Journal of Manufacturing Systems
Vol. 23/No. 4
2004

Heuristic Algorithm for Determining Optimal Gate and Vent Locations for RTM Process Design

Xugang Ye, Chuck Zhang (*chzhang@eng.fsu.edu*), **Zhiyong Liang,** and **Ben Wang,** Dept. of Industrial Engineering, Florida A&M University-Florida State University, Tallahassee, Florida, USA

Abstract

In the resin transfer molding (RTM) process, locations of gates and vents are important design parameters that have a great impact on the resin flow pattern and mold filling time. The resin flow pattern is crucial to the quality and properties of the final product. A good resin flow pattern can cover all dry areas in the mold with liquid resin; whereas, a poor flow pattern will lead to some dry spots, which directly result in defects. In this paper, a single gate, multiple vents location problem for 2-D RTM process design is formulated as an optimization model. A process performance index is introduced as the objective function to evaluate the flow pattern in a 2-D RTM process. The authors introduce the concept of general distance between any two locations within the mold. To avoid forming dry spots during the process, all potential vents are determined once a gate is given. A two-phase optimization method, called a graph-based two-phase heuristic (GTPH), is used to find the appropriate locations of the gate and associated vents. The results are compared with those from other methods such as genetic algorithms.

Keywords: *Resin Transfer Molding, Mold Design, Process Simulation, Optimization Technique, Heuristic Method*

Nomenclature

B	index set that denotes partition of the boundary of the mold geometry; its element, b_i, $i \in \{1, 2, \ldots, m\}$, denotes center of the ith triangle in the finite element mesh of the mold geometry boundary
$d_{i,j}$	Euclidean distance between v_i and v_j
F	control volume
$k(v_i, v_j)$	average principal permeability from v_i to v_j
$\bar{\bar{k}}$	permeability tensor (second order) (for isotropic case, $\bar{\bar{k}}$ can be replaced by its principal value, say, k)
l	general distance between centers of two adjacent finite element triangles (adjacent if and only if a common edge exists)
L	length of shortest path from x (gate) to y_j (jth vent)
M	simulation that determines the pressure profile and set of locations for the vents from the given location and the pressure of the gate
n_o	number of vents
$p_i(t_0)$	pressure profile against index of triangles of the finite element mesh of the mold geometry at time t_0
P	pressure
∇P	pressure gradient
$S(x, y, z, t)$	saturation rate at (x, y, z), which is the fraction of resin occupancy
μ	viscosity of liquid resin
$\vec{v}$	velocity vector of a small fluid element
V	index set that denotes partition of the mold geometry; its element, v_i, $i \in \{1, 2, \ldots, n\}$, denotes the center of the ith triangle in the finite element mesh of the mold geometry
X	location of gate
y_j	location of jth vent

1. Introduction

During the past two decades, widespread interest in the manufacture of composite parts has increased due to their light weight, superior properties, and excellent design flexibility. As a preferred method for manufacturing structural composite parts, the resin transfer molding (RTM) process has attracted increased attention because of its relatively low equipment and tooling costs, short cycle times, and ability to make parts with complex geometry. Although the RTM technique is still in the developmental stage, it has tremendous potential to produce consistently superior, quality components.

The RTM process can be generally divided into four steps, as shown in *Figure 1*. In the first step, dry

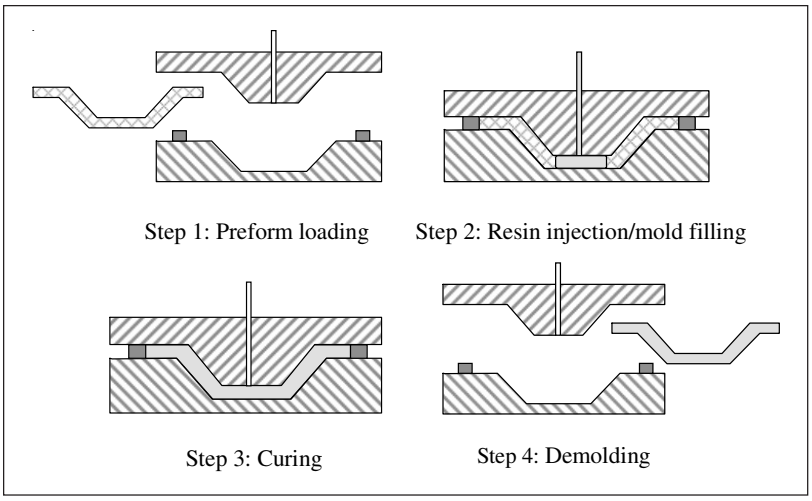

Figure 1
RTM Process

reinforcements are cut and/or shaped into preformed pieces and then placed in a prepared mold cavity. This is usually called preform loading. After the mold is closed and clamped tightly, resin is injected into the mold cavity, where it flows through the reinforcement preform, expelling the air in the cavity and "wetting out" or impregnating the reinforcement. This step, which is considered the most critical in the RTM process, is called mold filling. When excessive resin begins to flow out of the vent area of the mold, resin injection is stopped and the curing step begins. Part curing can take from several minutes to several hours. When part curing is complete, the component is then removed from the mold. This final step is called demolding.

Of the four steps in an RTM process, mold filling is the most critical step in the process. Among the various processing parameters, gate and vent locations are very important as they have a great impact on mold filling time and the resin flow pattern, thus directly affecting process efficiency and product quality. Traditionally, selection of gate and vent locations in RTM process/mold design is largely based on the experience of designers and an iterative trial-and-error process. Clearly, such a procedure is usually very time consuming and expensive. In the last decade, many studies had been conducted regarding the use of computer software to simulate the RTM mold filling process (Jiang et al. 1998; Lin, Hahn, and Huh 1998; Modi, Simacek, and Advani 2003; Phelan 1997; Young et al. 1991). With such simulation tools, a virtual experiment can predict pressure distribution, liquid resin flow front, possible void formation, and other phenomena of the process. From simulation results, one can determine ideal design parameters, such as locations of injection gates, pressures at injection gates, and locations of vents. Because these parameters are extremely important for minimizing or even eliminating the voids in the finished part, a systematic method is needed to determine at least some of the parameters. Some representative works that use optimization techniques in RTM process design include Yu and Young (1997); Mathur, Advani, and Fink (1999); and Luo et al. (2001). These studies extensively employed genetic algorithms (GAs) to optimize the arrangement of gates and vents locations. A major disadvantage of using a genetic algorithm is the expensive computation due to a considerable number of simulation runs. Gokce, Hsaio, and Advani (2002) developed a branch-and-bound method (B&B) and compared their results with enhanced GAs. Their method showed promising results.

