0.1 Analytical Solution

Bryan Kaiser

Given the advection-diffusion-heat flux problem for a bathtub:

$$c\Big(\frac{\partial T}{\partial t}+U\frac{\partial T}{\partial x}\Big)=\kappa\frac{\partial^2 T}{\partial x^2}-\frac{h}{d}(T-T_C),$$

where the left boundary condition at $x=0$ is the hot water faucet

$$UcT-\kappa\frac{\partial T}{\partial x}=UcT_H,$$

and the right boundary condition at $x=L$ is a drain

$$\kappa\frac{\partial T}{\partial x}=0,$$

we are asked to find the solution. Here T is the temperature, c is the heat capacity, κ is the thermal diffusivity, T_C is the (cold) air temperature, T_H is the (hot) water flowing into the tub from the faucet, h is a sensible heat flux bulk coefficient, and d is the depth of the tub. At steady state, the governing equation can be written as

$$\frac{\partial^2 T}{\partial x^2}-\frac{cU}{\kappa}\frac{\partial T}{\partial x}-\frac{h}{\kappa d}T=-\frac{h}{\kappa d}T_C$$

$$a=1,\quad b=-\frac{cU}{\kappa}\quad c=-\frac{h}{\kappa d}\quad T_p(x)=-\frac{h}{\kappa d}T_C,$$

and analytically solved by method of undetermined coefficients. The auxiliary equation for the complementary (homogeneous, which in this case is equivalent to $T_C=0$) solution has two real roots ($b^2-4ac>0$)

$$r^2-\frac{cU}{\kappa}r-\frac{h}{\kappa d}=0,\qquad r=\frac{-b\pm\sqrt{b^2-4ac}}{2a},\qquad r=\frac{cU}{2\kappa}\pm\frac{1}{2}\sqrt{\Big(\frac{cU}{2\kappa}\Big)^2+\frac{4h}{\kappa d}}=\lambda_1,\lambda_2,$$

$$\lambda_1=\frac{cU}{2\kappa}+\frac{1}{2}\sqrt{\Big(\frac{cU}{2\kappa}\Big)^2+\frac{4h}{\kappa d}},\qquad \lambda_2=\frac{cU}{2\kappa}-\frac{1}{2}\sqrt{\Big(\frac{cU}{2\kappa}\Big)^2+\frac{4h}{\kappa d}},$$

where

$$T_c(x)=\alpha_1 e^{\lambda_1 x}+\alpha_2 e^{\lambda_2 x},$$

and α_1 and α_2 are constant unknown coefficients that are obtained by using the boundary conditions at $x=0,L$. The particular solution is just $T_p(x)=T_C$. The sum of the complementary and particular solution is the complete solution for the inhomogenous partial differential equation:

$$T(x)=T_c(x)+T_p(x)\quad\rightarrow\quad T(x)=\alpha_1 e^{\lambda_1 x}+\alpha_2 e^{\lambda_2 x}+T_C.$$

Substitution into the boundary condition at $x=0$ yields

$$UcT-\kappa\frac{\partial T}{\partial x}=UcT_H,\tag{0.1}$$

$$Uc(\alpha_1+\alpha_2+T_C)-\kappa(\lambda_1\alpha_1+\lambda_2\alpha_2)=UcT_H\quad\rightarrow\quad \alpha_1=\frac{Uc(T_H-T_C)}{Uc-\kappa\lambda_1}-\alpha_2\frac{Uc-\kappa\lambda_2}{Uc-\kappa\lambda_1}$$

and $x=L$:

$$\kappa\frac{\partial T}{\partial x}=0\tag{0.2}$$

Therefore plug in

$$\kappa(\lambda_1\alpha_1 e^{\lambda_1 L}+\lambda_2\alpha_2 e^{\lambda_2 L})=0\quad\rightarrow\quad \alpha_2=-\frac{\lambda_1}{\lambda_2}e^{(\lambda_1-\lambda_2)L}\alpha_1$$

therefore plug in and the coefficients can be found

$$\alpha_1=\Big(1-\frac{\lambda_1}{\lambda_2}e^{(\lambda_1-\lambda_2)L}\frac{Uc-\kappa\lambda_2}{Uc-\kappa\lambda_1}\Big)^{-1}\frac{Uc(T_H-T_C)}{Uc-\kappa\lambda_1},$$

$$\alpha_2=-\Big(1-\frac{\lambda_1}{\lambda_2}e^{(\lambda_1-\lambda_2)L}\frac{Uc-\kappa\lambda_2}{Uc-\kappa\lambda_1}\Big)^{-1}\frac{Uc(T_H-T_C)}{Uc-\kappa\lambda_1}\frac{\lambda_1}{\lambda_2}e^{(\lambda_1-\lambda_2)L},$$

completing the analytical solution for $T(x)$.

