3D Conceptual Design of Sheet Metal Products by Sketching

M. Shpitalni, H. Lipson
Laboratory of Computer Graphics & CAD, Faculty of Mechanical Engineering
Technion -Israel Institute of Technology, Haifa, Israel 32000
shefi@tx.technion.ac.il

Abstract: Design of sheet metal products can be a complex and elaborate process. However, many important aspects of the manufacturing characteristics of a product are already determined at the early stages of the design. This paper reports on a system for conceptual design of sheet metal products by sketching, based on principles of *early incorporation* of CAD, *imprecise* analysis, and *natural, in-context* interaction. Using this approach, various preliminary aspects of a product, such as manufacturability, optimal flat pattern, weight and cost can be estimated automatically based on only a rough freehand sketch of a product, without requiring accurate details. This approach permits designers to explore new concepts more freely and to make a sounder decision when selecting a particular concept for detailed design. The paper describes the concept of the proposed methodology and its underlying algorithms for efficient flat-pattern determination based on fuzzy AI techniques. Examples from a working implementation are shown.

Keywords: sheet metal, flat pattern, conceptual design, sketch, expert system.

1. Introduction

Over the past decade, changes in product development processes and increased consumer demands have dramatically influenced engineering working practices and targets [Bar Cohen, 95]. Development cycles for products have been reduced from several years to several months. Marketing deadlines have become crucial, as most profits are earned within a short period after product release. Among other consequences, the demand for shortened cycles has narrowed the leeway allowed for exploration, as engineers need to converge earlier to a particular design concept.

In a recent paper, we have described possible directions for improving engineering tools to overcome the above difficulties [Kimura *et al*, 98]. This paper briefly reviews the proposed directions and demonstrates a system for conceptual design of sheet metal products based these principles. The paper then describes the algorithms used in the implementation and shows some examples.

2. Directions for future CAD/CAM systems

Three general directions for improvement were identified in [Kimura *et al*, 98] to alleviate difficulties introduced by escalating market demands: (a) bringing computer tools into the engineering process **earlier**, thereby permitting more extensive exploration, earlier detection of problems, and more informed selection of alternatives; (b) improving human-computer interfaces towards ***natural*** and ***in-context*** interaction, where information is presented with natural cues in its final context, thereby making information exchange more fluent and fast; and (c) transforming computers from passive tools to active ***self-initiated*** engineering aids that are capable of *suggesting* alternatives and actively seeking solutions, based on demanding circumstances rather than upon explicit user request. These directions are briefly described as follows:

2.1. Early analysis

In order to permit wide exploration of alternatives, computer aided engineering (CAE) tools must be incorporated *early* into preliminary design. Earlier incorporation of computerized engineering aids, however, is not simple. During the preliminary stages, concepts are vague and defined only roughly; in contrast, most expert software requires accurate and well defined models along with the expertise for interpreting the results. The following capabilities are therefore required:

1. Fast definition of **conceptual models** to convey key aspects of a design.
2. **Conceptual analysis** of rough models to reveal key qualities of a design.

Traditional CAD requires accurate data and explicit dimensions from the start, even if these can be subject to changes later on. The need to handle rough models means that the traditional CAD methodology of *"model and analyze"* is reversed. First, a rough concept is defined without dimensions or accurate information. Next, a rough analysis is performed to get an idea of the implications of that concept. Only then, after further exploration, is the specific rough concept transformed to an accurate model. Thus, a product model is constructed only *after* analysis.

A surprising amount of information can be predicted by analyzing a rough model at a preliminary stage. For example, Figure 1 shows how a rough conceptual design sketch of a sheet metal product has been interpreted and analyzed to convey approximate information about the number of required components, the optimal flat pattern, and other aspects of the product's production cost.

