*Proceedings of the 2014 Winter Simulation Conference*
*A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.*

# A CONTINUOUS PIECEWISE-LINEAR NHPP INTENSITY FUNCTION ESTIMATOR

David M. Nicol

Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
619 South Wright Street
Urbana, IL 61801, USA

Lawrence M. Leemis

Department of Mathematics
The College of William & Mary
200 Ukrop Way
Williamsburg, VA 23187, USA

## ABSTRACT

We consider the problem of using either grouped observations of a counting process over a period or prescribed constant mean counts to create an intensity function for a Non-Homogeneous Poisson Process (NHPP) that estimates the observed process, and satisfies several constraints. First, we require that the estimator's mean value over an observation interval be equal to the mean number of observations in that interval; second, we require that the intensity function be continuous; third, we require that the function be piecewise linear. Optionally, we may also require that the intensity value at the end of the period be identical to the intensity value at the beginning of the period, for application in contexts in which the period of interest is inherently cyclic, e.g., a day, or a week. An objective of the estimator is that it should be "smooth," which will be defined subsequently. Our approach is to define a class of continuous piecewise-linear intensity functions and formulate the problem as a constrained quadratic programming problem, approachable through the solution of a simultaneous set of linear equations. We describe the method, identify conditions under which feasible solutions are assured to exist, and study the behavior of the solutions on an example problem.

## 1 INTRODUCTION

An NHPP model is useful in a variety of settings, most notably queueing and reliability. The arrival stream to a queue often has "rush periods" that are inconsistent with a Poisson or renewal process. The fluctuating arrival rate can be captured in an NHPP model. In reliability, the rate of occurrence of failures of a repairable system typically increases with time. The NHPP model is often used to describe the probabilistic mechanism associated with the failure times of such a deteriorating system, where repair times are considered to be negligible and are not explicitly modeled. In general, we desire a stochastic model for a series of event times that occur at random points in time. The probability distribution of these event times are controlled by the non-negative intensity function $\lambda(x)$ of the NHPP, which, for example, is the arrival rate in a queueing setting and the rate of occurrence of failure in a reliability setting. NHPPs are also defined by the cumulative intensity function $\Lambda(x) = \int_0^x \lambda(y)\,dy$, which is the expected number of events that occur by time $x$; in Chen and Schmeiser (2013) one says that $\lambda(\,)$ is *mean-constrained.* Four major aspects of NHPP models are determining probabilistic properties of NHPPs, estimating the intensity function or cumulative intensity function from a data set, assessing the fit of the NHPP, and random process generation. More details concerning these aspects of NHPPs are given, for example, in Law (2007).

This paper focuses on the problem of constructing an estimate of the intensity function from a data set that consists of counts over time subintervals or prescribed target mean values over time subintervals. More specifically, we address the question of how to smooth the piecewise-constant estimator of the intensity function in a similar fashion to the I-SMOOTH estimator of Chen and Schmeiser (2013) and the piecewise-linear estimator of Nicol and Leemis (2014). The intensity function estimator that we formulate breaks

978-1-4799-7486-3/14/$31.00 ©2014 IEEE

each subinterval into equal-length pieces and determines a piecewise-linear intensity function estimate that satisfies non-negativity and is mean-constrained, while maximizing the smoothness of the estimator. The development here is not statistical in nature, but is rather a constrained optimization problem that converts target mean values, obtained from data or prescribed, to an intensity function that might more accurately reflect the population intensity function. The words "intervals," "subintervals," and "cells" are synonyms.

The attraction of a piecewise-linear intensity function is that sampling arrivals from an NHPP is computationally efficient using the inverse transform method. For an NHPP with any arbitrary cumulative intensity function $\Lambda(x)$, if $E_1$, $E_2$, ... denote the event times from a unit homogeneous Poisson process, then $\Lambda^{-1}(E_1)$, $\Lambda^{-1}(E_2)$, ... are the associated event times for the NHPP. Since the intensity function in the estimator that we are suggesting here is piecewise linear, the cumulative intensity function is piecewise quadratic. So inverting the cumulative intensity function involves just solving a quadratic equation. The inversion algorithm is synchronized in the sense that one random number produces one event time for the NHPP. The inversion algorithm is monotone in the sense that a small random number will result in a small time between events in the unit homogeneous Poisson process, which will result in a small time between events in the nonhomogeneous Poisson process. The process generation algorithm for the algorithm developed here will be given in a subsequent section.

Law (2007) gives the maximum likelihood estimator for a piecewise-constant intensity function from count data as the ratio of the total number of events observed in an interval over all realizations to the number of realizations times the length of the interval. In this paper, we denote this ratio for interval $i$ as simply $c_i$, for $i = 1, 2, \ldots, n$. The $c_i$ values will be calculated from data in most applications, but will be treated as prescribed constants for the rest of the paper.

