Proceedings of DETC'01
ASME 2001 Design Engineering Technical Conferences
Design for Manufacturing
Pittsburgh, Pennsylvania, September 9-12, 2001

DETC2001/DFM-21187

DESIGN OF AN EVALUATIVE TOOL FOR A MULTI-STATION INJECTION MOLDING SYSTEM USING AN EVOLUTIONARY ALGORITHM

A. R. Wolf, L. H. Shu *, R. D. Venter
Department of Mechanical and Industrial Engineering
University of Toronto
5 King's College Road
Toronto, Ontario, M5S 3G8
* T: 416-946-3028, F: 416-978-7753, E: shu@mie.utoronto.ca

ABSTRACT

A tool to facilitate the feasibility study of a newly proposed multi-station injection molding system is developed. The conceptual design and proposed embodiment of the new system are geared toward the development of a system flexible enough to handle multiple part types and production volumes. A comprehensive design model is used to structure the problem by identifying the desired design objectives and the effect the system variables have on the final design. An Evolutionary Algorithm optimization is used to find the combination of system variables that yields optimal system outputs. The algorithm uses a number of components customized to suit the design requirements of the proposed system. This optimization and evaluation process provides a basis by which the new system can be compared with traditional injection molding practices. Results confirm that the new multi-station system is less affected by the degree of product variety than traditional molding machines.

Keywords: Injection Molding, Lean Manufacturing, Design Optimization, Evolutionary Algorithms.

INTRODUCTION

The goal of this research is to design a tool to evaluate the feasibility of a newly proposed multi-station injection molding system. The conceptual development of the system is first presented including a description of the proposed embodiment of the design. Next, a comprehensive design model is developed. The inherent flexibility of the proposed system results in a large number of variables that affects the performance of the design. The design model identifies the desired design objectives and the effect the system variables have on the design. The design function is then used to optimize the design according to the desired objectives. This optimization and evaluation process provides a basis by which the new system can be compared with traditional injection molding practices.

The manufacturing sector has always been under pressure to produce parts quicker, at a lower cost in order to remain financially competitive. These demands have been further extended in recent years, by automotive Original Equipment Manufacturers (O.E.M.s) in particular, to include the ability to handle increased product variety and smaller batch sizes. In addition, injection molders are required to be able to handle frequent product changes without incurring large additional costs. To address these new demands, Just-In-Time and Flexible Production Systems have been cited as methods by which custom injection molders can remain competitive (Lankton, 1985, Packman, I.D., 1986). More recently, industry reports indicate that the production demands are still beyond the capabilities of many molders. Again, efforts focused on the production facility are suggested to help alleviate the problem (Grande, 1995, Offergeld, 1998).

The approach introduced in this paper focuses on the injection molding apparatus as opposed to the production facility. The injection molding process is reviewed in an attempt to identify inefficiencies as well as obstacles to increasing the variety of part types while still minimizing costs. The project scope encompasses the design and optimization of a newly proposed injection molding system as an alternative to traditional machines. However, to allow for a comparison of the two systems, an evaluative tool is first developed, facilitating a cost-based feasibility comparison of the traditional and proposed injection molding machines.

The following sections outline the steps taken in the development of an evaluative tool for the multi-station injection molding system. The system description, design model formulation, as well as the details of the Evolutionary Algorithm are described below. First, however, some background information and motivation for the research are discussed, followed by a review of related work.

BACKGROUND AND MOTIVATION

The design and evaluation of the proposed system required an established set of design objectives. These objectives were used to guide the conceptual development of the design, and later acted as measures by which the system's performance could be assessed. To establish these objectives, elements of 'Lean Production' were used. Lean Production can be defined as a manufacturing and managerial philosophy that focuses on reducing all forms of waste in a system to allow a company to handle increased levels of product mix and smaller, more frequent production runs. Of importance to the design model are concepts such as Takt Time, and Single-Piece Flow (Cochran, 1999).

(1) TAKT (TARGET) TIME

Lean manufacturing stresses the importance of building no more and no less than that demanded by the customer to avoid waste. To achieve this goal the production rate must match exactly the Takt Time or customer demand rate. Building too quickly creates finished goods inventory, while building too slowly generates the need for overtime, excess transportation, etc. The Takt Time becomes the driver of the production rates, and in turn, the material flow rates throughout the system. To measure a system's ability to produce parts at the Takt Time, a measure called Build-to-Schedule (B.T.S.) is used. Build-to-Schedule is defined below.

$$\begin{aligned}
&\textit{Given}: \\
&\quad x_i = \#\ \text{part type}\ i\ \text{produced} \\
&\quad y_i = \#\ \text{part type}\ i\ \text{demanded} \\
&\text{B.T.S.}_i = \begin{cases} \textit{if}\quad x_i \langle y_i \quad \textit{then} & \dfrac{x_i}{y_i} \\[2ex] \textit{if}\quad x_i \rangle y_i \quad \textit{then} & \dfrac{2y_i - x_i}{y_i} \end{cases} \\
&\text{B.T.S.}_{\text{system}} = \prod_{i=1}^{p} \text{B.T.S}_i
\end{aligned}$$

Definition 1 - Build-to-Schedule

B.T.S. provides a metric to determine how closely the production system matches the demanded rate. To ensure a useful measure is attained, the B.T.S. should be calculated over a maximum period of a day.

(2) SINGLE-PIECE FLOW

Once the Takt Time is established, the system must be designed to meet these production requirements. The constraint operation or the operation determining the cycle time must be capable of running at the shortest required Takt Time.

Producing multiple types of parts introduces an additional consideration in terms of the production run or batch size. Typically, parts are produced in large batches to avoid machine changeover times. This practice, however, creates swollen inventories, quality control problems, and delayed lead times and runs counter to Lean Production principles.

Single-Piece Flow is introduced to reduce the batch size to the lower limit of one piece per batch. Batches that run through multiple operations experience lot delay, as each part must wait until the batch is finished before moving to the next operation. Single-Piece Flow on the other hand, moves individual pieces through each station separately, allowing separate operations to run in parallel. Figure 1 shows Batch Production (top) compared to Single-Piece Flow (bottom) through multiple operations. The total production time for Single-Piece Flow is lessened due to the reduction of lot delay.

