A SIMULATION-BASED OPTIMIZATION APPROACH TO POLYMER EXTRUSION FILTER DESIGN

K.R. Fowler[1] S.M. LaLonde[1] E.W. Jenkins[2] C.L. Cox[2]

[1] Department of Mathematics, Clarkson University, Postdam, NY

[2] Department of Mathematical Sciences, Clemson University, Clemson, SC

Abstract

A challenge in polymer processing is the effective removal of debris, via filtration, from the polymer melt during the extrusion process. We propose methods for finding optimal parameters for the filter such that its lifetime is maximized, while placing reasonable restrictions on the amount of escaped debris. We make use of a three-dimensional simulation model that describes the deposition of debris particles in the filter. Optimization algorithms are used in conjunction with the simulator tool to identify the filter parameters. This is a difficult problem that is not described by a differentiable function, since function evaluations are given by a simulator which is treated as a black-box. Thus, we apply derivative-free techniques to analyze the behavior of the filter and to gain insight into its optimal design. We will discuss these techniques, along with approaches to formulating this problem mathematically. We present promising numerical results and point the way towards advancing this study.

1   INTRODUCTION

Researchers at the Center for Advanced Engineering Fibers and Films (CAEFF) [1] assist industrial partners in the design and production of polymer products. As part of this effort, they have developed a computational tool that simulates the main stages of a fiber-spinning process [3]. This tool includes a three-dimensional model of an extrusion filter [13], which is used to separate debris particles from the molten polymer before the material is spun into a fiber (see Figure 1). The diameter of the pores in the filter are small enough to capture particles a few microns in diameter. As these debris particles accumulate inside the filter, the maintenance of a constant mass flow rate through the system causes the pressure drop across the filter to increase. Once the pressure drop across the filter (i.e., the difference in pressure values at the entrance and exit of the filter) passes a certain threshold, there is a risk of damaging the pumping mechanism. Thus, the filter must be replaced before this threshold is reached.
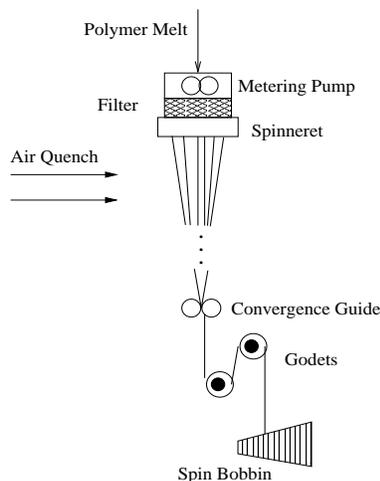
Figure 1: Fiber melt-spinning process

Filter performance can be evaluated using two measures. First, the filter must effectively remove debris particles as the presence of particles in the fiber may drastically reduces the quality of the finished product. Second, the filter should have a sufficient lifetime. Filter replacement is costly, often requiring that the entire spinline be taken out of production. In-line filters are being introduced, but analysts believe that the trend will be to combine these filtration methods with conventional filters located at the spinneret [9]. Thus we are faced with competing objectives, as extending the filter lifetime could be achieved by simply reducing the amount of trapped debris.

Previous work attempted to optimize the performance of the filter using a multi-objective genetic algorithm [7], which requires the use of two competing objective functions. In this

2

work, we combine the competing objectives into a single functional using barrier methods. This allows us to minimize the functional using single-search methods as opposed to multi-objective algorithms.

The problem is still a challenging one, however, since any objective function will be defined in terms of the output from the simulator. This means that we lack a continuous function that describes the filtration process; hence, we lack gradient information. Since we are unable to use classical calculus-based approaches, we must turn to optimization methods that do not require knowledge of derivatives. Our choice is an implementation of the implicit filtering algorithm [10, 11]. Implicit filtering is a sampling-based method; that is, the optimization is guided only by objective function values. The data obtained from this design space sampling will guide us towards optimal design parameters and provide insight into the effects of such parameters on the behavior of the filter. An advantage of this approach is that it provides a framework for a variety of simulation methods, including the three dimensional algorithm we apply here.

