Available online at www.sciencedirect.com

ScienceDirect

Composites: Part A 38 (2007) 2076–2085

composites Part A: applied science and manufacturing

www.elsevier.com/locate/compositesa

Optimising the flow pipe arrangement for resin infusion under flexible tooling

Josef F.A. Kessels [a,*], Attie S. Jonker [a], Remko Akkerman [b]

[a] *Department of Mechanical Engineering, North-West University, Private Bag X6001, Potchefstroom 2520, South Africa*
[b] *Faculty of Engineering Technology, Production Technology Group, University of Twente, P.O. Box 217, 7500AE, Enschede, The Netherlands*

Received 21 November 2006; received in revised form 11 April 2007; accepted 11 April 2007

Abstract

A method is presented to optimise the flow pipe arrangement for the RIFT process for complex $2\frac{1}{2}$D geometries. To this end, a mesh distance-based model is coupled to a genetic optimisation algorithm. The mesh distance-based model is based on the assumption that the resin fills the nodes closest to the inlets first. It was verified with cases known from literature and with the results from a physically based flow model. The genetic algorithm provides a stable and effective optimisation method. A variable crossover rate increased its effectiveness. Depending on the choice of fitness function, the method can be used to optimise the different production parameters such as flow pipe position and length, fill distance and number of vents.

Keywords: A. Preform; C. Computational modelling; E. Resin flow

1. Introduction

Resin infusion under flexible tooling (RIFT) is a method to manufacture good quality composite products with low investment costs. Dry fibre mats (or a preform) are draped into/onto a female/male mould and then covered by a flexible plastic sheet (bag). The mould and bag are sealed and put under vacuum. The resin is drawn into the mould by this vacuum and impregnates the preform. A sketch of the process is depicted in Fig. 1.

Nowadays, many manufacturers of large composite structures, such as wind turbine blades and boat hulls, use RIFT. For these large structures, flow enhancement structures are normally used to speed up the process. Common examples are a coarse infusion mesh and spiral bind infusion pipe (top right of Fig. 1), which are placed on top of the preform. Also flow enhancement layers which are integrated in the preform, can be used. However, these cannot be removed from the final product and thus come with a weight penalty. Large components can be infused in a relatively short amount of time, compared to hand lay-up, with the help of these flow enhancement structures. Brouwer et al. [1] and Gunnarsson [2] presented some very impressive examples, e.g., an infusion of an 11.8 m boat hull in only 195 min with 340 kg of resin.

The use of the infusion mesh is straightforward: nearly the entire top surface of the preform can be covered. Because the flow through the infusion mesh is much faster than in the preform, care has to be taken that the resin gets time to flow through the thickness. If this is not the case, the resin in the infusion mesh can arrive at the vent prematurely, causing a dry spot under the vent location, as demonstrated by Hsiao et al. [3].

The great effectiveness of the use of flow pipes can be explained by comparing the permeability of a pipe with the permeability of a preform. The effective permeability of a pipe can be found by using Poiseuille's law. According to this law, the steady volume flow rate Q through a pipe with a constant circular cross-section radius r, a resin viscosity μ and a pressure gradient ∇P can be written as [4]:

[*] Corresponding author.
E-mail address: mgijfak@puk.ac.za (J.F.A. Kessels).

1359-835X/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compositesa.2007.04.008

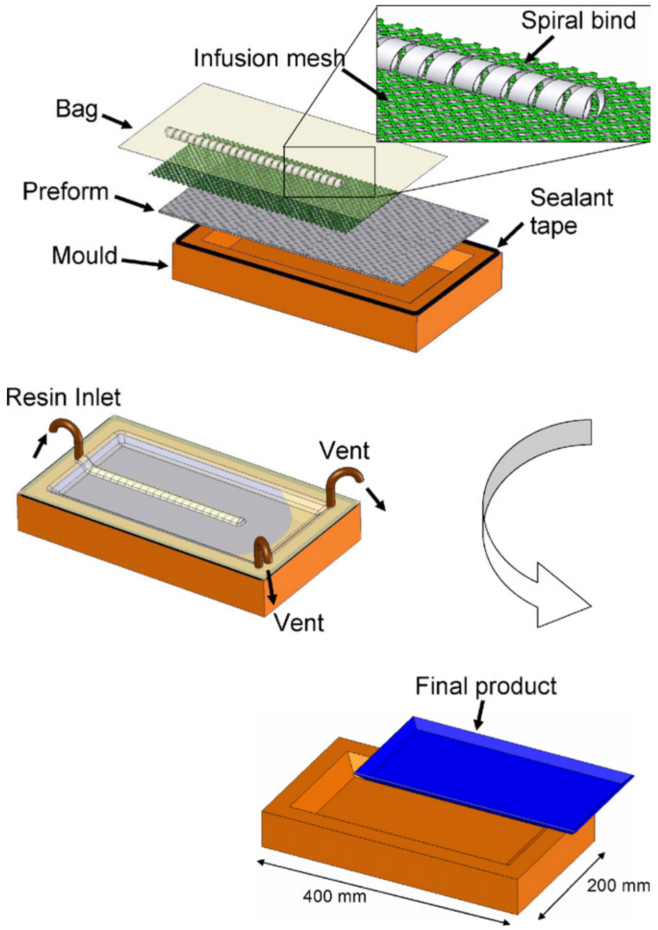


Fig. 1. Schematical representation of the RIFT process with flow enhancement structures.

$$Q = \bar{u}\pi r^2 = \frac{\pi r^4}{8\mu}\nabla P \qquad (1)$$

The flow in the preform can be described using Darcy's law [5]. According to this law the relation between the local resin flux density (also called superficial velocity), $\bar{u}$, the isotropic preform permeability K, the resin viscosity μ and the resin pressure gradient ∇P_{r}, can be written as

$$\bar{u} = -\frac{K}{\mu}\cdot\nabla P_{\mathrm{r}} \qquad (2)$$

Combining both laws gives an effective permeability for a pipe with radius r:

$$K = \frac{r^2}{8} \qquad (3)$$

Consider a pipe with $r = 12\times10^{-3}$ m, it has an effective permeability of $K = 1.8\times10^{-5}$ m^2. As presented by Kessels et al. [6], a preform of 10 layers of 280 g/m^2 Twill Weave (similar to Interglas 92125) has a permeability in the uncompressed state of 6×10^{-10} m^2. Even preforms with integrated flow enhancement layers and an optimal flow behaviour like CoreTEX have an uncompressed permeability of 1×10^{-8} m^2.