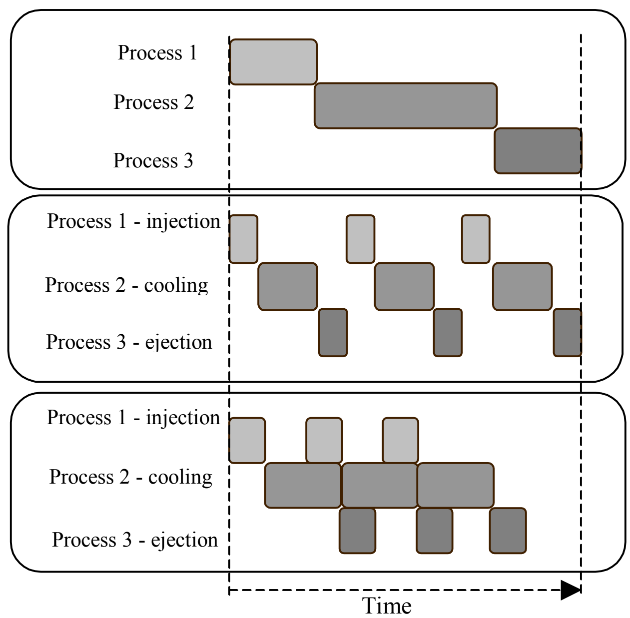

Figure 1 – Batch Production (top), Traditional injection molding using serial production (middle) versus Single-Piece Flow (bottom) through multiple operations

Current injection molding machines perform the injection, cooling, and ejection operations serially, as shown in Figure 1 (middle). This causes the system to experience lot delay equivalent to running the entire batch serially through each operation. In Figure 1, note that the time taken for the parts to run through all operations is the same for both batch (top) and serial production (middle), which is longer than the case with Single-Piece Flow (bottom).

With Single-Piece Flow in place, products are pulled through the system, with each downstream operation pulling parts from the upstream operation. In this way, all processes will produce at the rate of the constraint operation, which is set to match the Takt Time.

RELATED RESEARCH

A number of modifications to the injection molding process have been suggested to improve the performance and flexibility of current systems. These modifications can be divided into two main groups. The first deals with the optimal setting of the process variables of the machine and focuses on improving quality and throughput. The second deals with machine modifications and focuses on improving flexibility.

A number of research works based on various approaches have been performed in the domain of variable setting for injection molding. These approaches include expert systems (Bernhardt and Kassa, 1995), mathematical modeling, numerical simulations, Artificial Neural Networks, Case Based Reasoning and Genetic or Evolutionary Algorithms (Mok *et al.*, 1999). Non-exhaustive approaches are needed because typically over a dozen process variables are involved. The traditional method by which variables are set is trial-and-error, which is not conducive to a process requiring a high degree of flexibility.

Changes to the injection molding apparatus to improve flexibility typically include the addition of a quick-mold-change system to reduce the changeover time of a system (Rozema and Travaglini, 1995). The earliest form of quick-mold-change system, the S.M.E.D. (single-minute exchange dies) developed by Toyota Motor Corporation, involved a reduction in setup times through an improved changeover process (Szatkowski and Reasor, 1991). The S.M.E.D system focused on removing or modifying the manual die-changing process. More recently, a number of other approaches have been developed including magnetic-mold mount systems, floor-based systems that use an air table and robot-assisted mold change (Wilder, 1990). While some use these systems to augment a traditional injection-molding machine, others suggest using this technology as the basis for a flexible manufacturing system (F.M.S.) or cell (Henze, 1988). F.M.S. makes use of mold changers and industrial robots to achieve quick changeover times and allow for smaller production runs (Von Eysmondt, 1989). Some molders have achieved a level of flexibility that allows production run sizes as small as 6 pieces (Schut, 1999).

In addition to quick-mold-change systems, other manufacturers have developed a carousel, or horizontal mold rotation, injection-molding system. While these systems were originally developed for thick-part injection molding, the benefits of separate injection, cooling and ejection stages have become apparent with regard to flexibility. The use of a multi-stage rotary system requires lower clamping forces and injection pressures, reducing operating costs and tooling requirements (Neilley, 1997). Due to the multi-station nature of these systems, they are ideal for co-injection molding where different cores can be replaced during processing of a single part (Jaroschek and Steger, 1998). Despite the increased flexibility, these systems limit the number of molds that can move through the system. Furthermore, the movement of the molds is constrained by the rotation of the mold carousel. Lack of independent movement of the molds eliminates the potential of a mold exchange station where molds can be exchanged to accommodate mold maintenance and/or changes to the production requirements.

While considerable research has been conducted regarding the setting of process variables for injection molding, a more limited effort seems to have been applied to the redesign of the injection molding system itself. The focus continues to be increasing output, instead of streamlining the injection molding process by eliminating waste and improving the handling of multiple part types.

SYSTEM DESCRIPTION

Scope

The design of an injection molding system that could provide significantly improved flexibility began with a high-level functional decomposition of the injection molding process into the five key functional requirements shown in Figure 2.

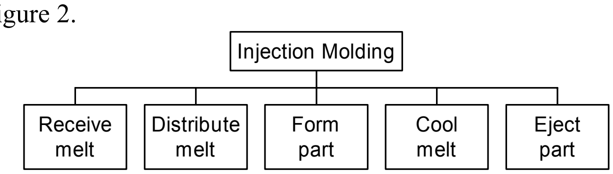

Figure 2 – Functional decomposition of the injection molding process

The physical entities found on a traditional injection molding machine that are used to satisfy these requirements are shown in the rightmost vector in Figure 3 below. In addition, a design matrix is included to illustrate the relationship between the functional requirements and the physical design features of the system. In the matrix, a strong relationship is indicated by the symbol ‘x’, while a weak, or no relationship is indicated by the symbol ‘o’. For example, the ‘x’ in the upper left corner of the design matrix indicates that the sprue and nozzle affects the receiving of the melt.

$$\begin{bmatrix} \text{Receive Melt} \\ \text{Distribute Melt} \\ \text{Form Part} \\ \text{Cool Melt} \\ \text{Eject Part} \end{bmatrix} = \begin{bmatrix} x & o & o & o & o \\ x & x & o & x & o \\ o & x & x & o & o \\ o & o & x & x & o \\ o & o & x & x & x \end{bmatrix} \begin{bmatrix} \text{Sprue and Nozzle} \\ \text{Runners and Gates} \\ \text{Cavity and Core} \\ \text{Cooling Channels} \\ \text{Ejector Pins} \end{bmatrix}$$

Figure 3 – Design of a traditional injection molding system

The design matrix reveals four problematic relationships that create a coupled design, which often implies inflexibility and difficulty in controlling machine operation. These relationships are highlighted and numbered in Figure 3. The first relationship involves the interaction of the cooling channels with the distribution of the melt. A cool mold can affect the distribution of the melt through the runners and gates and into the mold itself. This problem can create short shots due to melt freezing at the gates or thin-walled sections. Freezing is often addressed by increasing the injection pressure, which reduces its occurrence and reduces the processing time. The processing costs, however, are increased and the high pressure causes accelerated mold wear.

While the first problem created a direct coupling of the design, the remaining three relationships create a coupled system only if the functional requirements have to be satisfied in parallel. Since traditional injection molding systems do not operate these steps in parallel, this is not typically considered a problem. However, guided by the principles of Lean Production, this reliance on sequential or serial operations (temporal coupling) is itself a problem due to the creation of lot delay. The three remaining relationships highlight couplings in the system between the design features of the mold and the injection-molding machine. If eliminated, some functional requirements execute in parallel, reducing lot delay.