Our discussion is structured as follows. An overview of the filtration model and the simulation tool is given in Section 2. In Section 3, we outline the theoretical approaches taken and the derivations of three objective functions which measure the filter performance. We describe the implicit filtering algorithm used for optimization and provide numerical results in Section 4. Conclusions and future work are given in Section 5.

2   DEBRIS DEPOSITION MODEL

The filtration code (i.e., the "simulator") used in this work was developed at CAEFF. The filtration problem was studied in one-dimensional space in [5], and this was extended to three-dimensional space in [13]. The three-dimensional work led to the development of the simulator.

The material entering the filter consists of molten polymer, unmelted polymer gel particles (formed when the polymer does not completely melt in the heating/mixing stage), and other debris such as metal particles. The density of the mixture is assumed to be the same as the polymer density, i.e., the density of the debris is negligible in comparison to the polymer density. It is also assumed that the thickness of the filter is small so that gravitational effects can be ignored [5, 13].

The flow through the filter is governed by the continuity equation (mass conservation) and Darcy's law, modified to account for the non-Newtonian behavior of the polymer melt. The debris entering the filter is specified in terms of a mass fraction relative to the mass of the mixture. The distribution of debris particles is characterized statistically, as in [9], using either truncated or standard normal probability density functions [7].

The mass of captured debris is calculated for each computational cell. Particles with a larger diameter than the pore diameter are eligible for capture; the percentage of these particles retained by the filter is determined by an empirical retention function associated with the filter [13]. All particles not captured within a computational cell are transported to downstream cells.

The porosity of the filter and the average pore diameter of the filter are updated at the end of every time step using the information on the deposited debris.

The new values for porosity and pore diameter are used to calculate the permeability and pressure drop. The permeability parameter k is currently modeled using a Blake-Kozeny relationship that depends on the average diameter of the filter pore size, d_p, and the current filter porosity, η. The relationship for k, derived in [2] and used in [5, 13], is

$$k = \frac{d_p^2 \eta^3}{150(1 - \eta)^2}.$$

Filters are often composed of multiple layers, each of which is characterized by a porosity, an average pore diameter, and a retention function. We consider only single-layer filters in this work, though our simulator is capable of handling multiple layers.

As the filter clogs, the volume of the void space decreases, leading to a decrease in the permeability. The pressure drop must then increase in order to maintain a constant mass flow rate throughout the filter. The large values of pressure required at the inlet may lead to damage of the metering pump, so the filter must be replaced once the pressure drop reaches a threshold value. The threshold value is set at $35e6$ Pa in the simulator, which is comparable to existing industry thresholds [13]. A pressure drop curve for a filter with an initial porosity of 0.61 and an initial average pore diameter of 25.4 microns is given in Figure 2.
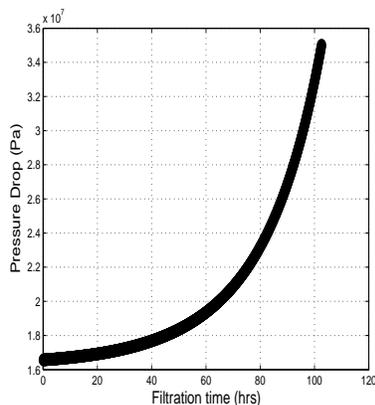
Figure 2: Pressure drop curve for representative filter

More detailed information on the model can be found in [4, 7, 13].

3 MODELLING THE EXTRUSION FILTER PERFORMANCE

The optimization problem is formulated as

$$\text{Maximize (or Minimize)} F(x), x \in \Omega, \tag{1}$$

where $F(x)$ represents the performance of the filter (e.g., lifetime), x is a vector of decision variables (filter parameter values), and Ω defines acceptable bound constraints on those parameters.