Enabling definition and analysis of rough models is a difficult challenge for two main reasons: *inaccuracy* and *implicitness*. In order to incorporate analysis tools earlier into preliminary exploratory design, conceptual modeling and analysis tools must rid themselves of their dependency on explicit and accurate specifications. While the subject of imprecision in design has been treated extensively, (see [Antonsson, 97, Wallace *et al*, 96] for overviews), the subject of implicitness has not attracted attention.

2.2. Natural Interaction

User interfaces and man-machine interaction have a substantial influence on the efficiency of the design process. When using computer-oriented tools, such as menus, commands or keyboards, the user is forced to convey intentions, describe ideas and obtain information through terms and means supported by computer software and hardware. This requirement calls for a cognitive conversion process whereby abstract information is translated into accurate descriptive language and operations. This additional load is detrimental to communication efficiency and effectiveness; it may impede creative flow and impair the cognitive problem-solving process taking place at that instant.

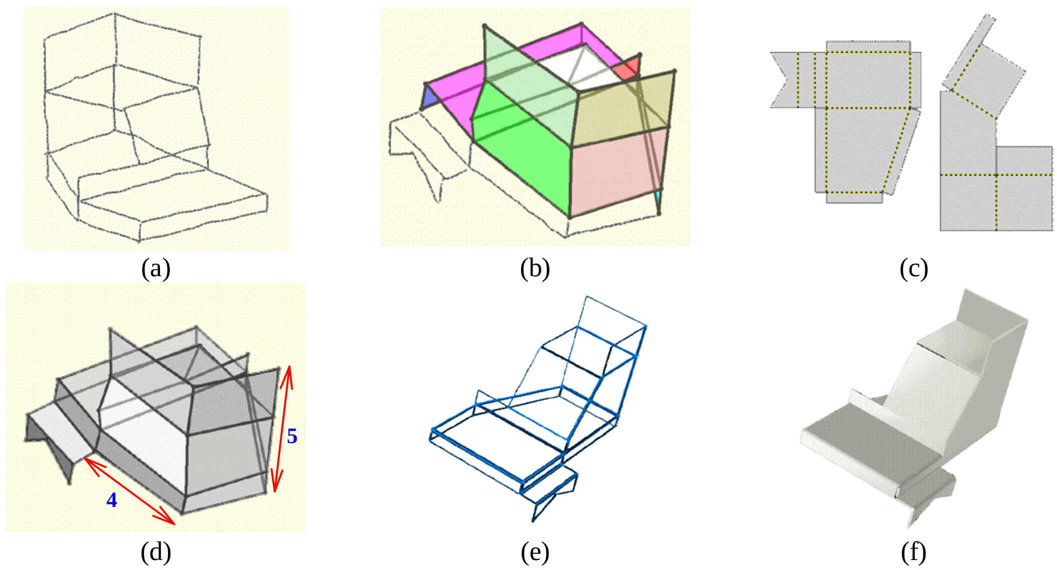


Fig 1. *A rough modeling and analysis sequence of a sheet metal product, (a) rough sketch, (b) 3D reconstruction, (c) flat pattern analysis, (d) refinement by variational geometry, (e) accurate model, (f) simulation of final product.*

Natural interaction involves allowing humans to communicate with the computer using the same means they use to communicate among themselves. Natural interaction, therefore, must address two main issues:

1. The need to convey information using **fast and loose** descriptive language rather than accurate and elaborate specifications
2. The need to use means which are natural and **effortless** for humans, rather than those oriented at computers

2.3. In context interaction

A significant amount of overload is caused by the need to embed or extract information with respect to its context. It is possible, however, to employ direct embedding to eliminate the extraction phase. For example, a technical engineering drawing for quality assurance provides critical dimensions placed in the context of a geometrical shape. If the dimensions were shown directly on the product (using, for example, augmented reality), that part of the information would be redundant. Presenting information in context can thus drastically reduce information overload.

