Estimating Transient Surface Heating using a Cellular Automaton Energy-transport Model

Vallorie J. Peridier
College of Engineering, Temple University,
1947 North Twelfth Street,
Philadelphia, PA 19122 USA

The use of subsurface temperature data to estimate an unknown transient surface-heating event is a classic, and practical, inverse problem in engineering. This paper describes a new inverse heat-transfer estimation procedure that is formulated using a cellular automaton model of energy transport into the medium. The quantitative heat-flux estimates generated by this new scheme appear to have accuracies comparable to conventional inverse procedures; consequently, this new method may prove a useful alternative and/or confirmational strategy in measurement situations where the data is noise-ridden or the heating event is sporadic.

1. Introduction

1.1 The model problem

To estimate an object's heat-flux absorption engineers utilize subsurface medium-embedded sensors such as thermocouples to measure the corresponding transient temperatures. The schematic of the model problem shown in Figure 1 employs notation that is commonly used in heat transfer texts (e.g., [1]).

In the model problem, the transient heat-flux event $q''(t)$ [W/m^2] is indirectly detected with temperature data T_i taken within the medium, a distance r from the surface, at regular sampling intervals $\triangle t$. The experiment is generally designed with a substrate medium of sufficient thermal capacity and thickness L that the "back" side of the object, at $x = L$, remains effectively at the substrate initial temperature T_0 throughout the transient event. In this application the substrate medium serves as a so-called "heat sink" that absorbs heat in a unidirectional fashion, and thus the one-dimensional idealization depicted in Figure 1 is a satisfactory approximation to the physical process.

1.2 The corresponding direct and inverse mathematical problems

The mathematical problem statement which governs the model problem of Figure 1 is the one-dimensional unsteady thermal diffusion equa-

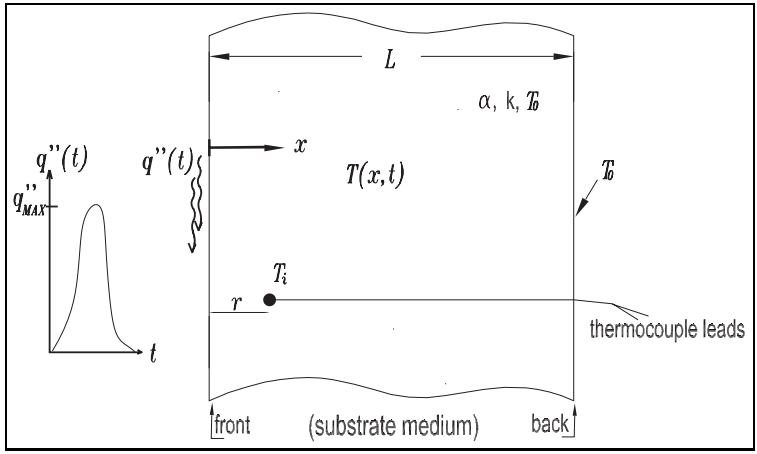

Figure 1. Schematic of the one-dimensional model problem. The unknown surface heating $q''(t)$ has dimensions [W/m^2]. The thermocouple that records temperature data T_i is located a distance r from the heated front surface. The medium has a characteristic thickness $L > r$, and material properties thermal conductivity k [W/m°C] and thermal diffusivity α [m^2/s]. The entire substrate medium is initially at temperature T_0, and the back (unheated) surface of the substrate effectively remains at temperature T_0 throughout the transient.

tion [1]:

$$\begin{aligned} \alpha \frac{\partial^2 T}{\partial x^2} &= \frac{\partial T}{\partial t}, \quad 0 \le t, \quad 0 \le x \le L; \\ T|_{t=0} &= T|_{x=L} = T_0; \\ q''(t) &= -k \left. \frac{\partial T}{\partial x} \right|_{x=0}. \end{aligned} \tag{1}$$

If the heat flux $q''(t)$ were known, these equations constitute the direct problem for $T(x,t)$ and the solution is unique.

However, in this application all that is known is T_i, the temperature data at a single location $x = r$, and the objective is to deduce $q''(t_i)$. Deducing an initiating configuration (here, $q''(t_i)$) from an observed result (here, T_i) is called an "inverse problem" and often the solution is not unique.[1] Inverse problems are common in engineering, and although they have nonunique solutions, analysis can nevertheless yield relevant engineering-design information. For example, in this model problem, knowledge of the exact temporal shape of $q''(t)$ is not really necessary if the objective is to estimate limiting-behavior parameters

[1]Note for example, in the model problem of Figure 1, that various forms of surface heating $q''(t)$ would yield effectively the same temperature data at $x = r$.

such as the peak heat-flux value $q''_{\max}$, or the approximate duration of the transient.

