Author: Yeh-Liang Hsu, Tzu-Chi Liu, Francis Thibault, Benoit Lanctot (2003-09-04); recommended: Yeh-Liang Hsu (2004-03-24).

Note: This paper is published in *Proceedings of the I MECH E Part B Journal of Engineering Manufacture*, Vol. 218, No. 2, 1 February 2004, p. 197-212(16).

Design optimization of the blow moulding process using fuzzy optimization algorithm

Abstract

Blow moulding is the forming of a hollow part by “blowing” a mould-cavity-shaped parison that is made by thermoplastic molten tube. Blow moulded parts often require a strict control of the thickness distribution in order to achieve the required mechanical performance and final weight. A fuzzy optimization algorithm for determining the optimal die gap openings and die geometry for the required thickness distribution in the blow moulding process is presented. The idea of the fuzzy optimization algorithm is that, instead of using purely numerical information to obtain the new design point in the next iteration, engineering knowledge and human supervision process can be modeled in the optimization algorithm using fuzzy rules. The structure of an optimization algorithm is still maintained to guide the engineering decision process and to ensure that an optimal solution rather than a trial and error solution can be obtained. It is shown that how a single fuzzy engine can be used in various cases and types of optimizations of the blow moulding process.

Keywords: blow moulding, fuzzy optimization algorithm, computer simulation

Notation

D_{im}	mandrel diameter at the neck point
D_{ob}	diameter of bushing
D_{om}	mandrel diameter at the die exit
f	optimization objective function
m	the total number of nodes in the finite element model
n	the total number of discrete die gap opening programming points
x_i	die gap opening at i-th programming point
Δx_i	the vector of change of die gap openings at i-th programming point
$\boldsymbol{x}^q$	vector of x_i in the q-th iteration
$\Delta \boldsymbol{x}^q$	the change in $\boldsymbol{x}^q$
$x_{i,\max}$	maximum allowable die gap opening for i-th programming point
$x_{i,\min}$	minimum allowable die gap opening for i-th programming point
Y	target thickness
y_j	the thickness at the j-th node
$\boldsymbol{y}^q$	vector of y_j in the q-th iteration
$\Delta \boldsymbol{y}^q$	the change in $\boldsymbol{y}^q$
$\bar{y}_i$	the average weighted thickness $\bar{y}_i$ of all nodes affected by x_i
$\bar{y}_{i,\max}$	maximum value of average weighted thickness for i-th programming point
$\bar{y}_{i,\min}$	minimum value of average weighted thickness for i-th programming point
α^q	step size in the q-th iteration

1. Introduction

Blow moulding is the forming of a hollow part by “blowing” a mould-cavity-shaped parison that is made by thermoplastic molten tube. It is the most popular and efficient process for manufacturing commodity hollow plastic parts such as bottles, containers, and toys. More recently, this forming process has been applied to the manufacture of complex automotive parts such as fuel tanks, seat backs, air ducts, windshield washer and cooling reservoirs.

The blow moulding process consists of three phases: parison extrusion, parison inflation and part solidification. As shown in Figure 1, the extrusion phase involves the extrusion of a polymer melt through an annular die to form a hollow cylindrical parison with a non-uniform material distribution and consequently non-uniform parison thickness along its length. Once the parison is extruded to the desired length, it is inflated to take the shape of an enclosing mould. The part then solidifies as a consequence of heat transfer to the cooling mould. The parison thickness distribution is modified significantly by the inflation and the solidification stages to yield the final part thickness distribution.

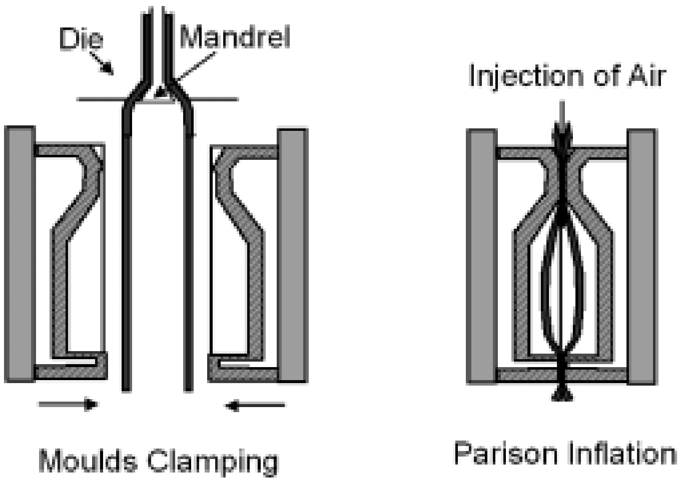
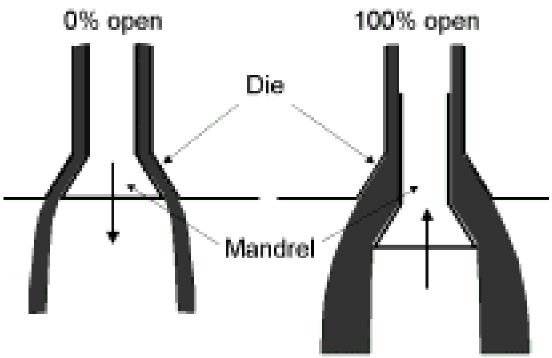
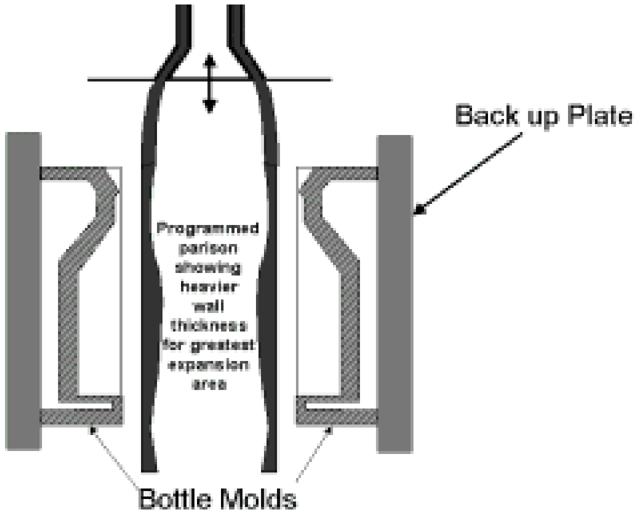


Figure 1. Blow moulding process

Blow moulded parts often require a strict control of the thickness distribution in order to achieve the required mechanical performance and final weight. Figure 2 shows the forming of an axisymmetric bottle. As illustrated in Figure 2(a), by moving the mandrel up and down, the die gap can be adjusted as a function of time. The movement of the mandrel can be programmed by the percentage of gap openings at discrete time. When gap opening is 0%, the mandrel is at the upper limit, which results in the minimum die gap; when gap opening is 100%, the mandrel is at the lower limit, which results in the maximum die gap. Manipulation of the programming of gap openings can lead to an optimal part thickness distribution. For example, in order to obtain uniform thickness distribution of the hollow part, the thickness of a programmed parison must be vary along its length. As shown in Figure 2(b), the parison thickness for the largest expansion area must be thicker than those of the other areas.



(a)



(b)

Figure 2. Illustration of parison programming

BlowSim is a finite element software package designed to simulate the extrusion blow moulding, injection stretch blow moulding, and thermoforming processes. It is developed by the Industrial Materials Institute (IMI) of the National Research Council (NRC), Canada. The blow moulding process simulation consists of modelling the successive process stages in order to predict the final part quality as a function of the operating conditions, the mould geometry and the material properties. Predictions of final part thickness were made by integrated simulation of the parison formation, clamping and inflation, and part cooling and solidification stages. Programming points, die dimensions, extrusion temperature, parting plane shape, and mould temperature were among the operating conditions considered **[1]**. BlowSim can be used to model the process phases: parison formation, clamping and inflation, part cooling and shrinkage, and part mechanical performance. The process modelling is based on a large displacement finite element formulation **[2]**. The parison deformation is modelled using a multi-layer membrane element type and a non-isothermal visco-elastic material model. The mechanical performance is modelled with the predicted thickness distribution, and the appropriate applied load. The simulation results of BlowSim have been validated with many industrial cases and show good agreement.