2.4. Self initiative

Computers are usually conceived as passive tools, yielding information upon request. This passiveness constitutes a major drawback, because it depletes one of our most limited resources: *attention*. Traditional direct-manipulation CAD interfaces alone are not sufficient. Future engineering software will not merely respond to user requests but, based on previous activities and built-in knowledge, will anticipate user needs and actively seek ways to support the user, exhibiting *autonomy* and intelligence. Software tools exhibiting these qualities are often termed ‘software agents’ acting on behalf of the user. An example of such a feature relevant to engineering design is automatic seeking of design alternatives, whether these are, in fact, used, or just serve to provoke the designer into thinking in new directions.

3. Implementation

In this paper we report on a system for conceptual design of sheet metal products by sketching, which adheres to and demonstrates the principles mentioned above. The idea behind the system architecture is that a designer will be able to explore designs for sheet metal products. The designer interacts with the system by sketching out the product geometry; qualitative and estimated quantitative aspects of the design are displayed back on the sketch, without need for further interaction or provision of details. The system consists of the following two components:

1. A sketching interface (hardware and software) to interact with the designer, interpret design specifications and display analysis results in context.
2. A sheet metal expert system to analyze the reconstructed model and derive the product's anticipated properties.

These components are described in the following sections.

3.1. Sketching interface and interpreter

The sketching interface is comprised of hardware and software. The hardware interaction device, shown in Figure 2, provides a natural environment for freehand interaction using a virtual drawing table (developed at the Technion) that supports natural pen-and-paper-like interactive sketching. This apparatus supports both input and output on the same real sheet of paper.

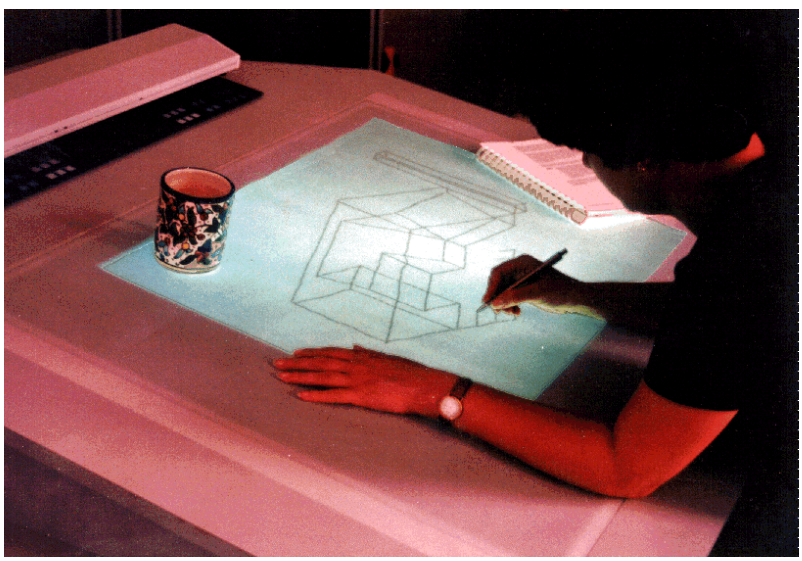


Fig 2. *Virtual pen-and-paper-like sketching environment.*

The sketch strokes themselves are first preprocessed and classified [Shpitalni and Lipson, 97] to create a *flat* edge-vertex graph corresponding to the projection of the product depicted by the drawing. However, in order to analyze the sketch as a product, it is necessary to reconstruct the three-dimensional object depicted by the flat two-dimensional sketch. Although restoration of a three dimensional object from its projection is a mathematically indeterminate step, it can be achieved using image cues. Details of this process are described in [Shpitalni and Lipson, 96] and [Lipson and Shpitalni, 96]. Figure 3 below presents some sample results.