## 2 MODEL

We choose units so that $x_0 = 0, x_1 = 1, \ldots, x_n = n$ denote the interval endpoints, and use $c_i$ to denote the average count of occurrences between $x_{i-1}$ and $x_i$, for $i = 1, 2, \ldots, n$. We will refer to $c_i$ as the "cell area." The problem of interest is to find a continuous piecewise-linear intensity function $\lambda(x)$ with the equal-area property that for every $i = 1, 2, \ldots, n$,

$$\int_{x_{i-1}}^{x_i} \lambda(x)\ dx = c_i.$$

We want $\lambda(x)$ to be relatively smooth. We use the term "cyclic context" from Chen and Schmeiser (2013) to describe an NHPP model which requires that $\lambda(0) = \lambda(n)$. When this constraint is not present, this will be referred to as the "finite-horizon" context. The cyclic context arises when the NHPP models event times over some cyclic period such as a day, or a week. From the point of smoothness, it recognizes that the function behavior near $\lambda(n)$ is close to the function behavior near $\lambda(0)$ because time "wraps around," and we want the same continuity of intensity between the last and first intervals as we do other adjacent intervals. This would be the case, for example, for modeling patient arrival times to an emergency room over a 24-hour cycle. It would be reasonable to require that the intensity-function values match at the beginning and end of a cycle: $\lambda(0) = \lambda(24)$.

Our approach will be to partition each cell into $w$ equal-width sub-cells; define $\lambda()$ at the boundary of each sub-cell; and define $\lambda()$ within sub-cell boundaries as the linear interpolation of intensities at its boundaries. The solution vector $\mathbf{y}$ thus has $n \cdot w + 1$ components, with $y_k = \lambda(x_i + j/w)$, where $i = \lfloor k/w \rfloor$ and $j = k \bmod w$, for $k = 0, 1, 2, \ldots, n \cdot w$. This is illustrated in Figure 1 for $w = 3$. The continuity of $\lambda()$ we require is achieved by construction; the non-negativity of $\lambda()$ will be a constraint on the solution vector, and, if desired, we can require $y_{n \cdot w} = y_0$ to induce a "wrap-around" solution when the model is used in the cyclic context.

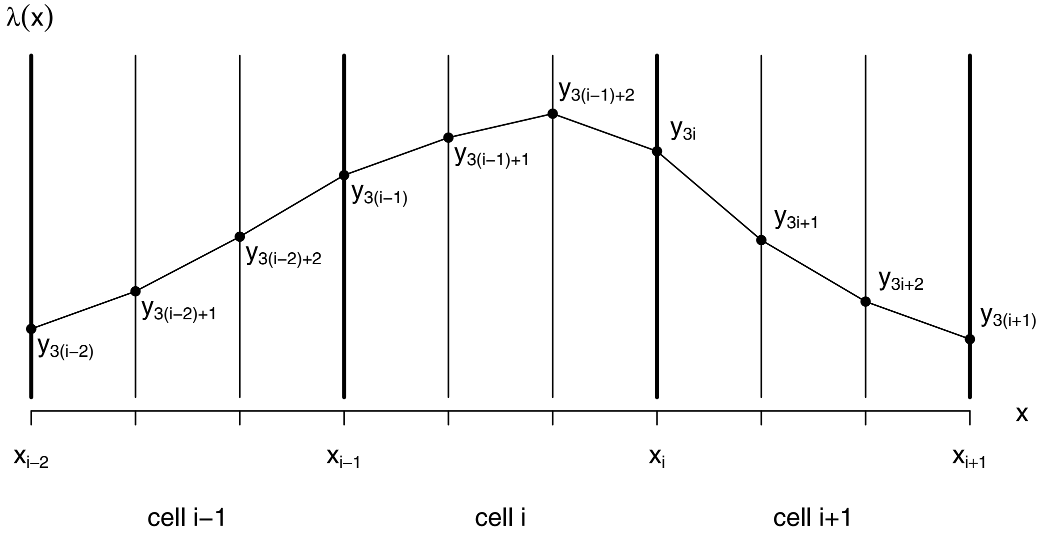

Figure 1: Intensity function structure.

### 2.1 Feasibility

We now turn to the feasibility of solutions, looking first at the feasibility of a solution over a single cell. From cell-focused results we can develop a global result.

Suppose we are given cell boundary intensities $\lambda(x_{i-1}) = y_l$ and $\lambda(x_i) = y_r$. A solution is *locally feasible* if we can find $w-1$ interior intensities such that the area under the linear segments so defined is $c_i$. It turns out that a locally feasible solution is possible if and only if $c_i$ is sufficiently large relative to $y_l$ and $y_r$.

**Lemma 1** Given intensity function values $\lambda(x_{i-1}) = y_l$ and $\lambda(x_i) = y_r$ and cell area $c_i$, a locally feasible solution satisfying the equal-area property exists if and only if $(y_l + y_r)/(2w) \le c_i$.

*Proof.* The necessity of the condition can be seen by observing that the minimal area of a locally feasible solution occurs when $\lambda(x)$ drops as rapidly as it can from $y_l$ to value 0 over a span of of $1/w$ on the $x$-axis, stays at 0 until distance $1/w$ before $y_r$, and only then rises as it eventually must to intensity $y_r$. This minimal area is $(y_l + y_r)/(2w)$.