In many industrial applications, combining simulation tools with optimization methodologies allows the designers to treat complex design criteria via simulation to pursue maximum part quality and minimum manufacturing costs. In the blow moulding process, it is desirable to manipulate the percentage of die gap openings to obtain a final part of constant thickness or a predefined thickness profile. It is, therefore, an optimization problem on how to control the die gap openings to minimize the deviation in the thickness of the final part from the target thickness.

Figure 3 shows the finite element model of the bottle case in Figure 2 by BlowSim. Given a set of die gap openings at n programming points x_i, i=1, 2, …, n, we are able to extract the thickness of all nodes from the simulation results by BlowSim, and apply them to the following equation to get the objective function value:

$$\text{min. } f = \left(\sum_{j=1}^{m} (y_j - Y)^2 \Big/ m-1 \right)^{0.5} \quad (1)$$

where y_j is the thickness at the j-th node in the simulation model, and Y is the corresponding target thickness, and m is the total number of nodes in the BlowSim finite element model. The die gap openings at discrete programming points x_i, i=1, 2, …, n, are the variables to be determined in the optimization process. Obviously, the thickness at a node is a function of the corresponding die gap openings.

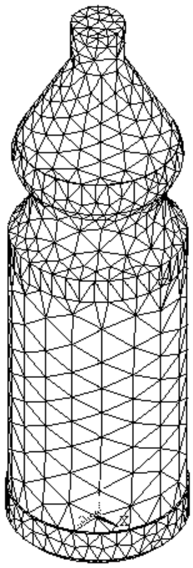


Figure 3. The finite element model of the bottle

DiRaddo and Garcia-Rejon **[3]** proposed an iterative optimization loop which combined a blow moulding process predictor and an updating technique to search for the parison thickness profile that results in the minimum overall difference between the specified final part thickness distribution and the individual iteration's output from the predictor. Lee and Soh **[4]** determined the optimal thickness profiles of a preform for a blow-moulded part having required wall thickness distribution. A finite element model is formulated to relate the preform wall thickness distribution to the wall thickness distribution in the blow-moulded part. The feasible direction method is used for optimization, and the design variables are the thickness of finite elements.

Gradient type numerical optimization algorithms can certainly be used to solve for the optimal die gap openings. On the other hand, manufacturing engineers usually adjust the die gap openings empirically: reduce the die gap opening if the corresponding portion of the final part is too thick, and vice versa.

When solving an engineering optimization problem using numerical optimization algorithms, we basically view the problem as a pure mathematical optimization model. Design modifications in the optimization process rely on numerical information rather than engineering heuristics, experience, and knowledge. This paper develops a “fuzzy optimization algorithm” for engineering optimization problems, which enables the use of engineering heuristics to generate the new design point of the next iteration. The structure of an optimization algorithm is still maintained to guide the engineering decision process and to ensure that an optimal solution rather than a trial and error solution can be obtained. Currently this fuzzy optimization algorithm is developed specifically for engineering optimization problems whose objective functions are in the form of Equation (1).

This paper first explains the concept of fuzzy optimization algorithms. The blow moulding process optimization results are presented to demonstrate the generality of this approach to various optimization cases in different application domains.

2. The concept of “fuzzy optimization algorithms”

As shown in Figure 4, the optimization process can be viewed as a closed-loop control system. In the case of blow moulding process optimization, BlowSim is analogous to the system process to be controlled, whereas an optimization algorithm is analogous to the controller. In the q-th iteration, BlowSim simulation results (thickness distribution $\boldsymbol{y}^q$) are input to the optimization algorithm, which in turn generates the change in die gap openings ($\Delta \boldsymbol{x}^q$) according to its search rules. Die gap openings for the next iteration are updated ($\boldsymbol{x}^{q+1} = \boldsymbol{x}^q + \Delta \boldsymbol{x}^q$) and simulated again using BlowSim to continue the iteration. Finally, a control system attempts to achieve a stable, predefined output. The optimization process pursues a converging objective function value.

Figure 4. General block diagram of a design optimization process

When we apply traditional numerical optimization algorithms to an engineering problem, we treat the engineering problem as a pure mathematical problem. Engineering heuristics are totally ignored. This motivates the idea that, in addition to crisp numerical rules, the engineering heuristics such as “reduction in the die gap opening if the corresponding portion of the final part is too thick, and vice versa” should also be modeled in the optimization algorithm using fuzzy rules. As suggested in Figure 4, the “controllers” in the optimization process may as well be fuzzy controllers!

A fuzzy system is characterized by a collection of linguistic statements based on expert knowledge. The linguistic statements are usually in the form of IF-THEN rules. As shown in Figure 4, if the relations between the system process input $\boldsymbol{x}^q$ (die gap openings) and system process output $\boldsymbol{y}^q$ (thickness distribution) and $\Delta\mathbf{y}^q$ are known empirically (reduce the die gap opening will reduce the thickness of the corresponding portion of the final part, and vice versa), a fuzzy logic engine instead of a numerical optimization algorithm can be used to generate the system process input change rate $\Delta\mathbf{x}^q$ according to a set of domain parameters given by the users.

Arakawa and Yamakawa **[5]** demonstrated an optimization method using qualitative reasoning, which makes use of the qualitative information that gives an approximate direction of the optimum search. Hsu et al. **[6]** proposed a fuzzy optimization algorithm and applied it for determining the “move limit”, which is an important optimization process parameter in the sequential linear programming algorithm. Mulkay and Rao **[7]** also proposed a modified sequential linear programming algorithm using fuzzy heuristics to control the optimization parameters. Arabshahi et al. **[8]** pointed out that many optimization techniques involve parameters that are often adapted by the user through trial and error, experience, and other insight. Instead, they applied neural and fuzzy ideas to adaptively select these parameters.

In these papers, fuzzy heuristics were used to control the parameters of the optimization algorithm to improve its performance. The following sections demonstrate how engineering heuristics can also be modeled into the fuzzy optimization algorithm for the optimization of the blow moulding process.

3. Die gap opening optimization for constant part thickness

The bottle case study in Figure 3 was first used to illustrate the fuzzy optimization algorithm. In this example, we hope to manipulate the die gap openings at 7 discrete programming points (x_i, i=1,.., 7) to obtain a uniform wall thickness part of 2 mm. Therefore, in the objective function Equation (1), Y = 2. Note that the die gap opening at a discrete time point x_i may affect the thickness of many nodes. BlowSim provides the “average weighted thickness” $\bar{y}_i$ of all nodes affected by x_i. As discussed earlier, designers usually adjust the die gap openings empirically: reduce the die gap opening if the corresponding portion of the final part is too thick, and vice versa. This engineering heuristic indicates that the average weighted thickness of a certain portion ($\bar{y}_i$) is a monotonic increasing function with respect to the corresponding die gap opening (x_j), and can be expressed by 5 rules:

(1) IF $\bar{y}_i$ is PB THEN Δx_i is NB;

(2) IF $\bar{y}_i$ is PS THEN Δx_i is NS;

(3) IF $\bar{y}_i$ is ZE THEN Δx_i is ZE;

(4) IF $\bar{y}_i$ is NS THEN Δx_i is PS;

(5) IF $\bar{y}_i$ is NB THEN Δx_i is PB.