0.2 Numerical Solution

Rearranging the governing equation and leaving only the time derivative on the left hand side of the equation yields an expression that can be integrated forward in time:

$$\frac{\partial T}{\partial t} = -U\frac{\partial T}{\partial x} + \mathcal{K}\frac{\partial^2 T}{\partial x^2} - \mathcal{H}T + \mathcal{H}T_C,$$

were $\mathcal{K} = \kappa/c$ and $\mathcal{H} = h/(cd)$. The left boundary condition is rearranged to form a Neumann boundary condition at $x = 0$:

$$\frac{\partial T}{\partial x} = \frac{U}{\mathcal{K}}(T - T_H),$$

and the right boundary condition for $\kappa > 0$ reduces to

$$\frac{\partial T}{\partial x} = 0.$$

Padé finite differences stencils for spatial derivatives

To integrate the governing equation forward in time, the spatial derivatives much be computed at each time step. To compute the first-order derivative, a fourth-order accurate Padé scheme (i.e. the truncation error is $\mathcal{O}(\Delta \mathrm{x}^4)$, where $\mathcal{O}()$ operator denotes “on the order of”). Padé finite difference stencils are just another combination of the same Taylor expansions of $T(x)$ at grid points used to derive central, forward, and backward finite difference stencils (cite Ferziger). In general, they take the form:

$$\alpha\frac{\partial T}{\partial x}\Big|_{j+1} + \frac{\partial T}{\partial x}\Big|_{j} + \alpha\frac{\partial T}{\partial x}\Big|_{j-1} = \beta\frac{T_{j+1} - T_{j-1}}{2\Delta \mathrm{x}} + \gamma\frac{T_{j+2} - T_{j-2}}{4\Delta \mathrm{x}}.$$

For a fourth-order stencil, $\alpha = 1/4$,$\beta = 3/2$, and $\gamma = 0$. The first derivative of the temperature profile is then obtained by using

$$\frac{\Delta x}{3}\Big(\frac{\partial T}{\partial x}\Big|_{j+1} + 4\frac{\partial T}{\partial x}\Big|_{j} + \frac{\partial \psi}{\partial x}\Big|_{j-1}\Big) = T_{j+1} - T_{j-1},$$

at the grid point indices $j = [2, 3, ..., N-1]$. At the locations $x = 0$ and $x = L$ (corresponding to cell edge grid point indices $j = 1$ and $j = N$) the given Neumann boundary conditions specify the first derivative of the temperature profile. Therefore the equations can be written as a set of matrix operations:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ \Delta \mathrm{x}/3 & 4\Delta \mathrm{x}/3 & \Delta \mathrm{x}/3 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & \Delta \mathrm{x}/3 & 4\Delta \mathrm{x}/3 & \Delta \mathrm{x}/3 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \Delta \mathrm{x}/3 & 4\Delta \mathrm{x}/3 & \Delta \mathrm{x}/3 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & \Delta \mathrm{x}/3 & 4\Delta \mathrm{x}/3 & \Delta \mathrm{x}/3 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \partial_x T_1 \\ \partial_x T_2 \\ \partial_x T_3 \\ \vdots \\ \partial_x T_{N-2} \\ \partial_x T_{N-1} \\ \partial_x T_N \end{bmatrix} = \begin{bmatrix} U(T_1 - T_H)/\mathcal{K} \\ T_3 - T_1 \\ T_4 - T_2 \\ \vdots \\ T_{N-1} - T_{N-3} \\ T_N - T_{N-2} \\ 0 \end{bmatrix}.$$

Let the square matrix (and invertible) on the left hand side be denoted as $\mathbf{A}$, the vector of unknown temperature derivatives on the left and side be $\mathbf{x}$ and the vector of boundary conditions and temperature differences on the right hand side be $\mathbf{b}$. The vector of temperature derivatives is then easily obtained simultaneously by solving the linear algebraic operation $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ for $\mathbf{x}$.

The second spatial derivative of the temperature profile is also computed by a fourth-order Padé finite difference stencil. The boundary conditions for the second derivative are obtained by using fourth-order accurate first derivative finite difference stencils on the (already computed) first temperature derivative vector. At $x = 0$:

$$\frac{\partial^2 T_1}{\partial x^2} = \frac{-25/12\partial_x T_1 + 4\partial_x T_2 - 3\partial_x T_3 + 4/3\partial_x T_4 - 1/4\partial_x T_5}{\Delta x} + \mathcal{O}(\Delta x^4),$$

where $\partial_x T_j$ denotes the discrete first x derivative of T at the j grid index. At $x = L$ the backward finite difference for the second derivative is

$$\frac{\partial^2 T_N}{\partial x^2} = \frac{25/12\partial_x T_N - 4\partial_x T_{N-1} + 3\partial_x T_{N-2} - 4/3\partial_x T_{N-3} + 1/4\partial_x T_{N-4}}{\Delta x} + \mathcal{O}(\Delta x^4),$$