The second problem identified in Figure 3 describes the relationship between the cooling of the melt and the core and cavity. Here the core and cavity, which are designed for part formation, can affect the cooling of the melt. The problem arises because additional parts cannot be formed while the melt is cooling. The operations are coupled due to the temporal reliance of two functional requirements on a single design feature. The third and fourth relationships also involve a temporal coupling of components. In this case, parts can only be ejected from a mold once the part has been formed and the melt has cooled. The problem caused by the third and fourth relationships arises because the core and cavity, cooling channels, and ejector pins are all physically coupled. Apart from the temporal coupling exhibited by the fourth relationship, the core and cavity affects part ejection due to part shrinkage onto the core. This problem, however, is reduced through proper mold design by incorporating adequate drafts and the proper treatment of undercuts.

The four coupled relationships identified provided the impetus and direction for the redesign of the injection molding system. Figure 4 shows the six functional requirements of the new system along with a new set of design features. The corresponding design matrix is included to indicate the relationships that now exist.

$$\begin{bmatrix}\text{Heat Mold}\\ \text{Receive Melt}\\ \text{Distribute Melt}\\ \text{Form Part}\\ \text{Cool Melt}\\ \text{Eject Part}\end{bmatrix} = \begin{bmatrix} x & o & o & o & o & o\\ o & x & o & o & o & o\\ o & x & x & o & o & o\\ o & o & x & x & o & o\\ o & o & o & o & x & o\\ o & o & o & o & o & x\end{bmatrix} \left[\begin{array}{l} \left.\text{Heating Station}\right\}\text{Heating}\\ \left.\begin{array}{l}\text{Sprue and Nozzle}\\ \text{Runners and Gates}\\ \text{Cavity and Core}\end{array}\right\}\text{Injection}\\ \left.\text{Cooling Station}\right\}\text{Cooling}\\ \left.\text{Ejection Station}\right\}\text{Ejection}\end{array}\right]$$

Figure 4 – Design of the newly proposed injection molding system

Each of the abovementioned problems was eliminated using the new set of design features. The first problem, relating to short shots due to the solidification of melt during its distribution, was overcome by introducing a new design feature to heat the mold prior to injection. A heated mold eliminates the risk of frozen gates, and allows thin-walled parts (<1mm) to be manufactured without the need for special polymers and high injection pressures. By grouping the second, third and fourth functional requirements together to form a single unit, the final design can be broken into four uncoupled units, a heating unit, an injection unit, a cooling unit, and an ejector unit. The components within the injection unit still exhibit a temporal coupling but because these functional requirements are to be satisfied serially by one physical unit, this relationship does not pose a problem. The second, third and fourth problems identified were solved by using multiple molds and stations, i.e. the cooling and ejection units, to allow the respective functional requirements to be temporally uncoupled. With multiple molds moving through a series of functionally and physically uncoupled operations, the system is able to perform operations in parallel. Furthermore, additional stations can be added to help balance the operating times to minimize idle times. Using multiple molds means that the number of cavities of each mold can be smaller, which leads to lower injection pressures and clamping forces and thus machines requiring less power.

The uncoupled nature of the four stations: heating, injection, cooling, and ejection, of the new injection molding system is consistent with Axiomatic Design (Suh, 1990). The proposed embodiment of the new system is described next.

Proposed Embodiment

The proposed embodiment of this system includes separate stations for each of the four units, connected by a mold conveyor system. The stations are arranged in a manner that allows a mold to be transferred through the system, sequentially visiting each of the stations until their combined series of processes form a completed part. The embodiment allows for the movement of multiple molds through the system, producing single- or multiple-part types. A mold will only move to the next station if an opening arises, that is, the molds are pulled through the system. Because the stations operate in parallel, the additional heating station could be added to the system with little additional processing time incurred.

Figure 5 shows the proposed embodiment of the design. Two mold types are shown, reflecting the system's ability to concurrently handle the varying part types. The number of molds per part and the number of cavities for each of those molds can vary, as can the capacity of each of the stations.

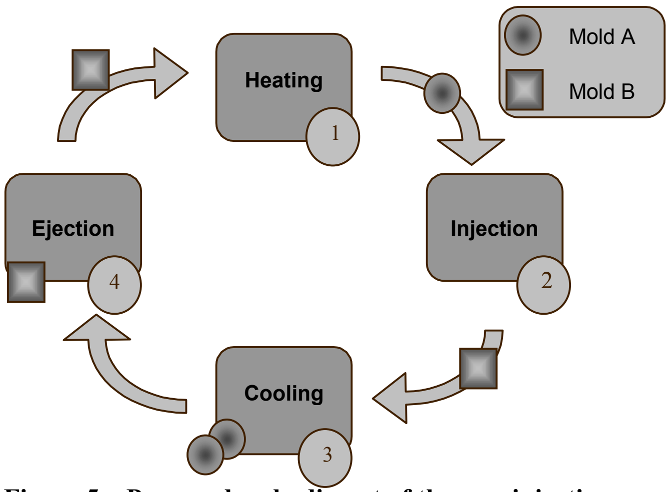

Figure 5 – Proposed embodiment of the new injection molding system

DESIGN MODELING

Design modeling provides a structure to the solution of the design problem at hand. This problem can be described as follows. Given the production volume and part types for a set of parts to be manufactured, find the best system configuration that will satisfy the production requirements and minimize costs. By varying the production volumes, part geometries, and number of part types, optimal system configurations can be found for the different scenarios. These configurations can then be used in cost analyses of the new machine versus the traditional equipment.

The design modeling process consists of five elements: design outputs, design objectives, design inputs, design range and the design function. Figure 6 shows the relationship between the design modeling elements.

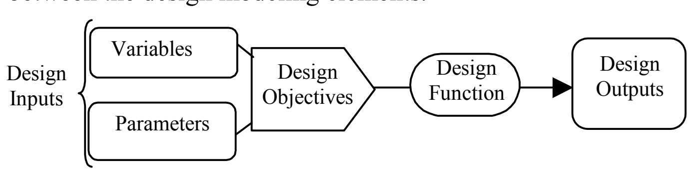

Figure 6 – Schematic of the Design Model

Design Outputs

Design outputs are the main features of the design that must be monitored or evaluated to ensure the design fulfills its requirements. The design outputs are:

1. Build to Schedule (B.T.S.) – a measure of the ability of the system to build the correct type, quantity, mix, and order of parts.
2. Total Cost – a measure of the system level cost.

The first criterion, Build-to-Schedule (B.T.S.), is aimed at achieving Single-Piece Flow. The second criterion, Total Cost, aims to attain the aforementioned flexibility at the lowest possible cost.