On the other hand, the lack of unique solutions for inverse problems does of course pose computational difficulties, and the conventional tactic is to devise a problem-specific optimized-search strategy. For example, say in the model problem that the temperature data T_i has N points: the corresponding iterative optimization strategy would entail searching for N total free parameters, say q''_i, each representing the average q'' in the "ith" time step. Such an iterative procedure would involve doing a complete solution of equation (1) with a progressively-optimized guess for q''_i in each iteration. The ultimate objective, then, would be an optimized set of q''_i parameters that produces the best fit of the calculated values $T|_{x=r}(t_i)$ as compared to the measured temperature time-series data T_i.[2] Thus, even for very basic arrangements such as Figure 1 the conventional strategy for obtaining inverse-problem parameter estimates is hardly trivial.

1.3 Specific objectives of this study

This study was undertaken as a preliminary investigation of cellular automaton modeling procedures within the larger context of inverse problems in engineering. While engineering practice is riddled with both measurement and design problems of the inverse type, the model problem of Figure 1 is a particularly attractive candidate for preliminary study because of the following.

- It is one of the simplest inverse problems, yet it has features representative of more complex situations.
- Test data for evaluating the method can be readily generated from the direct problem statement of equation (1).

Thus, here are the two specific objectives of this study.

1. To devise an inverse procedure for estimating heat flux, based on a cellular automaton energy-transport model.
2. To determine whether, using a cellular automaton inverse method, it is possible to obtain quantitative estimates of comparable accuracy to conventional inverse schemes which, in turn, are based on the exact solution of equation (1).

[2]The large number of free unconstrained parameters q''_i, the requirement that equation (1) be solved at all spatial and temporal locations at each iteration, and the sensitivity of this method to data error and noise, especially at small times, are difficulties encountered in practice. See Beck *et al.* [2] for specifics.

2. Analysis

2.1 Time discretization notation

The sampling time step $\triangle t$ of the embedded temperature sensor effectively discretizes the problem in time. Since $q''(t)$ cannot be determined to a finer time granularity than is present in the data T_i, we therefore seek:

$$q''(t_i), \qquad t_i \equiv \triangle t(i-1), \qquad 1 \le i \le N. \tag{2}$$

2.2 Dimensionless variables

1. Heat flux.
Let:

$$q''(t_i) = q''_{\max} \cdot f(t_i), \tag{3}$$

here $f(t)$ is taken without approximation to be a dimensionless scaled quantity so that $0 \le f(t) \le 1$.

Now, in view of the cellular automaton approach, the continuous function f is approximated by a boolean vector $\{q_i\}$, that is,

$$q''(t_i) \simeq q''_{\max} \cdot \{q_i\} = q''_{\max} \cdot \{q_i, q_2, \ldots, q_i, \ldots, q_N\}, \tag{4}$$

where the q_i are dimensionless and have value either {0,1}.

In this study the "solution," for a given input time-series T_i, entails finding both: (*i*) the magnitude of the surface heat flux $q''_{\max}$, and (*ii*) its approximate temporal distribution vector $\{q_i\}$.

2. Space and temporal scales.
The temperature-sensor depth r of Figure 1 and the thermophysical properties of the substrate provide the temporal and spatial scales, *viz*:

$$\begin{aligned} \xi &= \frac{x}{r}, \\ \tau &= t\frac{\alpha}{r^2}. \end{aligned} \tag{5}$$

Here, α is the thermal diffusivity[3] [m^2/s], and the characteristic length-scale r is the distance of the thermocouple from the surface.

Note that in the dimensionless variables so defined the temperature sensor is located at $\xi = 1$.

[3]Thermal diffusivity is defined as $\alpha = k/\rho c$, where k [W/m°C] is the thermal conductivity, ρ [kg/m^3] is the mass density, and c [J/kg°C] is the material heat capacity.