To see that $(y_l + y_r)/(2w) \le c_i$ is a sufficient condition as well, suppose that $(y_l + y_r)/(2w) \le c_i$, and consider the solution in which all interior intensity function heights are $h$ for some real-valued, non-negative constant $h$. $(2wc_i - (y_l + y_r))/(2(w-1))$ is the solution for $h$ in the equal-area equation

$$c_i = (y_l + h)/(2w) + (w-2)h/w + (h + y_r)/(2w),$$

which is non-negative and making this a locally feasible solution. □

*Global* feasibility is achieved when each cell has a locally feasible solution, subject to the continuity constraint. Imagine a solution in which $\lambda(x_{i-1}) = (c_{i-1} + c_i)/2$ for $i = 2, 3, \ldots, n$; we specify the intensities at the cell boundaries as the average of the adjacent cell areas. Lemma 1 indicates that a locally feasible solution over $x_{i-1}$ to $x_i$ exists if $((c_{i-1} + c_i)/2 + (c_i + c_{i+1})/2)/(2w) \le c_i$. This occurs if $c_{i-1} + c_{i+1} \le (4w-2)c_i$, leading to Lemma 2 below, which codifies this intuition that a global feasible solution exists if the cell areas do not change “too fast.”

**Lemma 2** Suppose that $c_{i-1} + c_{i+1} \le (4w-2)c_i$ for $i = 2, 3, \ldots, n-1$. Then there exists an intensity function $\lambda()$ which is continuous and piecewise linear with each cell having a locally feasible solution.

Observing that $w$ is a user-selected parameter, a corollary to Lemma 2 is that for any given problem instance we can always find a large enough $w$ to ensure the existence of a globally feasible solution. Lemma 3 provides guidance concerning the value of $w$ required to satisfy the non-negativity constraint.

**Lemma 3** Define

$$w^* = \max_i \left\{ \left\lceil \frac{c_{i-1} + 2c_i + c_{i+1}}{4c_i} \right\rceil \right\}.$$

Then constructing a solution using $w^*$ sub-intervals per cell ensures a globally feasible solution.

*Proof.* The argument to the maximum function is the integer solution for $w$ to the inequality expressed in Lemma 2. The maximum identifies the bottleneck limiting global feasibility; eliminate the bottleneck and the conditions of Lemma 2 are met. □

## 2.2 Smoothness

Experience experimenting with estimators showed us the potential for the computed intensity function to make sharp turns. Given the relative coarseness of the $c_i$ values, rapid changes beyond what is needed for consistency with that data seems to us to be an overly strong statement about the underlying model's behavior. The motivation for the present paper is to consider function smoothness as a desirable characteristic and find ways of computing smooth functions.

At each of our functions' knot points an angle is formed between the line segments joined at the knot. The larger the magnitude of that angle, the less smooth is the function at that knot point. In fact, if we score the function smoothness at a knot point as the square of the angle's magnitude and score the overall function smoothness as the sum of the knots' smoothness scores, we place the problem of finding an optimally smooth function in a tractable mathematical setting. Thus we define $\beta(\mathbf{y})$ to be the overall function smoothness. In the cyclic context, we equate $y_0 = y_{n\cdot w}$ and include comparison of the first and last segments:

$$\beta(\mathbf{y}) = w^2 \sum_{i=0}^{n\cdot w} \left( y_{(i+1) \bmod n\cdot w} - 2y_i + y_{(i-1) \bmod n\cdot w} \right)^2;$$

in the finite-horizon context, we take $y_{n\cdot w}$ as a free variable:

$$\beta(\mathbf{y}) = w^2 \sum_{i=1}^{n\cdot w-1} \left( y_{i+1} - 2y_i + y_{i-1} \right)^2.$$

The important point is that $\beta(\mathbf{y})$ is quadratic in $\mathbf{y}$, which means that we're able to formulate the problem of finding a solution that minimizes $\beta(\mathbf{y})$ as a constrained quadratic program. Before turning to formalization of these constraints, we consider expression of $\beta(\mathbf{y})$ in standard matrix form for quadratic equations.

There is a pleasing symmetry in $\beta(\mathbf{y})$ in the cyclic context; every component of $\mathbf{y}$ has the same dependency structure, on variables with indices within one and two values (modulo $n\cdot w$) of the variable's own. In the expansion of $\beta(\mathbf{y})$, the quadratic coefficient of $y_i$ is $6w^2$, and the linear coefficient is $(2y_{i-2} - 8y_{i-1} - 8y_{i+1} + 2y_{i+2})w^2$, where we understand the indexing arithmetic to be modulo $n\cdot w$. We transform these coefficients for expression in quadratic matrix form $(1/2)\ \mathbf{y}^T Q\ \mathbf{y}$:

$$Q_{i,j} = \begin{cases} 12w^2 & \text{if } i = j \\ -8w^2 & \text{if } j = (i+1) \bmod (n\cdot w), \text{ or } j = (i-1) \bmod (n\cdot w) \\ 2w^2 & \text{if } j = (i+2) \bmod (n\cdot w), \text{ or } j = (i-2) \bmod (n\cdot w). \end{cases}$$

