From *SIAM News*, Volume 41, Number 8, October 2008

Semi-infinite Optimization Meets Industry: A Deterministic Approach to Gemstone Cutting

By Karl-Heinz Küfer, Oliver Stein, and Anton Winterfeld

For five centuries artisans have been cutting faceted gemstones, creating jewels that sparkle with internally reflected light, showing off the "fire" of the stones. Working with stones that are transparent or translucent, a cutter makes the most of the color of a stone through careful choice of the angles between facets, depending on the refractive index of the material. At the same time, gem producers have their own goal: to use as much of the volume of the rough stones as possible. By applying modern methods of semi-infinite optimization to this problem, we were able to improve the volume yield significantly while guaranteeing optimal optical properties of the faceted gemstones.

Experienced cutters of such stones—rubies, sapphires, tourmalines, and others—have always done their work manually, de-ciding on shape and facet design without technical support. In the recent work, researchers at the Fraunhofer Institute for Industrial Mathematics in Kaiserslautern, to-gether with a consortium of mechanical engineering companies and a gem producer, developed a fully automatic process for industrial gem production based on an optimal balance of volume yield and ideal proportions of the resulting gemstones. The machine first maps the surface of the rough stone by projecting narrow bands of light onto it. Using the scan data, the optimization software chooses one of many basic shapes (e.g., emerald, trillion, or pear; see Figure 1) and a suitable arrangement of facets (e.g., brilliant, Ceylon, or Portuguese cut; see Figure 2), and finds an embedding of the faceted gemstone in the rough stone such that the volume yield is maximized. Once the optimal solution has been found, a grinding and polishing plan is automatically generated and transferred to a CNC machine. Finally, with no manual intervention, the faceted gemstone is ground and polished to a precision of 10 micrometers.

Figure 1. *From left, emerald, trillion, and pear cuts.*

Parameterization of a faceted gemstone begins with the position and orientation of the cut stone within the rough stone; other parameters describe the shape of the faceted gemstone, including height, radius, and aspect ratio. The first person to investigate the influence of different shape parameters on the appearance of the brilliant cut was Marcel Tolkowski, at the beginning of the 20th century [14]. The optimal proportions he calculated (called the Tolkowski Ideal Cut) have long served as a reference for the quality of a brilliant cut. Recently, numerous groups have studied the optics of faceted gemstones (collections of articles can be found, for example, in [9] and [10]).

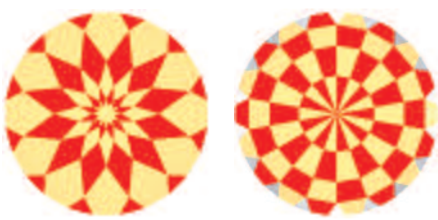

Figure 2. *Even for a fixed shape like the round cut, a large number of facet variations need to be taken into account.*

The volume-optimization problem has been studied much less extensively. The methods developed to date concentrate on diamond cutting and assume a fixed polyhedral geometry of the faceted gem. Few references are available (see [8] and [15] or, for commercial publications on problems of this type, [4, 5], [13]).

The available methods are appropriate for diamond-cutting problems (where one fixed facet arrangement, the so-called round brilliant cut, predominates) and cannot be applied to the cutting of colored gemstones because of subtle yet very important differences between these problem classes. On the one hand, the lapidary proportions are much less restrictive than the brilliant cut proportions. On the other hand, the assumption of a fixed facet arrangement is not appropriate for the lapidary cutting problem because of the large number (several hundreds) of possible geometries.

The requirement that several hundreds of parameterized cut variations be taken into account precludes the use of fixed polyhedral geometries and leads to a crucial question, one that is left unanswered by the optimization methods developed so far but that needs to be answered before we can tackle the lapidary cutting problem:

If the polyhedral description of a faceted gemstone cannot be used during optimization, what, then, do we optimize?

Modeling and Complexity Reduction

Generations of gemstone cutters have answered this question, as elaborated in most gemstone-cutting textbooks (e.g., [1, 2]). In fact, each stone is roughly pre-formed before the planar facets are cut. Figure 3 shows a typical pre-formed shape, the so-called calibration body.