3. Temperature scale.
Let θ be a scaled dimensionless temperature variable given by

$$\theta = \frac{T - T_0}{T_{\text{scale}}}. \tag{6}$$

Using the derivative boundary condition at $x = 0$ in the direct problem statement of equation (1), and the fact that ξ, τ have been scaled $O(1)$ in equation (5), the characteristic temperature scale T_{scale} is seen to be

$$T_{\text{scale}} = q''_{\max}\frac{r}{k}. \tag{7}$$

Because of the nature of the model problem (heating, not cooling) and due to the form of equations (6) and (7), the quantity θ is bounded by $0 \le \theta < 1$.

2.3 The test data

The exact solution of equation (1) was used in this study to generate test data for the inverse heat-flux estimation procedure. By using the scaled variables defined in section 2.2, the direct problem statement of equation (1) in dimensionless form is:

$$\begin{aligned} \frac{\partial^2\theta}{\partial\xi^2} &= \frac{\partial\theta}{\partial\tau}, \qquad 0 \le \tau, \qquad 0 \le \xi \le \frac{L}{r}; \\ \theta|_{\tau=0} &= \theta|_{\xi=L/r} = 0; \\ f(\tau) &= \left(\frac{q''(\tau)}{q''_{\max}}\right) = -\left.\frac{\partial\theta}{\partial\xi}\right|_{\xi=0}. \end{aligned} \tag{8}$$

An implicit second-order-accurate finite difference scheme (the Crank–Nicholson method[4]) was used to integrate equation (8) forward in time, and thus the test data for the inverse procedure was generated as follows.

1. A dimensionless "heat flux" function $f_i = f(\tau_i)$, is posed where

   $$\tau_i = (i-1)\triangle\tau.$$

2. The dimensionless temperature field $\theta(\xi, \tau_i)$ for all values $0 \le \xi \le L/r$ was systematically integrated forward in time using equation (8), with f_i the derivative boundary condition at $\xi = 0$ in time step i.

[4]The Crank–Nicholson method is a standard approach for the numerical solution of one-dimensional diffusion-type PDEs. For representative algorithms, see Press *et al.* [3] who describe an implementation in the `c` programming language. However, note that the specific implementation differs here because `Mathematica` was used throughout in this study.

3. The "temperature-sensor data," devised as input for the cellular automaton inverse procedure described below, was generated from the computed $\theta|_{\xi=1}(\tau_i)$ values by:

$$T_i = (150^\circ)\,\theta|_{\xi=1}(\tau_i).$$

where 150° is an arbitrary temperature scale selected to illustrate the procedure.

Having described the procedure for generating test "thermocouple data" from the exact solution of equation (8), let us now consider an entirely different, cellular automaton, model for thermal-energy transport into the medium.

2.4 The cellular automaton energy transport model

A one-dimensional cellular automaton[5] evolution is the core modeling construct for this inverse heat-flux estimation scheme. In this model each row of a conventional cellular automaton array represents a successive time step, and each column represents a spatial position deeper into the medium, with the leftmost column (the first cell of each row) corresponding to the surface position at $\xi = 0$. In this cellular automaton model, the leftmost column of cells are prescribed to correspond to a boolean heat flux at the surface; see Figure 2.

Energy transport (rather than, say, temperature) is modeled as a cellular automaton evolution because energy has several unambiguous modeling criteria.

- We require that the model represent the timewise introduction of new energy at the surface.
- We require that the model conserves energy (i.e., the sum of black cells in the first column is equal to the sum of black cells in the last row).
- We require that the model emulate the heat-sink characteristic of the medium.

Rule 226 with a right-shift[6] was ultimately selected because it satisfies all of the above energy-modeling criteria. Furthermore, by using Rule 226 (with right-shift) temporal spans of intense surface flux appear to propagate aggressively into the medium (see Figure 2), a model-specific feature that is consistent with physical intuition.

[5] See Wolfram [4], the encyclopedic discourse on cellular automaton modeling and its wide-ranging implications for scientific investigation.

[6] The cellular automaton rule-numbering convention used here follows Wolfram [4]. Note that both Rules 98 and 226 (each with a right-shift and the first cell prescribed) satisfied the energy-transport modeling criterion cited above and that the ultimate selection of Rule 226, over Rule 98, was arbitrary.