For example, when $n = 4$ and $w = 2$, the circulant $8 \times 8$ matrix $Q$ in the cyclic context is

$$\begin{bmatrix} 48 & -32 & 8 & 0 & 0 & 0 & 8 & -32 \\ -32 & 48 & -32 & 8 & 0 & 0 & 0 & 8 \\ 8 & -32 & 48 & -32 & 8 & 0 & 0 & 0 \\ 0 & 8 & -32 & 48 & -32 & 8 & 0 & 0 \\ 0 & 0 & 8 & -32 & 48 & -32 & 8 & 0 \\ 0 & 0 & 0 & 8 & -32 & 48 & -32 & 8 \\ 8 & 0 & 0 & 0 & 8 & -32 & 48 & -32 \\ -32 & 8 & 0 & 0 & 0 & 8 & -32 & 48 \end{bmatrix}.$$

Matrix $Q$ is slightly different in the finite-horizon context. To begin with, it has one more row and one more column because it has one more variable. The pattern above is retained (without the wrap-around effect) for all but the first two and last two rows. Here (and only here) the effect of $y_0 \neq y_{n \cdot w}$ is reflected in the equations, and we have different entries for the following elements:

$$\begin{aligned} Q_{0,0} &= 2w^2, Q_{0,1} = -4w^2, \\ Q_{1,0} &= -4w^2, Q_{1,1} = 10w^2, \\ Q_{n \cdot w+1, n \cdot w+1} &= 2w^2, Q_{n \cdot w+1, n \cdot w} = -4w^2, \\ Q_{n \cdot w, n \cdot w+1} &= -4w^2, Q_{n \cdot w, n \cdot w} = 10w^2. \end{aligned}$$

For the $n = 4$ and $w = 2$ case, the symmetric, pentadiagonal $9 \times 9$ matrix $Q$ in the finite-horizon context is

$$\begin{bmatrix} 8 & -16 & 8 & 0 & 0 & 0 & 0 & 0 & 0 \\ -16 & 40 & -32 & 8 & 0 & 0 & 0 & 0 & 0 \\ 8 & -32 & 48 & -32 & 8 & 0 & 0 & 0 & 0 \\ 0 & 8 & -32 & 48 & -32 & 8 & 0 & 0 & 0 \\ 0 & 0 & 8 & -32 & 48 & -32 & 8 & 0 & 0 \\ 0 & 0 & 0 & 8 & -32 & 48 & -32 & 8 & 0 \\ 0 & 0 & 0 & 0 & 8 & -32 & 48 & -32 & 8 \\ 0 & 0 & 0 & 0 & 0 & 8 & -32 & 40 & -16 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & -16 & 8 \end{bmatrix}.$$

The general quadratic form requires us to include terms that are constants times a single variable, and ordinary constants:

$$\beta(\mathbf{y}) = (1/2)\ \mathbf{y}^T Q\ \mathbf{y} + \mathbf{s}^T \mathbf{y} + K. \tag{1}$$

For our specific function there are no non-zero terms other than what is captured by $Q$, so that $K$ is 0 and $\mathbf{s}$ is a zero-vector with as many elements as there are solution variables. We will refer to $\mathbf{s}$ again when we solve the optimization problem.

### 2.3 Constraints

When $\lambda()$ is used to sample NHPPs, we want the expected number of counts that occur between time $x_{i-1}$ and $x_i$ to be the cell area, $c_i$. The expected number of counts induced by $\lambda()$ is the area under the curve, which by our construction is the area under $w$ linear segments. It is not difficult to see that the area under a linear segment passing from height $u$ to height $v$ over a span of length $1/w$ in the $x$-axis is $(1/w)(1/2)(u+v) = (u+v)/(2w)$. Applying this observation to each of $w$ segments, the equal-area constraint for cell $i$ gives us

$$c_i = \frac{\sum_{j=0}^{w-1} (y_{i \cdot w+j} + y_{i \cdot w+j+1})}{2w}, \text{ or equivalently, } y_{i \cdot w} + 2 \sum_{j=1}^{w-1} y_{i \cdot w+j} + y_{(i+1) \cdot w} = 2wc_i,$$

and we have one such constraint for each cell. We express this in matrix form as $E\ \mathbf{y} = \mathbf{d}$, where $\mathbf{y}$ is the solution vector, $\mathbf{d}$ is a vector with $n$ elements: $d_i = 2 \cdot w \cdot c_i$, and $E$ is a $n \times (n \cdot w)$ sparse matrix in the cyclic context and a $n \times (n \cdot w + 1)$ sparse matrix in the finite-horizon context. For all $i$, row $i$ of $E$ is zero from column 0 to column $i \cdot w - 1$, has a '1' in column $i \cdot w$, has a '2' in columns $i \cdot w + 1$ through $i \cdot w + w - 1$, a '1' in column $(i+1) \cdot w$, with the remaining columns being 0. When there are $n = 3$ cells and $w = 3$ sub-intervals within a cell, the $3 \times 9$ matrix $E$ in the cyclic case is

$$E = \begin{bmatrix} 1 & 2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 2 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 \end{bmatrix}$$

and the $3 \times 10$ matrix $E$ in the finite-horizon case is

$$E = \begin{bmatrix} 1 & 2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 1 \end{bmatrix}.$$

We have also the non-negativity constraint, which in vector form is $\mathbf{y} \geq \mathbf{0}$, where $\mathbf{0}$ is a vector of 0's, and the inequality comparison is done on a component-by-component basis.