Design Objectives

The design objective adds to each design output the designer's intent regarding the design. Design objectives can be grouped into one of two categories, *wish* objectives and *must* objectives. *Wish* objectives describe outputs which are considered smaller-the-better, larger-the-better, or on-target-the-better. *Must* objectives have a defined range of permissible values that will ensure the design requirements are met. The design objectives for this model are:

Wish Objective: Cost - smaller-the better

Must Objective: Build-to-schedule must equal 100% (B.T.S. = 100%)

Design Inputs

The design inputs are divided into two types, design parameters and design variables. Parameters are fixed during the solution of a particular design case. However, they can change from one design case to another, especially when a range of problems is examined. The design variables, however, are changed during the design process in order to achieve the desired output.

Design variables and parameters are represented using the following vector notation:

Design Variable Vector $\mathbf{X} = \{X_m\}^T \quad \{m \mid m \in I, [1, \#\text{ variables}]\}$

Design Parameter Vector $\mathbf{P} = \{P_n\}^T \quad \{n \mid n \in I, [1, \#\text{ parameters}]\}$

Design Input Vector $\mathbf{Y} = \{\mathbf{X}, \mathbf{P}\}$

where $[\ ]^T$ signifies a matrix transform

The parameters and variables used in our model are listed in Appendix I.

Design Range

This defines the range of possible values that could be taken on by each of the variables or parameters. With regard to parameters, the range typically refers to design alternatives, with the allowable values being based on existing materials and standard part sizes, etc. The variable range involves realistic limitations of the design variables that limit the design variable space. The range for the parameters and variables used in our model are listed in Appendix II.

Design Function

The design function is a mapping between the design inputs and the design outputs, incorporating the designer's intent captured by the design objectives. In this case, two design objectives exist, the must objective of 100% Build-to-schedule, and the wish objective to minimize the total cost. To achieve the first objective, the design function must ensure that the number of parts built during a given time period matches the customer demand rates for the various part types.

Because the number of cycles of molds through the system could vary, the design function requires that the ratio of parts built during each cycle matches the customer demand rate. Therefore, the frequency, or number of cycles through the system could be changed to ensure the correct quantity is produced. Table 1 shows the various acceptable cycle quantities with the respective frequency that yields the demanded part quantities (given customer demand for each part).

Customer Demand	Frequency of 1	Frequency of 3	Frequency of 6
Part A – 30 units	30	10	5
Part B – 18 units	18	6	3
Part C – 6 units	6	2	1

Table 1 – Number of parts per cycle based on frequency and customer demand

Within each of the frequency-ratio couplets, there exist a number of possible mold and cavity combinations to achieve the cycle quantity. For example, to produce 18 units of B with a frequency of 6 (last column of Table 1), the system must produce 3 parts (third row) of Type B in each cycle. This can be accomplished using any of the following mold combinations,

- 1 mold – 3 cavities
- 2 molds – 1 cavity, 2 cavities
- 3 molds – 1 cavity, 1 cavity, 1 cavity

N.B. The sum of cavities across all molds equals three.

Satisfying the must design objective of 100% Build-to-schedule translates to a constraint on the design where the possible mold cavity and frequency combinations are limited. The second design objective, minimizing total cost, forms the objective function of the system optimization. The formulation of the optimization problem using all the elements of the design modeling is shown in Figure A, Appendix III.

Due to the large number of design inputs and complex nature of the variable and parameter ranges, i.e., the fact that many can take on integer or fixed values, the optimization becomes difficult to solve using a closed form solution. The combinatorial nature of the problem introduced by the need for the production rate to meet the customer demand rate prompted the use of an Evolutionary Algorithm (E.A.). This algorithm provides the ability to search through a complex design space quickly and efficiently. Although the global optimal solution is not guaranteed, it was felt that the algorithm would be suitable for the feasibility analysis.

The following sections describe the E.A. used, including details of the objective, or cost function, and the development of new E.A. components (genome and operators) designed to generate mold, cavity, and station capacity values that fit the optimization constraints.

Algorithm Structure

The algorithm begins with the production volumes for the various part types as inputs. Next, the possible frequencies that the system may run are calculated based on the production volumes. The greatest common divisor is found for the production values and a list of possible frequencies is generated. The E.A. runs a separate time for each of the frequencies, finding the close-to-optimal solution. The resulting solutions for the different frequencies are then compared and the solution with the lowest cost is selected as the final solution.

The evolutionary algorithm operates as follows:

1. Create genome population
2. Test for fitness
3. Select genomes that are most fit
4. Apply evolutionary operators
5. Generate a new population
6. Go back to step 2 (repeat for n generations)

The algorithm begins by randomly generating a population of feasible genomes. The genomes are then tested for fitness using the cost objective function. The fittest genomes are selected using a specified selection scheme; these genomes form the basis for new members of the population. Evolutionary operators are then used to modify the genomes in the population, creating new genomes to complete the population. This process repeats for a specified number of generations, generating new genomes and giving the highest chance of survival to the fittest genomes of each population. The fittest genomes are thereby propagated to the next generation. Each generation, the maximum, minimum, and mean scores for the population are tracked, providing a method by which the success of the evolution can be monitored.

The algorithm was developed in the C++ programming language, using Genetic Algorithm Library 2.4.5 (Wall, 1996).

The Genome

A genome consists of a codified string of elements that characterize the input set. The genome used captures all the variable information of the system, i.e. $\mathbf{X} = \{X_m\}^T \{m| m \in I, [1,\# \text{ variables}]\}$. Because information regarding both the mold and station/buffer information must be captured, a new customized composite genome was developed. This genome consists of two parts. The first part is dynamic in size and covers mold information, while the second part has a fixed size and covers the station/buffer information.

The first part of the genome captures information regarding the number of molds of each of the part types, and the cavities per mold. This is achieved by storing the information for a mold in one genome element. The part type is stored as a letter, and the number of cavities for the mold is stored as a number within an element. The number of elements reflects the number of molds. Figure 7 shows an example of the first part of the genome, describing 6 molds for 3 part types (A, B, C) with the respective number of cavities per mold shown in each element.

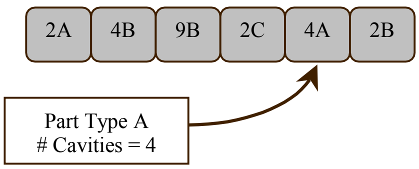

Figure 7 – First part of the composite genome

The second part of the genome captures the station capacities of each of the eight buffers and stations. Each buffer/station is represented by an element containing a letter to represent the buffer/station and a number to represent the buffer/station capacity. The buffers/stations and their respective letters are shown in Table 2 below.

Heating Station Buffer	Heating Station	Injection Station Buffer	Injection Station	Cooling Station Buffer	Cooling Station	Ejection Station Buffer	Ejection Station
S	T	U	V	W	X	Y	Z

Table 2 – Station and buffer identification letters

An example of the second part of the genome is shown in Figure 8.