When these permeability values are compared it is clear why flow pipes are such an effective way of reducing the infusion time. A disadvantage of the use of flow pipes is that the arrangement of the flow pipes is not always straightforward as, for example, the infusion mesh. An incorrect pipe/vent arrangement can lead to dry spots in the final product. Therefore, a method was developed which optimises the flow pipe, inlet and vent position in advance.

2. Previous modelling effort

The number of developed optimisation tools for the RIFT process is significantly lower than those for the RTM process. In RIFT the preform compacts (due to the flexible bag) during the process, influencing the permeability of the preform. In RTM both moulds are solid and hence mould filling times will be different for the RTM and RIFT process under similar conditions (e.g., same resin, pressure and preform). Nevertheless this preform compaction will not be taken into account here, since it does not influence the optimisation significantly. Acheson et al. [7] showed that the RIFT process can successfully be predicted with RTM models (which do not include preform compaction) by using an "effective" permeability, which accounts for the change in permeability due to compression. Therefore, the relative difference in computed mould filling times for two different scenarios will be the same whether the preform compaction is included or not. For this reason, some RTM optimising models will also be discussed.

Most researchers focussed on the optimal location of the inlet and vents. Cai [8] was one of the first to look at these optimal locations and came up with some useful closed form solutions for the wet length, mould filling time and pressure distribution of rectangular, trapezoidal and circular sections. Two years later Young [9] published an algorithm to optimise the inlet location on any $2\frac{1}{2}$D geometry. A previously developed non isothermal flow simulation program was coupled with a genetic search algorithm. Thin film part assumptions were used for the simulation program and, for example, the resin flow in the thickness direction was neglected. Therefore, this model, although describing 3D geometries, is called an $2\frac{1}{2}$ dimensional ($2\frac{1}{2}$D) flow model. The disadvantage of using a physically based flow model was also shown: The calculation time of 600 generations with a population size of 30 on a 448 element model was over 75 h.

Boccard et al. [10] addressed the issue of excessive calculation times and therefore presented a fast geometrically based model to determine the location of the vents on flat (2D) RTM moulds. However, the major drawback of their model was, that it was limited to 2D shapes and the calculation for complex parts was very difficult [11].

Jiang et al. [11] used a genetic algorithm to, again, optimise the location of the inlet and vent with a mesh distance-based approach model. Such a model is fast and can accommodate any type of mesh (e.g., the mesh of a structural analysis). The model could only optimise 2D geometries. Although not mentioned explicitly it was shown how ineffective a genetic algorithm can be. In a case where one inlet was allowed on a model with 930 nodes, it took the algorithm 1000 trials. It would have been faster to try all 930 nodes successively. However, optimising times in order of minutes instead of hours, like Young [9], were achieved, with the mesh distance-based model.

Hsiao et al. [3] were some of the few researchers who tried to optimise the RIFT process. They used RTM software and a genetic algorithm to find the optimum for the diameter of the flow runner channels and the amount of layers of a flow distribution medium. Because these values could be zero, solutions without these flow enhancement structures could also be found. Good agreement between simulation and experiments was achieved and the unwetted area was reduced by optimising the size of the flow enhancement structures. They focused only on one design case where the position of the flow channels/pipes and distribution medium were fixed and determined by the user in advance.

Especially with larger and more complex structures the optimal position of the flow pipes is not always straightforward and the presented model here will therefore optimise the inlet/vent position as well as the arrangement of flow pipes.

3. Method of optimisation

Before the optimisation model was developed, a couple of assumptions were made. Firstly, preform compaction is neglected, an isotropic preform permeability is assumed and there is no pressure gradient in the z direction. The latter restricts the model to preforms with a uniform flow through the thickness. There are two situations where this occurs: The different plies have a uniform permeability over the thickness (like the aforementioned 10 plies of twill weave) or the flow is dominated by the layer with the highest permeability. An example of the latter are preforms which consist of a thick flow enhancement core which is covered by just a few single plies of (woven) fabric. In such a preform, the single plies of fabric are wetted instantaneously as the resin reaches the underlying core and hence the flow can be assumed to be uniform [12]. The CoreTEX fabric is an example of such a preform.

The model is also based on the assumption that the resin first fills the nodes that are closest to the inlet, then the next closest, etc. Although locally true, Cai [8] showed that this assumption is not globally true for the RIFT or the RTM process. If different types of flow exist (radial or linear/channel), it is possible that a node further away from the inlet is filled earlier than a closer one. Still it is valid to make this assumption, when the objective of the optimisation is only to minimise the distance between inlet and vent (and hence minimise fill time) and an accurate estimation of the fill time is of second order importance. This distance between inlet and vent is defined here as the fill distance.

Based on these assumptions, a mesh distance-based model was developed to simulate the process, based on a given inlet/vent and flow pipe arrangement. The mesh distance-based model is similar to the one presented by Jiang et al. [11], but with an extension to the use of $2\frac{1}{2}$D geometries. It is much faster and hence better suited for optimising purposes than physically based flow models. The input parameters of the mesh distance-based model are a meshed surface representation of the part and the flow pipe arrangement. It is assumed that the inlet is at the first node of the flow pipe. The output parameters of the mesh distance-based model are:

- x: the fill distance;
- N_{vents}: the number of the vents;
- N_{pipes}: the number of nodes on the pipe (pipe nodes).

The number of pipe nodes, N_{pipes}, is being used as an estimate for the relative pipe length, instead of calculating the flow pipe length precisely. This is much faster and gives a fair indication as long as the lengths of the element edges are more or less uniform.