## 3 OPTIMAL SMOOTHNESS

We have now all the machinery in place to express the problem and solve it. We seek $\mathbf{y}$ that minimizes the quadratic form expressed as Equation (1), subject to the equality constraint $E\ \mathbf{y} = \mathbf{d}$ and the inequality constraint $\mathbf{y} \geq \mathbf{0}$.

There is a large literature on the general solution to this problem, e.g., Wolfe (1959), Boyd and Vandenberghe (2004), Dostal (2009), and Fletcher (2013). For the purposes of conceptual demonstration, our solution approach will be to ignore (temporarily) the inequality constraint and proceed with only the linear equality constraint. When $Q$ is well-behaved (and our $Q$ is) a standard technique (Boyd and Vandenberghe 2004) in such a situation is apply to Karush–Kuhn–Tucker conditions that use Lagrange multipliers to develop an equivalent system

$$\begin{bmatrix} Q & E^T \\ E & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{g} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{d} \end{bmatrix},$$

where $\mathbf{g}$ is a vector of Lagrangian multipliers to be solved for as well. Solution of this system of linear equations yields the vector $\mathbf{y}$ that minimizes Equation (1), subject to the equal-area constraint.

This is an ordinary numerical problem that can be handled by any linear algebra software package. Existence of a solution using this technique can be an issue owing to potential numerical issues, but we can be confident that the quadratic programming problem has a unique solution to be found, and if the method of Lagrange multipliers does not work out, there are a host of other options available. An R function for calculating $\mathbf{y}$ in the case when the non-negativity constraint is not violated is given in the appendix.

If all components of $\mathbf{y}$ are non-negative then we've found the optimal solution subject to all our constraints. Intuition suggests that if all the cell areas have values well away from 0 then there can be no advantage with respect to smoothness to have a negative intensity value, and the solution found will be non-negative. In our experience we have yet to encounter a problem derived from real data where the unconstrained solution has negative componenet. Still, if one or more components of the $\mathbf{y}$ so found are negative, we can either use more sophisticated solution techniques that explicitly account for the non-negativity constraint, or find a potentially suboptimal solution by forcing the negative intensity values to be zero, reformulate $Q$ and $E$ to reflect this assignment, and solve again using the same technique.

## 4 EXAMPLE

We now illustrate the results of this algorithm on an example. Call center counts over a 13-hour period were collected over five realizations in Leemis (2004). A preliminary analysis suggests that the five realizations can be considered to be independent observations from a single target NHPP. There are $n = 13$ cells with areas $c_1, c_2, \ldots, c_{13}$ given below.

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 68.6 | 126 | 140.2 | 139.4 | 125.4 | 115 | 126.8 | 140.2 | 140 | 119.4 | 100.6 | 70.6 | 70.2 |

The areas of the first and last cell are close enough to be consistent with a circular model (even though 13 hours does not span a circular epoch) and so we will compute the circular solution for the purposes of illustration.

Figure 2 plots the input cell areas as constants over their respective domains and four solutions, two when the model is non-cyclic, two when it is considered to be cyclic. For both model types we used $w = 2$ and $w = 4$. The key gives the average absolute value of the difference in slopes of adjacent segments for each case. One interesting (and surprising result) for this set is that while there is a very noticeable drop in average slope change going from 2 subintervals per cell to 4, the magnitude of those averages is quite small, with little impact on the visual perception of the curve. This could be important if such an observation holds true more widely, since the time and memory costs of a linear system solution grow more than linearly in the number of variables. There is a visual impact of using a non-cyclic model versus a cyclic one. If the underlying model *is* cyclic but the non-cyclic solution is used, there is a large gap between the first and last intensity (35 vs. 80), showing that the cyclic formulation can have an important effect on model behavior.

Examining the finite-horizon case more closely, a potentially problematic situation occurs in the final cell. There is an increase in the arrival rate of phone calls between 8 PM and 9 PM that might not be warranted for the traffic at the call center. The final two $c_i$ values, $c_{12} = 70.6$ and $c_{13} = 70.2$, indicate