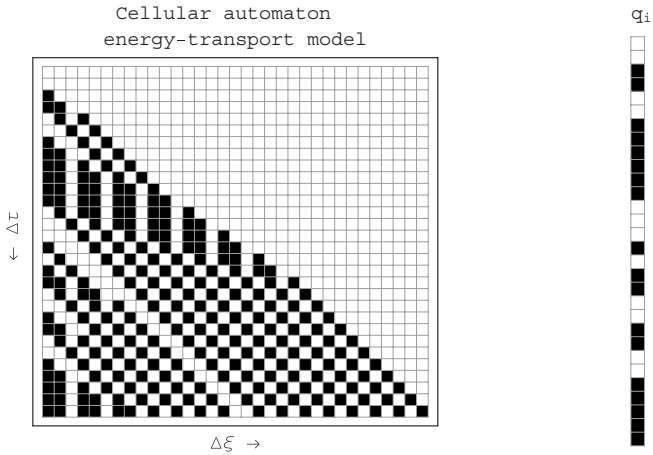

Figure 2. Schematic of the cellular automaton energy-transport idealization of the model problem depicted in Figure 1. The $\Delta\xi, \Delta\tau$ array of cells represent the transport of energy into the substrate medium of Figure 1, with time progressing down (top to bottom), and the physical penetration depth of thermal energy from left to right. The cells in the leftmost column of the array correspond to the heated surface of Figure 1, and are prescribed to be a boolean heat flux $\{q_i\}$. The right side of the schematic depicts this same boolean heat-flux vector q_i on an enlarged scale. In this schematic "1" cells are black, and "0" cells white. This cellular automaton model is Rule 226 with a right-shift at each successive time step and the cells of the first column prescribed. The right-shift emulates the heat-sink characteristics of the physical medium of the model problem depicted in Figure 1.

The estimation of temperature time-series data, using this cellular automaton energy-transport model, will now be considered.

2.5 Calculating dimensionless temperature θ from the cellular automaton array

Because the objective data are thermocouple-temperature measurements, use of this cellular automaton model in an inverse heat-transfer estimation procedure necessitates the capability to calculate the temperature θ at the thermocouple position $\xi = 1$. Recall that, in this study, the energy penetration into the medium is represented by a cellular automaton array (see Figure 2) in which time progresses down the rows (in $\Delta\tau$ steps), and the spatial depth corresponds to the number of columns from the left (in, say, steps of $\Delta\xi$). This section explains how to calculate $\theta|_{\xi=1}(\tau_i)$ from the cellular automaton array, for a particular experimental arrangement.

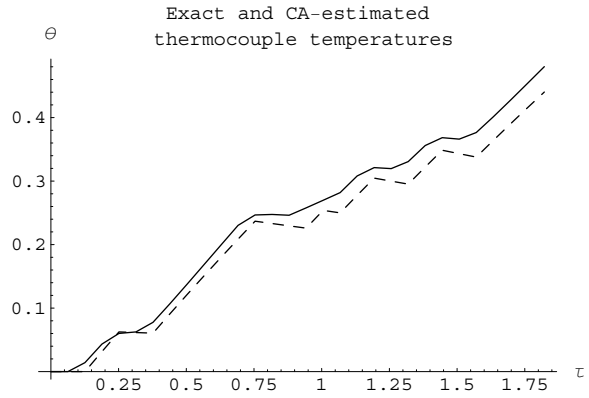

Figure 3. The exact (solid) and cellular-automaton-model estimated (dashed) calculations for $\theta|_{\xi=1}$ for the boolean flux $\{q_i\}$ depicted in Figure 2. In calculating both curves $\triangle\tau = \pi/50$; for the exact solution of equation (8) the medium thickness was taken to be $L/r = 10$.

Say, for example, that an experimental setup has a sampling rate $\triangle t = 1/50$[sec], a substrate thickness $L = 10$[cm], a substrate thermal diffusivity $\alpha = \pi(10^{-4})$[m^2/sec], and that the thermocouple is located a distance $r = 1$[cm] from the surface. Assume further that the dimensionless surface heating is the specific boolean $\{q_i\}$ sequence shown in Figure 2. Given this sequence as the surface heating transient, then exact solution (using equation (8)) for the dimensionless temperature at the thermocouple $\theta|_{\xi=1}(\tau_i)$ is shown as a solid line in Figure 3; it was computed using the finite-difference Crank–Nicholson method with dimensionless time step $\triangle\tau = \pi/50$, and $L/r = 10$. The dashed line on this same graph is the temperature curve as calculated from the cellular automaton array, computed (as described next) for these same parameters. The good agreement evident in this particular case turns out to be representative of the agreement obtained for a variety of input heat-flux distributions $\{q_i\}$.