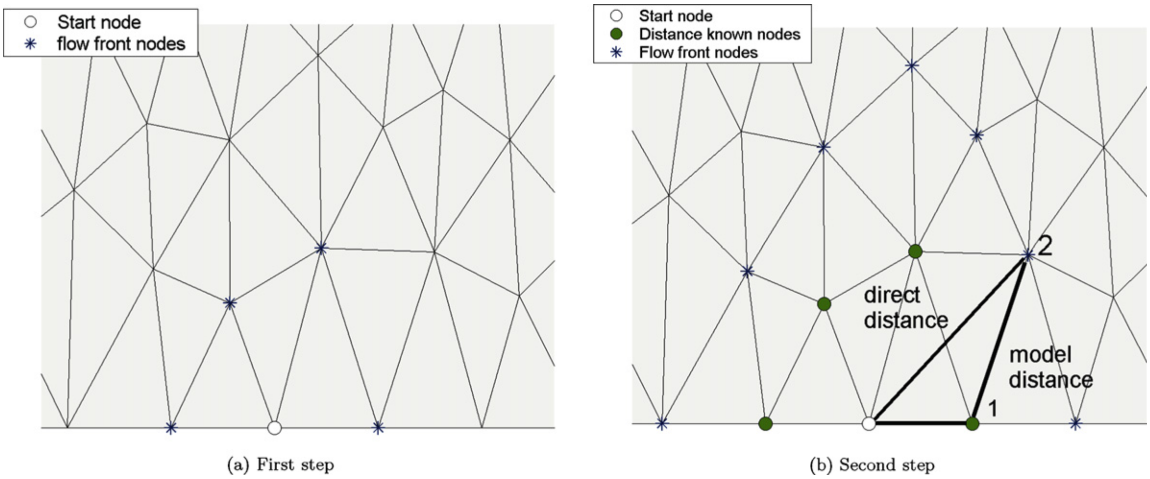
In the next sections, the calculation of these output parameters will be discussed.

4. Calculation of the fill distance

For each vent the calculations of the fill distance are straightforward:

1. The neighbour nodes of the inlet(s) are identified as flow front nodes and the inlet nodes are defined as distance known nodes with distance 0.
2. The distance of the flow front nodes to the distance known nodes is calculated. In case of a triangular mesh, as used in this research, the distance between two neighbouring nodes equals the edge length of the element they both belong to.
3. The minimum of the calculated distances for every flow front node is taken as the distance from this node to the inlet.
4. The distance of the flow front nodes is now known and therefore these nodes are defined as distance known nodes. Their neighbours are defined as flow front nodes.
5. With this new set of flow front nodes the calculation continuous with step 2 until the distance of all nodes in the model are known.

This process is depicted in Fig. 2. Significant errors can occur between this calculated distance and the geometrical, direct, distance. Looking at node 2 of Fig. 2, it can easily be seen that the distance directly from the start node and the calculated distance, using the interlying node 1, differ quite a lot. This mesh-structure dependent error should be borne in mind when creating the mesh for a model. Ideally, every



(a) First step

(b) Second step

Fig. 2. Calculation of the distance from the start node.

node in the mesh should have connections to the other nodes in every quadrant and at different angles than its surrounding nodes. Modern preprocessors, such as MSC-PATRAN as used here, have the possibility to generate such meshes, which reduces this error significantly.

On 2D parts, the direct distance could be used, like Jiang et al. [11] did, but it would restrict the model to the use of geometrically described surfaces. Techniques have been developed to compute geodesics between two points on triangular meshes (e.g., the algorithm presented by Martinez et al. [13]), but are, since they are based on iterative schemes, time consuming and would reduce the advantages of using a mesh distance-based model instead of a physically based flow model.

The calculated distances by the mesh distance-based model were verified with the geometrical, direct, distances for several geometries, similar to the cases presented by Jiang et al. [11]. In general, the relative error between the geometrical distance and the mesh based distance for a certain node decreases if the amount of nodes between the inlet and that node increases. At the points furthest away from the inlet (they are used to position the vents and are only of interest), appropriately small errors (3%) were obtained.

5. Definition of the flow pipes

If the model contains flow pipes, these have to be defined in advance. Here, the flow pipes are defined by their start and end node. The path of the flow pipes is defined by the shortest path along the element edges between the start and end node. This path is found with the same routine to calculate the fill distances. The start node is defined as the inlet and for every node the shortest distance to this inlet is stored but also a ‘‘bread-crumb’’ containing the neighbouring node ID where it gets its shortest distance from. For example, assuming that the start node in Fig. 2b is the beginning of a flow pipe, then for node 2, its shortest distance to the start node is being stored, but also where it gets its distance from: in this case node 1.

At the end of the calculation routine this trail of bread-crumbs is traced back, starting at the end node and ending at the beginning of the pipe. This gives the shortest distance (along the element edges) from the end node to the start node, but also gives the intermediate nodes on the pipe (pipe nodes). For example, assuming, again, that the start node in Fig. 2b is the beginning of a flow pipe and node 2 is the end node of the flow pipe, then the trail of bread-crumbs will go from node 2 to node 1 and end at the start node.

As soon as all the pipe nodes are known, the fill distance for every node can be calculated, using the routine presented in the beginning of this section. In order to accommodate the higher permeability of the pipe compared to the permeability of the fibrous preform (in this case a factor 1×10^3), the distance between every pipe node is divided by the same factor.

6. Positioning of the vents

Once the fill distance of every node is known, the position of the vents is determined. Vents should be placed in such a way that all the air can evacuate during the process and no air gets trapped in a region. This is ensured if the vents are positioned at the point which is last filled in a certain region. For the model, this can be interpreted such that the vents should be positioned at the nodes which have a higher fill distance than all their neighbouring nodes (local maxima). This reduces the determination of the vent to a simple matrix operation. This method of vent determination was verified with the cases presented by Boccard et al. [10] and Jiang et al. [11] and showed 100% agreement.

