connected together. In order to produce a parametric steering of these elements, they are supplemented with measures that may be varied. One powerful method to enable a generic sketch setup is to express these measures as geometric constraints and solve the sketch entity by means of a geometric constraint solver.

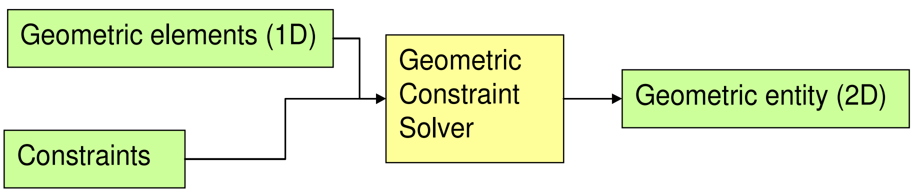

Figure 2-1: Illustration of the concept of 2D geometric constraint solving. One-dimensional geometric elements are positioned and connected together by means of constraints. The geometric elements my initially be placed arbitrarily.

The concept of geometric constraint solving is exhibited in Figure 2-1. In principal, the initial placement of the one-dimensional geometric elements can be arbitrarily and the geometric constraint solver takes care of positioning the elements and providing transition conditions such as tangency between two curve elements. However, as demonstrated in the subsequent sections, the constraint solving turns out to be a not very well posed problem. At least in some well known CAD systems the sketch implementation exposes non-unique solution behavior.

2.2 Sketcher of a prominent commercial CAD System

The aforementioned non-uniqueness behavior is shown in Figure 2-2. The designer draws his design intent into the sketch and puts measures as well as subsidiary constraint conditions until the sketch is supposed to be iso-constrained. In principal, in an iso-constrained system all degrees of freedom are fixed similar to a statically defined system in structural mechanics.

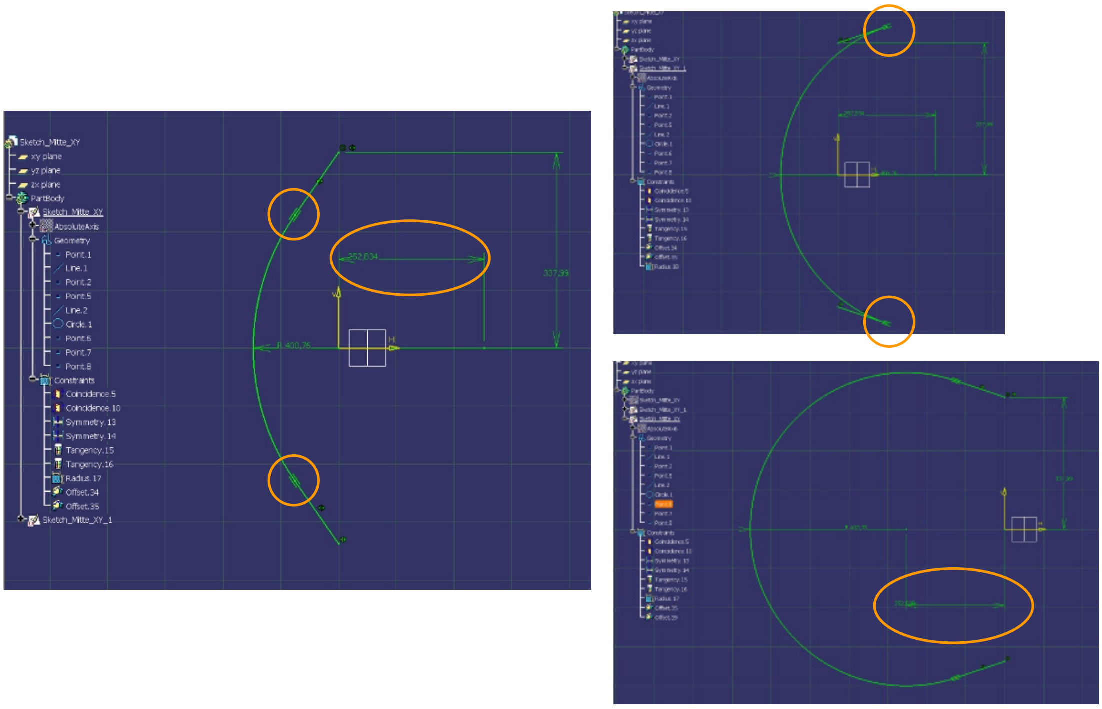

Figure 2-2: Exposition of non-uniqueness of a simple iso-constrained sketch. The three results shown are based on exactly the same constraint setup. On the left it is shown the original design intent. On the right (upper part) the tangent condition flips into a non-continuous (C1) position, on the right (lower part) the distance is flipped from right to left.