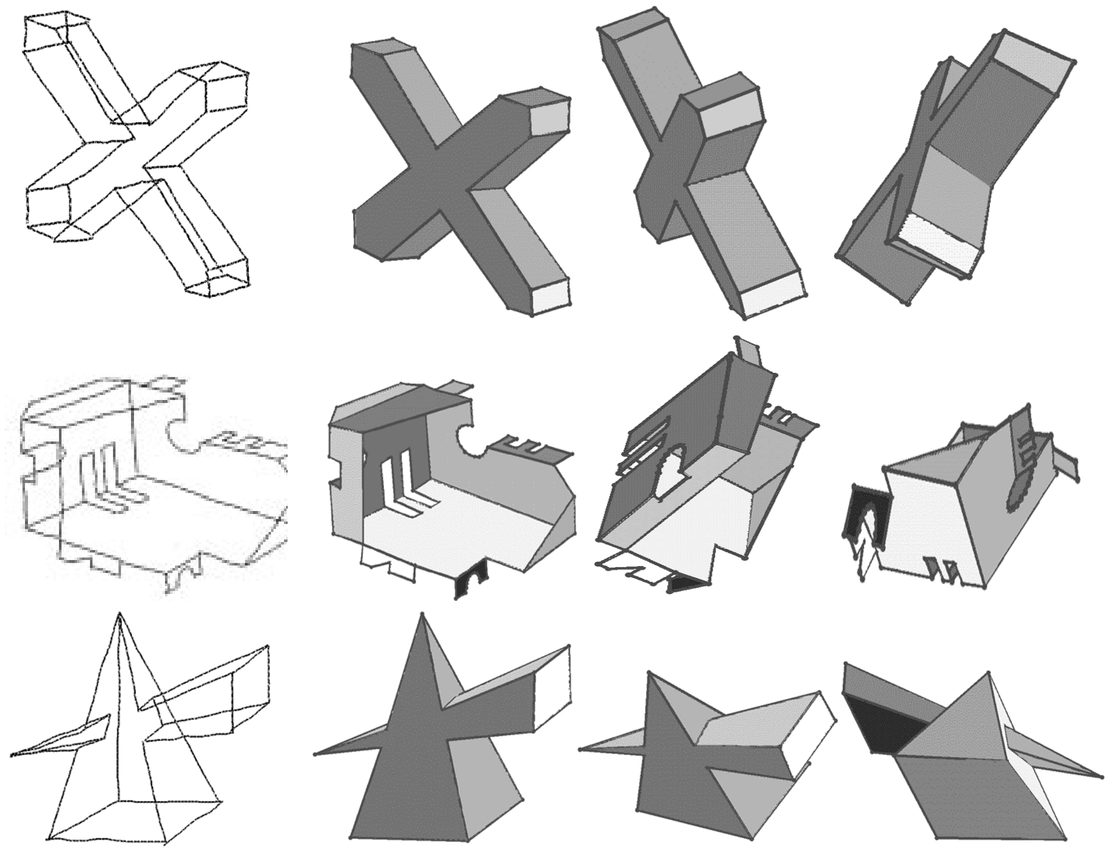

(a) (b)

Fig 3. *(a) product sketches, (b) the reconstructed 3D models.*

3.2. Sheet metal expert system

The system for analyzing the three-dimensional geometry as a product is based on concepts of classic expert systems for sheet metal products such as [DeVin *et al*, 92] and [Shpitalni, 93]. The three dimensional geometry obtained in the previous section is first decomposed into planar *facets* and *links* (between adjacent facets) in order to enable calculating some preliminary aspects of product cost and properties. Given a scale factor, a material and a thickness, it is immediately possible to display a preliminary three dimensional simulation of the product, as well as estimate the following overall properties:

- Number of bending operations
- Total facet area (for painting)
- Total material volume and product weight
- Overall packing volume

The product is then analyzed for manufacturing. This stage determines the optimal unbent flat pattern associated with the product, under various criteria. The flat pattern (or sets of flat patterns) may predict additional information, such as:

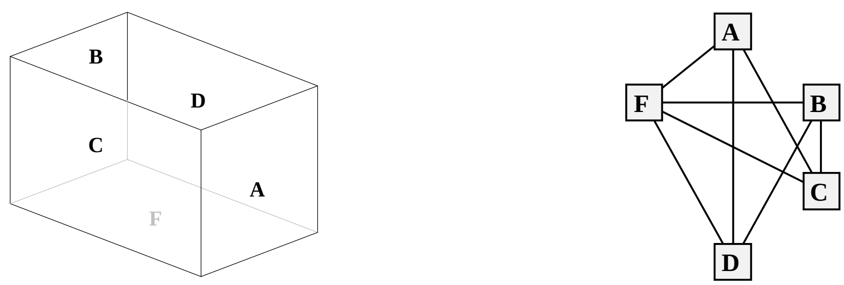
- Number of components
- Estimated flat pattern shape
- Nesting efficiency

- Raw material needed
- Overall manufacturing cost

In the following section, we describe the algorithm used to determine the flat pattern, and subsequently, amendments to this algorithm necessary to handle the inherent inaccuracy of the model.

3.2.1. Determination of the optimal flat pattern

The algorithm for determining the flat pattern is based on a heuristic search on the connectivity graph of the product. The topological connectivity of a sheet metal product can be represented by a graph with nodes corresponding to facets and edges corresponding to connections between facets, or *potential* bends. An example of a product and its corresponding graph is shown in Figure 4 below.



(a) (b)

Fig 4. (a) An open box, and (b) its graph representation

In principle, any flat pattern is a spanning tree of the graph. Depending on the optimality criteria, weights can be assigned to the links corresponding to their desirability as bends. Determination of the flat pattern then is reduced to finding a maximum weight spanning tree. However, in contrast with abstract spanning trees, the flat pattern must comply with certain constraints. The primary constraint is due to collisions: facets in the unbent flat pattern must not overlap, and sheets cannot cross at joints. This restriction introduces the possibility that a compliant spanning tree will not exist at all. It is therefore also necessary to consider the possibility of multiple components (corresponding to a “spanning forest”). When the product contains surface forks, two distinct “sides” cannot be defined uniquely, and therefore some facets may need flipping flat patterns. This makes the problem domain *dynamic*.

The algorithmic implementation of the solution procedure is based on the A* approach. The states in the problem domain correspond to all valid flat patterns. Each state is represented by a binary vector with a digit set to one for actives links. Initially, all links are detached, and the state vector is zero. This corresponds to a flat pattern composed of the individual facets, each as a detached unit, which is always valid. The transition from state to state is done by joining a link while ensuring no collision is introduced, no edges are used more than once and no graph circuit is closed. This ensures that all valid multi-component layouts are considered “on the way” to evaluating single component flat patterns.

In order to define the search goal, a measure of state cost is necessary. In accordance with the A* algorithm, the cost associated with each state is composed of two terms: *g(n)*, corresponding to the actual cost required to reach the current state from the initial state, and *h(n)*, corresponding to an *optimistic* estimation of the remaining cost required to reach a goal state from the current state, where n is the current state. The cost functions are dependent on the overall goal of the unbending procedure. In a simple case, the cost $g(n)$ is a sum of the costs of individual bend lines used so far, plus a cost c_d associated with the number of detached components. Typically, selecting the long bend lines gives better layouts, so cost is assigned in inverse proportion to length. Since it is usually simpler to manufacture a product with fewer components, it is desirable that $c_d \gg c_b$. The remaining cost $h(n)$ can be estimated optimistically by assuming that in the best case, for k facets with m links already selected, the best $k\text{-}m\text{-}1$ of the remaining links will be used. A bound on the number of components can be obtained from topological considerations.

The above formulation yields good solutions, including products requiring multiple components. However, the search is inefficient because it permits multiple states corresponding to permutations of the same flat pattern created by activating the same links in a different order. This situation is overcome by assigning arbitrary indices to the links, and connecting links only in monotonous order. Moreover, a significant improvement in performance is obtained by assigning the indices in order of decreasing costs, so that the preferred links are, in general, 'tried' first. This has the effect of pruning the search tree early in the search.