7. Objective function

For a given pipe arrangement, the number of vents and the maximum fill distance are calculated using the method described above. These output parameters are coupled to the process properties, as process costs and process time, using weighting functions. With these weighting factors, the output parameters are fitted into an objective function. A genetic algorithm (GA) is being used for the optimisation of this objective function, since the design space is non-uniform and discrete. In genetics, the objective function is called the fitness function. The fitness function used here to calculate the fitness value F for each scenario is:

$$F = \frac{w_{\text{distance}}\left(1 - \frac{x}{x_{\max}}\right) + w_{\text{pipes}}\left(1 - \frac{N_{\text{pipes}}}{N_{\text{nodes}}}\right)^4}{P_{\text{vents}} \cdot P_{\text{distance}}} \tag{4}$$

where

$$\sum w = 1 \tag{5}$$

In Eq. (4), w are the weight functions with, respectively, w_{distance} for the fill-distance and w_{pipes} for the number of pipe nodes. N_{nodes} is the number of nodes in the model and hence the maximum number of pipe nodes and vents. Because the total number of nodes in the model is usually much larger than the number of pipe nodes, N_{pipes}, the term $N_{\text{pipes}}/N_{\text{nodes}}$ is not very sensitive to a change in the number of pipe nodes. A 4th power was therefore applied to the pipe evaluation in order to magnify a change in the number of pipe nodes. Hsiao et al. [3] also used this 4th power to distinguish the good results from a batch of results more clearly.

The parameter, $x_{\max}$, is the maximum calculated distance (by the mesh distance-based model) between any two nodes in the model.

The variables P_{vents} and P_{distance} act as a penalty in case an individual has, respectively, more vents or a higher fill distance than allowed. For example, in case an individual has more vents, N_{vents}, than the maximum allowable number of vents, N_{a}, its fitness should be penalized and hence P_{vents} was defined as:

$$P_{\text{vents}} = \max\left\{\begin{matrix} N_{\text{vents}} - N_{\text{a}} + 1 \\ 1 \end{matrix}\right\} \tag{6}$$

The optimum of the fitness function is 1, when there is only one vent, no pipe is being used and the fill distance is 0.

The values for the weight functions are chosen according to the objective of the optimisation and the desired optimum process properties. If a short fill time and hence fill distance is important, the weight function w_{distance} will be larger than in the case where the number of vents has to be reduced or the costs of consumables has to be reduced.

8. Genetic algorithm

As already mentioned, a genetic algorithm is being used for the optimisation of the fitness function. The basic principles of genetic algorithms (GAs) were proposed by Holland [14]. It is based on the mechanism of natural selection and natural genetics. The combination of design parameters is represented by a single bit string, analogous to the genes of a chromosome. Several of these bit strings, also called individuals, generated by the different combinations of design parameters, form a population. Here, each population consisted of 20 of these individuals. The degree of "goodness" or how well the individual fits into the environment is represented by a fitness value. Throughout a genetic evolution, the fitter chromosome has a tendency to yield better quality offspring, meaning a better solution to the problem. Through natural selection and reproduction, the population improves and only those who fit in the environment best (highest fitness value) will survive and represent the optimal solution. The main strength of GAs is that they are robust, can deal with a wide range of problem types and generally produce global optimal solutions in a large search space.

A more detailed description of GA can be found in [15,16]. The GA developed here, uses the Roulette Wheel Scheme for selection and simple crossover and one point random mutation for reproduction. The start and end nodes of the flow pipe (which are the input parameters for the mesh distance-based model) are the chromosomes of every individual. The probability of mutation was set to 0.2. The probability of crossover ($P_{\text{crossover}}$) is normally also a fixed value. The problem with a fixed crossover probability, is that after the population evolves it becomes quite homogeneous and offspring produced by crossover are becoming clones instead of new samples. Booker [17] presented a variable crossover rate, depending on the spread of fitness. When the population converges, the crossover rate is reduced to give more opportunity for mutation to find new variations. DeJong and Spears [18] showed that having such an "adaptive" crossover operator enhances long term performance significantly. Here, the crossover rate varied linearly from 1 to 0.3, depending on the spread of the population.

Due to crossover and mutation, it is possible that the best individual is not conserved. Therefore, the best individual is reinserted after 20 steps, if in these 20 steps no individual had a better fitness than this best individual.

The GA was terminated if the solution converged or the maximum amount of iteration steps was achieved. The solution was considered to be converged, if the fitness of the best individual did not improve since the last 200 steps. The maximum amount of iteration steps was set to 1000. The results presented in the following section will show that this number was never reached.

9. Design cases

The combination of the mesh distance-based model and the genetic optimisation algorithm was implemented into the MATLAB programming environment. The effectiveness of the developed optimisation tool was validated with a number of design cases. Two of these cases, a flat rectangular plate and a glider seat will be presented here.

A flat rectangular plate, with dimensions 0.6 × 1.0 m, was used to examine the influence of the fitness functions.

Furthermore, the optimal solution for this plate was easier to recognize, since it is a 2D product. The model of the plate was meshed with 1581 nodes and 3000 first order triangular elements.

Three scenarios were simulated. For the first two scenarios, the objective of the optimisation was to minimise the maximum fill distance. For the last scenario the amount of consumables had to be minimised as well.

For the first scenario, the optimisation tool had to find the minimum fill distance (thus maximising the fitness function F of Eq. (4), with $w_{\text{distance}}=1$ and $P_{\text{distance}} = 1$) using only one pipe. The pipe was defined by only its Begin- and End node ID-number. One vent was allowed and hence in Eq. (6), $N_a = 1$.

Convergence was reached after 17 min on a 2.01 GHz PC with 512 MB of RAM. The best solution, after convergence was reached, is presented in Fig. 3. In this figure, and also in the following figures, $\circ\circ\circ$ are pipe nodes, $\otimes$ marks the position of a vent and $\odot$ is the beginning of the flow pipe and inlet. The intensity of the gray scale represents the fill distance (m).

The solution presented in Fig. 3a confirms the obvious result and corresponds with the design rules of Cai [8]. One would perhaps expect that the pipe should be positioned on the centerline of the plate (as depicted in Fig. 1), since this would result in the minimum fill distance for only one pipe. However, since the resin would spread from this centerline to the sides, 2 vents would be required. Fig. 3b shows the maximum, minimum and average fitness of the population during the optimisation cycles. After 117 generations the optimal solution was found. Every generation consisted of 20 individuals, thus 2340 individuals (or iteration steps) were calculated to obtain the optimal solution. With 1581 nodes in the model, there were 1,249,780 possibilities to position this pipe. This shows that the genetic algorithm is over 500 times more effective than trying all possibilities successively. However, as shown in Fig. 3b, after the optimal solution was found, the algorithm calculated another 200 generations to ensure convergence and the solution obtained was the optimal solution. This resulted in a total of 318 generations (6360 individuals), which is still almost 200 times less than the total number of possibilities.