However, as can be seen in Figure 2-2 (right side), the degrees of freedom are not eliminated rigorously. With exactly the same constraint setting, the tangency condition might flip, Figure 2-2 (right side, upper), to a configuration where the orientation at the tangency point changes sign. Another situation is shown in Figure 2-2 (right side, lower) where the distance flips from H>0 to H<0 leading to unintended sketch entity. While the tangency issue can be resolved only interactively by trial and error, an extended menu option (Figure 2-3) is provided in the interactive session to steer the distance flip.

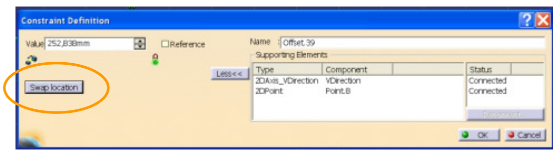

Figure 2-3: Extended distance constraint menu that allows for swapping location.

Although the described behavior might be less relevant in an interactive system where the designer can control the layout visually, a well-defined constraint set would provide better assistance in interactive work as well. As stressed already in the introduction, uniqueness and robustness is of utmost importance in an automated system.

2.3 Ledas Sketcher

Another prominent candidate to investigate the behavior described above is the 2D constraint solver LGS2D of the Ledas company [Ref. 1] that ought to be the base technology of the sketcher analyzed in section 2.2. The present study is carried out on the basis of the sample demonstration code for the ledas constraint solver library which can be found on the ledas web site [Ref. 1]. Beside this demo code that needs to be installed and requires OpenCascade base libraries, the Ledas website contains a flash demo which can be started simply from the web browser.

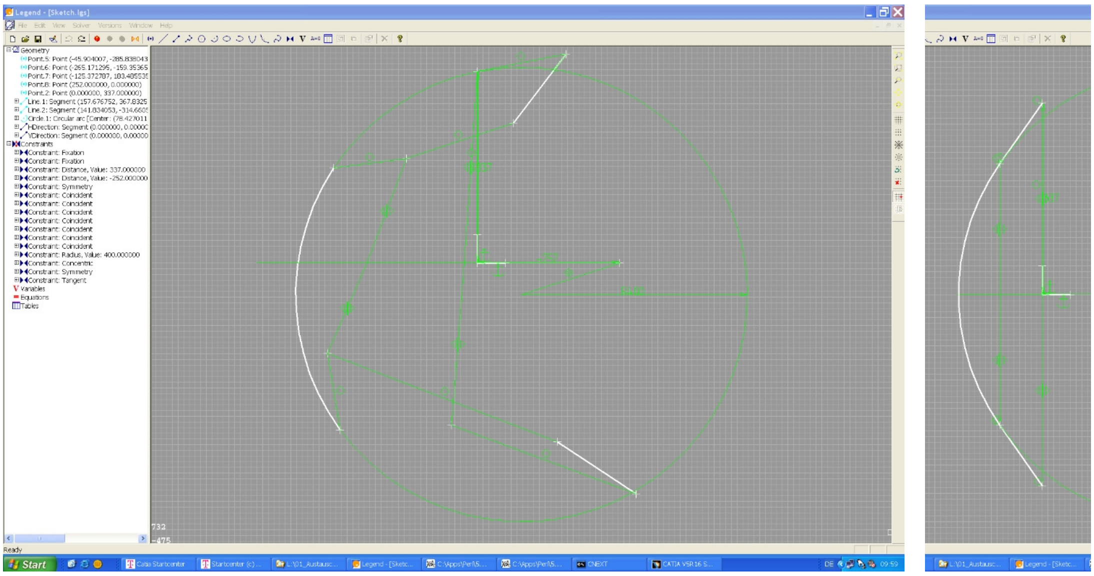

Figure 2-4: Geometric elements and constraint configuration generated with the Ledas demonstration application. A more or less random positioning of input geometries (on the left) is solved (right) upon the defined constraints.