The cellular-automaton-array temperature-computation algorithm employed in this study assumes that a specific array column—let us call it column "j"—corresponds to the thermocouple position $\xi = 1$, and "integrates" the temporal energy contributions $q_{i,j}$ down this column to generate a temperature time series $\theta_j(\tau_i)$. The algorithm[7] used to estimate this unsteady dimensionless temperature at cell position j is:

$$\theta_j(\tau_i) \simeq \theta_j(\tau_{i-1})e^{-\triangle\xi\triangle\tau} + \kappa q_{i,j}\triangle\xi\triangle\tau, \tag{9}$$

[7]The form of this expression draws upon "lumped capacitance" approximations utilized in conventional heat-transfer analysis [1].

with initial condition $\theta_j(0) = 0$. In equation (9) the numerical value for $\triangle\tau$ is fixed (from the experimental sampling rate), and the values $q_{i,j}$ are determined from the cellular automaton array evolution. However, use of equation (9) also requires quantitative estimates for $\triangle\xi$, j, and κ, which are now considered in turn.

An order-of-magnitude estimate can be made for $\triangle\xi$ using the following two observations.

- The cellular automaton energy-transport model is contrived so that energy propagates one cell ($\triangle\xi$) to the right each time step ($\triangle\tau$).
- For diffusion phenomena in a semi-infinite medium, the signal scales as $(x^2/\alpha t) \sim$ constant, and thus $(\triangle\xi^2/\triangle\tau) \sim O(1)$.

So if we assume that $(\triangle\xi^2/\triangle\tau) \sim 1$ this leads to a rough estimate for $\triangle\xi$:

$$\triangle\xi \sim (\triangle\tau)^{1/2}. \tag{10}$$

Furthermore, we need to determine which column j corresponds to the thermocouple depth in the medium $\xi = 1$. Since $\xi = j(\triangle\xi)$ we get:

$$j \sim \frac{1}{(\triangle\tau)^{1/2}}. \tag{11}$$

It is obvious that the result of equation (11), which computes the column number corresponding to the thermocouple depth, must be rounded to the nearest integer. Consequently, were this scheme to be used in conjunction with a particular experiment, the accuracy would be enhanced if the data-sampling rate $\triangle t$ was adjusted so that equation (11) evaluates to a nearly integer value. In this example, for $\triangle\tau = \pi/50$, we get $j \sim 4$.

Finally, the parameter κ accounts for the fact that the $O(1)$ propagation of the $(\triangle\xi^2/\triangle\tau)$ signal is not identically 1, but an $O(1)$ constant. This parameter κ was estimated by visually fitting the cellular-automaton array-derived temperature series equation (9) with the exact solution calculated from equation (8). For the case plotted in Figure 3 ($\triangle\tau = \pi/50$, $L/r = 10$) the value $\kappa \simeq 2$ evidently gave good agreement with the exact solution.

2.6 The search procedure for the unknown surface heat flux

The search algorithm for the unknown heat flux $q''(t_i)$, given the input temperature data T_i, is described in this section. Recall that this cellular automaton inverse method represents surface heat flux as $q''(t_i) \sim q''_{\max}\{q_1, q_2, \ldots, q_i, \ldots, q_N\}$, where $q''_{\max}$ is the dimensioned scaling quantity and the boolean $\{q_i\}$ values idealize the temporal behavior. Consequently, this inverse-search procedure needs to estimate both: (*i*) the value of $q''_{\max}$, and (*ii*) the boolean vector $\{q_i\}$. In our discussion the following notation is used.

- The measured thermocouple data is designated T_i.
- The notation $\theta_{\mathrm{CA}}(\tau_i)$ represents the dimensionless temperature $\theta(\tau_i)$, evaluated at the dimensionless thermocouple depth $\xi = 1$ as calculated from the cellular automaton (CA) energy-transport model.

The search procedure begins by constructing an initial guess for the $\{q_i\}$ vector using the input T_i data. The method employed for constructing the initial guess[8] consistently and by design understates the total number of ones in the solution vector $\{q_i\}$. The search procedure then iterates the following sequence of three steps.

1. The procedure systematically rearranges eight-cell subsequences of $\{q_i\}$, starting at large times in the vector ($i \to N$) and working its way backwards (to $i \to 1$) in steps of four cells, so that the rearrangements overlap. The objective here is to find the "optimal" overall arrangement of the (0,1) values in $\{q_i\}$, that is, the arrangement which yields a computed shape of $\theta_{\mathrm{CA}}(\tau_i)$ which most closely resembles T_i.