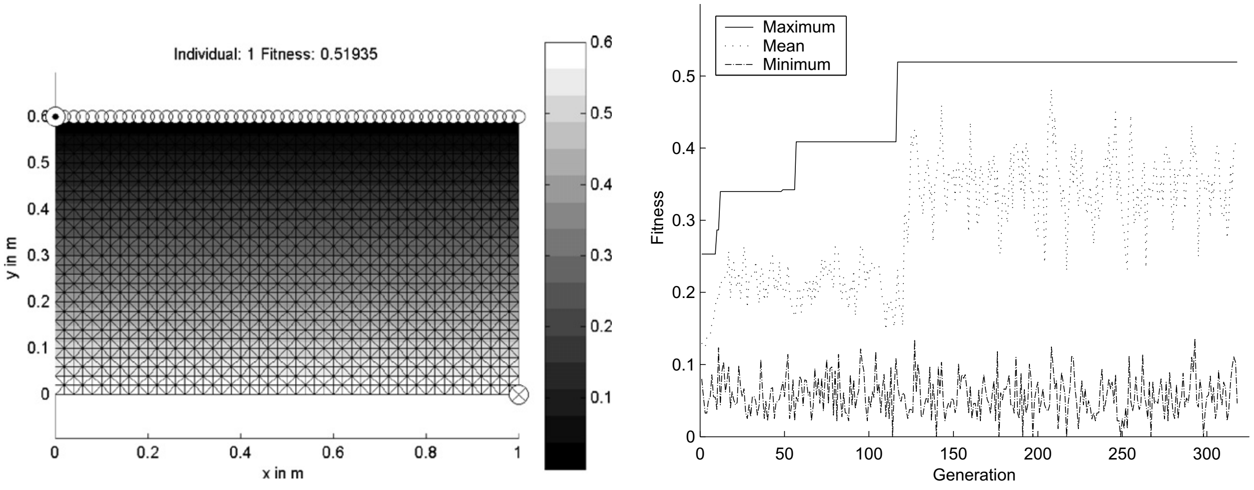
The objective of the second scenario was still to minimise the maximum fill distance. However, a total of 3 pipes and 2 vents were allowed on the product. The first pipe was defined by an extra middle point and two other pipes had to connect to this main pipe. The same fitness function and weighting factors were used, but in order to allow 2 vents on the product $N_a = 2$.

The results after convergence are presented in Fig. 4. The algorithm needed about the same number of iteration steps (2640 individuals) to come to convergence compared to the first scenario, although the number of possibilities for this scenario is much larger. The figures show only one case. Therefore, the first scenario was simulated 50 times, as well as the second scenario. On average the first scenario converged after 326 generations and the second scenario after 353 generations.

The result for this scenario also confirmed the obvious solution but also gave rise to the question: Is this the optimal solution with the minimum number of pipe nodes required? It can be expected that for example, the middle pipe could be shorter without reducing the maximum fill distance of 0.14 m.

Therefore, the objective of the third scenario was to minimise both the fill distance and the flow pipe length. The same boundary conditions (number of pipes and vents) applied as for scenario 2. By reducing the pipe length, the amount and cost of consumables as well as the amount of resin needed to fill the product will be reduced.

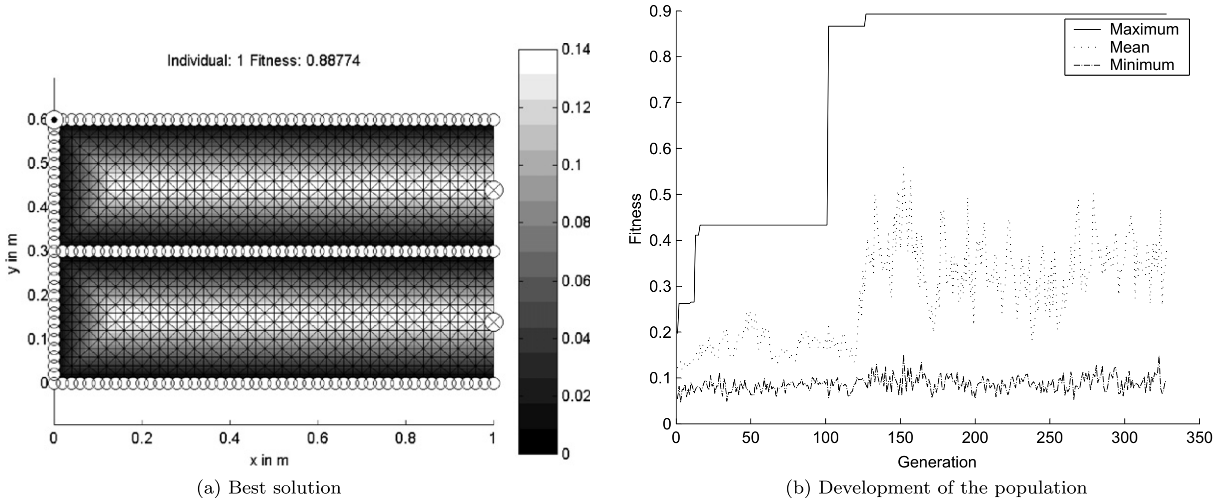
If a reduction of consumables and costs is desired, the genetic algorithm can easily be adjusted to minimise the flow pipe length, simply by modifying the weighting factors



(a) Best solution

(b) Development of the population

Fig. 3. Optimising the fill distance if 1 vent and 1 pipe are allowed.



(a) Best solution

(b) Development of the population

Fig. 4. Optimising the fill distance if 2 vents and 3 pipes are allowed.

of the fitness function. The weighting factors w_{distance} and w_{pipes} were, respectively, set to 0.1 and 0.9 in order to emphasise the optimisation of the flow pipe length.

The penalty function, P_{distance}, was defined in such a way that it reduced the fitness if the fill distance was larger than the allowable fill distance, which was arbitrarily set to 0.2 m.

$$P_{\text{distance}} = \begin{cases} 10, x > 0.2 \\ 1, x \leqslant 0.2 \end{cases} \tag{7}$$

The result with the highest fitness factor after convergence is depicted in Fig. 5. All three horizontal pipes were shorter and the length of the flow pipes was reduced by 0.34 m compared to the previous case. The maximum fill distance was 0.2 m, as could be expected due to the penalty function.