An optimization model, in general, consists of a set of decision variables (continuous or discrete), which represents the controllable parameters of a process; an objective function (multiobjective functions must be mathematically reduced to a single objective), which acts as the process performance index; and a collection of constraints, which indicates if a specific solution is feasible. In this study, a simulation-based black box optimization method is employed. Some functions (for example, the objective function) in the optimization model can only be evaluated by running RTM simulations. In the model, the decision variables are: location of the single gate that is fixed during the process, locations of vents that are fixed during the process, and the pressure at the gate that is also fixed during the process.

The objective function is formulated by minimizing the maximum distance from the single gate to all vents. Once the location of the gate is given, the location of every vent can be determined by running a simulation. This indicates the explicit relationship among the decision variables. The constraints identify the general distance between any two locations within the mold. Because one can determine the locations of vents by the gate location,

the model here can use unconstrained optimization techniques.

The region defined by mold geometry is a continuous single connected area. The optimization model is continuous when determining the location of the gate and the locations of the vents. However, computer simulation can only treat a partial differential equation (PDE) discretely (by discretizing the domains and boundary conditions). Thus, the decision variables in a simulation-based optimization model can only lie in a discrete space. This will definitely lead to combinatorial optimization, which is not tractable. In this study, a directed weighted graph (DWG) is constructed as the simplified representation of a mold geometry. To decrease the computational complexity, the gate location in the optimization model is confined at the boundary of the mold.

2. Problem Formulation

Let $V = \{v_1, v_2, \ldots, v_n\}$ be the index set that denotes the partition of the mold geometry, where v_i, $i \in \{1, 2, \ldots, n\}$ denotes the center of the ith triangle in the finite element mesh of the mold geometry. Let $B = \{b_1, b_2, \ldots, b_m\} \subset V$ be the index set that denotes the partition of the boundary of the mold geometry, where b_i, $i \in \{1, 2, \ldots, m\}$ denotes the center of ith triangle in the finite element mesh of the mold geometry boundary.

Let $x \in B$ be the location of the gate. Let $y_j \in V$ ($j = 1, 2, \ldots, n_o$) be the locations of vents, where n_o is the number of vents.

Let M: $(B, R^+, R^+) \rightarrow \{(R_-^+)^n, \mathcal{P}(V)$ (power set of $V)\}$ denote the simulation that determines the pressure profile and the set of locations of the vents from the location and the pressure of the gate at some time. So, $M(x, P_x, t_0) = \left\{\left[p_1(t_0), p_2(t_0), \ldots, p_n(t_0)\right], \left\{y_1, y_2, \ldots, y_{n_o}\right\}\right\}$, where x is the location of the gate; P_x is the pressure fixed at the gate; t_0 is the time until local minima of the pressure profile are isolated; $[p_1(t_0), p_2(t_0), \ldots, p_n(t_0)]$ denotes the pressure profile against the index of triangles of the finite element mesh of the mold geometry at time t_0; and $\left\{y_1, y_2, \ldots, y_{n_o}\right\}$ is the set of isolated locations of local minimum pressure at time t_0. Obviously, $\left\{y_1, y_2, \ldots, y_{n_o}\right\}$ is also the set of locations of the vents.

Let

$$l\left(v_i, v_j, t_0\right) = \frac{\mu \cdot d_{i,j}^2}{k\left(v_i, v_j\right) \cdot \left|P_i\left(t_0\right) - P_j\left(t_0\right)\right|}$$

be the general distance between centers of two adjacent finite element triangles (adjacent if and only if a common edge exists), where μ is the viscosity of the liquid resin; $k(v_i, v_j)$ is the average principal permeability from v_i to v_j; and $d_{i,j}$ is the Euclidean distance between v_i and v_j.

Let $\left(x, v_{i_1}, v_{i_2}, \cdots, v_{i_s}, y_j\right)$ be a directed path from x (gate) to y_j (jth vent) if and only if for each $k = 1, 2, \ldots, s-1$, $P_{i_k} - P_{i_{k+1}} > 0$.

Let $L(x, y_j)$ be the length of shortest path from x (gate) to y_j (jth vent).

The optimization model can be written as:

$$\min f\left(x, y_1, \cdots, y_{n_o}\right) = n_o \cdot \max_{j=1:n_o}\left(L\left(x, y_j\right)\right) \tag{1}$$

S.t.

$$M\left(x, P_x, t_0\right) = \left\{\left[P_1\left(t_0\right), P_2\left(t_0\right), \cdots, P_n\left(t_0\right)\right], \left\{y_1, y_2, \cdots y_{n_o}\right\}\right\} \tag{2}$$

$$L\left(x, y_j\right) = \min_{\text{all paths from } x \text{ to } y_j}\left[l\left(x, v_{i_1}, t_0\right), + \sum_k l\left(v_{i_k}, v_{i_{k+1}}, t_0\right) + l\left(v_{i_s}, y_j, t_0\right)\right] \quad \text{for all } j = 1, 2, \cdots, n_o \tag{3}$$

$$l\left(v_i, v_j\right) = \frac{\mu \cdot d_{i,j}^2}{k\left(v_i, v_j\right) \cdot \left[P_i\left(t_0\right) - P_j\left(t_0\right)\right]} \quad \text{for all } i, j = 1, 2, \cdots, n \tag{4}$$

$$x \in B \subset V \tag{5}$$

$$y_j \in V,\ j = 1, 2, \cdots, n_o \tag{6}$$

3. Algorithm

The model (1)–(6) is a nonlinear resource allocation problem, which is NP-hard (see Gokce, Hsiao, and Advani 2002). The model (1)–(6) is a black box optimization problem because of the constraint (2). Thus, for efficiency, an efficient heuristic algorithm is needed to search for satisfactory solutions.