The quantization table (Table 1) gives quantitative definitions for PB (positive big), PS (positive small), ZE (zero), NS (negative small) and NB (negative big). There are 5 "domain parameters" in Table 1 to be determined by the user according to the application problem. The definition of the 5 domain parameters and their numerical values for the bottle case example are

Y: target thickness (2mm);

$\bar{y}_{i,\min}$: minimum value of average weighted thickness (0mm for all programming points);

$\bar{y}_{i,\max}$: maximum value of average weighted thickness (4mm for all programming points);

$x_{i,\min}$: minimum allowable die gap opening (5% for all programming points);

$x_{i,\max}$: maximum allowable die gap opening (95% for all programming points).

Table 1. The quantization table

Boundaries of fuzzy input, $\bar{y}_i$	**Boundaries of fuzzy output, Δx_i**	**Quantized Level**
$Y+(\bar{y}_{i,\max}-Y)$	$x_{i,\max}-x_i$	2
$Y+(\bar{y}_{i,\max}-Y)/2$	$(x_{i,\max}-x_i)/2$	1
Y_j	0	0
$Y-(Y-\bar{y}_{i,\min})/2$	$(x_{i,\min}-x_i)/2$	1
$Y-(Y-\bar{y}_{i,\min})$	$x_{i,\min}-x_i$	2

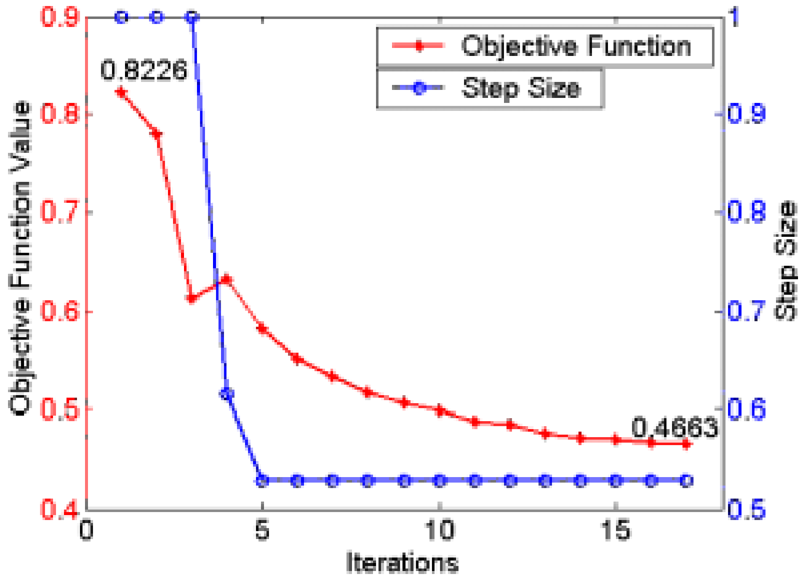
Table 2 shows the results for the first two iterations of the bottle example. Initially, the die gap openings are set at 50% for all 7 programming points. From BlowSim simulation, the average weighted thickness varies from 1.341mm to 5.119mm, and the objective function value is 0.82. Then the fuzzy engine generates the change in die gap openings Δx_i and the die gap openings at the 7 programming points are updated. From BlowSim simulation, the average weighted thickness now varies from 1.374mm to 3.262mm, and the objective function value is reduced to 0.78. The fuzzy engine then generates Δx_i for this iteration, and the die gap openings are updated again.

Table 2. Results for the first two iterations of the bottle example

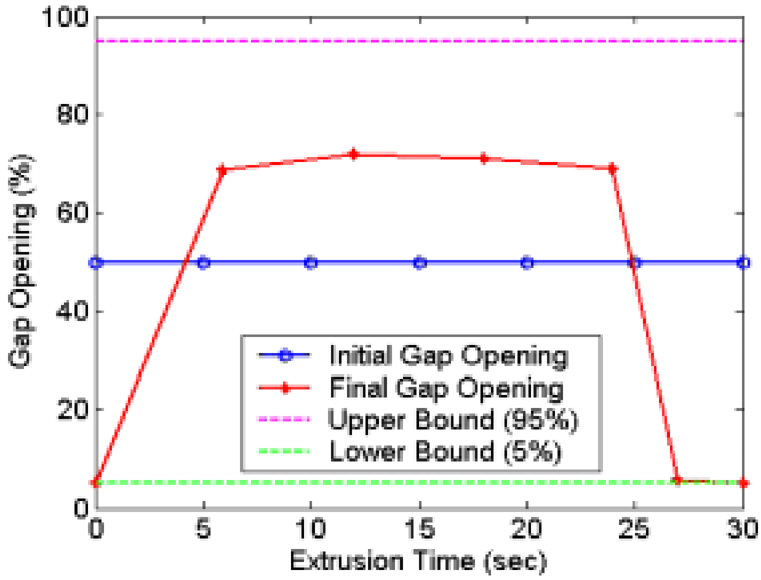
Programming points		1	2	3	4	5	6	7	Objective function value
Initial values	x_i(%)	50	50	50	50	50	50	50	0.82263
	$\bar{y}_i$ (mm)	5.119	2.671	1.362	1.341	1.858	3.957	4.828	
	Δx_i(%)	-34.604	-15.623	14.336	14.791	2.787	-33.998	-34.604	
1^{st} iteration	x_i(%)	15.396	34.377	64.336	64.791	52.787	16.002	15.396	0.77913
	$\bar{y}_i$ (mm)	3.258	1.374	1.938	2.29	2.015	2.846	3.262	
	Δx_i(%)	-5.151	18.928	0.835	-18.767	-0.3119	-4.475	-5.159	
2^{nd} iteration	x_i(%)	10.256	53.305	65.171	46.025	52.475	11.528	10.237	0.61265
	$\bar{y}_i$ (mm)	3.453	2.292	1.978	1.212	1.652	2.696	3.042	
	Δx_i(%)	-2.842	-6.330	0.2905	18.902	6.788	-2.263	-2.398	

Referring to Figure 4, in the optimization iterations, we expect the objective function to flatten out when approaching convergence. However, in reality, the objective function value might "overshoot" when approaching convergence. In many numerical optimization algorithm, a scalar multiplier α^q (often called "step size") determining the amount of change for this iteration is introduced, and $\boldsymbol{x}^{q+1}= \boldsymbol{x}^q+\alpha^q\Delta\boldsymbol{x}^q$**[9]**. Usually α is adjusted dynamically throughout the iteration process. The heuristics for adjusting α is simply, reduce if the change in objective function value is big, and vice versa. Obviously this can also be expressed by the same 5 rules previously discussed. In the examples in this paper, initially α^0=1, and in the iteration process, α is adjusted using the same fuzzy engine. If the change in objective function is big (objective function increases rather than decreases), α in the next iteration will be reduced to 0.5~1.0 times of that of the current iteration. The current iteration will be given up if the increase in the objective function is larger than 10%.

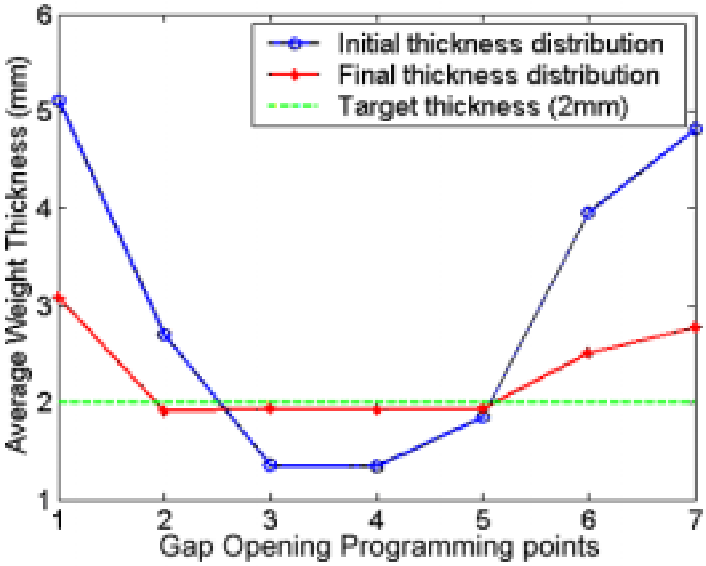
Finally, Figure 5(a) shows the iteration history for the bottle case, including the history of the objective function value and the step size to show the effect of the step size control. Figure 5(b) compares the initial (50% die gap openings for all programming points) and final die gap openings, and Figure 5(c) compares the average weighted thickness of the initial and final design on the 7 programming points.