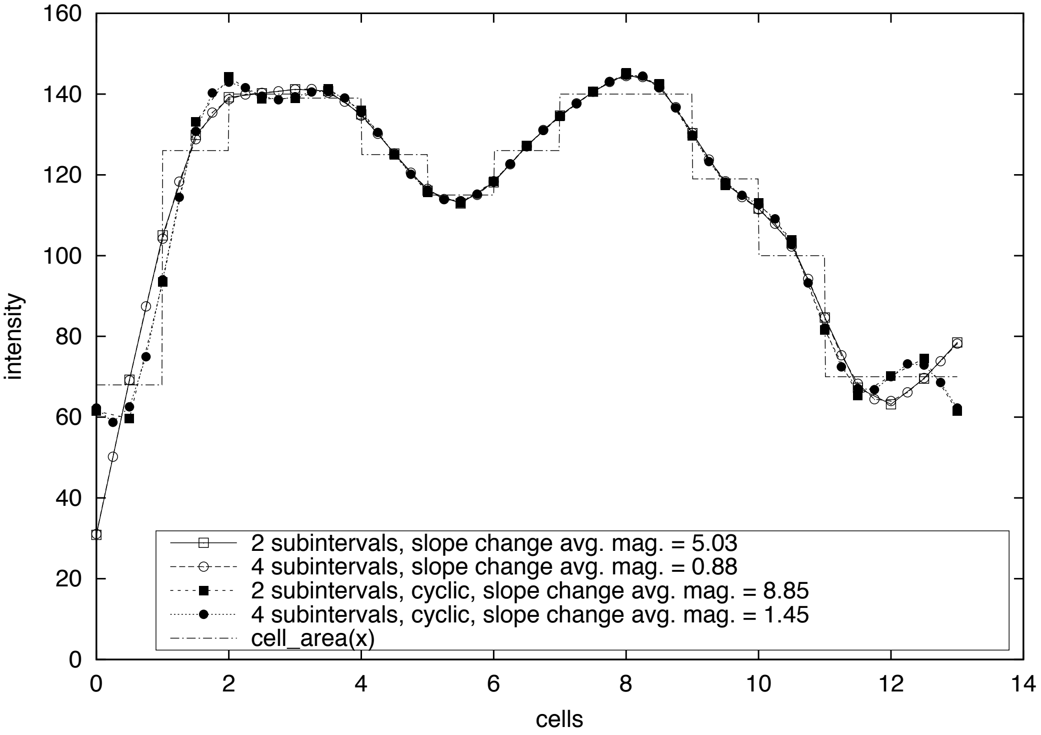

Figure 2: Example solutions.

a trailing off of call volume, rather than a surge, at the end of the day. Both I-SMOOTH and the first algorithm from Nicol and Leemis (2014) also reflect this surge at the end of the day. The second algorithm in Nicol and Leemis (2014) provides a way to avoid this problem.

## 5 PROCESS GENERATION

We now return to the problem of generating a process from the estimated NHPP that was briefly mentioned in the Introduction. Klein and Roberts (1984) consider process generation from an NHPP with a piecewise-linear intensity function via inversion; the derivation here parallels their work.

Inversion will be used to generate the event times for the NHPP as $\Lambda^{-1}(E_1)$, $\Lambda^{-1}(E_2)$, ..., where $E_1$, $E_2$, ... are the event times of a unit homogeneous Poisson process. The algorithm for generating event times associated with the estimated NHPP is identical in the cyclic and finite horizon contexts, so that distinction can be dropped in this section. The $x$-coordinates of the knot points for the piecewise-linear intensity function estimator are

$$0, \frac{1}{w}, \frac{2}{w}, \ldots, \frac{nw}{w}.$$

The $y$-coordinates (e.g., the heights) of the knot points for the piecewise-linear intensity function estimator are

$$y_0, y_1, y_2, \ldots, y_{nw},$$

which have been calculated by the constrained optimization problem. (In the cyclic context, $y_0 = y_{nw}$.) The slope of the $j$th linear segment of the piecewise-linear intensity function is $(y_j - y_{j-1})/(1/w) = w(y_j - y_{j-1})$. Since the point $(j/w, y_j)$ is the right-hand endpoint of the $j$th linear segment, the intensity function is

$$\lambda(x) = y_j + w(y_j - y_{j-1})x - j(y_j - y_{j-1}) \qquad \frac{j-1}{w} < x \le \frac{j}{w},$$

for $j = 1, 2, \ldots, nw$. The cumulative intensity function $\Lambda(x)$, on the other hand, is piecewise quadratic with the same $x$-coordinates for the knot points as the intensity function. The heights of the knot points for the cumulative intensity function are

$$0, \frac{y_0 + y_1}{2w}, \frac{y_0 + y_1}{2w} + \frac{y_1 + y_2}{2w}, \ldots, \frac{y_0 + 2(y_1 + y_2 + \cdots + y_{nw-1}) + y_{nw}}{2w}.$$

The height of the last knot point must equal $\sum_{i=1}^{n} c_i$ because of the equal-area constraint. For notational compactness, the heights of the cumulative intensity function at the knot points will be referred to as $\Lambda_0$, $\Lambda_1$, $\Lambda_2$, ..., $\Lambda_{nw}$.

Generation of samples of the NHPP are driven by samples from a unit homogeneous Poisson process, each simulated event time of which is transformed as follows. The $i^{th}$ sample event time of the NHPP is the solution $x$ in the equation $E_i = \int_0^x \lambda(t)\, dt$, where $E_i$ is the time of the $i^{th}$ event in the unit homogeneous Poisson process. We know and can pre-compute the values $\Lambda_k = \int_0^{y_k} \lambda(t)\, dt$ for $k = 1, 2, \ldots, nw$, so that given $E_i$ we can use an efficient search algorithm to find the smallest $j$ such that $E_i \le \Lambda_j$. For $\frac{j-1}{w} < x \le \frac{j}{w}$ and $j = 1, 2, \ldots, nw$, the cumulative intensity function is

$$\begin{aligned}
\Lambda(x) &= \Lambda_{j-1} + \int_{(j-1)/w}^{x} \big(y_j + w(y_j - y_{j-1})t - j(y_j - y_{j-1})\big)dt \\
&= \Lambda_{j-1} + \left[\frac{w}{2}(y_j - y_{j-1})t^2 + y_j t - jt(y_j - y_{j-1})\right]_{(j-1)/w}^{x} \\
&= \Lambda_{j-1} + \frac{w}{2}(y_j - y_{j-1})x^2 + \big(y_j - j(y_j - y_{j-1})\big)x - \frac{j-1}{w}\left(\frac{(j-1)(y_j - y_{j-1})}{2} + y_j - j(y_j - y_{j-1})\right).
\end{aligned}$$