Reeves (1993), Pardalos and Resende (2002), and Pardalos and Romeijn (2002) categorized "black box" optimization (discrete or continuous) as optimization problems with the objective function ana-

lytically unknown. They proposed several general techniques, including general branch-and-bound, local search by multi-starts, pattern search by multi-starts, sequential constructive algorithm, tabu search, recursive sampling, simulated annealing, and genetic algorithm. All of these techniques are meta-heuristic, namely, they are independent of problem properties. Therefore, for the model (1)–(6), all of the techniques referred to above can be implemented to obtain a solution heuristically. However, the essential difficulty is the considerable number of simulation runs required. In the family of combinatorial black box optimization problems, the number of runs of simulation grows exponentially with the increase of feasible solution space, as does the number of constraints checking. A successful simulation-based optimization implementation requires an efficient solution algorithm.

3.1 Simulation-Based Optimization

Simulation-based optimization involves finding optimal inputs to a process or a system such that the responses can be optimized. In this case, the simulation takes the role of a process or a system.

Generally, a simulation is built by numerical techniques to approximate (or simulate) a real process qualitatively and quantitatively. Reasonable assumptions are made to simplify the mathematical formulation of the problem of interest such that the approximate solution can be easily obtained by numerical techniques. Once the simulation can approximate the real process, it can be combined with optimization methods to assist optimal design of the process. *Figure 2* characterizes the relation between optimization and simulation.

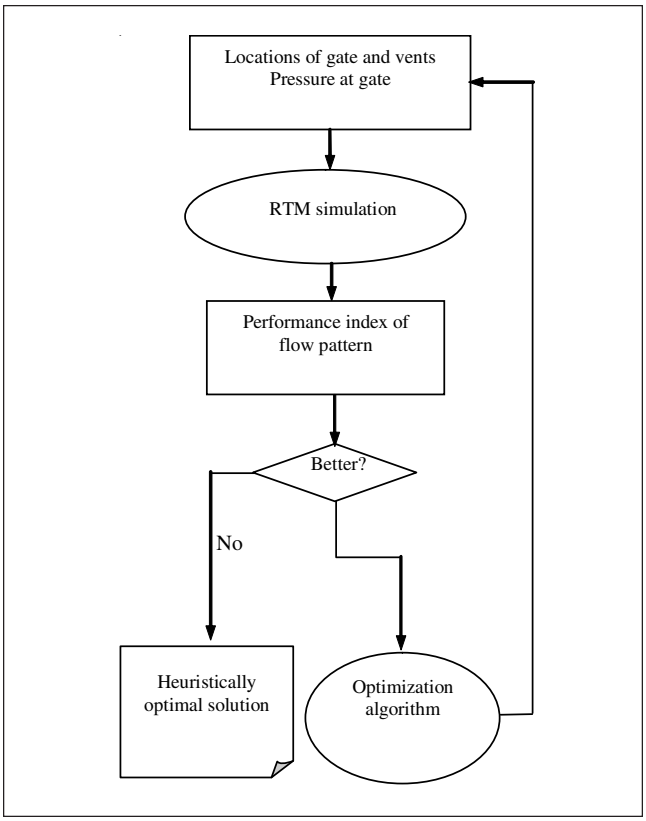

Figure 2
Schematic of Simulation-Based RTM Process Design Optimization

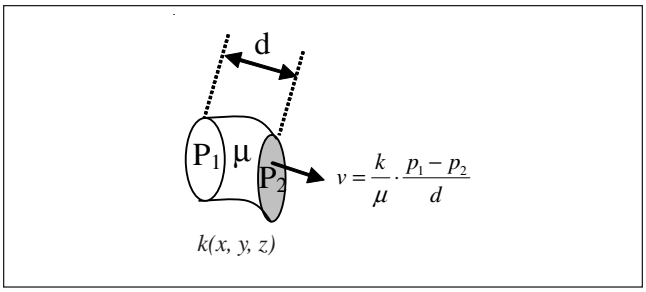

Figure 3
Darcy's Law

3.2 Simulation of RTM Process

The RTM process simulation is based on solving a time-dependent partial differential equation (PDE) that describes the liquid resin flow in a porous medium under some initial and boundary conditions. If the location of the gate and the pressure at the gate are fixed during the process, the boundary condition can be viewed as time independent (or steady).

Before the PDE is derived, some basic assumptions are made to simplify the mathematical formulation of the liquid resin filling process. The assumptions include: incompressible liquid resin, no chemical reaction, negligible inertia effect, negligible surface tension, isothermal, and no fiber deformation. If the thickness of the part is much smaller than the size of the surface, the simulation model can be reduced to 2-D. With these assumptions, Darcy's law [Eq. (7)] can replace momentum equations in general Navier-Stokes equations for a small element of fluid, as shown in *Figure 3* (Young et al. 1991).

$$\vec{v} = \frac{\overline{\overline{k}}}{\mu} \cdot \nabla P \tag{7}$$

where $\vec{v}$ is the velocity vector of small fluid element at (x, y, z, t), ∇P is the pressure gradient at (x, y, z, t), $\bar{\bar{k}}$ is the permeability tensor (second order) at (x, y, z) (for isotropic case, $\bar{\bar{k}}$ can be replaced by its principal value, say, k), and μ is the viscosity of fluid, which can be viewed as a constant for a specific liquid resin.

The filling process is a time-dependent process. When considering the process as single-phase flow, a filling rate at (x, y, z) is defined as: F(x, y, z, t) (∈ [0, 1]) : F(x, y, z, t) = 0 if the control volume at (x, y, z) is empty; F(x, y, z, t) = 1 if the control volume at (x, y, z) is fully filled. Those control volumes with 0 < F(x, y, z, t) < 1 are viewed as belonging to the flow front. Then mass conservation can be formulated as:

$$\varepsilon \frac{\partial \mathrm{F}}{\partial t} = \nabla \vec{v} = \nabla \cdot \left(\frac{\bar{\bar{k}}}{\mu} \cdot \nabla P \right) \tag{8}$$

where ε is a constant. At gate: P (or $\frac{\partial P}{\partial n}$) is given; at mold boundary: $\frac{\partial P}{\partial n} = 0$; beyond the flow front: P = 0. The filling process ends with $F = 1$ throughout the mold geometry.