(a) Iteration history



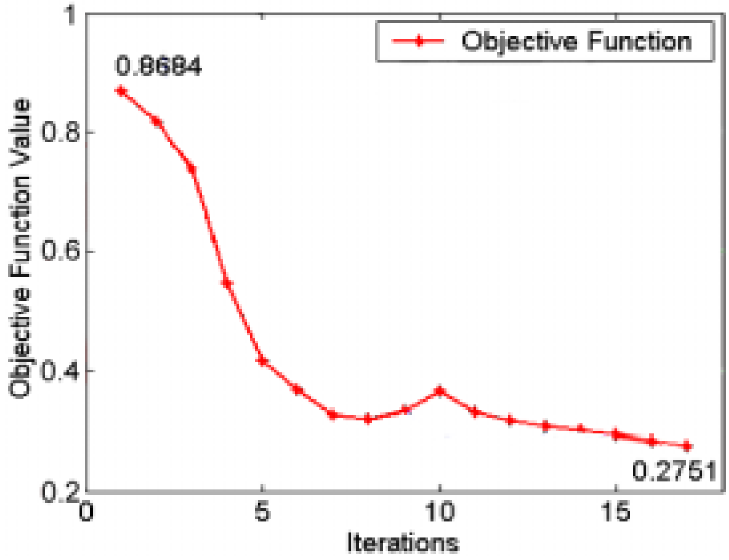
(b) Initial and final die gap openings



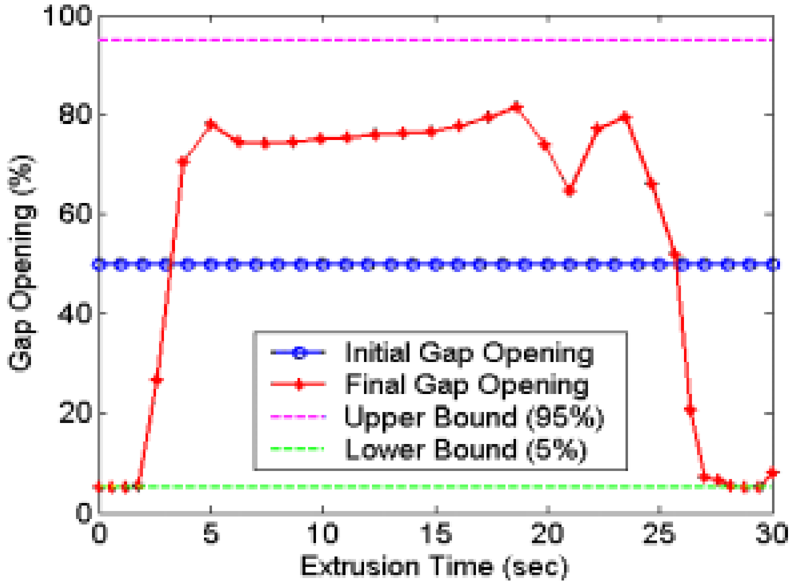
(c) Initial and final average weighted thickness

Figure 5. Die gap opening optimization results for the bottle case study using 7 programming points

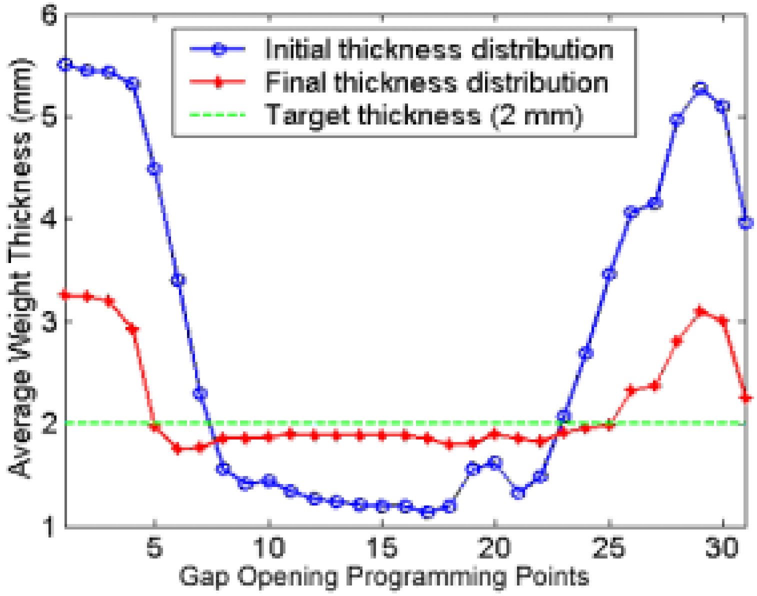
In this example, the optimization process terminated after 17 iterations, when the change in objective function value was less than 0.1%. Only 18 BlowSim simulations were needed (one simulation was given up between iterations 3 and 4 because the overshoot was too large), and no sensitivity calculation was required. Ideally the objective function should converge to zero upon obtaining a part with uniform thickness. However, the objective function at the end of the iteration is 0.47, and as shown in Figure 5(c), the average weighted thickness of the top and bottom portions of the bottle are still higher than the target value. As shown in Figure 5(b), the average weighted thickness at these two portions cannot be further reduced because the corresponding die gap openings are already close to the lower bound 5%. Figure 6 shows the optimization result of the bottle example with the same domain parameters using 31 programming points. The fuzzy optimization algorithm terminated after 18 iterations, and 20 simulations were needed in this case. Increasing the number of programming points further reduces the objective function value. However, too many programming points in a short parison extrusion time is sometimes not practical since the pneumatic mandrel movement into the die head is limited by its response time. Note that the computation cost in each iteration of the fuzzy optimization algorithm is independent of the number of design variables.



(a) Iteration history



(b) Initial and final die gap openings



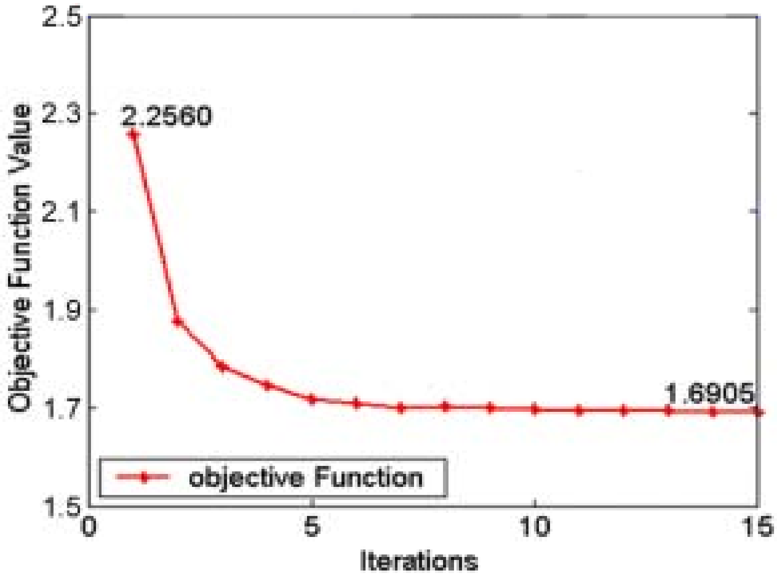
(c) Initial and final average weighted thickness

