Automation of Patterned and Solar Glass Cutting and Stacking Line

Taner İncirci
Altınay Robot Technologies
İstanbul Deri ve Endüstri Serbest Bölgesi
Orjin Cad. 34957 Tuzla Istanbul, Turkey
taner.incirci@altinay.com.tr

Dilek (Bilgin) Tükel
Doğuş University, Control Engineering
Acıbadem,Kadıköy 34722
İstanbul Turkey
dtukel@ dogus.edu.tr

Abstract

In the glass industry, automation is a widely used and well accepted technology in each step of the production. Product modularity and standardization during the design phase allow for easy and fast system integration by combining standard components. Robots work best when the product development and manufacturing processes are designed for robotic application. We define and develop an optimization algorithm based on big bang-big crunch methodology that solves robotized glass cutting problem.

1. Introduction

Glass has been naturally formed when rocks rich in silicates melt at high temperatures and cool before they can form crystalline structures for thousands of years and has been manufactured for human use since 12,000 BCE[1]. Early examples of glass were glass beads, jars, and eating materials first appeared in ancient Egyptian culture. This most versatile and wonderful substances on Earth, used in many applications and in a wide variety of forms, from plain clear glass to tempered and tinted varieties.

Turkey is among the top glass manufacturers of the world. The Turkish glass industry has a share of 8 percent in total global production. Building glass production in 2006 was around 1 million tons, 85 percent of which belongs to flat glass products. In 2009, Şişecam Mersin decided to establish a new production line for patterned and solar glass. Altınay Robot Technologies Inc. which is specialized [2][3] in mechatronic projects, automation technology, robotic handling systems for automotive and glass industry designed and built cold end equipment for the production of patterned and solar glass for Şişecam, Mersin. In this application oriented paper, our automation design concepts for the patterned and solar glass cutting line will be presented.

The designed line from lehr to warehouse consists of several connected sections (Fig. 1):

- Conveying the glass ribbon

Figure 1. Patterned and solar glass cutting line.

- Quality control of the glass ribbon manual or automatic system
- X and Y cutting the glass ribbon according to the production plan into the sheets of optimum size while minimizing the waste using the optimization software: Optimay
- Snapping
- Size and position control using vision
- Rejection of faulty glass sheets
- Sorting using robots (with conveyor tracking ability) according to size and quality
- Papering, dusting or stringing the glass sheets.

Control of the the designed line from lehr to warehouse are achieved using PC, PLC and Motion Controllers. Our control system meets quick reaction and full use of the line capacity requirements at its best. Using a vision system, our robots are capable of taking off glass plates directly from the moving transport conveyor, correcting their alignment.

Optimay is designed for solving cutting optimization problems for glass ribbon lines. This recognizes and facilitates the elimination of defects ensuring minimum wastage and a high glass yield. It is based on .NET technology and runs on 32-bit Windows platforms. Using innovative heuristics and power of multiprocessor architecture of modern PCs, the system can handle complicated optimization problems in noteworthy times. Due to its flexible parametric design, it can be integrated into any glass cutting automation system that can communicate through TCP/IP and/or OPC.

2. Optimization

For each shift, a production plan is made in which the dimensions, qualities and amount of glass plates are listed. When the shift started line supervisors and operators have additional parameters to control the order of the glass plates. All of these parameters are called glass demand parameters and for a specific glass plate demand, the parameter set is called a channel. Thus a channel consists of all the parameters about a specific glass plate order and for more than one glass plate order, which is usually the case, a channel list is used. In addition to channel parameters, there are general parameters such as quality class definitions, optimized area dimension, minimum scrap width and maximum optimization duration. Combining data from general parameters, channel list and defects received from fault detection system as the glass ribbon moves on conveyors, optimization is performed for the purpose of deciding optimum placement of glass plates in the optimization area in a time limited by maximum optimization duration. Optimization area is located in between fault detection system and glass cutting systems towards the beginning of the cold-end line.