When considering the process as two-phase flow, a saturation rate at (x, y, z) is defined as S(x, y, z, t), which is the fraction of resin occupancy. The mass conservation can be similarly formulated as:

$$\alpha \frac{\partial \mathrm{S}}{\partial t} = \nabla \vec{v} = \nabla \cdot \left(\frac{\bar{\bar{k}}}{\mu} \cdot \nabla P \right) \tag{9}$$

where α is a constant. Those control volumes with saturation close to some threshold are viewed as the flow front. At the gate: P (or $\frac{\partial P}{\partial n}$) is given; at mold boundary: $\frac{\partial P}{\partial n} = 0$; beyond the flow front: $P = P_0$. The filling process ends with S > some threshold throughout the mold geometry.

Both Eqs. (8) and (9) have the form:

$$(\circ) \frac{\partial (\circ)}{\partial t} = \nabla \cdot \left(\frac{\bar{\bar{k}}}{\mu} \cdot \nabla P \right) \tag{10}$$

In the isotropic case, Eq. (10) can be reduced to:

$$(\circ) \frac{\partial (\circ)}{\partial t} = \nabla \cdot \left(\frac{k}{\mu} \cdot \nabla P \right) \tag{11}$$

Generally, the coefficient k/μ has an order of magnitude of about 10^{-5}. This will make the filling process simulation run very slowly when dealing with a large mold geometry. A method of accelerating the virtual filling process is to take advantage of a similar transformation of the governing PDE. From Eq. (11), multiply a magnifying factor γ on both sides; a new PDE is obtained as:

$$(\circ) \frac{\partial (\circ)}{\partial \tau} = \nabla \cdot (D \cdot \nabla P) \tag{12}$$

where $D(x, y, z) = \gamma \cdot k(x, y, z)/\mu$, $\tau = t/\gamma$.

Finding the solution of Eq. (11) from time 0→1 is the same as solving Equation (12) from time $0 \rightarrow 1/\gamma$. Hence, the virtual process is greatly accelerated when $\gamma >> 1$. However, some details will be lost when the same time step is used.

3.3 Optimization Method

With the simulation accelerated, the next step is to accelerate the optimization process. The key idea is to decrease the number of objective function evaluations while maintaining the quality of the heuristically optimal solution.

3.3.1 Mold Geometry Simplification

In this section, a method of directed weighted graph (DWG) construction is introduced to approximately analyze the mold geometry. Once the mold geometry is simplified, dealing with constraints (3)–(4) can be greatly simplified. *Figures 4* to *9* show six 2-D cases that have been studied.

To simplify the mold geometry, the following heuristic principles are employed:

(1) A mold geometry can be represented by a graph. *Figures 10* to *15* are graph representations of the mold geometries of *Figures 4* to *9*, respectively.
(2) The graph can be transformed into a directed graph (edges are transformed to arcs), according to computed pressure profile (at time $t = t_0$) at the vertices. The pressure at each vertex can be treated as its label. The local maximum can be viewed as the source or the gate; the local minima can be viewed as sinks or vents.
(3) In constructing the representative directed graph, the scale of the graph (number of vertices, number of arcs) should be minimized. Oth-