The final design case was a pilot seat for a glider. Where the previous design cases were only 2D, this product was 3D and had a more complex shape. The seat is basically a bath-tub shape with a back rest and armrests on the side. Opposite the back rest there are two leg rests with a gap in between for the control stick. The arm rests are on different levels, thus the product is not symmetrical. The CAD model is depicted in Fig. 6a. This CAD model was meshed with MSC-Patran using 1463 first order triangular elements and 807 nodes. The meshed finite element model is presented in Fig. 6b.

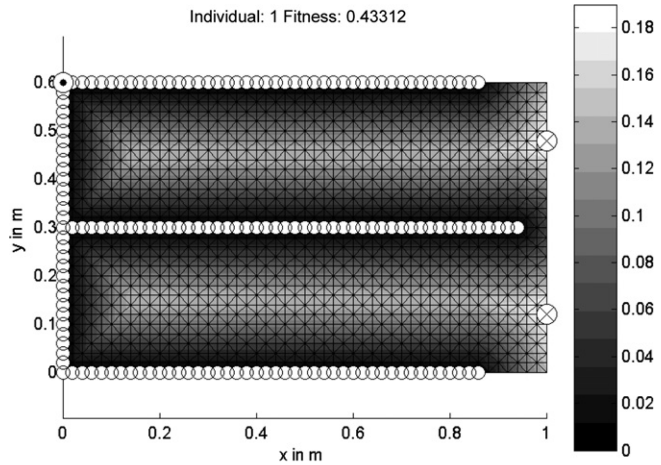
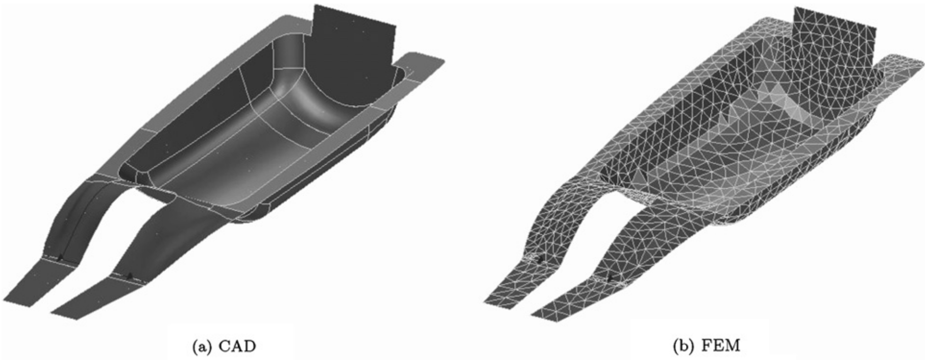


Fig. 5. Optimising the fill distance and the amount of consumables if 2 vents and 3 pipes are allowed.

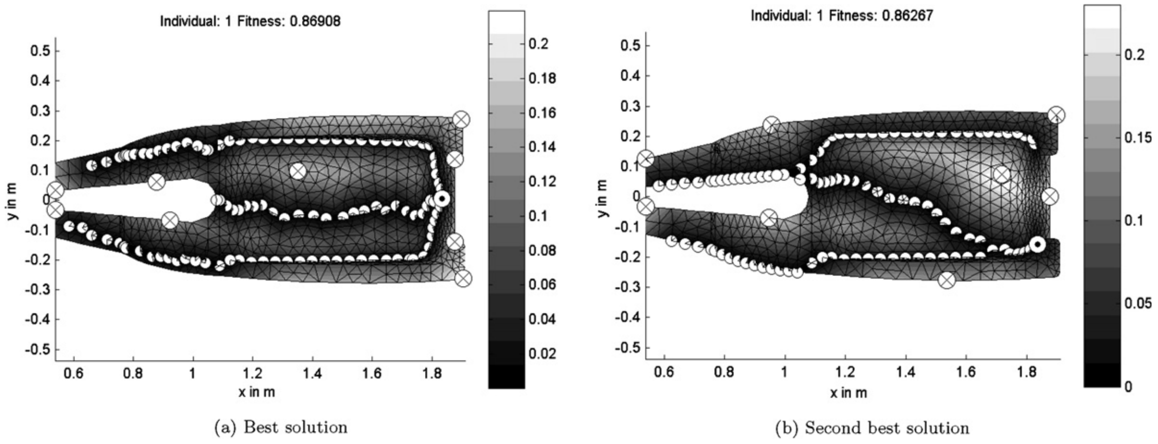
The seat consists of a combination of woven carbon- and Kevlar fibres and production numbers are only a few per year. Since the permeability of this carbon/Kevlar



(a) CAD

(b) FEM

Fig. 6. The CAD and finite element model of the glider seat.



(a) Best solution

(b) Second best solution

Fig. 7. Optimal pipe position on a glider seat if 3 pipes are allowed.

combination is quite low, infusion times and thus the maximum fill distance had to be as short as possible. The number of vents was kept free ($N_a = \infty$) and the fitness function of Eq. (4) was used with $w_{\text{distance}} = 1$ and $P_{\text{distance}} = 1$. Due to the two leg rests, it was clear that at least two pipes should be used, otherwise one of the two leg rests would not contain a pipe which will increase the fill distance by at least the full length of the leg rest ($\approx$0.5 m). Therefore, three pipes were allowed.