2.1. Quality Class Definitions

Both manual control platform and vision system provides feedback about the location and categorization of glass defects. Depending on the classification and quantification of these defects, it is possible to determine whether or not a specific glass plate can belong to a specific quality class.

A table, namely quality class definitions table, which maps defect counts per a specific area to quality classes, is filled in by the line supervisors. Quality class is found as the lowest index quality class satisfying all type of defect counts with less than or equal to relationship. In this paper, for the purpose of simplification, we will represent quality class definition table mapping as a black box function. The function will take as input the glass plate specifications and defects and will give binary output indicating whether or not the specific glass plate satisfies its quality requirement.

2.2. Channel Parameters:

All of the channel parameters have some effect on the production order and amount:

Dimension: Indicates height (measured vertical to the flow of the line), width (measured along the flow of the line) and thickness of the demanded glass plates. During a production shift, height and thickness usually do not change; hence these parameters are the same for all glass plates except for split glass types. For the given height of split glasses, longitudinal cutting head is adjusted accordingly. For this specific application there is only one split cutting head, hence when used, two pieces with different heights can be produced. Even though in the real application split types are included here we will exclude them for the sake of clarity.

Quality class: This parameter indicates the demanded quality of glass plates and must be strictly satisfied. In the decision process of placing glass plates to the optimization area if a specific area in the optimized area has too many defects and demanded quality of the glass plates cannot be satisfied on that area, then scrap glass is produced. This means, defected area will be cut and then broken for recycling. Minimum width of the sheets that can be cut by traversal cutting heads is limited. For the scrap glass plates, this width is called minimum scrap width.

A table, namely quality class definitions table, which maps defect counts per a specific area to quality classes, is filled in by the line supervisors. This table maps defect counts per 10 m² glass surface area to the quality classes. If, more than one quality class satisfies the condition, lowest quality (meaning a better quality) class will be accepted. In short, quality class is found as the lowest index quality class satisfying all type of defect counts with less than or equal to relationship. However, glass plate areas calculated from channel dimension specifications are not necessarily 10 m², and in fact 10 m² is determined so that maximum possible glass plate area possible in the glass line is less than 10 m², hence a similar but two phase mapping algorithm is applied to find the correct quality class for a given glass plate. In this paper, we will not be considering this mapping as it will be regarded as a black box function.

Priority: Glass plate belonging to a channel with higher priority compared to other channels will be laid out continuously on the optimization area unless traded off by other channel parameters.

Percentage: Indicates the demand for the percentage of a specific channel glass plates to the total production. If calculated percentage is less/more than the specified percentage, priority of the channel is increased/decreased accordingly.

Consecutiveness: Specifies the maximum allowed number of consecutive glass plates of a specific channel on the line.

Production amount: Production for the specified channel is ended (channel closed), when the production amount is reached.

3. Optimay optimization algorithm

In this optimization problem, there are many parameters that are traded-off and those parameters ultimately determine the optimum layout. Therefore for most cases, laying out glass plates on the optimized area with minimum scrap configuration is not the optimum solution. The problem is also different from cutting stock and guillotine cutting problems that are widely studied: There is continuous flow of glass ribbon and placement of glass plates on the optimization area is constrained only by channel parameters and defects. In addition height and thickness of the glass ribbon is fixed, therefore the placement is 1D. There is nothing that constrains the placement of the glass plates edge by edge without any scraps in any order but the channel parameters and layout constraints. Among these parameters, only the quality parameter must be strictly satisfied.

Our algorithm for optimization is in three stages:

- Generate all the possible placements of glass types on the optimization area.
- For each configuration/sub-problem run an optimization algorithm to find minimum scrap placement with quality constraint and layout.
- Evaluate each optimized placement with channel parameters, compare and find best placement.