   In this study, the "most closely resembles" criterion means minimizing the `EstimatedVariance` parameter of the (`Mathematica 5.1`) `Regress` function for a linear least-squares fit of $\theta_{\mathrm{CA}}(\tau_i)$, T_i.

2. A least-squares fit of $\theta_{\mathrm{CA}}(\tau_i)$, with respect to the input data values T_i, produces a dimensioned multiplicative factor (say, T_{scale}), that is:

$$T_i \sim T_{\mathrm{scale}} \cdot \theta_{\mathrm{CA}}(\tau_i) + T_0.$$

3. The scheme notes at which temporal position τ_i the difference $(T_i - T_{\mathrm{scale}} \cdot \theta_{\mathrm{CA}}(\tau_i))$ is greatest, and replaces a "0" with a "1" at this (suitably time-lagged) position in $\{q_i\}$.

Steps 1 through 3 are repeated until $\{q_i\}$ does not change, or the `EstimatedVariance` parameter ceases to decrease. Thus, at the conclusion of this search the procedure has identified both the $\{q_i\}$ vector and the value T_{scale}, from which $q''_{\max}$ can be calculated using equation (7).

3. Sample computed results

Three sets of computed results are reported in this section. In all three sets, the "thermocouple time-series data" T_i was generated (as described

[8] In this study, the initial guess for $\{q_i\}$ was produced as follows.

(a) Take differences $T_{i+1} - T_i$.

(b) Convolve (a) with {1/4, 1/2, 1/4} to smooth it.

(c) Normalize (b) to [0, 1] values.

(d) Create a boolean pulsed signal from (c) which conserves area under the curve.

in section 2.3) by:

(*i*) prescribing a dimensionless heat-flux function $f(\tau_i)$;

(*ii*) using this f_i to calculate the solution $\theta|_{\xi=1}(\tau_i)$ from equation (8);

(*iii*) creating the thermocouple data by setting $T_i = 150 \cdot \theta|_{\xi=1}(\tau_i)$.

In each case considered the inverse solution procedure generates a $\{q_i\}$ vector to approximate f_i, and also reports a T_{scale} value for which, in this set of calculations, 150 would be the "exact" answer.

3.1 T_i due to a smooth surface heating function

Figures 4 and 5 show the computed results due to input temperature time-series data T_i that was generated using a smoothly varying input dimensionless heat-flux function f_i. Figure 4 shows the input T_i (as a solid line) and the inverse-solver's best-fit $T_{\text{scale}} \cdot \theta_{\text{CA}}$ (the dashed line); the temperature scaling T_{scale} was estimated to be 141.4 (150 would have been exact). Figure 5 shows the originating f_i function (smooth) and the inferred boolean approximation $\{q_i\}$ on the same graph.

3.2 T_i due to a sporadic (random) surface heating function

Figures 6 and 7 show the computed results due to input temperature time-series data T_i that was generated using the q_i sequence of Figure 2, in effect a random, sporadic heat-flux function f_i. Figure 6 shows this input T_i (as a solid line) and the inverse-solver's best-fit $T_{\text{scale}} \cdot \theta_{\text{CA}}$ (the

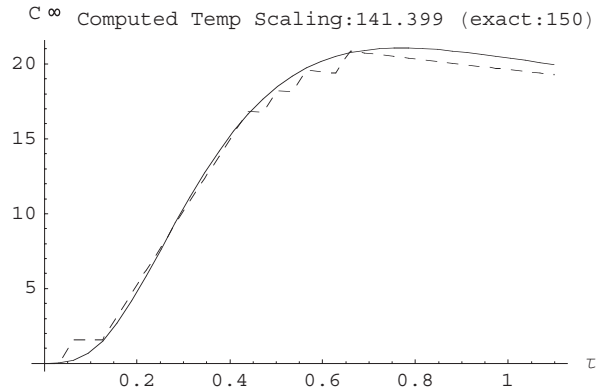

Figure 4. The input T_i (solid line) and the inverse-solver corresponding best-fit (dashed line) temperature profiles for the case where the input T_i data was generated from a smooth dimensionless heat-flux function f_i (Figure 5).
The input T_i was created with a temperature-scaling factor $T_{\text{scale}} = 150$; the inverse solver's estimate of this value was 141.4.