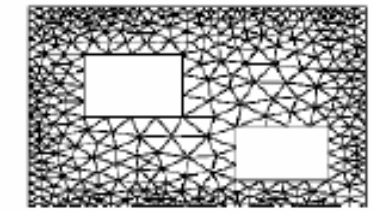

Figure 4
Mold Geometry 1

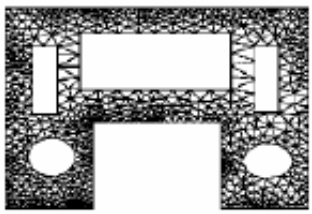

Figure 5
Mold Geometry 2

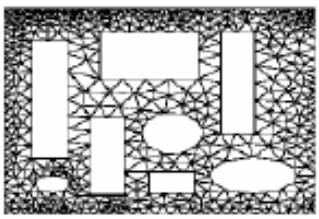

Figure 6
Mold Geometry 3

Figure 7
Mold Geometry 4

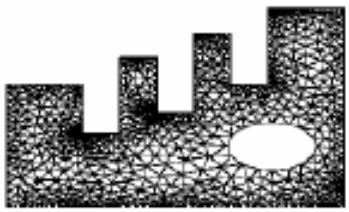

Figure 8
Mold Geometry 5

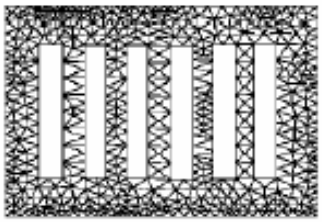

Figure 9
Mold Geometry 6

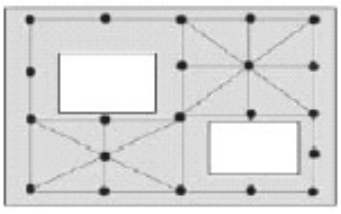

Figure 10
Graph Representation for Mold Geometry 1

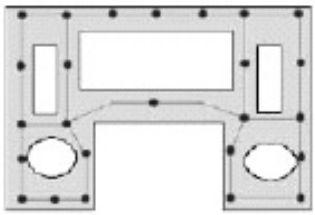

Figure 11
Graph Representation for Mold Geometry 2

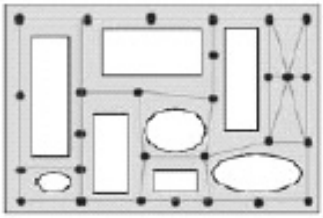

Figure 12
Graph Representation for Mold Geometry 3

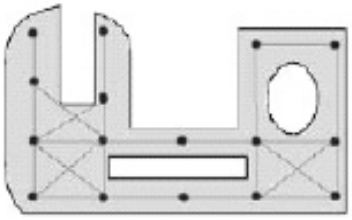

Figure 13
Graph Representation for Mold Geometry 4

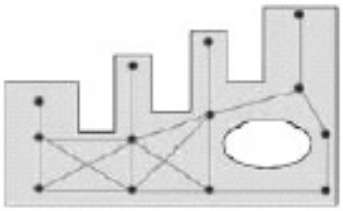

Figure 14
Graph Representation for Mold Geometry 5

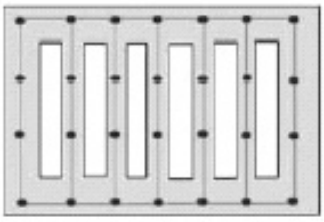

Figure 15
Graph Representation for Mold Geometry 6

erwise, the algorithms for finding the shortest path in the directed positively weighted graph (Dijkstra algorithm, Bellman-Ford, and Acyclic SP algorithm) would be extremely computationally costly (West 2001).

(4) In constructing the weight for each arc of the directed graph, the length of the arc and the average permeability along the arc must be considered.

Suppose v_1 and v_2 are two adjacent vertices in the constructed directed graph with pressure label P_1 and P_2, respectively. Denote (v_1, v_2) as the arc from v_1 to v_2, and $c(v_1, v_2)$ as the curve of the arc (v_1, v_2). Denote $d_{1,2}$ as the Euclidean length of $c(v_1, v_2)$. Then:

$$d_{1,2} = \int_{\theta_1}^{\theta_2} \sqrt{x_\theta'^2 + y_\theta'^2} \cdot d\theta \tag{13}$$

where the integration is on the curve equation of $c(v_1, v_2)$ with parameter θ, say $c(v_1, v_2) = (x = x(\theta), y = y(\theta):\theta_1 \rightarrow \theta_2)$.

The average permeability along the curve of arc $c(v_1, v_2)$ is:

$$k(v_1, v_2) = \frac{\int_{\theta_1}^{\theta_2} k(x(\theta), y(\theta)) \cdot \sqrt{x_\theta'^2 + y_\theta'^2} \cdot d\theta}{d_{1,2}} \quad (14)$$

The general length of the arc (v_1, v_2) is:

$$l(v_1, v_2) = \frac{\mu \cdot d_{1,2}^2}{k(v_1, v_2) \cdot |P_1 - P_2|} \quad (15)$$

where μ is the viscosity of the liquid resin.

The direction of the arc is determined by its vertex labels P_1 and P_2: $(v_1 \rightarrow v_2)$ if and only if $P_1 > P_2$. The vertex with only outbound arcs is the source. Those with only inbound arcs are sinks.

Once the weights and the directions are constructed, the next step is to find the lengths of the shortest paths from a presumed source to each of the sinks.

3.3.2 Optimization in Graph

Once a vertex of the constructed DWG is labeled as a source, one simulation run can determine all of the sinks. The best candidate algorithm—the Dijkstra algorithm—for finding the lengths of the shortest path from the single source to each sink is employed. Back to the optimization model (1)–(6), the equations in the set of constraints (3) are redefined in the constructed DWG as:

$$L(x, y_j) = \min_{\text{all paths from } x \text{ to } y_j} \left[l(x, v_{i_1}, t_0) + \sum_k l(v_{i_k}, v_{i_{k+1}}, t_0) + l(v_{i_s}, y_j, t_0) \right] \quad (16)$$
$$\text{for all } j = 1, 2, \cdots, n_1$$

where x, $y_j \in V(G)$: set of vertices of constructed DWG, $j = 1, \ldots, n_1$ (number of sinks).

One time of computing the objective function is equivalent to one time of calling the Dijkstra algorithm. Because the constructed DWG is much simpler than the original finite element mesh of the mold geometry, the computational cost of running the Dijkstra algorithm is largely decreased.

If the single source is not presumed, the optimal one must be found, which is indicated by the objective function (1). Because the set of constraints (3) is redefined in the constructed DWG, the objective function is accordingly redefined:

$$f(x, y_1, \cdots, y_{n_1}) = n_1 \cdot \max_{j=1:n_1} (L(x, y_j)) \quad (17)$$

where x, $y_j \in V(G)$: set of vertices of constructed DWG, $j = 1, \ldots, n_1$ (number of sinks).

However, finding the optimal source in the constructed DWG according to objective function (17) is a problem of global optimization. Every vertex can be tried to compute the index. This is an exhaustive enumeration. Obviously, trying every vertex is not necessary. A heuristic principle is to confine the candidate source to the boundary of the mold because a source at the boundary usually generates less sinks than a source at the interior.

The heuristic method used in this study is called a graph-based two-phase heuristic (GTPH). In this heuristic solution algorithm, completing the search in the vertices at the boundary is finishing the first phase of the optimization. The second phase is to refine the search of the optimal source in the arcs adjacent to the satisfactory source obtained in the first phase. The first phase is nothing more than a partial enumeration. The second phase is certainly the technique of pattern search. After completing the second phase, a heuristically ideal flow pattern of one gate versus multiple vents is found. In the next section, the numerical results will be provided and analyzed for mold geometries 1 to 6 in detail.

4. Numerical Examples

In this section, each of the six mold geometries is investigated using the filling process simulation and numerical optimization. Each case is studied with a given permeability profile. The pressure at the gate is set as a constant because the value of the pressure at the gate mainly affects the speed of the filling process other than the flow pattern. The results are shown in *Figures 16* to *21*.