Figure 6. Die gap opening optimization results of the bottle case study using 31 programming points

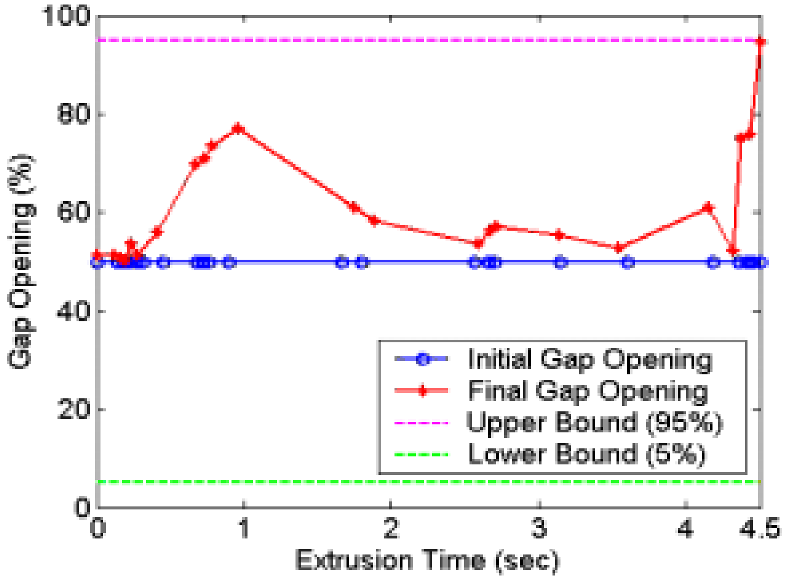
The fuzzy optimization algorithm was then applied to the process optimization of a fluid reservoir shown in Figure 7, which is a more complex automotive part. In this case the target thickness was 5mm, and 23 programming points were used. The values of the domain parameters in this case were assigned as: Y=5mm, $\bar{y}_{i,\min}$=0mm, $\bar{y}_{i,\max}$=10mm, $x_{i,\min}$=5% and $x_{i,\max}$=95%.

Figure 7. Geometry of the windshield washer fluid reservoir

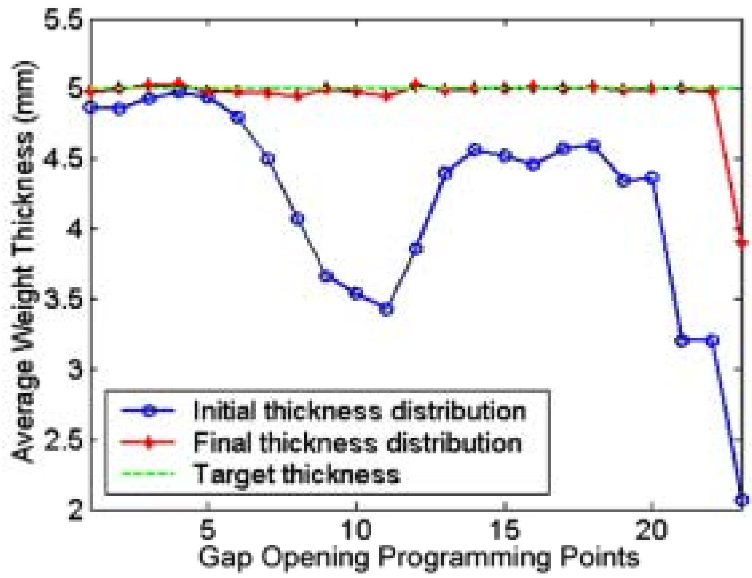
The optimization process terminated after 15 iterations, and Figure 8 shows the optimization results. In Figure 8(c), the average weighted thickness of all programming points after optimization are close to 5mm, but the objective function value in Figure 8(a) at the end of optimization is still high (1.69). This is because the fluid reservoir is not symmetric. For unsymmetrical parts whose cross sections are not circular, it is not possible to obtain a part with uniform thickness using a circular die. When the parison is inflated to take the shape of an enclosing mould, the thickness varies along the cross-section of the part. The die geometry has to be manipulated before the die gap opening optimization in order to obtain the desired part thickness. This will be discussed in the next section.



(a) Iteration history



(b) Initial and final die gap openings



(c) Initial and final average weighted thickness

Figure 8. Die gap opening optimization results for the fluid reservoir case

4. Die geometry optimization

In BlowSim, the geometry of the die in the closed and open positions is defined by the minimum and maximum die gap (GapMin and GapMax) at a number of die sections or "die points" at different angular positions. Figure 9 shows the geometry of a typical bushing and mandrel die head components. GapMin is defined as the effective die gap for 0% die gap opening

$$\text{GapMin} = (D_{ob} - D_{om})/2, \tag{2}$$

and GapMax is defined as the effective die gap for 100% die gap opening

$$\text{GapMax} = (D_{ob} - D_{im})/2, \tag{3}$$

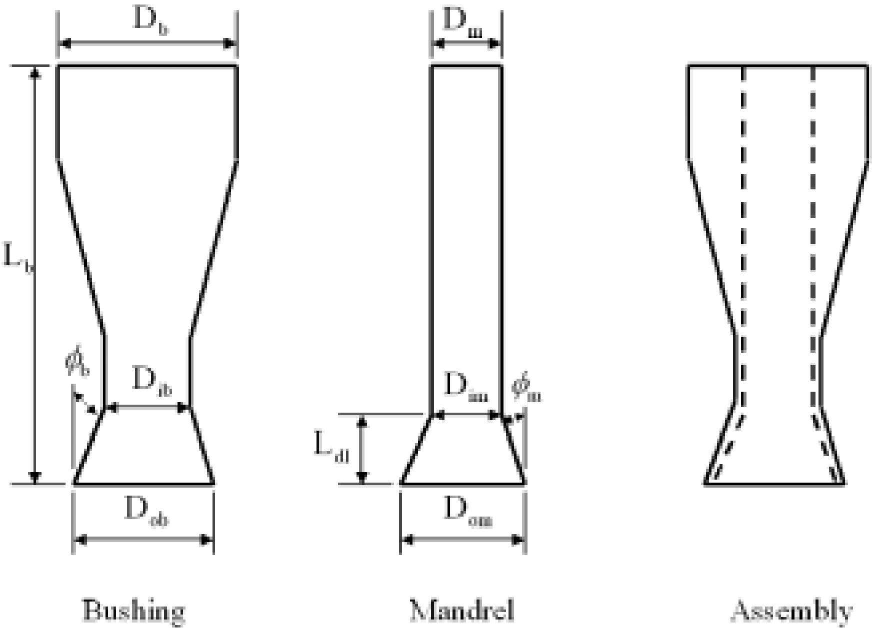
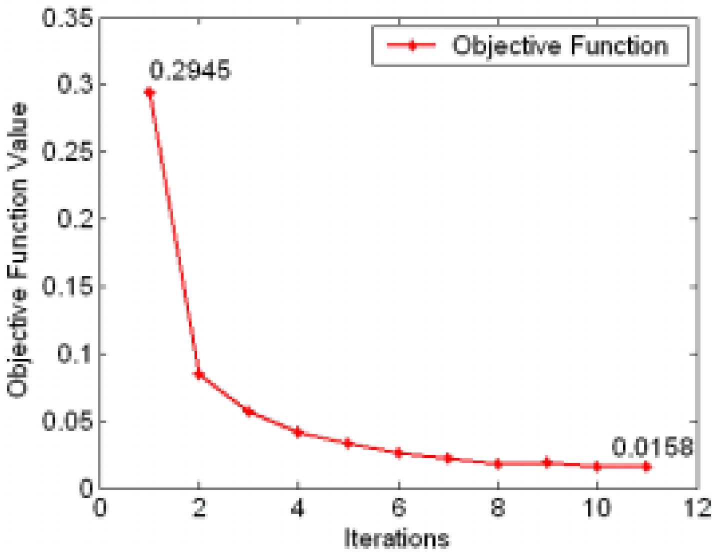