When this expression is equated to $E_i$, the resulting equation

$$E_i = \Lambda_{j-1} + \frac{w}{2}(y_j - y_{j-1})x^2 + \big(y_j - j(y_j - y_{j-1})\big)x - \frac{j-1}{w}\left(\frac{(j-1)(y_j - y_{j-1})}{2} + y_j - j(y_j - y_{j-1})\right)$$

is a quadratic equation that can be solved by the quadratic formula with

$$a = \frac{w}{2}(y_j - y_{j-1}), b = y_j - j(y_j - y_{j-1}),$$

and

$$c = \Lambda_{j-1} - E_i - \frac{j-1}{w}\left(\frac{(j-1)(y_j - y_{j-1})}{2} + y_j - j(y_j - y_{j-1})\right).$$

The sign choice in the ± part of the quadratic formula should be made based on the sign of $y_j - y_{j-1}$ (use $+$ when this quantity is positive and $-$ when this quantity is negative). If $y_j - y_{j-1} = 0$, then the quadratic portion of $\Lambda(t)$ happens to be linear on this subinterval, so the event time generated associated with $E_i$ is

$$x = \frac{j-1}{w} + \frac{E_i - \Lambda_{j-1}}{y_j}.$$

## 6 RELATED WORK

Our original interest in this problem was motivated by I-SMOOTH, described first in a 2011 WSC paper by Chen and Schmeiser (2011) and later in Chen and Schmeiser (2013). The foundational idea of the SMOOTH algorithm is to refine a piecewise-constant intensity function with $k$ segments into one with $2k$ segments, as follows. For segment $i$ with constant intensity $\lambda_i$, the segment is transformed into two, one with rate $c_i - \gamma_i$, the other with rate $c_i + \gamma_i$. The problem considered is choice of $\gamma_i$ for all $i$. For any given choice of $\gamma$'s one can compute the difference in intensity levels of adjacent intervals. The measure of smoothness is the sum of the squares of these differences; the $\gamma$ sought minimize this sum. The problem is that of minimizing a quadratic objective function, subject to constraints that keep the intensity levels non-negative. This problem is solved repeatedly, doubling the number of segments with each step, until the user is satisfied with the result.

Our initial work (Nicol and Leemis 2014) considered two adaptations of the problem. First we sought solutions that are piecewise linear rather than piecewise constant, and second we sought solutions that use fewer segments than I-SMOOTH needs to achieve (visually) smooth curves. Our insight was that with piecewise-linear intensity functions, the equal-area constraint can sometimes (with judiciously chosen intensities at the cell boundaries) be met with one line segment over a cell, can usually be met using only two segments, and can always be met using three segments. While efficient with respect to numbers of segments, our solutions sometimes exhibited non-intuitive rapid changes in segment slope. That observation lead to the present paper, where we focus on smoothness rather than minimization of function segments. There are parallels between the smoothness definition of I-SMOOTH, and that which we employ. While our formulation is closer to the definitions of differential geometry (because our intensity function is continuous and so explicitly has slopes) the conceptual approach of the present paper is very much in line with Chen and Schmeiser (2013). Chen and Schmeiser (2014) use a continuous, piecewise quadratic function to approximate the intensity function. In addition to the equal-area and non-negativity constraints, they match the first derivatives at the knot points. For every interval that results in a negative intensity function, non-negativity is enforced by taking the maximum of zero and the quadratic-function value, modifying the quadratic to maintain the equal-area constraint.

## 7 SUMMARY

Estimation of NHPP intensity functions is a practical problem. An over-riding constraint on estimators is that the mean value of the number of occurrences over a period of observation be equal to mean number of actual observations in that period. An obvious estimator computes a constant intensity over each period, but this leads to sometimes large step function changes in the intensity passing between periods. Past work has looked at smoothing this: Chen and Schmeiser (2013) refines the number of periods, computing a piecewise-constant intensity function with smaller steps; Nicol and Leemis (2014) propose piecewise-linear intensity functions, aiming to minimize the number of function segments. Closed-form solutions are available in the case when non-negativity is irrelevant. This paper formulates the problem of maximizing smoothness using piecewise-linear intensity functions. We show that the problem can be approached using easily accessible mathematical tools, and observe empirically that visually smooth intensity functions might be discovered using as few as two line segments per period.

## ACKNOWLEDGMENTS

This work was supported in part by the Boeing Corporation.