We consider three problem formulations, each using a different measure of filter performance, resulting in different definitions of $F(x)$. The performance of the filter is analyzed with respect to the filter parameters porosity, denoted by η, and average pore diameter, denoted by d_p so that $x = (\eta, d_p)$. The bound contraints are defined by the set

$$\Omega = \{(\eta, d_p) : 0.4 \le \eta \le 0.65, 23 \le d_p \le 40\}.$$

3.1 $F_1(x)$: Filter lifetime

The primary objective in our first problem formulation is to maximize the lifetime of the filter. A secondary, competing objective, is to minimize the amount of debris escaping the filter. This secondary objective is handled through the use of a barrier method. Taking into consideration the viewpoint of an engineer or production manager, we wish to define an acceptable bound on the escaped debris. This bound represents a level of debris that the manager would allow to make its way into the final product. Let b represent the bound, and let $\xi(x)$ represent the total mass of debris that escapes over the lifetime of the filter. Naturally, we wish to construct the barrier function in such a way that $\xi(x)$ is bounded away from b. In addition, the function should attain its minimum when $\xi(x) = 0$. These stipulations lead to the function

$$B(\xi) = \frac{\xi(x)}{b - \xi(x)}, \tag{2}$$

which is defined on the half-open interval $[0, b)$. We can see that $B(\xi) \to \infty$ as $\xi \to b^-$. Likewise, $B(\xi) \to 0$ as $\xi \to 0^+$. Thus $B(\xi)$ is minimized when $\xi = 0$ and grows without bound as ξ approaches b.

Let $t(x)$ represent the lifetime of the filter. Since we wish to maximize $t(x)$, we will minimize its negative. This lends itself to the function

$$F_1(x) = -t(x) + \rho_1 B(\xi), \tag{3}$$

where ρ_1 is a constant with units of hours. Note that, from the definition of $B(\xi)$, we see that $F_1(x)$ is not defined for values of x for which $\xi(x) \ge b$. We handle such cases in a computational sense by setting $F_1(x)$ to an unreasonably large value.

3.2 $F_2(x)$: **Pressure drop across filter**

As previously stated, we assume that the filter is replaced when the pressure drop reaches a certain threshold. For the second formulation, which is motivated by Figure 2, the primary objective is to minimize the maximum rate of change of the pressure drop across the filter. This offers a way of indirectly maximizing the lifetime of the filter. We retain the secondary objective of minimizing the total escaped debris.

To quantify this primary objective,let p_n denote the difference in pressure (in Pascals) across the filter at the nth timestep t_n. The change in pressure between two successive timesteps can be written as

$$\Delta p_n = \frac{p_n - p_{n-1}}{t_n - t_{n-1}}, \tag{4}$$

which has units of Pascals per hour. If $P = \{\Delta p_1, \Delta p_2, \ldots, \Delta p_k\}$, where k is the total number of time steps, then clearly there is an $\alpha_{min} = \sup P$. Since x is fixed, it follows that there is associated with each value of x one and only one α_{min}. Hence, α_{min} can be written as a function of x.

In order to incorporate the secondary objective, we keep the barrier function $B(\xi)$ that was introduced in (2). This leads to the objective function

$$F_2(x) = \alpha_{min}(x) + \rho_2 B(\xi), \tag{5}$$

where ρ_2 is a constant with units of Pascals per hour. As with $F_1(x)$, $F_2(x)$ is defined only for values of x where $\xi(x) < b$; that is, where $B(\xi)$ is defined. Computationally, this is handled in the same manner for both functions.