Figure 9. Illustration of the die geometry

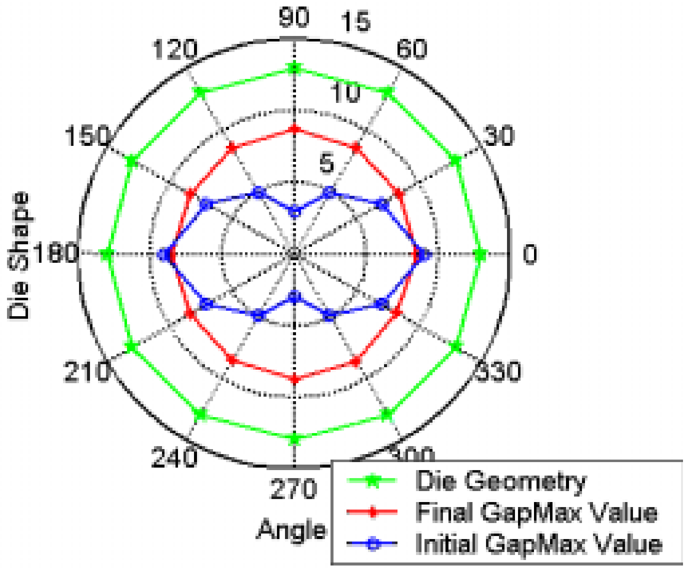
For symmetrical parts with circular cross-sections, a uniform thickness can be obtained using circular die geometry. For unsymmetrical parts, GapMin and GapMax at each die point should be optimized first to obtain a die geometry that is suitable for the shape of the unsymmetrical part. Then the die gap opening optimization is carried out to obtain a part with constant thickness using this die geometry. BlowSim also provides the average weighted thickness of all nodes affected by a die point.

Here the die geometry optimization manipulates only GapMax while keeping GapMin fixed. The objective is to obtain constant average weighted thickness for all die points. The engineering heuristics for adjusting GapMax is the same as those for adjusting die gap openings: reduce GapMax if the corresponding average weighted thickness is too large, and vice versa. Obviously this can also be expressed by the same 5 fuzzy rules previously discussed. Note that the objective function used here is to minimize the deviation of the average weighted thickness of the die points from the target thickness. Moreover, in die geometry optimization, x_i becomes the GapMax at die point *i*.

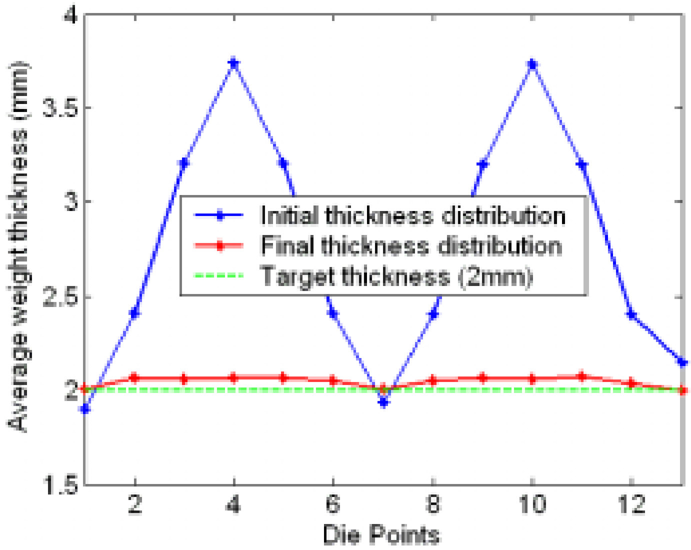
The bottle case in Figure 3 was used again to verify the results of die geometry optimization using the fuzzy optimization algorithm. As shown in Figure 10(b), we deliberately created a noncircular initial die geometry for validation purpose. The domain parameters are: Y =2mm, $\bar{y}_{i,\min}$ =0mm, $\bar{y}_{i,\max}$ =4mm, $x_{i,\min}$=3mm and $x_{i,\max}$=13mm. Note that the definitions of some of the parameters have been changed, though the same fuzzy engine is used. In this case, the die gap openings were kept at 50% during the die geometry optimization, and GapMin was fixed at 3mm. Figure 10(a) shows that after 11 iterations, the fuzzy optimization algorithm converged to the expected circular die geometry shown in Figure 10(b) because the bottle is a symmetrical part. A total of 11 BlowSim simulations were needed. Figure 10(c) shows that the final average weighted thickness of all die points are close to the target thickness of 2mm.



(a) Iteration history



(b) Initial and final die geometry

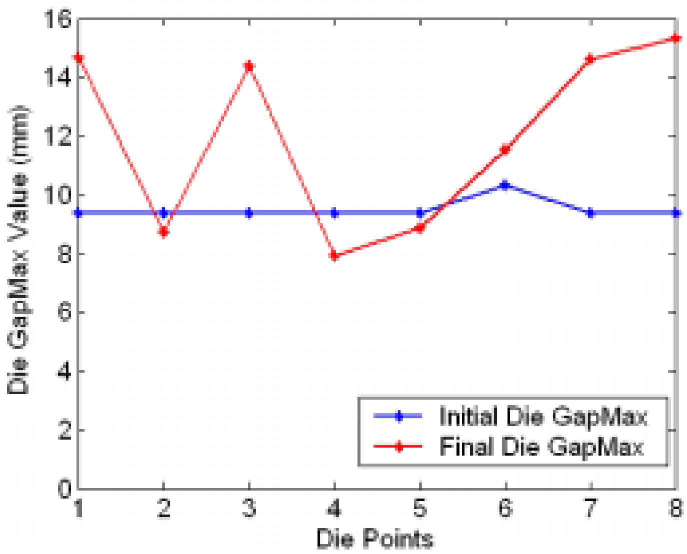


(c) Initial and final average weighted thickness

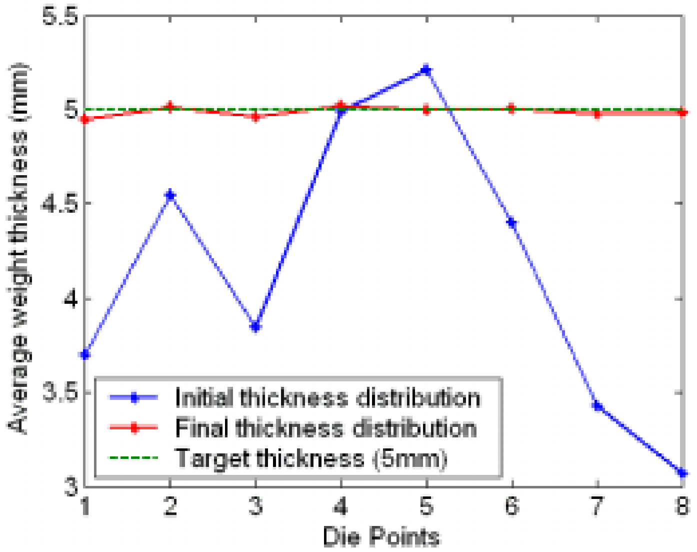
Figure 10. Die geometry optimization results of the bottle case

The die geometry optimization process was then applied to the windshield washer fluid reservoir in Figure 7. In this case the die gap openings were kept at 50% during the die geometry optimization, and GapMin was fixed at 0.17mm. The domain parameters in the die geometry optimization of the fluid reservoir are assigned as follow: Y =5mm, $\bar{y}_{i,\min}$ =0mm, $\bar{y}_{i,\max}$ =10mm, $x_{i,\min}$=0.17mm and $x_{i,\max}$=20mm. The die geometry optimization process terminated after 15 iterations. Figure 11(a) compares the initial and final GapMax. In this example (and the following examples), the diameter of the die is relatively large compared to the die gap, and therefore the die geometry is not shown here. Figure 11(b) shows that the average weighted thickness of all die points are close to the target thickness of 5mm.