3.1. Generating all possible glass plate configurations:

In order to generate the possible glass plate configurations, depth first search (DFS) algorithm [5] is used. Each node of the search tree represents a glass plate configuration. Root node is a single plate configuration belonging to a specific channel. Branching of DFS is done by adding glass plates from the channel list to the configuration. Nodes go deeper as long as the total width of the configuration does not exceed the optimization area width. To cover all possible plate configurations, a new root node is selected from channel list and DFS is repeated. For faster computation times, each DFS is processed in parallel.

3.2. Optimization of glass plate configurations:

The output of first step is an ordering of glass plates and do not have placement information. This means plates do not have specific coordinates on the optimization area. Optimization goal is defined as finding minimum scrap placement without violating quality requirement.

Here we try to give a mathematical description of the algorithm. Assuming the glass flow from left to right, glass placement is made from right to left. A 2D coordinate system is assigned on the optimization area, x axis from right to left direction, y axis from down to up direction and origin at right down corner of the rectangular optimization area. Position of glass plates are accepted as the right down corner of their 2D rectangular representations.

Initial glass placement on the optimization area is done by placing each plate in the configuration back to back with no scrap spaces in between.

We denote the width and height of the optimization area as W and H. Assuming we have n glass plates in a placement configuration then x positions of the plates are $x_1, x_2, \cdots x_n$ respectively.

Similarly, $(w_1, h_1), (w_2, h_2) \cdots (w_n, h_n)$ the width and height pairs of the plates which are given in the channel list and are regarded constant in the optimization. Then the total width of the scrap, call it W_s, can be found as:

$$W_s = x_1 + [x_2 - (x_1 + w_1)] + [x_3 - (x_2 + w_2)] \cdots [x_n - (x_{n-1} + w_{n-1})] = x_n - w_1 - w_2 \cdots w_{n-1} = \sum_{i=1}^{n} [x_i - (x_{i-1} + w_{i-1})] \quad (1)$$

where $x_0, w_0 = 0$

After cancelling out the terms, (1) becomes:

$$W_s = x_n - \sum_{i=1}^{n-1} w_i \quad (2)$$

Optimization variables are $x_1, x_2, \cdots x_n$ width of the plates are regarded constant. Therefore widths of the plates are dropped out from (2) and objective function for the optimization is:

Minimize:

$$x_n \quad (3)$$

Constraints of this optimization problem are layout constraints and quality constraints. Layout constraints result from the requirements that no overlapping of glass plates are allowed, glass plates must be placed within optimization area and minimum distance between two consecutive plates are, if not zero (side by side), must not be lower than minimum scrap width. That is, for all such $i = 1,2,\cdots,n$

$$x_i \in \{x_{i-1} + w_{i-1}\} \cup [x_{i-1} + w_{i-1} + W_s, W - \textstyle\sum_{k=1}^{n} w_k[\quad (4)$$

where $x_0, w_0 = 0$ and W_{ms} represents minimum scrap width parameter.

Quality constraints result from the quality requirement of each glass plate. For each glass plate in the placement, defects are counted and quality class is looked up from the quality class definitions table. The placement of the glass plates must be such that the required quality class of each glass plate must match the looked up quality. In other words, for all $i = 1,2,\cdots,n$ calculated quality class for i_{th} plate must be equal to the required quality class for i_{th} plate.

Hence, our representation for the problem is complete. We have simple linear objective function but sophisticated, nonlinear constraints. In order to solve the problem, for global converge, among possible heuristic optimization algorithms, we applied Big Bag Big Crunch algorithm [6]. It is shown that this algorithm provides faster global convergence than classical and combat genetic algorithms.

In our application Big Bang-Big Crunch (BB-BC) algorithm is slightly modified for faster convergence for our case. For a glass plate configuration of n glass plates, a point is represented by$x_1, x_2, \cdots x_n$. Therefore each generated point of BB-BC algorithm corresponds to a glass placement. Generated points are first checked for constraints. If constraints are satisfied fitness function is found from (3). One difference from the original algorithm is that, initially instead of a random placement, glass plates are placed with no scrap configuration which we think is close to the optimum point for most scenarios. Center of mass in our application is chosen to be the best placement among possible candidate placements. Number of generated points for Big Bang phase of the algorithm is determined by dividing free space available for each glass plate to a heuristically determined search distance. For i_{th} plate, free space available W_i^f is:

$$W_i^f = W - \sum_{i=1}^{n}[W_k - (x_{i-1} + w_{i-1} + W_{ms})] \quad (5)$$

With each iteration, this number is decreased heuristically together with the standard deviation of points around center of mass. Heuristic parameters are determined by trial and error and are customized for our specific application but can be tuned for other applications with different channel and general parameter values.