3.3 $F_3(x)$: **Overall pressure change**

As a variation of the pressure approach, we consider a linear estimate of the total change in pressure over the lifetime of the filter. Suppose the lifetime of the filter consists of N time steps. We measure the total change in pressure by constructing a line passing through the points (t_0, p_0) and (t_N, p_N) (motivated again by Figure 2). A reasonable objective would be to minimize the slope of this line, which can be expressed as

$$m = \frac{p_N - p_0}{t_N - t_0}, \tag{6}$$

where m has units of Pascals per hour. Note that $t_0 = 0$, and t_N is simply the overall lifetime of the filter, so we have $t = t_N - t_0$. Also note that we can easily express p_N, p_0, and t as functions of x, which gives the objective function

$$F_3(x) = \frac{p_N(x) - p_0(x)}{t(x)} + \rho_3 B(\xi). \tag{7}$$

6

Here $B(\xi)$ is the barrier function defined in (2), and ρ_3 is a constant with units of Pascals per hour. Once again, $F_3(x)$ is defined for values of x where $B(\xi)$ is defined and where $t(x) \neq 0$. The first case is handled as it was previously; the second can be neglected, since $t(x) = 0$ is impractical.

4   NUMERICAL RESULTS

We solve the optimization problem given by (1) using a MATLAB implementation of the implicit filtering algorithm developed by C.T. Kelley at North Carolina State University [10]. Implicit filtering is entirely derivative-free and is based on a quasi-Newton approach, which uses the objective function to build approximate gradients of $F(x)$ and a secant-like method for the Newton-step [8, 12]. The advantage of implicit filtering is that the size of the finite-difference interval on which the gradient is approximated is decreased as the optimization progresses. This makes implicit filtering particularly effective for minimizing "noisy" functions; that is, functions that have a large number of local minima, which are not necessarily known in advance. By using large difference increments early in the optimization, the algorithm may avoid local extrema and close in on the global minimum. The algorithm requires a means to compute the objective function, a feasible initial iterate, and termination criteria either in the form of a function evaluation budget or a fixed number of times that the finite-difference stencil is reduced.

Extensive numerical experiments were performed on all of the formulations of the objective function, using a variety of values for ρ. For the results presented here, we use $\rho = 10$ for $F_1(x)$ and $\rho = 10^4$ for $F_2(x)$ and $F_3(x)$. We used an initial iterate of $x_0 = (0.5, 25)$ and a budget of 60 evaluations of $F(x)$. The allowable debris, b, was 8×10^{-5} kg. We present and discuss selected results from these tests.

Table 4 contains results from the problem formulations. The data in the first column are the optimal values of (η, d_p) for each objective function. The data in the remaining columns provide the corresponding lifetime of the filter, α_{min} as described in Section 3.2, the mass of the escaped debris (ξ), and the slope of the secant line (m) for the overall pressure drop.

Objective Function (η, d_p)	t (hours)	α_{min} $(10^4$ Pa/hr)	ξ $(10^{-5}$ kg)	m $(10^5$ Pa/hr)
$F_1(0.6500, 26.9057)$	76.0	7.1749	4.6591	2.0407
$F_2(0.6500, 23.0000)$	67.8	6.8644	3.3308	2.2207
$F_3(0.6500, 28.6087)$	79.3	7.3072	5.1403	1.9747

Table 1: **A comparison of filter performance for** $F_1(x), F_2(x), F_3(x)$.

All formulations result in identical values for porosity which lie on the upper bound constraint, but the values of pore diameter differ greatly between the three. In particular, note that $F_2(x)$ offers the lowest values for both lifetime and escaped debris, $F_3(x)$ offers the highest

values for both, and $F_1(x)$ lies in the middle. It is apparent that $F_2(x)$ is the most conservative of the three in terms of controlling the escaped debris, which may mean this formulation is most sensitive to the barrier. The expense is that the lifetime is significantly shorter than that of $F_1(x)$ or $F_3(x)$

More investigation is needed to explain why $F_2(x)$ results in a pore diameter that is pushed towards the opposite bound constraint as the other two. It appears that the choice of pore diameter is highly dependent on the problem formulation. The varying results imply that the advantage of one formulation over another would be determined entirely by the preferences of a production manager, who must determine the relative importance of each objective. A more balanced solution may be reached by combining all three objectives into a single weighted objective function.