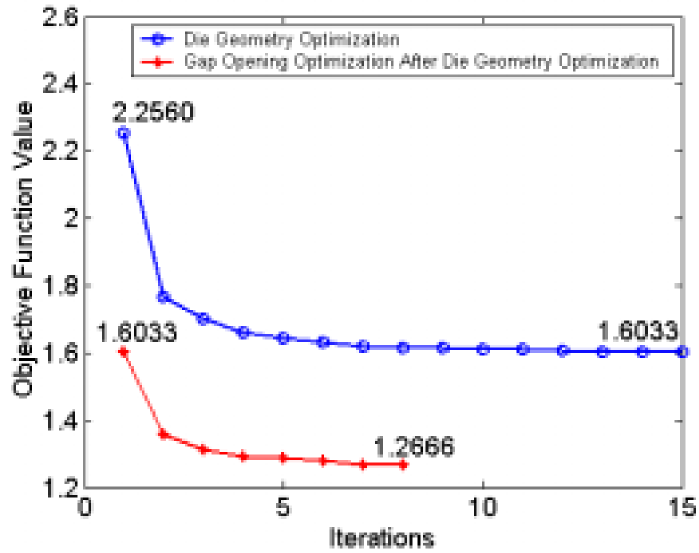
To get a better design, die gap opening optimization was then applied to the fluid reservoir example using the final die geometry in Figure 11(a). The domain parameters are Y =5mm, $\bar{y}_{i,\min}$ =0mm, $\bar{y}_{i,\max}$ =10mm, $x_{i,\max}$=95%, and $x_{i,\min}$=5%. Figure 11(c) shows the iteration history of both die geometry optimization and die gap opening optimization. The objective function in Figure 11(c) is the deviation of the thickness of the final part from the target thickness (Equation (1)). Compared to the final die gap opening objective function value of 1.69 obtained from the gap opening optimization (Figure 8(a)), the die geometry optimization objective function drops from 2.26 to 1.60, and further drops to 1.27 after die gap opening optimization. A total of 23 BlowSim simulations were needed, 15 for die geometry optimization and 8 for die gap opening optimization. Figure 11(d) shows the initial and final die gap openings.



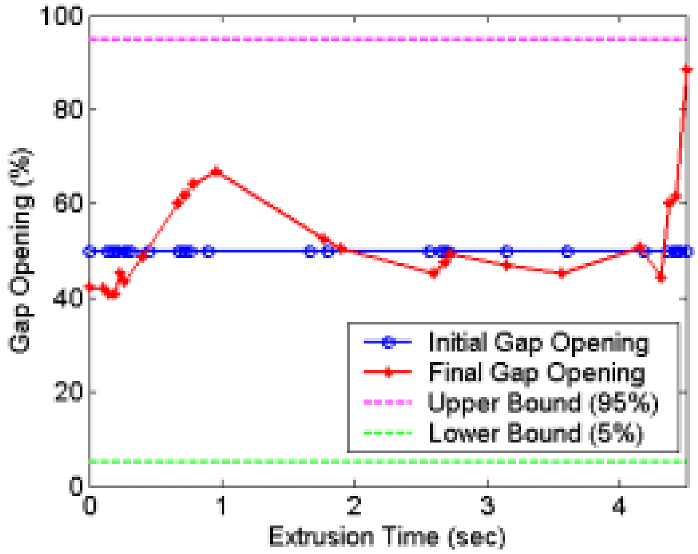
(a) Initial and final GapMax



(b) Initial and final average weighted thickness of the die geometry optimization



(c) Iteration history of the die geometry and die gap opening optimization



(d) Initial and final die gap openings

Figure 11. Optimization results of the fluid reservoir case

5. Application Examples

In this section, the process of die geometry optimization followed by die gap opening optimization is applied to the other two unsymmetrical blow moulded parts shown in Figure 12 and 13, the jerry can and the gas tank, respectively. Again, the same fuzzy engine can be applied to both examples for die geometry optimization and die gap opening optimization after a simple domain parameter mapping shown in Table 3 and Table 4.

Figure 12. Geometry of the jerry can

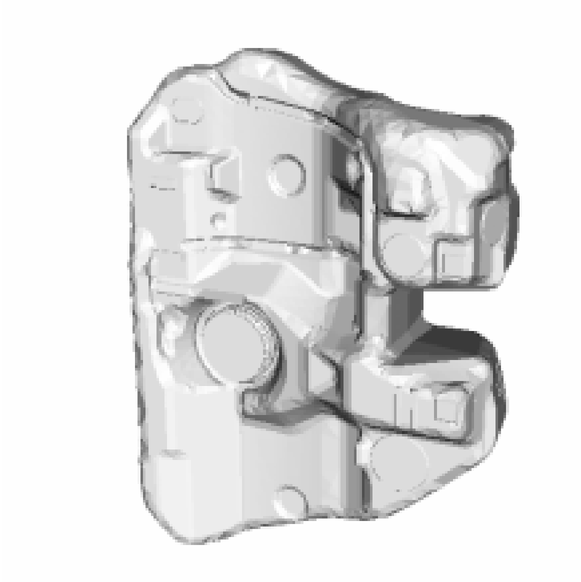


Figure 13. Geometry of the fuel gas tank

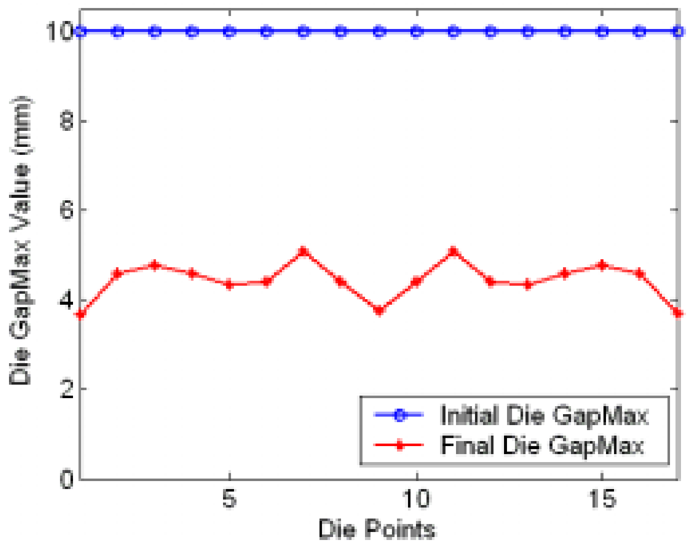
Table 3. Domain parameter mapping for the jerry can example

Domain Parameters	Die Geometry Optimization	Die gap opening Optimization
Y	2mm	2mm
$\bar{y}_{i,\max}$	4mm	4mm
$\bar{y}_{i,\min}$	0mm	0mm
$x_{i,\max}$	20mm	95%
$x_{i,\min}$	2mm	5%

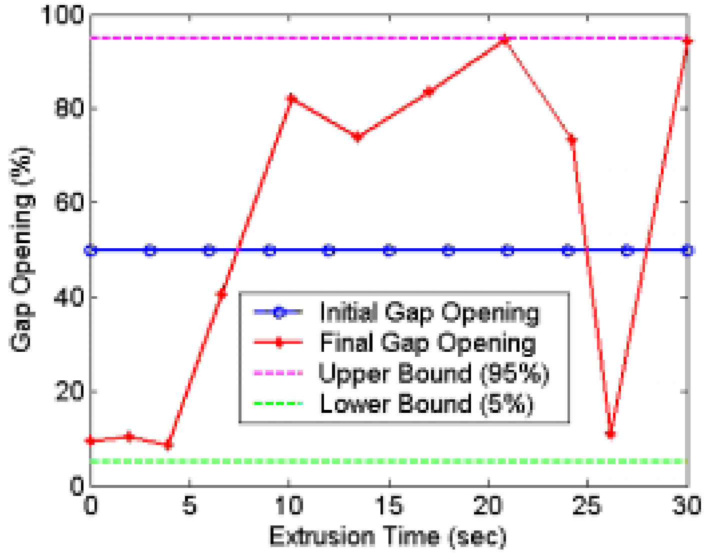
Table 4. Domain parameter mapping for the gas tank example