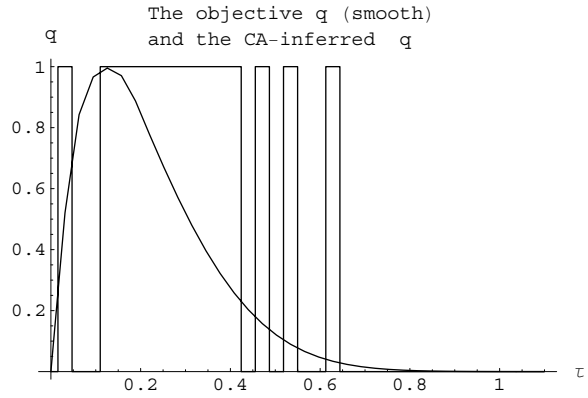

Figure 5. The dimensionless heat-flux function f_i used to generate the T_i values in Figure 4, and the inferred boolean approximation $\{q_i\}$ that the inverse-solver developed from the input T_i time series.

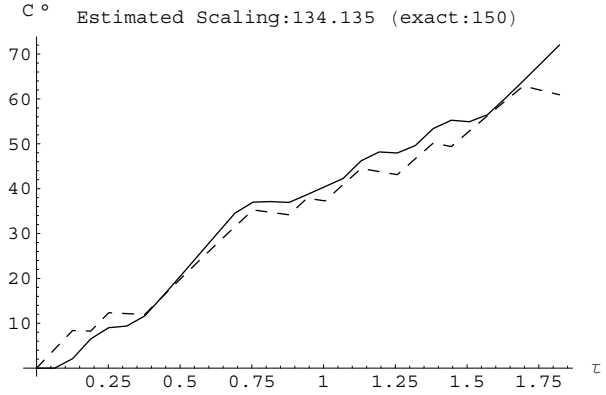

Figure 6. Input T_i (solid line) and inverse-solver best-fit (dashed line) temperature profiles for the case where the input T_i data was generated by the random boolean data of Figure 2 (also the f_i shown in the top graph of Figure 7).

dashed line), and with the temperature scaling T_{scale} estimated to be 134.1 (150 would be exact).

Figure 7 shows the originating f_i heat flux on the top graph, and the inferred boolean approximation $\{q_i\}$ aligned on the lower graph. Figure 7 demonstrates how the inverse solver failed to "see" the surface heat flux for times $\tau > 1.7$. This is because the surface thermal energy at these latter times had not yet propagated to the thermocouple depth in the medium.

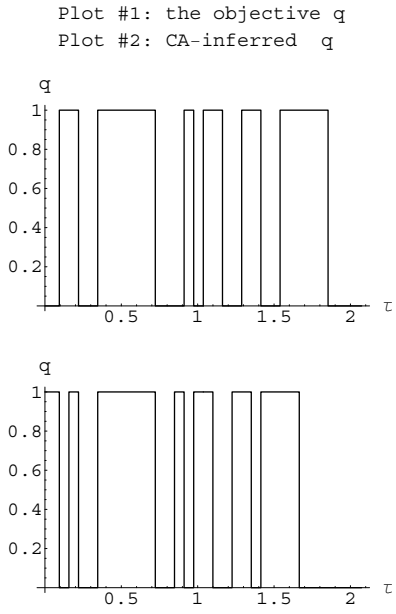

Figure 7. The top graph is the dimensionless heat-flux function f_i used to generate the T_i values in Figure 6, and the lower graph is the inferred boolean approximation $\{q_i\}$ that the inverse-solver developed from the input T_i time series.

It is worthwhile to note that sporadic heat-flux phenomena are difficult to estimate using conventional inverse heat-transfer procedures (see Beck *et al.* [2]).

3.3 Where the T_i data is noise-ridden

Figures 8 and 9 show the computed results due to an input temperature time-series data T_i (Figure 8) that, again, was generated using a smooth input dimensionless heat-flux function f_i (Figure 9). However, this time the input temperature signal T_i was "randomized" at random locations by as much as ±10%. Figure 8 shows the input T_i (as a solid line) and the inverse-solver's best-fit $T_{\text{scale}} \cdot \theta_{\text{CA}}$ (the dashed line), and with the temperature scaling estimated to be 140.5 (150 would be exact). Figure 9 shows the originating f_i function (smooth) and the inferred boolean approximation $\{q_i\}$ on the same graph.