In Figure 4 we show $F_1(x)$ as it varies with the pore diameter and porosity value pairs sampled by the optimizer and we examine the corresponding lifetime of the filter for the various values of $F_1(x)$. The figure on the left shows the various choices of (η, d_p) sampled by the optimizer. The porosity is clustered about $\eta = 0.65$ while the pore diameter varies more in an attempt to find the maximum. However, a majority of the design points are clustered near the best point found, which implies that the optimizer likely converged early within the function evaluation budget.

The figure on the right shows that, as one would expect, $F_1(x)$ varies nearly linear with the lifetime of the filter until the effects of the barrier term in $F_1(x)$ become dominant. For design points where the filter lifetime is longer than 80 hours, we see there is a steep penalty for the amount of debris escaping the filter by the increase in the function value.
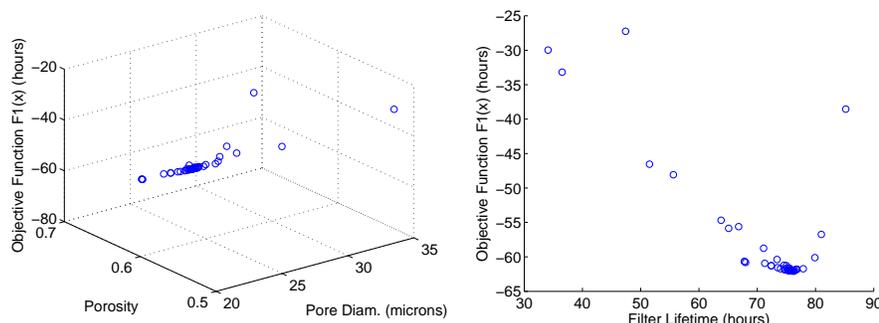
Figure 3: Scatter plots of the value of $F_1(x)$ versus (η, d_p)(left) and lifetime (right).

The same data is displayed in Figure 4 for $F_2(x)$. In this case, the optimizer sampled a wider range of values for the porosity and the pore diameter prior to convergence, but we still see a cluster of points close to the minimum. The plot on the right demonstrates that this formulation

is slightly less sensitive to the barrier approach as there is less of a linear trend.
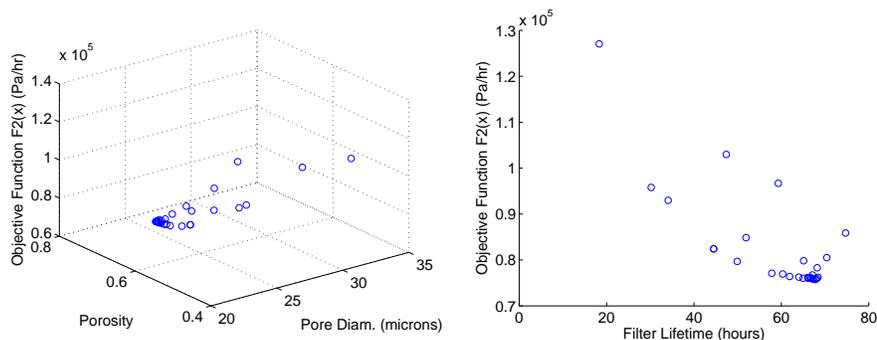
Figure 4: **Scatter plots of the value of $F_2(x)$ versus (η, d_p)(left) and lifetime (right).**

Finally, we consider the results using the third objective function. As previously mentioned, the behavior of this function is somewhat similar to that seen for $F_1(x)$. However, the optimizer sampled a wider range of design points in both η and d_p. The optimal point identified for $F_3(x)$ occurs at $x = (0.6500, 28.6087)$, compared to $x = (0.6500, 26.9057)$ for $F_1(x)$.