Domain Parameters	Die Geometry Optimization	Die gap opening Optimization
Y	5mm	5mm
$\bar{y}_{i,\max}$	10mm	10mm
$\bar{y}_{i,\min}$	0mm	0mm
$x_{i,\max}$	40mm	95%
$x_{i,\min}$	2mm	5%

In the jerry can example, 17 die points were used, GapMin was fixed at 2mm, and the die gap openings were kept at 50% during the die geometry optimization. The die geometry optimization process terminated after 5 iterations. Die gap opening optimization is then applied to the jerry can example using the final die geometry shown in Figure 14(a). Eleven programming points were used. The die gap opening optimization terminated after 20 iterations. Figure 14(b) shows the final die gap openings. The objective function value drops from 0.84 of the initial design to 0.62 of the final design, and a total of 5+20=25 BlowSim simulations were needed.



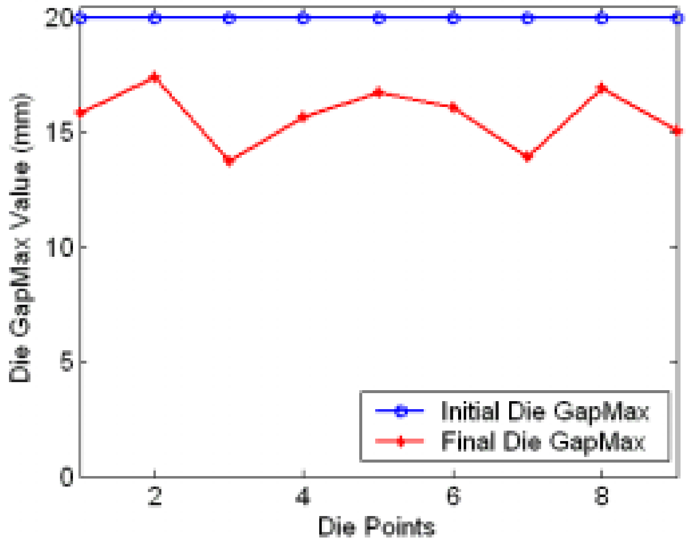
(a) Initial and final GapMax



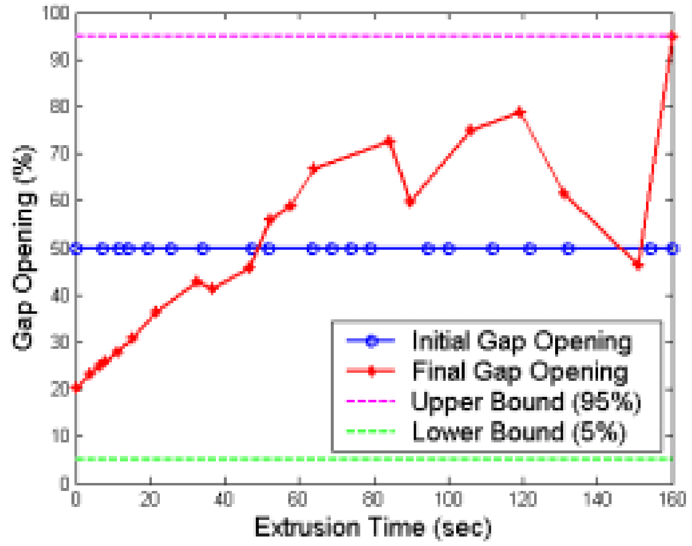
(b) Initial and final die gap openings

Figure 14. Die geometry and die gap opening optimization results for the jerry can case

In the gas tank example, 8 die points were used, GapMin was fixed at 2mm, and the die gap openings were kept at 50% during the die geometry optimization. The die geometry optimization process terminated after 11 iterations. Figure 15(a) shows the initial and final die geometry. Die gap opening optimization is then applied to the gas tank example using the final die geometry shown in Figure 15(a). Twenty programming points were used. The die gap opening optimization terminated after 13 iterations. Figure 15(b) shows the final die gap openings. The objective function value drops from 1.89 of the initial design to 1.32 of the final design, and a total of 11+13=24 BlowSim simulations were needed.



(a) Initial and final GapMax



(b) Initial and final die gap openings

Figure 15. Optimization results for the gas tank case

6. Conclusions

This paper presents a fuzzy optimization algorithm for determining the optimal die gap openings and die geometry in the blow moulding process. This fuzzy optimization algorithm has been integrated with the computer simulation software BlowSim, and has been tested on a number of blow moulding examples. The fuzzy optimization algorithm does not require sensitivity information and are completely external to BlowSim. Using a set of user-defined parameters, it is shown that a single fuzzy engine can perform die gap opening optimization and die geometry optimization for various cases. This characteristic makes the fuzzy optimization algorithm easily expandable to the integration with other simulation software in other application domains.

Comparing to traditional numerical optimization process, the fuzzy optimization algorithm tries to utilize engineering heuristics and is closer to the engineering decision process. The structure of an optimization algorithm is still maintained to guide the engineering decision process and to ensure that an optimal solution rather than a trial-and error solution can be obtained.

7. Acknowledgement

This research was supported by the National Science Council, Taiwan under grant No. NSC91 2212-E-155-007, and by National Research Council, Canada under grant No. NRC 75-05K-01.

References

1. **Laroche, D.**, **DiRaddo, R. W.**, and **Aubert, R.**, “Process modeling of complex blow-moulded parts,” *Plastics Engineering*, 1996, **52**(12), p. 37-39.

2. **Laroche D.**, **Kabanemi K.**, **Pecora L.**, and **DiRaddo R**, “Integrate numerical modeling of the blow moulding process,” *Polymer Engineering & Science*, 1999, **39**(7), p.1223-1233.

3. **Diraddo, R.W.** and **Garcia-Rejon**, A., “Profile optimization for the prediction of initial parison dimensions from final blow moulded part specifications” *Computers & Chemical Engineering*, 1993, **17**(8), p751-764.

4. **Lee, D. K.** and **Soh, S. K.**, “Prediction of optimal perform thickness distribution in blow moulding,” *Polymer Engineering and Science*, 1996, **36**(11), p. 1513- 1520.

5. **Arakawa, M**. and **Yamakawa, H.**, “Study on the optimum design applying qualitative reasoning,” Transactions of the Japan Society of Mechanical Engineers, Part C, 1990,**56**(522), Mar, p.398-403.

6. **Hsu, Y. L.**, **Lin, Y. F.** and **Guo, Y. S.**, "A Fuzzy Sequential Linear Programming Algorithm for Engineering Design Optimization," *Design Engineering Technical Conferences*, ASME 1995, **1**, p. 455-462.

7. **Mulkay, E. L.** and **Rao, S. S.**, "Fuzzy Heuristics for Sequential Linear Programming," *Journal of Mechanical Design*, 1998, **120**, p. 17-23.

8. **Arabshahi, P**., **Choi, J. J.**, **Marks, R. J. II**, and **Caudell, T. P.**, "Fuzzy parameter adaptation in optimization: some neural net training examples," *IEEE Computational Science & Engineering*, 1996, **3**(1), p 57-67.

9. **Vanderplaats, G. N.,** Numerical Optimization Techniques for Engineering Design, McGraw-Hill, Inc., 1993.