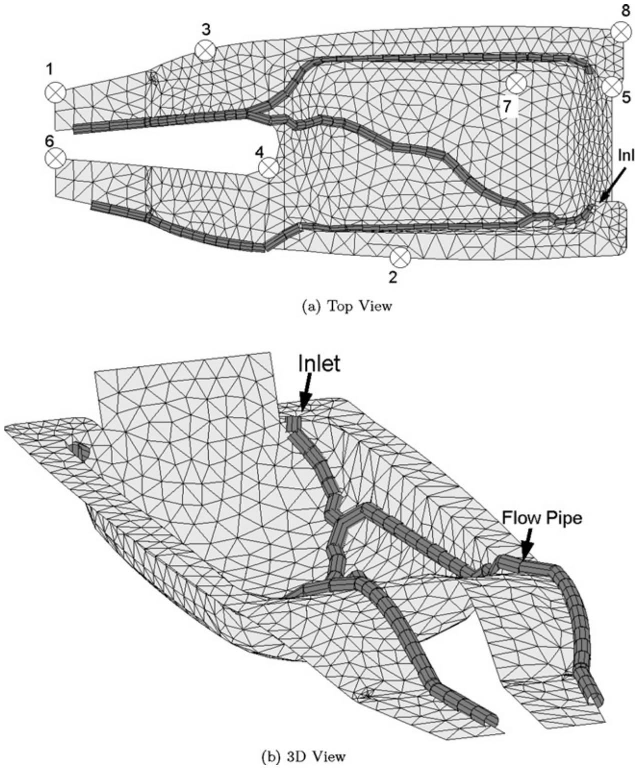
Convergence was reached after 551 generations and 14 min of calculation time. The solutions with the best and second best fitness after convergence are depicted in Fig. 7. The maximum fill distance was 0.21 m. The solution with the highest fitness required a total of 9 vents, whereas with the solution with the second best fitness only 8 vents were necessary. Therefore, the second best solution was favoured over the best solution.

10. Verification with a physically based flow model

Fig. 7b shows the required position of the vents for this flow pipe set up. Since the mesh distance-based model is only geometrically based, the position of the vents were verified with a simulation, using a physically based flow model. Here, the flow model presented by Kessels et al. [6] was used, without the preform compaction, but with an extension to accommodate the 1D flow pipes.

Fig. 8a shows the meshed model of the seat with the flow pipes and Fig. 8b shows a 3D view. The resin viscosity and the isotropic preform permeability were both normalised to unity ($\mu = K = 1$). The flow pipe had a permeability which was a factor 1000 higher than the rest of the preform. This is the same factor as used by the genetic based optimisation tool.

The simulation of the mould filling with the physically based flow model took 4 min. The simulated propagation of the flow front is depicted in Fig. 9, where t_n is the normalised time, which is defined by the time at that moment, t, divided by the total filling time, t_{total}.



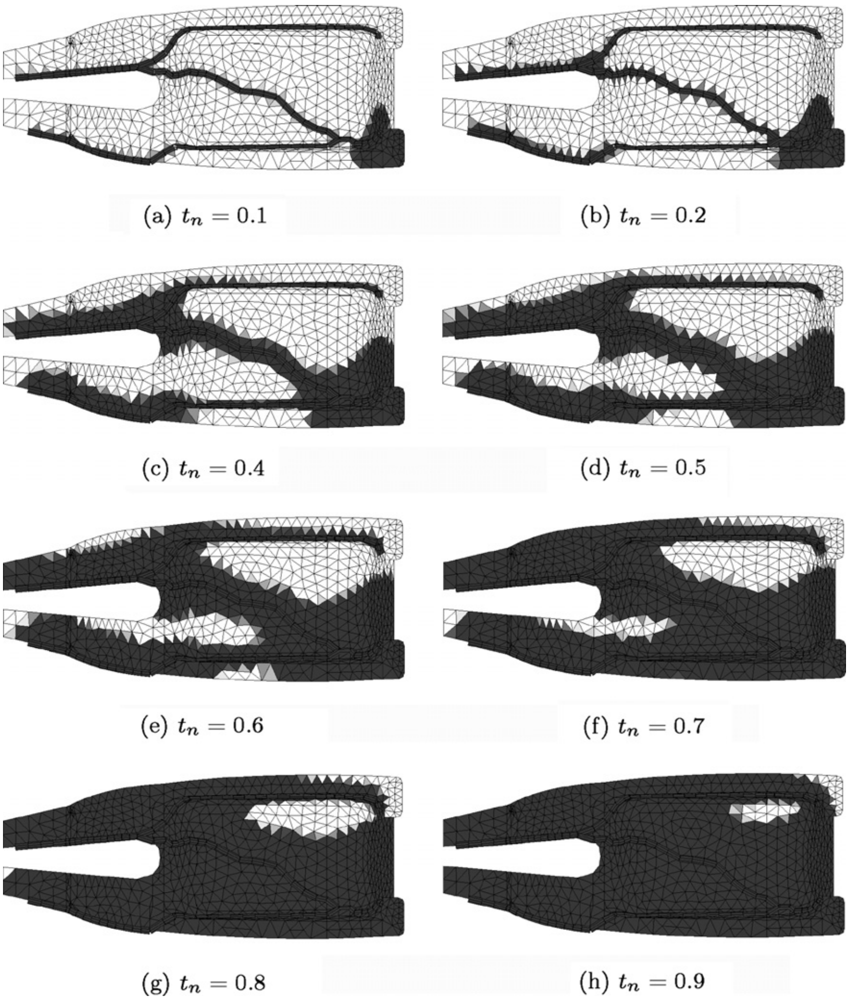
(a) Top View

(b) 3D View

Fig. 8. The flow pipe and inlet position and predicted vent position calculated by the flow model.

The prediction of the position of the vents by the physically based flow model can be derived from the flow-front propagation by looking at the nodes which are not filled yet, while all their neighbouring nodes are already filled. At these nodes, vents should be positioned, as presented in Fig. 8a.

The number of vents and the area where these vents had to be positioned agreed quite well with the results from the mesh distance-based model. Only the exact positions differed slightly compared to the predicted positions by the genetic based optimisation model.



(a) $t_n = 0.1$

(b) $t_n = 0.2$

(c) $t_n = 0.4$

(d) $t_n = 0.5$

(e) $t_n = 0.6$

(f) $t_n = 0.7$

(g) $t_n = 0.8$

(h) $t_n = 0.9$

Fig. 9. Simulated mould filling of a glider seat at different times, where normalised time $t_n = t/t_{\text{total}}$.

11. Discussion

The flow pipe position on a rectangular plate was optimised in the first scenario of the first design case. A total of 2340 individuals were calculated to come to convergence. The best solution after convergence agreed very well with the obvious optimal solution. It also showed that the genetic algorithm was more effective than the one used by Jiang et al. [11]. They needed 3071 trials to optimise the position of 2 inlets (which is similar to a pipe defined by its begin and end point) for a 1036 node model. This is mainly due to the variable crossover rate which allowed the solution to converge faster to the optimal solution [18].