In analyzing the second plot in Figure 4, we see that the slope defined in (6) is minimized when t is large, which is desirable. Also, we see in the first plot in Figure 4 that the optimizer samples more values for both the porosity and pore diameter in comparison to the other two.
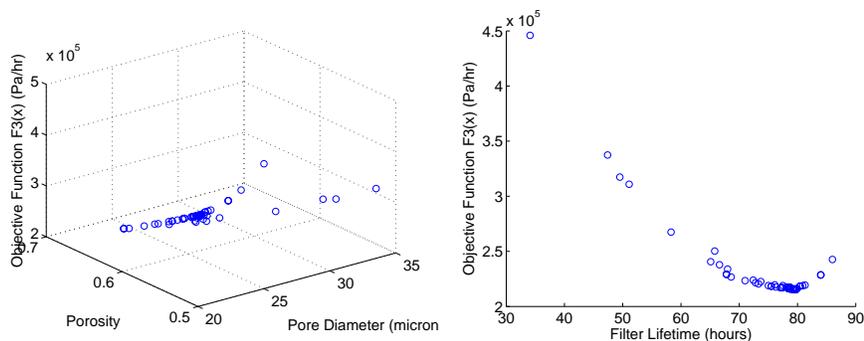
Figure 5: **Scatter plots of the value of $F_3(x)$ versus (η, d_p)(left) and lifetime (right).**

We also considered the same three formulations using an alternative derivative-free method, the DIRECT algorithm [6], for optimization and obtained essentially the same set of optimal

9

parameter sets for each formulation, implying that we have converged to at least a local (if not global) minimum.

5   CONCLUSIONS

A benefit of using a derivative-free sampling method for optimization is that intermediate function values can be used to analyze the behavior of the objective function and a wide variety of simulation methods can be used in a black-box fashion.

Our goal is to better understand the measures of filter performance by investigating new objective functions and incorporating more decision variables. Previous work with the simulator indicates there is a huge benefit to using multi-layered filters, but we know of no organized study that has fully explored the design space associated with these more complicated structures. Optimization methods coupled with the simulator give us a mechanism to understand the interactions between our variables and deposition rates. We plan to extend our current work on single objective functions to give industrial partners an effective tool with which to reduce the operating costs associated with one of the components of fiber production.

References

[1] http://www.clemson.edu/caeff.

[2] R.B. Bird, W.E. Stewart, and E.N. Lightfoot. *Transport Phenomena*. Wiley, New York, 1960.

[3] C.L. Cox, E.B Duffy, and J.B. von Oehsen. FISIM: The CAEFF integrated model for simulation of fiber and film processes. *Plastics, Rubber, & Composites: Macromolecular Engineering*, 33:426–437, 2004.

[4] C.L. Cox, E.W. Jenkins, and P.J. Mucha. Modeling of debris deposition in a polymer extrusion filter. In *Proceedings of PPS-21*, Leipzig, Germany, June 2005.

[5] D.D. Edie and C.H. Gooding. Prediction of pressure drop for the flow of polymer melts through sintered metal filters. *Industrial and Engineering Chemistry Process Design and Development*, 24:8–12, 1985.

[6] D.E. Finkel. *DIRECT Optimization Algorithm User's Guide*, March 2003.

[7] K.R. Fowler, B. McClune, E.W. Jenkins, C.L. Cox, and B. Seyfzadeh. Design analysis of polymer filtration using a multi-objective genetic algorithm. *Sep. Sci. Technol.*, 43(4):710–726, 2008.

[8] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, 1982.

[9] D.C. Hookway. How to design your deep bed polymer filter. *Filtration and Separation*, pages 161–166, February 1996.

[10] C.T. Kelley. *Iterative Methods for Optimization*. SIAM, 1999.

[11] C.T. Kelley. *User's Guide for imfil Version 0.5*, November 2005.

[12] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer-Verlag, New York, 1999.

[13] B. Seyfzadeh, D.A. Zumbrunnen, and R.A. Ross. Non-Newtonian flow and debris deposition in an extrusion filter medium. In *Proceedings of Plastics – The Lone Star*, pages 340–344. Society of Plastics Engineers, 2001.