therefore the Padé matrix equation for the second finite difference solution is

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ \Delta\mathrm{x}/3 & 4\Delta\mathrm{x}/3 & \Delta\mathrm{x}/3 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & \Delta\mathrm{x}/3 & 4\Delta\mathrm{x}/3 & \Delta\mathrm{x}/3 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \Delta\mathrm{x}/3 & 4\Delta\mathrm{x}/3 & \Delta\mathrm{x}/3 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & \Delta\mathrm{x}/3 & 4\Delta\mathrm{x}/3 & \Delta\mathrm{x}/3 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \partial_{xx}T_1 \\ \partial_{xx}T_2 \\ \partial_{xx}T_3 \\ \vdots \\ \partial_{xx}T_{N-2} \\ \partial_{xx}T_{N-1} \\ \partial_{xx}T_N \end{bmatrix} \dots$$

$$= \begin{bmatrix} (-25/12\partial_x T_1 + 4\partial_x T_2 - 3\partial_x T_3 + 4/3\partial_x T_4 - 1/4\partial_x T_5)/\Delta x \\ \partial_x T_3 - \partial_x T_1 \\ \partial_x T_4 - \partial_x T_2 \\ \vdots \\ \partial_{xx} T_{N-1} - \partial_x T_{N-3} \\ \partial_x T_N - \partial_x T_{N-2} \\ (25/12\partial_x T_N - 4\partial_x T_{N-1} + 3\partial_x T_{N-2} - 4/3\partial_x T_{N-3} + 1/4\partial_x T_{N-4})/\Delta x \end{bmatrix},$$

and once again the vector of the second derivative of temperature is then easily obtained solving by the linear algebraic operation $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ for $\mathbf{x}$.

A fourth-order Runge-Kutta scheme for time advancement

Once the first and second x derivatives of T are computed for all grid points, the forcing terms (advection, diffusion, and sensible heat flux) on the right hand side of equation BLANK can be computed. The forcing terms are summed four times per time step to solve for each fourth-order Runge Kutta coefficent. To obtain the Runge Kutta coefficients, the time derivative in equation BLANK is discretized

$$\frac{T^{n+1} - T^n}{\Delta t} = \frac{\partial T^n}{\partial t} = f(T^n, t^n),$$

$$f(T^n, t^n) = -U(t^n)\frac{\partial T^n}{\partial x} + \mathcal{K}\frac{\partial^2 T^n}{\partial x^2} - \mathcal{H}T^n + \mathcal{H}T_C,$$

into a forward difference scheme. The Runge Kutta coefficients are then computed as

$$k_1 = f(T^n, t^n) = -U(t^n)\frac{\partial T^n}{\partial x} + \mathcal{K}\frac{\partial^2 T^n}{\partial x^2} - \mathcal{H}T^n + \mathcal{H}T_C,$$

$$k_2 = f(T^n + \frac{\Delta t}{2}k_1, t^n + \frac{\Delta t}{2}),$$

$$k_3 = f(T^n + \frac{\Delta t}{2}k_2, t^n + \frac{\Delta t}{2}),$$

$$k_4 = f(T^n + \Delta t k_3, t^n + \Delta t).$$

Finally, to advance to the next step to coefficients are summed

$$T^{n+1} = T^n + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

Numerical solutions for U=0.001 m/s

Two odd behaviors were observed for the numerical solutions for the given velocities. First, Figure 0.1 shows that the error actually *increases* as the grid resolution increases (note how on the error plot the red line, at the highest resolution of 201 grid points, is above the blue lines everywhere). The simulations where integrated forward in time to the same time (they use the same amount of water) so it is possible that computational round-off error out-weighs the truncation error of the fourth-order accurate stencils. Round-off errors increase with every time step, and higher grid resolutions require smaller time steps in order to satisfy the Courant-Friedrichs-Lewy (CFL) condition that the Courant number (Co) be less than one. For an advection-diffusion problem, a Courant number less that one also constrains the dimensionless diffusion number (Di):

$$\mathrm{Co} = U\frac{\Delta t}{\Delta x} < 1 \qquad \mathrm{Di} = \mathcal{K}\frac{\Delta t}{\Delta x^2} < \frac{1}{2}.$$

The second odd behavior may be due to an incorrect analytical solution, and is discussed in the next section. The Courant and diffusion numbers for the $U = 0.001$ m/s simulation are:

N	Co	Di	$Ud^2 \cdot 1000$ (liters)
11	0.03	0.187	752 liters
21	0.05	0.625	752 liters
51	0.025	0.781	750 liters
101	0.0025	0.156	750 liters
201	0.005	0.625	750 liters