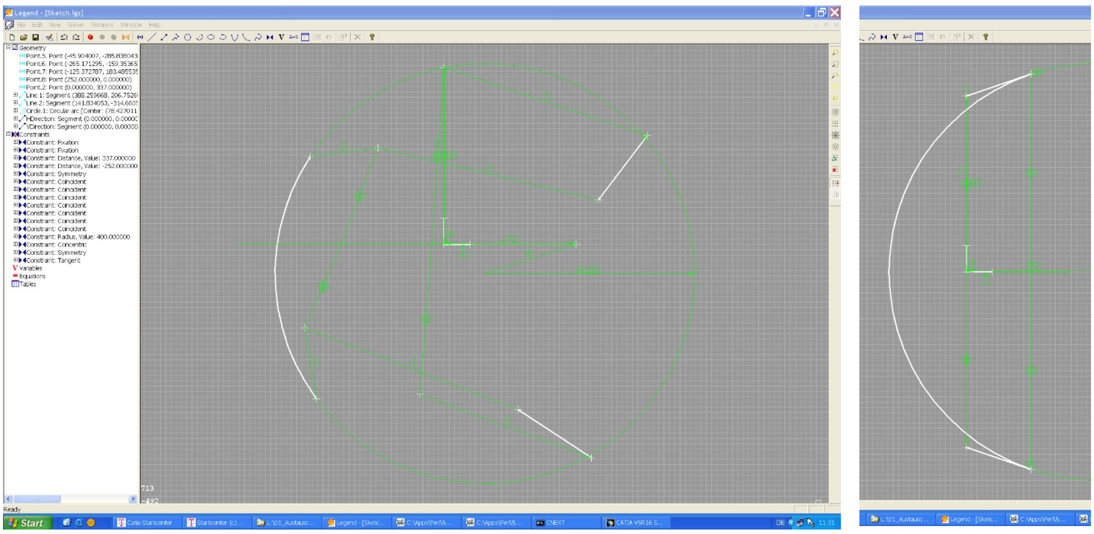

Figure 2-5: Configuration with different initial conditions of geometric random set leads to different sketch entity result.

In contrast to the sketcher paradigm analyzed in section 2.2, where frequently the constraints are imposed implicitly, any constraint has to be set explicitly in the Ledas demo application. This approach results in a rigorous distinction between geometric elements and constraint conditions. However, similar to Figure 2-2, the result of solving the constraint system is non-unique and obviously depends on the initial position of the geometric elements as can be seen in Figure 2-4 and Figure 2-5. In these Ledas examples the symmetry conditions are changed. Whereas in Figure 2-2 the line symmetry is imposed explicitly, the same symmetry is imposed in Figure 2-4 and Figure 2-5 by setting the symmetry condition on the two points of the line. The experimenting with different constraints again depicted another effect, i.e. the non-uniqueness of the solution depends on the kind of constraints. In addition, it was figured that the solution depends on the sequence of constraint setting.

3 Discussion

Geometric constraint solving turns out to be a difficult task that involves a sophisticated combination of symbolic and numerical methods [Ref. 1]. The constraint set for a bulk of geometric elements can be under-determined or over-determined. Even for a fully constrained system that might be compared to a statically defined system in structural mechanics, the solution may be non-unique. The sketch systems investigated expose chaotic behavior in the sense that the solution depends on several conditions and a minimal change in these conditions can lead to dramatic changes in the resulting sketch entity. These dependencies are the initial placement of the geometric elements (I), the kind of constraints (K), and the sequence of constraints processing (S). Accordingly, the sketch entity can be expressed by the following formula:

$$\text{Geometric Sketch Entity} = f(I,K,S)$$

The two issues of the exemplary CAD sketcher systems identified in section 2.2 and section 2.3 are:

- non-uniqueness of tangency orientation,
- distance location with respect to the reference geometry element.

While the distance location can be swapped via a menu option, see Figure 2-3, no switch to easily swap the tangency orientation exists. Simultaneously, a more rigorous and non-interactively controlled approach is demanded, if parametric variation of the geometry should be applicable in the context of mechanical optimization.

In summary, the experimentation with the constraint system leads to the assumption that the non-uniqueness is attributed more to the geometric interpretation of the constraints than to the constraint solving process itself. A smarter interpretation of the constraints would lead to a fully determined natural solution of geometric problem, i.e. the solution that is expected by the user. One step towards more rigorous constraint interpretation appears to be implemented in the Ledas demonstrator with regard to the distance location which is fixed by an additional sign operator, see Figure 2-4 and Figure 2-5. There appears to be no restriction to introduce a similar extra condition for the tangency based on the direction of the bonding segments at the joint.

4 Conclusion

In the present investigation some limitations of parametric CAD systems are figured, if intended to be applied in the context of numerical optimization. Specifically, the non-uniqueness of geometric constraint solving leads to robustness issues when applied in automatic geometry update cycles upon parameter variation.

The experiments suggest that the non-uniqueness issue is associated to non-rigorous geometric interpretation of the supplied constraints finally leading to unintended or non-natural solutions. The corresponding non-unique constraints are identified to be the distance and the tangency conditions. These may be amended by additional conditions, i.e. location and direction respectively.

Although the non-natural or unintended solution may be identified visually in an interactive system and resolved in a more or less straightforward fashion, a rigorous interpretation of the constraint set would provide better assistance in interactive work as well. However, the uniqueness and robustness is of utmost importance in an automated system. The suggested enhancement with regard to the geometric interpretation of distance and tangency strives towards a more reliable parametric CAD system.

5 References

Ref. 1 www.ledas.com, The Ledas website with lots of information concerning geometric constraint solving technology of Ledas, in addition incorporating some feature rich demonstration code, i.e. http://ledas.com/products/lgs2d/applications/, provides links to
→ flash demo of Ledas constraint solver,
→ demonstration application of a sketcher based on OpenCascade.

Ref. 2 http://ledas.com/products/lgs2d/technology/, Summary of Ledas technology.