5. Discussions

5.1 Effectiveness of Optimization

Figures 22 and *23* show two cases of unoptimized versus optimized RTM simulation processes. *Figure 22* shows two computational results of mold geometry 2. As *Figure 22a* shows, the gate is arbitrarily set at the edge of the mold. The computational result

Case 1: Mold Geometry 1

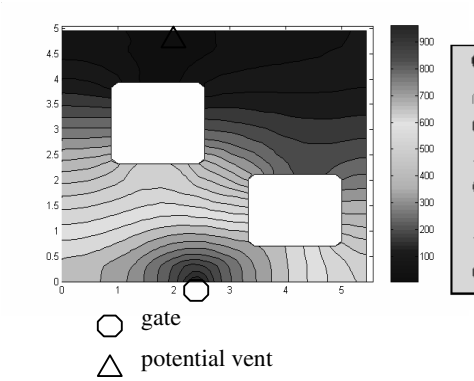

Figure 16a
Near Saturated Pressure Profile of Heuristically Optimal Flow Pattern

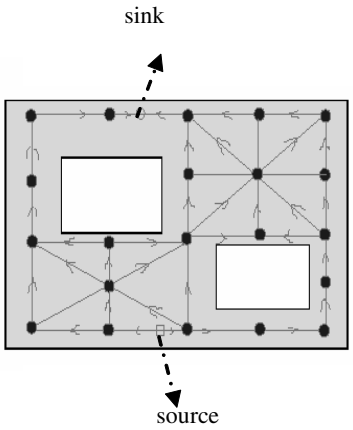

Figure 16b
Source and Sink in Weighted Directed Graph

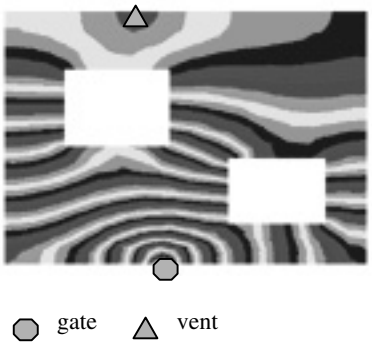

Figure 16c
Heuristically Optimal Flow Process Found After 19 Simulations

Case 2: Mold Geometry 2

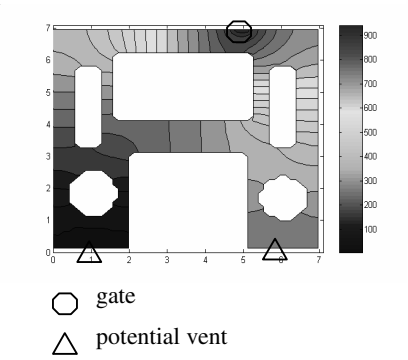

Figure 17a
Near Saturated Pressure Profile of Heuristically Optimal Flow Pattern

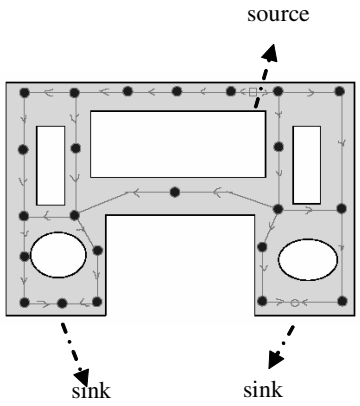

Figure 17b
Source and Sink in Weighted Directed Graph

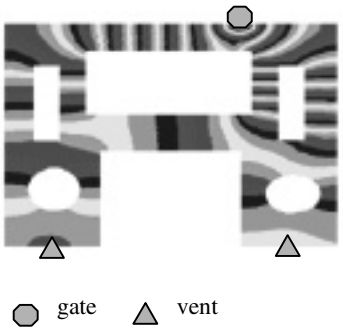

Figure 17c
Heuristically Optimal Flow Process Found After 27 Simulations

Case 3: Mold Geometry 3

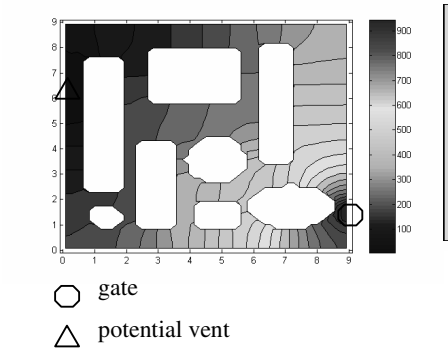

Figure 18a
Near Saturated Pressure Profile of Heuristically Optimal Flow Pattern

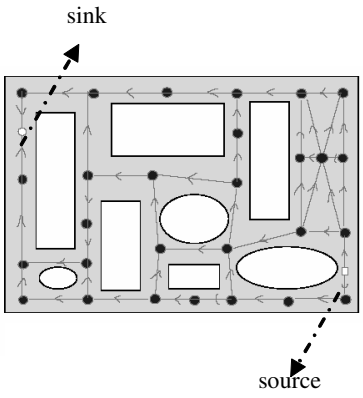

Figure 18b
Source and Sink in Weighted Directed Graph

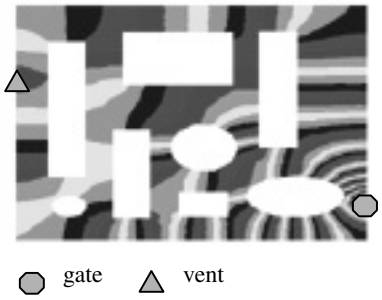

Figure 18c
Heuristically Optimal Flow Process Found After 23 Simulations

Case 4: Mold Geometry 4

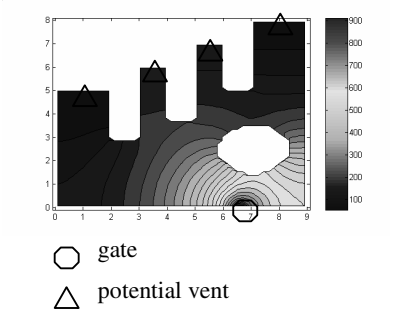

Figure 19a
Near Saturated Pressure Profile of Heuristically Optimal Flow Pattern

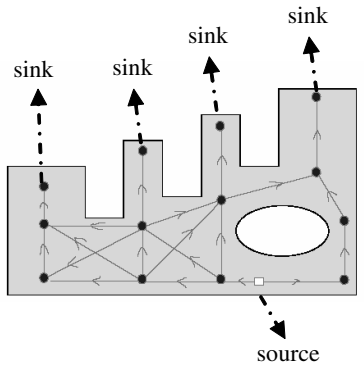

Figure 19b
Source and Sink in Weighted Directed Graph

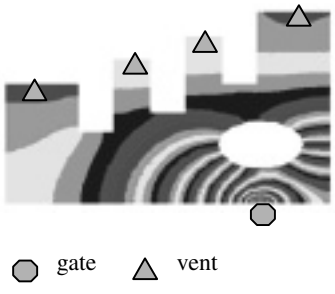

Figure 19c
Heuristically Optimal Flow Process Found After 14 Simulations

Case 5: Mold Geometry 5

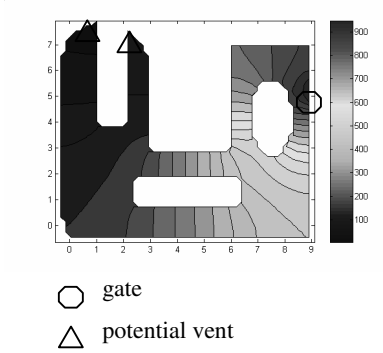

Figure 20a
Near Saturated Pressure Profile of Heuristically Optimal Flow Pattern

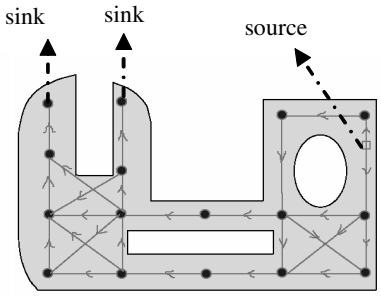

Figure 20b
Source and Sink in Weighted Directed Graph

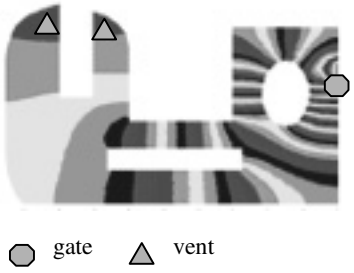

Figure 20c
Heuristically Optimal Flow Process Found After 16 Simulations

Case 6: Mold Geometry 6

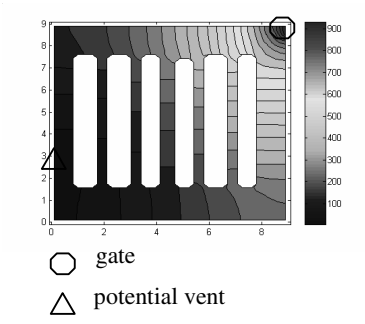

Figure 21a
Near Saturated Pressure Profile of Heuristically Optimal Flow Pattern

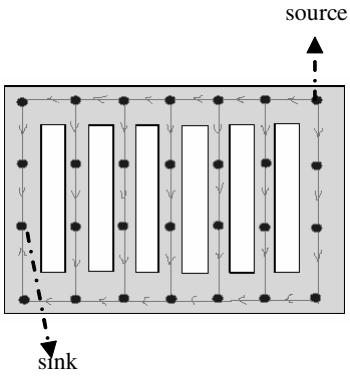

Figure 21b
Source and Sink in Weighted Directed Graph

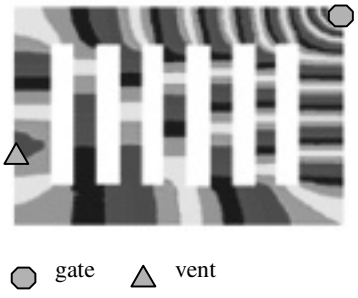

Figure 21c
Heuristically Optimal Flow Process Found After 22 Simulations

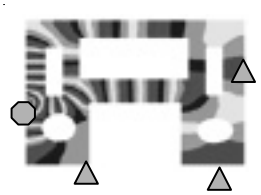

Figure 22a
Unoptimized Pattern of One Gate with Three Necessary Vents

Figure 22b
Optimized Pattern of One Gate with Two Necessary Vents

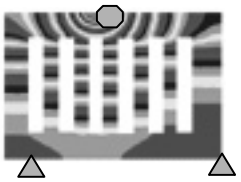

Figure 23a
Unoptimized Pattern of One Gate with Two Necessary Vents

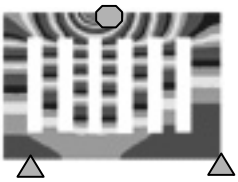