## APPENDIX: R Code

The R function named `nhpp` given below requires three arguments: `cc`, a vector of length $n$ containing $c_1, c_2, \ldots, c_n$; `w`, the number of equal-width subintervals per cell; and `cyclic`, a Boolean argument which is `TRUE` in the cyclic context and `FALSE` in the finite-horizon context.

```
nhpp = function(cc, w, cyclic = TRUE) {
  n = length(cc)
  if (cyclic == TRUE) {
    m = n * w
    Q = diag(rep(12, m))
    Q[cbind(2:m, 1:(m - 1))] = Q[cbind(1:(m - 1), 2:m)]
                             = Q[m, 1] = Q[1, m] = -8
    Q[cbind(3:m, 1:(m - 2))] = Q[cbind(1:(m - 2), 3:m)] = Q[m - 1, 1]
                             = Q[m, 2] = Q[1, m - 1] = Q[2, m] = 2
  } else {
    m = n * w + 1
    Q = diag(rep(12, m))
    Q[cbind(2:m, 1:(m - 1))] = Q[cbind(1:(m - 1), 2:m)] = -8
    Q[cbind(3:m, 1:(m - 2))] = Q[cbind(1:(m - 2), 3:m)] = 2
    Q[1, 1] = Q[m, m] = 2
    Q[2, 2] = Q[m - 1, m - 1] = 10
    Q[1, 2] = Q[2, 1] = Q[m, m - 1] = Q[m - 1, m] = -4
  }
  Q = Q * w ^ 2
  if (cyclic == TRUE) {
    E = matrix(0, n, m + 1)
    rowpat = c(1, rep(2, w - 1), 1)
    for (i in 1:n) {
      j = (i - 1) * w + 1
      E[i, j:(j + w)] = rowpat
    }
    E = E[ , -(m + 1)]
    E[n, 1] = 1
  } else {
    E = matrix(0, n, m)
    rowpat = c(1, rep(2, w - 1), 1)
    for (i in 1:n) {
      j = (i - 1) * w + 1
      E[i, j:(j + w)] = rowpat
    }
  }
  A = cbind(Q, t(E))
  B = cbind(E, matrix(0, n, n))
  C = rbind(A, B)
  rhs = c(rep(0, m), 2 * w * cc)
  yg = solve(C, rhs)
  return(yg[1:m])
}
```

## REFERENCES

Boyd, S., and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

Chen, H., and B. Schmeiser. 2013. “I-SMOOTH: Iteratively Smoothing Mean-Constrained and Nonnegative Piecewise-Constant Functions”. *INFORMS Journal on Computing* 25:432–445.

Chen, H., and B. W. Schmeiser. 2011. “I-SMOOTH: Iteratively Smoothing Piecewise-constant Poisson-Process Rate Functions”. In *Proceedings of the 2011 Winter Simulation Conference*, edited by S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. C. Fu, 469–480. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chen, H., and B. W. Schmeiser. 2014. “Piecewise-Quadratic Rate Smoothing: The Cyclic Context”. In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc. To appear.

Dostal, Z. 2009. *Optimal Quadratic Programming Algorithms*. Springer.

Fletcher, R. 2013. *Quadratic Programming*. Wiley.

Klein, R. W., and S. D. Roberts. 1984. “A Time Varying Poisson Arrival Process Generator”. *Simulation* 43 (4): 193–195.

Law, A. M. 2007. *Simulation Modeling & Analysis*. 4th ed. New York: McGraw-Hill, Inc.

Leemis, L. M. 2004, July. “Nonparametric Estimation and Variate Generation for a Nonhomogeneous Poisson Process from Event Count Data”. *IIE Transactions* 36 (12): 1155–1160.

Nicol, D.M. and Leemis, L.M 2014. “Continuous Piecewise-Linear Intensity Function Estimation for Nonhomogeneous Poisson Process Count Data”. Technical Report, Department of Mathematics, The College of William & Mary.

Wolfe, P. 1959. “The simplex method for quadratic programming”. *Econometrica* 27 (3): 382–398.

## AUTHOR BIOGRAPHIES

**DAVID M. NICOL** is the Franklin W. Woeltge Professor of Electrical and Computer Engineering at the University of Illinois at Urbana–Champaign, and is Director of the Information Trust Institute. He holds a B.A. in mathematics from Carleton College (1979), and M.S. and Ph.D. degrees in computer science from the University of Virginia (1983, 1985). Prior to joining UIUC, he taught at The College of William & Mary, and Dartmouth College. He has served in many roles in the simulation community (e.g., Editor-in-Chief of ACM TOMACS, General Chair of the Winter Simulation Conference Executive Board of the WSC), was elected Fellow of the IEEE and Fellow of the ACM for his work in discrete-event simulation, and was the inaugural recipient of the ACM SIGSIM Distinguished Contributions award. His current research interests include application of simulation methodologies to the study of security in computer and communication systems, and any fun little math problem that the field of DES throws his way that he can work on during interminable administrative meetings. His email address is dmnicol@illinois.edu.

**LAWRENCE M. LEEMIS** is a Professor in the Department of Mathematics at the College of William & Mary. He has formerly held faculty positions at The University of Oklahoma and Baylor University. He holds a B.S. in mathematics (1978), M.S. in mathematics (1980), and Ph.D. in industrial engineering (1984) from Purdue University. His email address is leemis@math.wm.edu.