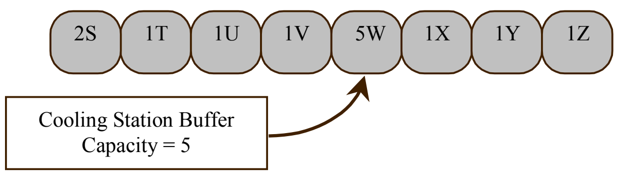

Figure 8 – Second part of the composite genome

Combined, the two components form the new composite genome, shown in Figure 9.

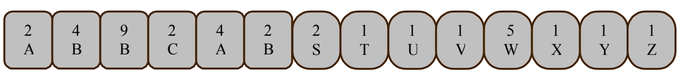

Figure 9 – Complete composite genome

Evolutionary Operators

Two operators are used to mutate the composite genomes to generate a new population, the 'split/join' and 'increase/decrease' mutators. These mutators were designed specifically for this design model and differ drastically from mutation operators traditionally used in E.A.s. The 'split/join' mutator applies to the first part of the genome, modifying mold information, while the 'increase/decrease' mutator applies to the last eight elements that deal with station capacities. Both operators require only one genome to generate a new member of the population, and are considered asexual. Both operators are applied to each genome that remains in the population after the selection process. Each element in the genome is visited and is mutated by the relevant mutator according to the mutation probability parameter.

Split/join mutator

The split/join mutator is designed to change the mold information of a genome while still maintaining 100% build-to-schedule as required by the second design objective. To achieve this, the mutator modifies the number of molds and the number of cavities per mold, but ensures that the total number of cavities dedicated to a specific part type remains constant. This in turn ensures that the number of parts of each type produced every cycle remains constant.

The mutator affects each element of a genome with the mutation probability, and then uses a 50% 'coin-toss' to decide whether to try to split the element into two, or join the element to the next element. The split operation first ensures that the cavity number for the element is greater than one, in which case the element is split into two elements. The mutator divides the number of cavities evenly; if the number of cavities is odd, the second element takes the extra cavity. Figure 10 shows the split operation applied to the third element of the first part of the genome.

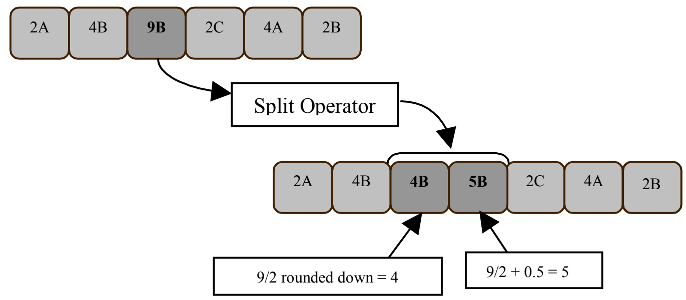

Figure 10 – Split Operator

When the join operation is applied to an element the operator checks to see if the next element has the same part type letter. If not, the operation fails and moves on to the next element. If, however, the part types match, the operator combines the two elements into one element with the same part-type letter and a cavity number equal to the sum of the cavities in the two initial elements. Figure 11 illustrates the join operation applied to the second and third elements of the first part of the genome.

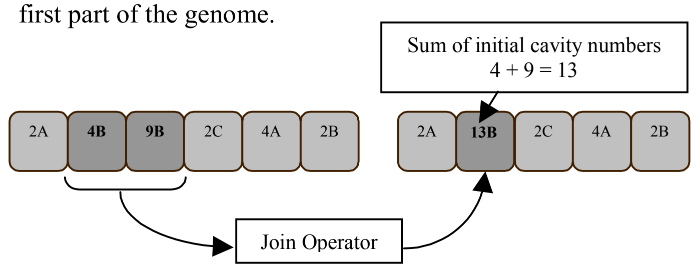

Figure 11 – Join Operator

Increase/decrease mutator

The increase/decrease mutator is designed to change the station capacity information within a genome. The operator successively visits each of the last eight elements in the genome and if applied, either increases or decreases the station capacity based on a 50/50 'coin-toss'. When the increase operator is applied, the station/buffer capacity is increased by a random value between 1 and 3. An example of the increase operator being applied to station/buffer-capacity-related elements of a genome is shown in Figure 12.

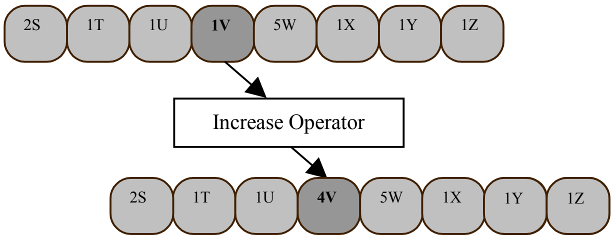

Figure 12 – Increase Operator

The decrease operation works in much the same way as the increase operator except the operator checks to make sure that the resulting capacity does not drop below 1. An example of the decrease operator being applied to the station/buffer-capacity-related elements of a genome is shown in Figure 13.

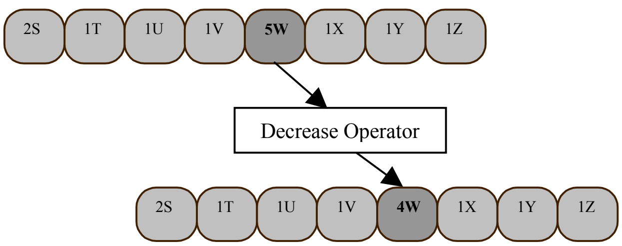

Figure 13 – Decrease Operator

Objective Function

The objective function provides an evaluative measure of the fitness of a genome. The function provides an output score for a set of variables and parameters. The genome and the evolutionary operators are designed to satisfy the *must* design objective of 100% build-to-schedule. The objective function helps to achieve the *wish* design objective of minimizing the total cost of the system by providing an evaluation of the cost for a given genome. That is, the objective function will provide the total cost of the system based on a given codified set of variables. The E.A. uses this fitness measure to ensure that the lower cost options are given a better chance of survival in each generation.

The total cost of the system consists of mold costs, processing costs, and station costs. The calculation of these costs is described in detail in the following sections.

Mold Cost

The mold cost calculation begins with the cost analysis of a single cavity mold, based on part geometry. The cost of a multi-cavity mold is then calculated using Equation A, Appendix III, that accounts for the design and CNC programming cost savings made when machining duplicate cavities (Boothroyd, *et al.*, 1994).

Once the cost of the multi-cavity molds has been found, the costs of all the molds for a specific part type are summed to find the mold cost for a particular part type. Because similar savings regarding duplicated machining and design may be captured across the molds with the same number of cavities for the same part type, a multi-mold index is used. The mold costs for each part type are then added to generate the overall mold costs for the mold-cavity combination specified by the genome. The total cost calculation is shown in Equation B, Appendix III.