Noise-ridden signals like the T_i of Figure 8 are not uncommon with thermocouples, and present difficulties for the several conventional inverse heat-flux estimation strategies which utilize temporal derivatives of the time-series data (see Beck *et al.* [2]).

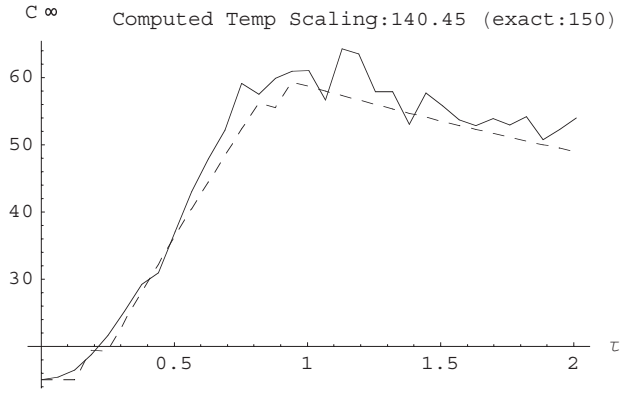

Figure 8. Input T_i (solid line) and inverse-solver best-fit (dashed line) temperature profiles for the case where the input T_i data is noisy and generated from the smooth dimensionless heat-flux function f_i shown in Figure 9.

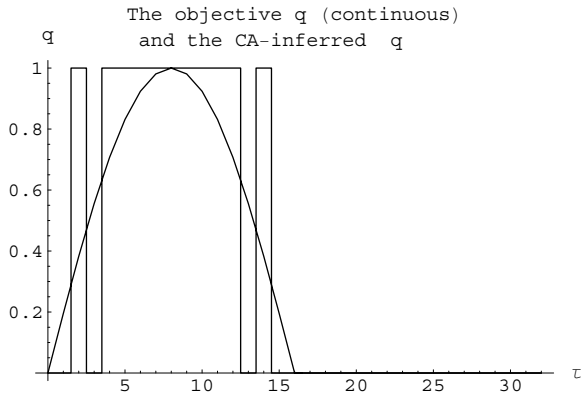

Figure 9. The dimensionless heat-flux function f_i used to generate the T_i values in Figure 8, and the inferred boolean approximation $\{q_i\}$ that the inverse-solver developed from the (noise-ridden) input T_i data.

4. Conclusions

This paper describes a new procedure for inverse transient heat-flux estimation which, remarkably, does not use the basic equations of heat transfer as in equation (1). Instead, this method utilizes a basic cellular automaton energy-transport model that embodies core phenomena attributes such as conservation of energy and heat-sink behavior. This cellular automaton energy-transport model, together with a suitable dimensionless-scales analysis, are the foundation of this inverse estima-

tion procedure which appears to have quantitative accuracy comparable to conventional inverse estimation procedures.

The approach described in this paper could be customized to a particular experimental investigation in several ways. First, the search procedure might be optimized to look for specific heat-flux distributions anticipated in the individual application under study. Second, the data-sampling rate should be fixed so that the cellular automaton column corresponding to the thermocouple position equation (11) evaluates to an integer value. Finally, while a number of procedural programming languages could in principle be used to implement the method, it emerged that `Mathematica` proved a particularly natural language for both the search algorithm and for the cellular automaton computation and thus it is recommended. All the numerical work in this study was carried out in `Mathematica 5.1`.

Acknowledgments

The author carried out this study at the NKS Summer School 2005 sponsored by Wolfram Research, Incorporated. Todd Rowland, Oyvind Tafjord, and Stephen Wolfram are all gratefully acknowledged for their insights and helpful suggestions.

References

[1] F. P. Incropera and D. P. DeWitt, *Fundamentals of Heat and Mass Transfer*, Second Edition (John Wiley and Sons, New York, 1981). [Chapter 5 considers unsteady heat conduction.]

[2] J. V. Beck, B. Blackwell, and C. R. St. Clair, Jr., *Inverse Heat Conduction: Ill-posed Problems* (Wiley Interscience, New York, 1985).

[3] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Programming* (Cambridge University Press, Cambridge, 1988). [See sections 2.6 "tridiagonal systems," and 17.2 "diffusive initial-value problems." Please note that for this study all computational algorithms were implemented in `Mathematica` rather than `C`.]

[4] Stephen Wolfram, *A New Kind of Science* (Wolfram Media, Inc., Champaign, IL, 2002).