The second scenario of the first design case showed that the genetic algorithm got more effective if the number of design parameters, i.e., number of pipes, increased. The reason can be found in the nature of the genetic algorithm and the problem itself: the start and end positions of the pipes can be optimised independently from each other and. Furthermore, the solutions for the different pipes can be interchanged as well due to crossover. Hence the position for the different pipes is optimised simultaneously (parallel and not serial) increasing the effectiveness of the algorithm as the number of design parameters increases.

An isotropic preform permeability was assumed for the optimisation tool. Anisotropic permeabilities can be accounted for in the same way as the different permeability of the flow pipe. The element edge lengths can virtually be reduced or increased according to the permeability of the preform in the direction of the element edge. However, in most practical cases, woven fabrics and/or flow enhancement layers, like the infusion mesh, are being used for the RIFT process. In these cases, the preform permeability can be assumed to be isotropic and the presented model will be adequate.

The calculation of the total pipe length was simplified by taking the number of flow pipes nodes as an indication for the total flow pipe length for the third scenario. Although this assumption saved significant calculation time, it required a constant element edge length in the model. In the case of the rectangular plate, the diagonal element edges were a factor $\sqrt{2}$ longer than the horizontal and vertical ones. Therefore, it could be expected that the model would favour flow pipes along the diagonal edges since more distance is covered with fewer flow pipe nodes. In the presented case, this phenomenon did not occur which justified the simplification.

On a more complex shape, like the glider seat in the second design case, the optimal pipe position is not always obvious and this is where the developed optimisation tool becomes very useful. After the optimisation tool converges, the final population contains the optimal solution, but also a variety of sub-optimal solutions. The design case showed that these solutions, which have a fitness close to the optimal solution, can be different but still of interest. After an optimisation cycle it is therefore recommended to consider all individuals of the last population and a final solution should be selected in the light of experience. For example in this design case, the second best solution was better suited for a reason which was not included in the fitness function.

12. Conclusion

The genetic based optimisation tool presented here is based on a mesh distance-based model and a genetic optimisation algorithm. The mesh distance-based model gives a fast prediction of the fill distance and position and number of the vents and is therefore very well suited for optimisation purposes. Although differences between the geometrical distance and the calculated distances occurred close to the inlet, the error for the maximum fill distances at the vents was within an acceptable range (3%). The position and number of vents, predicted by the model, agreed 100% with the test cases known from literature.

As soon as different types of flow exist, as with the gilder seat, the predicted positions of the vents differ from the actual ones. This did not influence the optimisation process, since the number and region were still correct, but requires a final calculation with a physically based flow model to give a more accurate vent position.

The genetic algorithm provides a stable and effective optimisation method. A variable crossover rate increases the effectiveness. Depending on the choice of fitness function, the algorithm is capable of optimising the different production parameters as flow pipe position and length, fill distance and number of vents. The genetic algorithm not only provides the optimal solution but also the sub-optimal solutions, which are also of interest.

Acknowledgement

The support of AeroEnergy and Jonker Sailplanes is gratefully acknowledged.

References

[1] Brouwer WD, van Herpt ECFC, Labordus M. Vacuum injection moulding for large structural applications. Compos: Part A 2003;34:551–8.

[2] Gunnarsson AT. Icelandic boat builders switch to resin infusion. Reinforced Plast 2004(May):34–6.

[3] Hsiao Kuang-Ting, Devillard Mathieu, Advani Suresh G. Simulation based flow distribution network optimization for vacuum assisted resin transfer moulding process. Model Simul Mater Sci Eng 2004;12:175–90.

[4] Sutera SP, Skalak R. The history of Poiseuille's law. Ann Rev Fluid Mech 1993;25:1–19.

[5] Gutowski TG, Morigaki T, Cai Z. The consolidation of laminate composites. J Compos Mater 1987;21:172–88.

[6] Josef FA Kessels, Attie S Jonker, Remko Akkerman. Fully 2 1/2 d flow modeling of resin infusion under flexible tooling using unstructured meshes and wet and dry compaction properties. Compos Part A: Appl Sci Manuf, in press. Available online 15 March 2006.

[7] Acheson JA, Simacek P, Advani SG. The implications of fiber compaction and saturation on fully coupled VARTM simulation. Compos: Part A: Appl Sci Manuf 2004;35:159–69.

[8] Cai Zhong. Analysis of mold filling in RTM processes. J Compos Mater 1992;26(9):1310–38.

[9] Young WB. Gate location optimization in liquid composite molding using genetic algorithms. J Compos Mater 1994;28(12):1098–113.

[10] Boccard A, Lee WI, Springer GS. Model for determining the vent locations and the fill time of resin transfer molds. J Compos Mater 1995;29(3):306–23.

[11] Jiang Shunliang, Zang Chuck, Wang Ben. Optimum arrangement of gate and vent locations for RTM process design using a mesh distance-based approach. Compos Part A: Appl Sci Manuf 2002;33:471–81.

[12] Grimsley BW, Hubert P, Song X, Cano RJ, Loos AC, Pipes RB. Flow and compaction in the vacuum assisted resin transfer molding process. In: Proceedings of the 33rd International SAMPE Technical Conference, Seattle, Washington: November 4–8; 2001.

[13] Martínez Dimas, Velho Luiz, Carvalho Paulo C. Computing geodesics on triangular meshes. Comput Graphics 2005;29:667–75.

[14] Holland John. Adaption in natural and artificial systems. Ann Arbor: The University of Michigan Press; 1975.

[15] Man KF, Tang KS, Kwong S. Genetic algorithms,concepts and designs. London: Springer Verlag; 1999.

[16] Goldberg David Edward. Genetic algorithms in search,optimization and machine learning. Reading, Mass: Addison-Wesley; 1989.

[17] Booker L. Improving search in genetic algorithms. In: Davis L, editor. Genetic Algorithms and Simulated Annealing. Pitman; 1987. p. 61–73 [Chapter 5].

[18] De Jong Kenneth A, Spears William M. An analysis of the interacting roles of population size and crossover in genetic algorithms. In: Proceedings of the First Workshop of parallel Problem Solving from Nature. Berlin: Springer Verlag; 1990. p. 38–47.