The overall mold cost is then amortized over a defined period with a set interest rate in order to calculate the annual cost component of the molds.

Processing Cost

The traditional processing cost calculation (Appendix III, Equation C) is based on Boothroyd, *et al.* (1994). The total cost is found by multiplying the machine hourly rate by the total processing time, which consists of the cycle time multiplied by the number of cycles.

The formula to evaluate the proposed system finds the machine hourly rate but multiplies this value with the total available time for production. The new machine by definition uses the time available to build parts to satisfy the customer's demand. The modified formula is shown in Appendix III, Equation D.

Processing cost is a function of the required clamp force, found by first calculating the maximum separating force generated during the injection of the molten plastic.

Two conditions exist that would cause a particular mold-cavity combination specified by a genome to be infeasible. The objective function checks for these two conditions and if found, the genome is assigned a large (reject) cost which causes it to be rejected. The first condition arises as a result of the machine being unable to meet the customer demand rate due to the production volume requirements and/or processing time constraint caused by part geometry. The total processing times for each of the molds represented by a genome are calculated. The longest resulting time forms the upper limit for the cycle time that can be achieved by the machine. If this time is greater than the Cycle Takt Time, then the machine will be unable to produce the required number of parts in the available time, and a reject cost is assigned.

Rejection condition 1:

$$\begin{pmatrix}\text{Longest cycle time of} \\ \text{any mold in the genome}\end{pmatrix} > \text{Cycle Takt Time}$$

The second condition to be checked involves the flow of molds through the system. The genome specifies the total number of positions available within the system (a summation of the buffer and station capacities). If the total number of molds moving through the system is equal to or greater than the positions available, a deadlock situation occurs (Onvural, 1990) and a reject cost is assigned to the genome to ensure it is rejected from the population during selection.

Rejection condition 2:

$$\text{Number of molds} \geq \begin{pmatrix}\text{Number of positions available} \\ \text{in the system}\end{pmatrix}$$

Station Cost

The station/buffer cost is the summation of the cost of each station multiplied by the respective station/buffer capacity. The calculation is shown in Equation E, Appendix III.

As with the mold cost, the station costs are then amortized over a defined period with a set interest rate to calculate the annual cost component of the molds.

Selection Scheme

A ‘roulette wheel’ selection scheme is used to help ensure that the genomes with the highest fitness level, i.e. lowest cost, have the greatest chance of survival.

RESULTS

Using the evaluative tools developed, the total annual costs were calculated for the new multi-station injection molding machine and the traditional system. A full factorial design of experiments was used to identify the effect of four factors on the total cost. First, however, a number of tests were conducted to ensure that the evaluative algorithms were working properly.

Output data indicates that the algorithm consistently moves in the direction of lowest cost, and never violates the *must* design objective. Various boundary conditions have been tested to ensure that the reject conditions properly filter infeasible system configurations. The algorithm is typically run for 250-500 generations, but convergence of the fitness scores, i.e. the total cost, typically occurs after approximately 150 generations. Computation time for a single run depends heavily on the number of feasible frequencies based on production volumes. A number of final results have been verified using manual calculations.

Design of Experiments (D.O.E.)

A 2^4 full factorial design was used to evaluate cost under different system conditions and to investigate the effect on system output of four selected factors: part wall thickness, part projected area, production volumes, and part variety. The four factors and their respective levels are shown in Table 3.

Part Wall Thickness **(T)**	Part Projected Area **(A)**	Production Volumes **(Vo)**	Part Variety **(Va)**
1 mm	4 cm^2	1000	2
6 mm	100 cm^2	10000	8

Table 3 - Four factors and their respective levels

These values were chosen to represent a realistic range of parts manufactured using injection molding. The 16 trials of the full factorial design were run with number of generations set to 500. The part’s shape selected for those trials, which dictates the complexity and in turn affects the mold cost, is shown in Figure 14.

In an attempt to fairly compare the final costs of the new and traditional machines, two scenarios were developed.

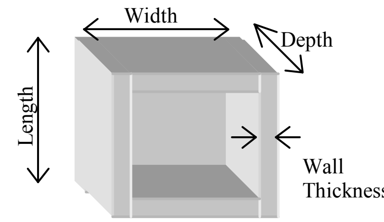

Figure 14 - Part shape

The first scenario constitutes the best case where the machine and processing (operating) costs of the new machine are equal to that of the traditional machine. Assumed here is that the machine and processing costs of the four stations are equal to those of the traditional machines. The second scenario, worst case, assumes that the machine and processing costs of each of the stations of the new machine equals the machine and processing costs of the traditional machine. This translates to a four-fold increase in machine and processing costs for the new machine.

The 16 trials were run to evaluate each of the three cases, (1) the traditional system, and (2) the new machine best-case and (3) worst-case scenarios. These three optimizations were repeated three times, and the minimum values of the three replications (R1, R2, R3) were used in the final data sets for analysis. The data-collection scheme is shown in Figure 15.

Traditional			**New – Best Case**			**New – Worst Case**		
R1 16 trials	**R2** 16 trials	**R3** 16 trials	**R1** 16 trials	**R2** 16 trials	**R3** 16 trials	**R1** 16 trials	**R2** 16 trials	**R3** 16 trials

Minimum data set Traditional system 16 trials	Minimum data set New - Best Case 16 trials	Minimum data set New – Worst Case 16 trials

Figure 15 – Data-collection scheme

An Analysis of Variance (A.N.O.V.A.) was performed on each of the final three sets of data. Higher order interactions, i.e. fourth order interactions, were pooled to establish a ‘within’ or error term. The Pareto plots of the I-hat/2 values for each data set are shown in Figures 16,17 and 18. (I-hat shows the difference in the mean response of the two levels or sets of levels that are being compared.) This value is referred to as the “estimated contrast effect” in some D.O.E. literature.