Pre-forming achieves two main goals: It removes major impurities and inclusions from the rough stone, and fixes the basic shape, approximate proportions, and orientation axis of the eventual gemstone so as to yield maximal volume. This led us to the idea of replacing the polyhedral description of a gemstone by smooth nonlinear, flexibly parameterized calibration bodies. Another question arose immediately: What mathematical method can be used to find the optimal embedding of a parameterized calibration body in the rough stone? Such problems of optimal embedding, which are called "design centering" problems, are known to be very hard in the general case. However, a numerical method proposed for solving general semi-infinite programs (GSIPs) [11] includes design centering as a special case (see [12]).

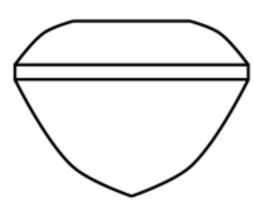

Figure 3. *A typical calibration body.*

In fact, in semi-infinite programming the decision variable of an optimization problem is subject to infinitely many inequality constraints. In the context of design centering, these inequalities stem from the inclusion constraint (in gemstone cutting: each of the infinitely many points of the calibration body has to lie within the rough stone).

In general, as opposed to standard, semi-infinite programming, the index set of these inequality constraints is also allowed to depend on the decision variable. For any semi-infinite program, a fundamental problem is to guarantee feasibility of the decision variable, which obviously involves the verification of infinitely many constraints. This crucial feasibility problem is equivalent to a global optimization problem called the "lower-level problem." In particular, the index set of inequality constraints serves as the feasible set for the lower-level problem. The main numerical challenge is to solve this lower-level problem to global optimality, as merely local solutions will not guarantee feasibility. Unfortunately, in applications of standard semi-infinite programming the lower-level problems are typically nonconvex.

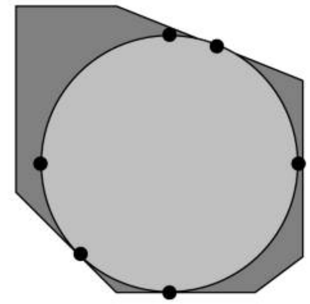

Figure 4. *Maximal disk in a two-dimensional container. The optimal solutions of the lower-level problems are indicated by black dots.*

A straightforward solution approach is to discretize the index set (that is, the feasible set of the lower-level problem), and refine the discretization adaptively. A broad survey of standard semi-infinite programming, such discretization approaches, and other solution methods can be found in the review paper [3].

The discretization approach is hard to implement for general semi-infinite programs, however, as the discretization points would depend on the decision variable. Fortunately, in many applications of general semi-infinite programming, including certain design centering problems, the lower-level problems do turn out to be convex. The algorithm presented in [11] takes advantage of this situation by reformulating the feasibility constraint: The restriction that a global solution of the lower-level problem has to be found is replaced by the (equivalent) first-order optimality conditions of the lower-level problem. To verify feasibility, it is now enough to evaluate the infinite constraint at only one point—namely, the point implicitly defined by the lower-level optimality conditions. Figure 4 shows an optimal design and the solutions of the lower-level problems.

But there's a catch: We have introduced complementarity conditions to the problem, transforming the original semi-infinite program into a mathematical program with complementarity constraints (for background, see [6]). Several numerical solution techniques have been developed since the 1990s for this latter problem class, including a regularization approach, as described in [11]. This approach makes it possible to solve general semi-infinite programs with convex lower-level problems at least to (upper-level) local optimality.

It has been verified [16] that the smoothness and convexity assumptions of the method can be satisfied by appropriate modeling, and initial numerical results for oval calibration bodies have been reported [16, 17]. For a functional description of the rough stone, linear and concave quadratic functions are used. It turns out that the number of these container constraints is, in general, far too large for numerical treatment. To obtain practically solvable problems, we use only a small fraction of the original constraints in the initial semi-infinite model.