Figure 23b
Optimized Pattern of One Gate with One Necessary Vents

shows three necessary vents. *Figure 22b* shows the optimized gate location. There are two necessary vents and the general distances from gate to the vents are more uniform.

Looking at mold geometry 6, as *Figure 23a* shows, the gate is arbitrarily set at the edge of the mold. Computational results show two necessary vents. *Figure 23b* shows the optimized gate location, requiring only one necessary vent.

The objective function (17) or (1) indicates that: (1) a single gate with as few necessary vents as possible is desirable; and (2) the shortest possible general distance from the single gate to every necessary vent is desirable (that is, as little filling time as possible at a fixed gate pressure). The numerical results in section 4 strongly support these choices.

5.2 Efficiency of Optimization

Of course, two obvious answers to the question as to why a heuristic is used are the NP-hardness of the nonlinear combinatorial optimization model and the computational cost of the objective function evaluations (runs of simulation). For computer-aided decision making in manufacturing, "quick and dirty" algorithms produce better results than technically detailed algorithms that may have better chance to find an exactly optimal solution. One reason for this is that an exact optimal solution is only exactly optimal to the model that describes the corresponding process. In the case that the model can only describe the process with some degree of approximation, the "optimal" solution may not be actually optimal. Hence, technically detailed algorithms that need huge numbers of function evaluations are not necessary due to their time inefficiency and pseudo-optimality. Besides, decision makers are more likely to seek some improvements over current practices. At this point, heuristic algorithms that find satisfactory solutions with as few computational efforts as possible would be more beneficial.

Because two meta-heuristic methods, genetic algorithms (GAs) and branch-and-bound (B&B) search, have been used extensively in the field of optimization for computer-aided manufacturing such as RTM, Gokce, Hsiao, and Advani (2002) compare the numbers of function evaluations of exhaustive search, GAs, and B&B. Several studies employed basic GAs to optimize the flow front to minimize the potential dry areas (Jiang, Zhang, and Wang 2002; Luo et al. 2001; Mathur, Advani, and Fink 1999; Yu and Young 1997). For comparison, some results are listed in *Table 1*. It can be seen that the methods that analyze the mold geometry best can lead to fewer simulations. The reason is very simple: more knowledge on the feasible solution space means less randomness in searching for better solutions, thus less waste of computational efforts.