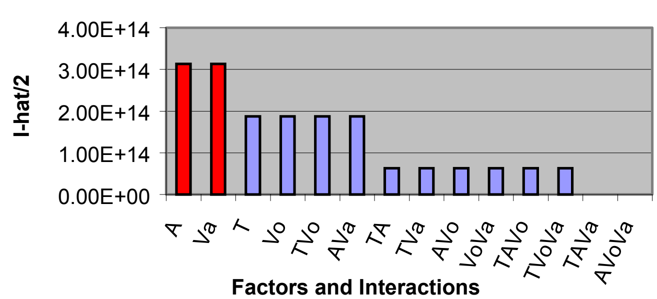

Figure 16 - Half Effects for Traditional Machine

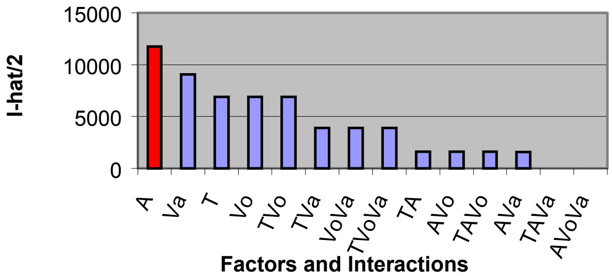

Figure 17 - Half Effects for New Machine (Best-Case)

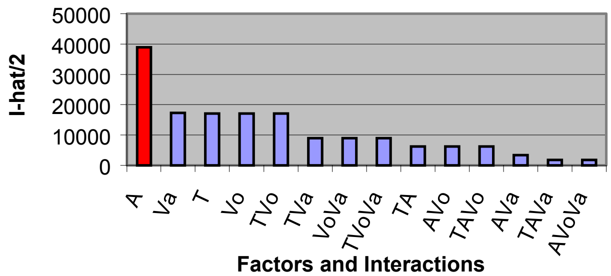

Figure 18 - Half Effects for New Machine (Worst-Case)

The results indicate that the traditional system costs are affected most by the area of the part (A) and part variety (Va). The new machine is less affected by part variety especially in the worst-case scenario, where the impact of part variety is equal to the second tier effects. The difference between the best and worst-case scenarios arises from the fact that an increase in part variety leads to an increase primarily in the mold cost component of the total cost. In the worst-case scenario, the processing and machine costs are quadrupled causing the mold cost to make up a smaller portion of the total cost. Therefore, the increase in part variety has less of an impact on the total cost.

As expected, these results reveal that a key difference between the current and proposed injection molding systems is the proposed system's flexibility regarding part variety.

These initial results confirm the operation of the traditional and new system evaluation tools. Additional investigation will focus on a direct cost comparison of the new and traditional systems. In addition, the ability of each system to handle a combination of part thicknesses, or production volumes, will be investigated.

CONCLUSION

An optimization and evaluation tool was developed to assess the benefits of a new multi-station, multi-mold injection molding system. The tool used an evolutionary algorithm with a genome and evolutionary operators designed to address the unique variables of this system. This research provides the base for further study involving additional simulations to determine the benefits of the proposed system versus the traditional system.

ACKNOWLEDGMENTS

This work used the GALib genetic algorithm software package, written by Matthew Wall at the Massachusetts Institute of Technology. The authors gratefully acknowledge the financial support of Materials and Manufacturing Ontario.

REFERENCES

Bernhardt, A., Kassa, N., 1995, "Benchmarking the Injection Molding Process for Continuous Improvement," *SPE, Conference Proceedings at ANTEC*, 570-574.

Boothroyd, G., Dewhurst, P., Knight, W., 1994, *Product Design for Manufacture and Assembly,* Marcel Dekker Inc., New York.

Cochran, D.S., 1999, "Production System Design Course," MIT.

Grande, J. A., 1995, "Thermoset Processors Fight to Maintain Margins," *Modern Plastics*, Feb.: 55-59.

Henze, H., 1988, "Flexible Manufacturing Cells for Injection Molding," *Kunststoffe German Plastic,* 78/9:785-788.

Jaroschek, C., Steger, R, 1998, "Horizontal Mould Rotation," *Kunststoffe Plastics Europe*, 88/9:1412-1416.

Lankton, G. B., 1985, "Just-in-Time Manufacture," *SPE, Eighth Annual Pacific Technical Conference,* 30-34.

Mok, S.L., Kwong, C.K., and Lau, W.S., 1999, "Review of Research in the Determination of Process Parameters for Plastic Injection Molding," *Advances in Polymer Technology,* 18/3:225-236.

Neilley, R., 1997, "Molding on a Carousel," *Injection Molding Magazine,* Nov.:15-16.

Offergeld, H., 1998. "Economic Benefits Through Use of Advanced Technologies," *SPE, Conference Proceedings at ANTEC*, 535-540.

Onvural, R.O., 1990, "Survey of Closed Queueing Networks with Blocking," *ACM Computing Surveys*, 22/2:83-121.

Packman, I. D., 1989, "Just-in-Time Control for a Polymer Plant," *The Plastics and Rubber Institute, Polymer processing: Automation,* 11:1-8.

Quick, N.J., 1989, "Case for Small Injection Moulding Machines," *Plastics and Rubber International,* 14/1:26-29.

Rozema, H., Travaglini, V., 1995, "Quick Mold Change Systems for High Volume Stack Molds," *SPE, Conference Proceedings at ANTEC*, 1011-1014.

Schut, J. H., 1999, "Why are these men smiling?" *Plastics Technology,* 45/1:100.

Suh, N.P., 1990, *The Principles of Design*, Oxford University Press, New York.

Szatkowski, P.M., Reasor, R.J., 1991, "SMED system for setup reduction. A case study," *International Industrial Engineering Conference Proceeding,* 123-129.

Von Eysmondt, B., 1989, "Flexible Automation in Injection Moulding Shops – State of the Art and Strategy for Introduction," *SPE, Conference Proceedings at ANTEC*, 535-540.

Wall, M., 1996, "GALib: A C++ Library of Genetic Algorithm Components", MIT.

Wilder, R. V., 1990, "Quick-change systems: Key to flexible molding operations," *Modern Plastics,* Aug.:34-37.

APPENDIX I

The parameters are grouped into six categories: part, material, machine, processing, amortization, and optimization.

Design Parameters

Part		
P_1	No. part types	numparts
P_2	Production Volume i	$V_{production}$
	Part Geometry	
P_3	Max Wall Thickness	$L_{thickness}$
P_4	Total Projected Area i	$A_{total\ projected}$
P_5	Width	L_{width}
P_6	Height	L_{height}
P_7	Depth	L_{depth}
P_8	Complexity *	C
P_9	Mold Base Cost i **	$C_{mb\ i}$
	Part Processing Times**	
P_{10}	Heating Time i	$T_{heat\ i}$
P_{11}	Injection Time i	$T_{inj\ i}$
P_{12}	Cooling Time i	$T_{cool\ i}$
P_{13}	Ejection Time i	$T_{ej\ i}$

* As defined by (Boothroyd *et al.*, 1994) including: No. of Side-Pulls, No. of Internal Lifters, No. of Unscrewing Device, Surface Finish/ Appearance, Tolerance, Texture, Parting Plane.

** Refers to calculated parameters. These values were not directly entered into the model but were calculated using other parameters and tracked during the design process.