3.2.2. Incorporating imprecision

Imprecise analysis is usually achieved by multiple executions of a deterministic ('crisp') analysis while varying the input parameters and examining the effect on the results. However, a multiplicity of crisp analyses is not guaranteed to capture the true fuzzy nature of a design. This failure is not a result of statistical error; rather, it is a fundamental problem which can only be overcome if the analysis *itself* is carried out using fuzzy/soft arithmetic. For example, consider the flat pattern illustrated in Figure 5(a) corresponding to the corner configuration shown in Figure 5(b). If the dimensions *a*, *b* and *c* correspond to the *outer* dimensions of the corner, then no matter how the parameters are varied, there will always be a collision along the highlighted line ***b*** in Figure 5(b) when the product is bent. Therefore, an analysis program determining adequacy of this design using a series of *crisp* collision checks will determine that this design is *never* satisfactory because it *always* generates a collision. However, a soft-collision analysis will determine that there is a slight collision, and moreover, that the volume of collision is proportional (and therefore probably related) to the parameter *b*, but is not a function of *a* or *c*. The latter conclusion is more informative and is the kind of result required from a conceptual analysis.

Figure 5. (a) A flat pattern, (b) the corresponding bent product.

The issue of imprecision is addressed in this implementation twice: first, in incorporating soft collision checks, and second, by introducing uncertainty into the cost function.

Instead of using crisp collision checks by checking for boundary intersections, we apply a soft collision check by determining the *amount* of overlap as area of intersection. The computation of area of intersection of two polygons is accomplished by a simple and efficient scan-line algorithm. The important question is how to use this information. The collision area cannot be introduced into the cost function, as it may be merely a result of imprecision and should not influence the choice of solution. It is necessary to use this information to determine whether the detected overlap is accidental (due to imprecision in the specification) or an inherent property of the particular flat pattern being considered. In the first case, the search should proceed without change to the cost. In the second case, the flat pattern should be deemed invalid, and the solution backtracked.

The significance of the intersection can be estimated as follows. Observations show that accidental collisions tend to occupy relatively elongated areas along boundaries, while significant overlaps tend to occupy more uniform shapes in the interior of facets. We estimated the slenderness factor of an arbitrary area using the relationship

$$f = c^2/4\pi A$$

where c denotes the circumference of the shape and A denotes the shape area. It can easily be seen that for a circle, $f = 1$, and for a square $f \sim 1.27$. For a slender rectangle with sides l » d, then $f \sim l/\pi d$ » 1 and $d \sim 2A/c$. Intersections are considered significant if their slenderness factor is less than a certain threshold, if the intersection area occupies a significant proportion of both facet areas, and if the facets involved are closely linked. These thresholds should be incorporated as fuzzy limits; however, a better approach (though more time consuming) would be to ensure that the threshold used is not critical (i.e. perturbations in it do not change the final result).

The cost function being optimized must also be made to consider imprecision by including *optimistic* upper and lower bounds. Note that these bounds denote uncertainty due to the data, not due to the prediction. At any given state of the search, all solutions with lower bounds below the upper bound of the currently optimal state are considered equally viable. Therefore, the search must continue beyond the first solution and output other solutions as well until the first solution which violates this criteria is encountered. This approach can be further extended by having each state include a full probabilistic distribution, rather than just upper and lower bounds.

4. Results

The results of the analysis are conveyed to the user by displaying them back on the sketch. The results contain numeric data, drawings of possible flat patterns, and illustrations of selected bend lines, as well as error estimators for some of the results. Although these results can be expressed in an organized and ‘neat’ output format, it has been attempted to convey the results *in context* of their original specification, so as to make them more easily accessible, according to the principle described in section 2.3. This has been achieved by (a) highlighting bend lines as rough marks overlaid on top of the original sketch using the reverse projection which was applied for reconstruction, and (b) displaying the output flat patterns using rough synthesized sketch strokes, with amplitude corresponding to overlap error, in order to convey the notion that the results are not accurate and to indicate the expected uncertainty. Figure 6 below illustrates an analysis sequence.