Also, numerical experiments show that the similarity analysis in section 3.2 helps to decrease the overall CPU time for one simulation. However, a balance between the acceleration and loss of filling process detail is required. Detailed research on this will be developed in future studies.

6. Conclusions

In this paper, an optimization model was developed for RTM process design with one gate and multiple vents. The model was solved using a graph-based two-phase heuristic method introduced by the authors. Through numerical examples, the authors illustrate that the proposed method is more efficient than the existing algorithms reported in the literature for solving similar problems. The model can handle cases of multiple vents. In addition, the model can be extended for solving multiple gates and multiple vents location problems for RTM process design, which could be treated as the problem of post-solution processing of the one gate and multiple vents location problem. This optimization model

Table 1
Comparison of Computational Efficiency

Algorithm	Average Number of Simulations Required
Exhaustive Search	> 800
Basic GAs	> 200
Enhanced GAs	> 150
B&B	25~100
GTPH	20~50

and solution algorithm can also be extended for similar process design applications of other processes such as injection molding where simulations are also involved for optimization. These applications will be the subject of future work.

Acknowledgments

This work has been supported by GKN Aerospace Services, Office of Naval Research, and NSF I/UCRC Program (Award #0224612).

References

Gokce, A.; Hsiao, K.T.; and Advani, S.G. (2002). "Branch and bound search to optimize injection gate locations in liquid composite molding processes." *Composites Part A: Applied Science and Mfg.* (v33, n9), pp1263-1272.

Jiang, S.; Zhang, C.; and Wang, B. (2002). "Optimum arrangement of gate and vent locations for RTM process design using a mesh distance-based approach." *Composites Part A: Applied Science and Mfg.* (v33, n4), pp471-481.

Jiang, S.; Han, K.; Zhang, C.; and Wang, B. (1998). "Process design of resin transfer molding with computer simulation." *Proc.* of 1998 IMEC&E Symp. on Concurrent Product and Process Design, Anaheim, CA, Nov. 15-20, 1998.

Lin, M.; Hahn, H.T.; and Huh, H. (1998). "A finite element simulation of resin transfer molding based on partial nodal saturation and implicit time integration." *Composites Part A: Applied Science and Mfg.* (v29A), p541.

Luo, J.; Liang, Z.; Zhang, C.; and Wang, B. (2001). "Optimum tooling design for resin transfer molding with virtual manufacturing and artificial intelligence." *Composites Part A: Applied Science and Mfg.* (v32, n6), pp877-888.

Mathur, R.; Advani, S.G.; and Fink, B.K. (1999). "Use of genetic algorithms to optimize gate and vent locations for the resin transfer molding process." *Polymer Composites* (v20), pp167-78.

Modi, D.; Simacek, P.; and Advani, S.G. (2003). "Influence of injection gate definition on the flow-front approximation in numerical simulations of mold filling processes." *Int'l Journal for Numerical Methods in Fluids* (v42), pp1237-1248.

Pardalos, P.M. and Resende, M. (2002). *Handbook of Applied Optimization*. Oxford Univ. Press.

Pardalos, P.M. and Romeijn, E. (2002). *Handbook of Global Optimization – Vol. 2: Heuristic Approaches*. Kluwer Academic Publishers.

Phelan Jr., F.R. (1997). "Simulation of injection process in resin transfer molding." *Polymer Composites* (v18, n4), p460.

Reeves, C.R. (1993). *Modern Heuristic Techniques for Combinatorial Problems*. New York: Halsted Press.

West, D.B. (2001). *Introduction to Graph Theory*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall.

Young, W.B.; Han, K.; Fong, L.H.; Lee, L.J.; and Liou, M.J. (1991). "Flow simulation in molds with preplaced fiber mats." *Polymer Composites* (v12, n6), pp391-403.

Yu, H.W. and Young, W.B. (1997). "Optimal design of process parameters for resin transfer molding." *Journal of Composite Materials* (v31, n11).

Authors' Biographies

Mr. Xugang Ye is a PhD student in the Industrial and Manufacturing Engineering Dept. at Florida A&M University-Florida State University College of Engineering. His research interests include development of optimization algorithms and their applications to manufacturing processes and systems.

Dr. Chun (Chuck) Zhang is professor and interim chair of the Industrial and Manufacturing Engineering Department at Florida A&M University-Florida State University College of Engineering, where he teaches courses in computer-aided manufacturing, computer-integrated manufacturing, and manufacturing systems modeling. Dr. Zhang's research interests include intelligent processing of composite materials, geometric tolerancing and metrology, and integration of manufacturing execution systems. His research projects have been sponsored by a number of organizations, including the National Science Foundation, National Institute of Standards and Technology, Air Force Office of Scientific Research, Army Research Laboratory, and Office of Naval Research, as well as industrial companies such as Cummins Engine Co., GKN Aerospace Services, and Lockheed Martin Co.

Dr. Zhiyong Liang is an associate professor in the Industrial and Manufacturing Engineering Department at Florida A&M University-Florida State University College of Engineering, where he teaches courses in manufacturing processes and composite materials manufacturing. He received his bachelor of science, master's, and doctoral degrees in materials science and engineering from Beijing University of Aeronautics and Astronautics (China). Dr. Liang's research projects have been sponsored by the Air Force Office of Scientific Research, Army Research Laboratory (ARL), The Boeing Company, National Science Foundation, and Lockheed Martin Co.

Dr. Ben Wang is professor of industrial engineering in the Industrial and Manufacturing Engineering Department at Florida A&M University-Florida State University College of Engineering, and assistant vice president for research in engineering with Florida State University. He holds the Simon Ostrach Professorship and Massie Chairship. He is a fellow of the Institute of Industrial Engineers and the Society of Manufacturing Engineers. He received his bachelor of science in industrial engineering degree from Tunghai University (Taiwan) and master's and doctoral degrees in industrial engineering from Pennsylvania State University.