Material		
P_{14}	Material Type i	Mat_i
Machine		
P_{15}	Buffer Cost	C_{buffer}
P_{16}	Heating Unit Cost	C_{heatu}
P_{17}	Injection Unit Cost	$C_{injectu}$
P_{18}	Cooling Unit Cost	C_{coolu}
P_{19}	Ejection Unit Cost	C_{ejectu}
	Size – Clamping Force**	
P_{20}	Clamping Force	F_{clamp}
Processing		
P_{21}	Processing Cost coefficient	K1
P_{22}	Processing Cost coefficient	K2
P_{23}	Available time per day	$T_{available}$
P_{24}	Production days per year	days
Amortization		
P_{25}	Mold Amortization Period	MoldPeriod
P_{26}	Machine Amortization Period	MachinePeriod
P_{27}	Interest Rate	Interest

Optimization		
P_{28}	Population size	population_size
P_{29}	Mutation probability	mutation_probability
P_{30}	Crossover Probability	crossover_probability
P_{31}	Number of Generations	number_of_generations
P_{32}	Replacement Percentage	replacement_percentage
P_{33}	Replacement Number	replacement_number

Design Variables – Traditional

Molds		
X_{1T}	% $parts_i$ handled by $machine_j$	$i \in [1, n] \quad j \in [1, 2n]$
X_{2T}	#cavities per $mold_{ij}$	$i \in [1, n] \quad j \in [1, 2n]$

Design Variables – New

Station Capacity		
X_1	Heat	$SCap_{heat}$
X_2	Inject	$SCap_{inject}$
X_3	Cool	$SCap_{cool}$
X_4	Eject	$SCap_{eject}$
Buffer Capacity		
X_5	Heat	$BCap_{heat}$
X_6	Inject	$BCap_{inject}$
X_7	Cool	$BCap_{cool}$
X_8	Eject	$BCap_{eject}$
Molds		
X_9	No. molds for each part	
X_{10}	No. cavities per mold	

APPENDIX II

Parameter Range

Part		
P_1	No. part types	$\{i \mid i \in I, [1, n]\}$
P_2	Production Volume i	$(V_{production})$
	Part Geometry	
P_3	Max Wall Thickness	$1\ mm \le L_{thickness} \le 10\ mm$
P_4	Total Projected Area i	$1\ mm^2 \le A_{total\ projected}$
P_5	Width	$0.1\ cm \le L_{width} \le 50\ cm$
P_6	Height	$0.1\ cm \le L_{height} \le 50\ cm$
P_7	Depth	$0.1\ cm \le L_{depth} \le 50\ cm$
Material		
P_{14}	Material Type i	HDPE, ABS, PC, etc.
Machine - Station		
P_{15}	Buffer Cost	(C_{buffer}) = $8000
P_{16}	Heating Unit Cost	(C_{heatu}) = $40000
P_{17}	Injection Unit Cost	$(C_{injectu})$ = $80000
P_{18}	Cooling Unit Cost	(C_{coolu}) = $40000
P_{19}	Ejection Unit Cost	(C_{ejectu}) = $40000
P_{20}	Clamping Force	(F_{clamp}) = [300, 500, 800, 1100, 1600, 5000, 8500] KN

Processing		
P_{21}	Processing Cost coefficient	(K1) = 33.80 $/hr
P_{22}	Processing Cost coefficient	(K2) = 0.0001 $/hr/kN
P_{23}	Available time per day	16 hr
P_{24}	Production days per year	250 days
Amortization		
P_{25}	Mold Amortization Period	3 yrs
P_{26}	Machine Amortization Period	5 yrs
P_{27}	Interest Rate	10 %
Optimization		
P_{28}	Population size	2 ≤ population_size
P_{29}	Mutation probability	0.00 ≤ mutation_probability ≤ 1.00
P_{30}	Crossover Probability	crossover_probability = 0.00
P_{31}	Number of Generations	1 ≤ number_of_generations
P_{32}	Replacement Percentage	0.00 ≤ replacement_percentage ≤ 1.00
P_{33}	Replacement Number	replacement_number

Variable Range – Traditional

Molds			
X_{1T}	% parts$_i$ handled by machine$_j$	$i \in [1, n]$	$j \in [1, 2n]$
X_{2T}	# cavities per mold$_{ij}$	$i \in [1, n]$	$j \in [1, 2n]$

Variable Range - New

Station Capacity		
X_1	Heat	$1 \le (SCap_{heat})$
X_2	Inject	$1 \le (SCap_{inject})$
X_3	Cool	$1 \le (SCap_{cool})$
X_4	Eject	$1 \le (SCap_{eject})$
Buffer Capacity		
X_5	Heat	$1 \le (BCap_{heat})$
X_6	Inject	$1 \le (BCap_{inject})$
X_7	Cool	$1 \le (BCap_{cool})$
X_8	Eject	$1 \le (BCap_{eject})$
Molds		
X_9	No. molds for each part$_i$	$\{k \mid k \in I, [1, m_i]\}$
X_{10}	No. cavities per mold$_k$	$\{l \mid l \in I, [1, molds_k]\}$

APPENDIX III

Find $\mathbf{X}^* \in R^n$

To minimize cost $f(\mathbf{X}^*, \mathbf{P})$

= Mold Cost + Processing Cost + Station Costs

subject to

$B.T.S._{system}(\mathbf{X}^*, \mathbf{P}) = 100\%$

$\Rightarrow B.T.S._{i}(\mathbf{X}^*, \mathbf{P}) = 100\%$

for $\{i \mid i \in I, [1, \text{number of parts}]\}$

$\Rightarrow$ # units produced $_i$ = # units demanded $_i$

$X_l \le \mathbf{X}^* \le X_u$

Figure A – Optimization Formulation

$$C_{\text{multi cavity mold}} = (C_{\text{single cavity mold}})\, n^m$$

where m is the multi-cavity mold index = 0.7
n is the number of cavities

Equation A – Multi-cavity mold cost calculation

$$C_{\text{overall mold cost}} = \sum_{i=1}^{\#\text{part types}} \left(\sum_{j=1}^{\substack{\#\text{molds for part i}\\ \text{with different}\\ \#\text{cavities}}} (C_{\text{multiple cavity mold}})\, n^{0.7} \right)$$

where n = number of molds for part$_i$ with the same number of cavities

Equation B – Overall mold cost calculation

$$C_{\text{processing}} = (K_1 + K_2 F)(N_t / n)t$$

where: K_1, K_2 = machine rate coefficients
F = clamp force (kN)
N_t = number of molded parts required
n = number of cavities in the mold
t = total cycle time

Equation C – Processing cost calculation for the traditional system

$$C_{\text{daily processing cost}} = (K_1 + K_2 F)(\text{available time per day})$$

where: K_1, K_2 = machine rate coefficients
F = clamp force (kN)

Equation D – Processing cost calculation for the new system

$$C_{\text{station/ buffer}} = (\Sigma\ \text{Buffer capacities})(\text{Cost of buffer}) + \sum_{k=1}^{4}(\text{Station capacity})(\text{Cost of station})$$
$$= (X_5 + X_6 + X_7 + X_8)(P_{15}) + (X_1)(P_{16}) + (X_2)(P_{17}) + (X_3)(P_{18}) + (X_4)(P_{19})$$

Equation E – Station/Buffer cost calculation