Clearly, the relaxation results in a perturbed problem, and a solution of the reduced GSIP cannot be expected to fit into the original container; see Figure 5. The violated original constraints can be identified by means of the set-containment characterization (see [7]), i.e., by solving all lower-level problems. Our GSIP solver, however, allows much faster identification of violated constraints. In fact, the optimal solutions of some original lower-level problems are available at no cost after the optimization step, as they are part of the finite reformulation of the GSIP. By evaluating all original constraints at these "corner points," we can eventually see violations. The drawback of this procedure is that violated constraints can be overlooked, and it thus needs to be combined with the first approach.

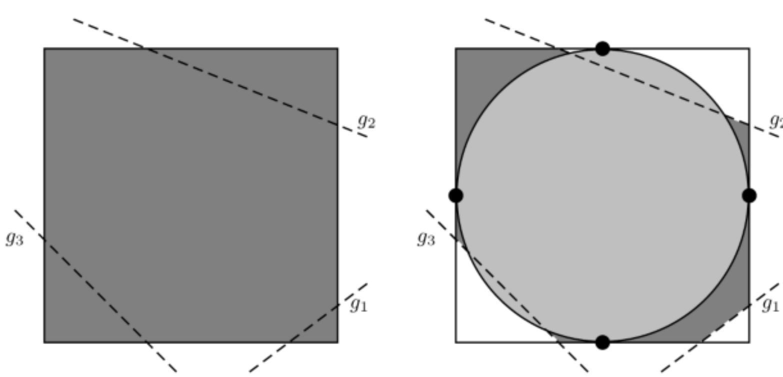

Figure 5. *Dropping some container constraints results in a reduced general semi-infinite program (left), although the solution is not necessarily feasible for the original problem (right).*

If violated constraints are found, they need to be added to the model and the problem needs to be re-solved. We emphasize that the description of the rough stone becomes more accurate only in a few critical regions. Because these regions depend on the values of the decision variables and thus are not known a priori, we refer to the procedure as adaptive refinement. As an example, we apply the refinement procedure to the solution shown in Figure 5. In the first iteration, when g_1, g_2, and g_3 are evaluated for the available solutions of the lower-level problems, the constraint g_2 is found to be violated. Figure 6 shows the optimal solution after the problem has been refined.

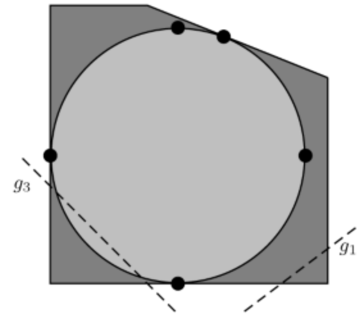

Figure 6. *Optimal solution after the first refinement step.*

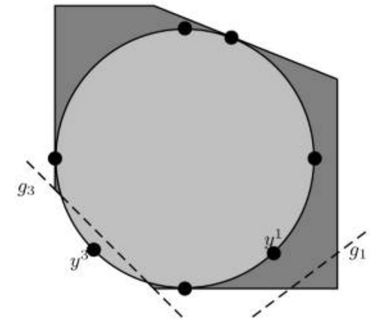

Figure 7. *The violated constraint g_3 is found by solving the corresponding lower-level problem.*

Now, the container constraint g_3 is satisfied in all lower-level solutions in the current model, although it cuts the optimal design. Thus, all lower-level problems not yet in the model have to be solved in order to identify this violated constraint (see Figure 7). Finally, adding g_3 to the model and re-solving the reduced problem lead to the solution of the original problem, as shown in Figure 4. Notice that the constraint g_1 never enters the reduced model.

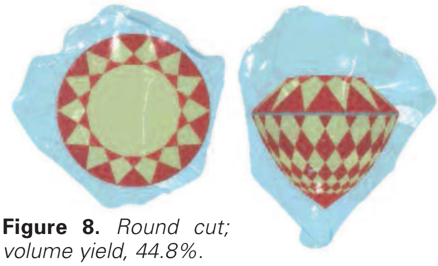

Figure 8. *Round cut; volume yield, 44.8%.*

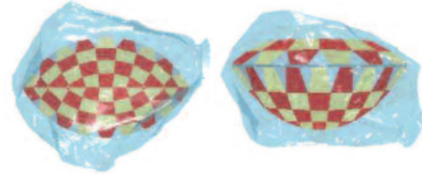

Figure 9. *Navette cut; volume yield, 50.3%.*

Computational Results

We implemented the adaptive refinement method with a GSIP solver in C++ and tested the implementation on a data set consisting of 50 irregularly shaped rough gemstones. Different calibration bodies were used to represent nine different cut shapes. Figures 8–10 show some of the optimal solutions. Interested readers can find information about the test settings, as well as detailed statistics, in [18]. We conclude here with a few comments on the test results:

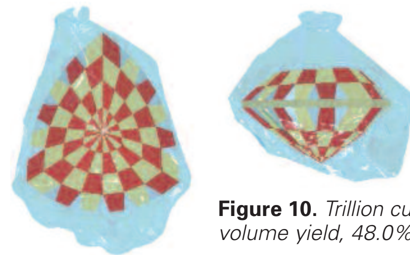

Figure 10. *Trillion cut; volume yield, 48.0%.*

- A typical lapidary cutting problem can be solved in minutes on a standard desktop computer.
- On average, the adaptive refinement requires less than 1% of the original container constraints to find a solution for a typical lapidary cutting GSIP.
- The volume yield—the ratio between the volume of the faceted gem and that of the rough stone—is well above 40% for the test set. This is a substantial improvement over the volume yields achieved by experienced human cutters (33%–36%).

References

[1] K. Fischer, *Edelsteinbearbeitung 1*, 3rd ed., Rühle-Diebener-Verlag, Stuttgart, 1996.
[2] K. Fischer, *Edelsteinbearbeitung 2*, 3rd ed., Rühle-Diebener-Verlag, Stuttgart, 1996.
[3] R. Hettich and K.O. Kortanek, *Semi-infinite programming: Theory, methods, and applications*, SIAM Rev., 35 (1993), 380–429.
[4] OctoNus Software Ltd., http://www.octonus.com.
[5] Sarin Technologies Ltd., http://www.sarin.com.
[6] Z. Luo, J. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, Cambridge, UK, 1996.
[7] O.L. Mangasarian, *Set containment characterization*, J. Global Optim., 24:4 (2002), 473–480.
[8] V.H. Nguyen and J.-J. Strodiot, *Computing a global optimal solution to a design centering problem*, Math. Program., 53 (1992), 111–123.
[9] Gemological Institute of America, http://www.gia.edu.
[10] Russian Gemological Server, http://www.gemology.ru.
[11] O. Stein, *Bi-level Strategies in Semi-infinite Programming*, Kluwer, Boston, 2003.
[12] O. Stein, *A semi-infinite approach to design centering*, in *Optimization with Multi-valued Mappings*, S. Dempe and V. Kalashnikov, eds, Springer, New York, 2006, 209–228.
[13] OGI Systems, http://www.ogisystems.com.
[14] M. Tolkowsky, *Diamond Design: A Study of the Reflection and Refraction of Light in a Diamond*, E. and F.N. Spon, Ltd., London, 1919; Web edition edited by Jasper Paulsen, Seattle, 2001.
[15] A. Viswambharan, *Optimisation in diamond cutting*, Tech. Rep., University of Auckland, 1998.
[16] A. Winterfeld, *Maximizing volumes of lapidaries by use of hierarchical GSIP-models,* Master's Thesis, Technical University of Kaiserslautern, 2004.
[17] A. Winterfeld, *Application of general semi-infinite programming to lapidary cutting problems*, Tech. Rep. 91, ITWM, 2006.
[18] A. Winterfeld, *Large-scale semi-infinite optimization applied to industrial gemstone cutting*, PhD Thesis, Technical University of Kaiserslautern, 2007.

Karl-Heinz Küfer heads the optimization department at the Fraunhofer Institute for Industrial Mathematics (Fraunhofer Institut für Techno- und Wirtschaftsmathematik) in Kaiserslautern, Germany, where Anton Winterfeld is a senior scientist who runs a research group on optimization in virtual engineering. Oliver Stein is a professor in the Faculty of Economics and Business Engineering at the University of Karlsruhe, Germany.