As a result, in this stage, glass plate configurations are turned into optimum glass placements based on layout and quality requirements.

3.3. Evaluation of the optimum placements:

Final evaluation of placements involves channel parameters priority and percentage. Each optimum placement is evaluated with an evaluation function. For an optimum placement of n glass plates, following function is used for evaluation:

$$E = \sum_{i=1}^{n-1} p_i w_i h_i \quad (6)$$

where p_i represents priority for the i_{th} plate and w, h are the width and height. Initially $p$is the same as priority parameter but it can be automatically changed depending on the percentage parameter requirement. If i_{th} plate production percentage is less than the required percentage, p is increased, if it is more then p is decreased.

Finally, by comparing the evaluation results (6) of the optimum placements, best placement is chosen.

3.4. Optimay Run Results:

On a quad-core PC, many test runs are performed for an optimization area of 6000mm width and 2600mm height with different channel parameters and randomly generated defects:

- Glass widths ranged from 800mm to 3000mm.
- Different channel parameters are tested.
- Since there are 4 stacking stations, number of active channels ranged from 1 to 4.
- Defects per 10 m^2 ranged from 1 to 100.
- Targeted maximum optimization duration was 3 seconds.

Test results indicated optimization algorithm gave satisfactory results. For most of the cases, optimization duration was under 1 second.

4. Conclusions

A fully automatic patterned and solar glass line from lehr to warehouse is designed and implemented. To achieve high performance and maintainable scheduling for such a complicated production line, great care has to be taken to ensure the quality, modularity and standardization of the products essential to system integration. Optimization in this regard is of extreme importance for glass manufacturers, as it directly affects production yield and quality. The main contribution of this paper is to introduce many parameters that provide control over order and quality of the production for the line supervisors and how those parameters come to affect the outcome of the optimization. In addition, algorithm of an already applied and working solution for finding the best glass placement is explained. In order to avoid complicated nature of the problem due to many parameters, a three stage algorithm is suggested which reduced the optimization problem as a whole to a collection of sub-problems. On second stage, sub optimization problem is modeled and as a solution, adaptation of Big Bang-Big Crunch algorithm to our problem is explained. Hence starting from written text specifications, optimization problem is modeled, solved and implemented in software, Optimay, with successful results.

References

[1] JF. Stirling, "The Manufacture of Heat Resisting Glassware," *IET, Production Engineer*, 41 Issue:6, 1962, pp. 300-323.

[2] DB. Tukel, H. Altinay, C. Bayar, G. Gök, G. Hurmalı, A. Şimşek, "Robotized Palletizing of Wine Glasses", 2nd *International Conference on recent advances in mechatronics ICRAM*, 1999, pp. 175-180 .

[3] DB. Tukel, Ö. Eminoğlu, AK. Ayar, "The rear axle production line for Clio and new modus, " *37th international symposium on Robotics*, 2006.

[4] The Fletcher-Terry Company, "The Principles of Glass Cutting, " *Technical Guide*, 2010, pp. 6-9.

[5] Th. Cormen, C. Leiserson, RL Rivest, and C. Stein, C, "Introduction to Algorithms," *MIT Press and McGraw-Hill, Second Edition*, 2001, Section 22.3: Depth-first search, pp.. 540–549.

[6] OK. Erol, I. Eksin, "A new optimization method: Big Bang–Big Crunch," *Elsevier, Advances in Engineering Software*, 2006, 37 (2) pp. 106–111.