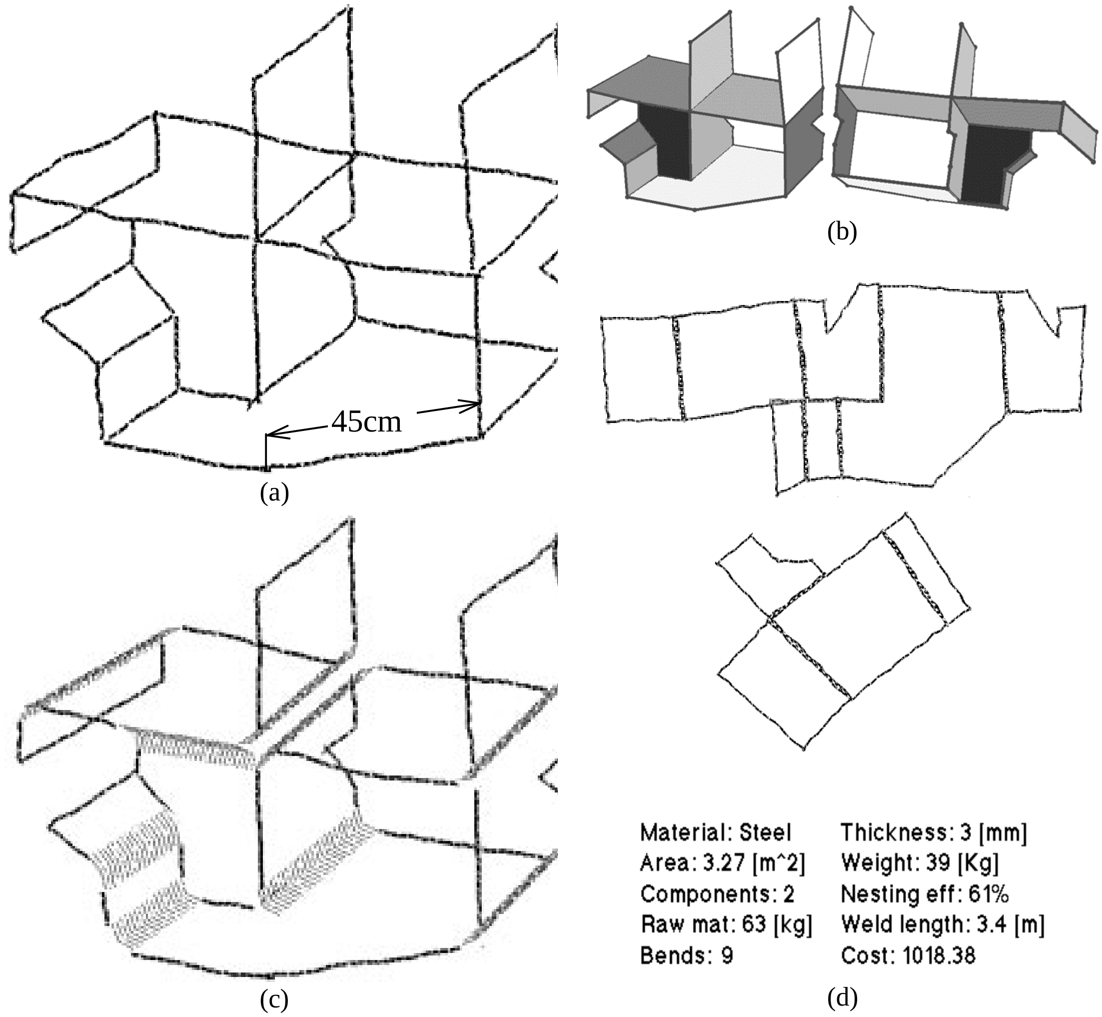


Figure 6. (a) original sketch, (b) reconstructed 3D model, (c) optimal bend assignments overlaid on original sketch, and (d) optimal flat pattern and predicted product properties

5. Conclusions

Design of sheet metal products can be a complex and elaborate process. However, many important aspects of a product's manufacturing characteristics are already determined at the early stages of the design. It is therefore important to allow the designer to try out and investigate many concepts of a product before starting the detailed design. This paper has reported on a system for conceptual design of sheet metal products by sketching, based on principles of *early incorporation* of CAD, *imprecise* analysis, and *natural, in-context* interaction. We have also described an efficient algorithm for finding the optimal flat pattern(s) of a sheet metal product using AI techniques which include modifications to account for imprecision in the design specification. Using this approach, various preliminary aspects of a product, such as manufacturability, optimal flat pattern, weight and cost, can be estimated automatically based on only a rough freehand sketch of a product, without requiring accurate details. This approach permits the designer to explore new concepts more freely and to make sounder decisions when selecting a particular concept for detailed design.

Acknowledgments

This research has been supported in part by the Mitchell Soref Foundation (project No. 6-030-11) and by Fund for the Promotion of Research at the Technion (Research No. 033-028).

References

[Bar Cohen, 95] Bar Cohen, A., 1995, "Mechanical Engineering in the Information Age", Mechanical Engineering, Vol.117 No. 12, pp. 66-70

[Kimura et al, 98] Kimura F., Lipson H., Shpitalni M., 1998, "Engineering Environments in the Information Age - Research Challenges and Opportunities", Submitted to Annals of the CIRP

[Ullman et al, 90] Ullman, D.G., Wood, S., Craig, D., 1990, "The Importance of Drawing in the Mechanical Design Process," Computers & Graphics, Vol. 14 No. 2, pp. 263-274

[Antonsson, 97] Antonsson E. K., 1997, "Imprecision in Engineering Design", Proceedings of IDEA'97, Symposium on Intelligent Design in Engineering Applications, 5th European Congress on Intelligent Techniques and Soft Computing - EUFIT'97, September 97, Aachen, **Germany**

[Wallace et al, 96] Wallace D. R., Jakeila M., Flowers W., "Design search under probabilistic specifications using genetic algorithms", Computer Aided Design, Vol. 28, 1996

[Lipson and Shpitalni, 98] Lipson H., Shpitalni M., "Implicit specification in conceptual design", submitted to Computer Aided Design, 1998

[Shpitalni and Lipson, 96] Shpitalni, M. and Lipson, H., 1996, "Identification of Faces in a 2D Line Drawing Projection of a Wire frame Object", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 10, pp. 1000-1012

[Lipson and Shpitalni, 96] Lipson, H. and Shpitalni, M., 1996, "Optimization Based Reconstruction of a 3D Object From a Single Freehand Line Drawing", Journal of Computer Aided Design, Vol. 28 No. 8, pp. 651-663

[**Shpitalni and Lipson, 97]** Shpitalni, M. and Lipson, H., 1997, "Classification of Sketch Strokes and Corner Detection Using Conic Sections Adaptive Clustering," Trans. of ASME J. of Mechanical Design, Vol. 119, No. 2, March 1997

[**Shpitalni, 93]** Shpitlani M., 1993, "A new concept for design of sheet metal products", *Annals of the CIRP*, Vol. 42/1, pp.123-126

[DeVin et al, 92] DeVin, L.J., de Vries J., Streppel A.H., Kals H.J.J., 1992, "PARTS-S, A CAPP system for small batch manufacturing of sheet metal components", *24th CIRP int. sem. on mfg. sys.*, Copenhagen, June 92, pp. 171-182