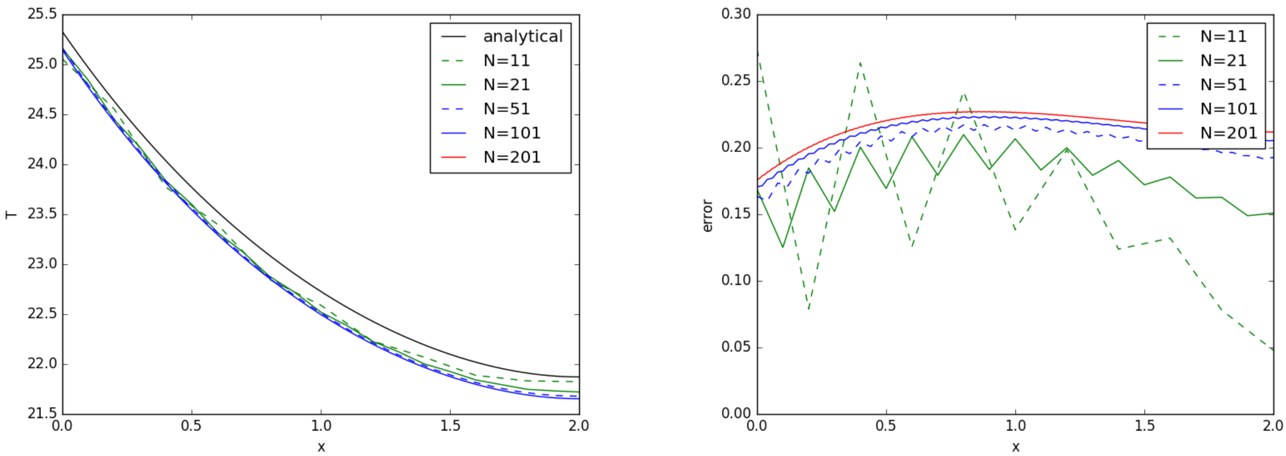

Figure 0.1: The solution (in units °C) and the solution error ($|T - T_{\text{analytical}}|$) for $U = 0.001$ m/s. Note that the solution converges with increasing grid points (increasing N) to a slightly larger error than for lower resolutions. This could be either due to larger round-off error than truncation error or a incorrect analytical solution.

Numerical solution for U=0.01 m/s

For $U = 0.01$ the analytical solution jumped to values far above T_H. Since that situation seems unphysical (there is no heat source within the tub, therefore the solution cannot be hotter than T_H) and the numerical solution converges to a profile that seems intuitively correct: the water is warmer near the hot faucet at $x = 0$ and slopes down to a flat profile just before the drain at $x = L$, consistent with the boundary conditions. As the velocity increases, the residence time of the water in the tub decreases and so the sensible heat flux term has less time to affect an individual parcel of water as it is advected through the tub. Therefore, the slope of the profile is flatter than for the slower velocity case.

N	Co	Di	$Ud^2 \cdot 1000$ (liters)
11	0.3	0.187	750 liters
21	0.09	0.125	750 liters
51	0.125	0.391	750 liters
101	0.05	0.313	750 liters
201	0.02	0.25	750 liters

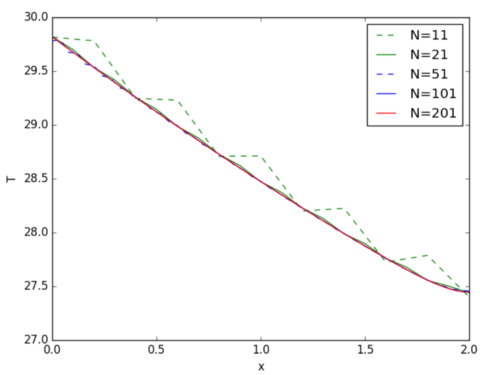

Figure 0.2: The gently sloping solution for $U = 0.01$ m/s (in units °C).

0.3 Time dependent velocity for an optimal bath

To add a time dependent solution that increases the smoothness of the profile for the least amount of water, it is helpful to think recast the thermal properties of the fluid at hand into characteristic time scales. For example, the amount of time it would take for an entire profile at $T(x) = T_H$ to cool by sensible heat flux to T_C is the sensible heat flux time scale:

$$\text{sensible heat time scale:} \qquad \tau_s = \frac{cd}{h} = 667\text{s}.$$

Similarily, the amount of time it would take for a warm anomaly to diffuse laterally across the bathtub to a flat profile in x is the diffusion time scale:

$$\text{diffusion time scale:} \qquad \tau_d = \frac{cdL}{\kappa} = 800\text{s}.$$

The residence time (which we can specify!) of the bathwater in the tub is

$$\text{residence time:} \qquad \tau_u = \frac{L}{U}.$$

Therefore in the $U = 0.01$ m/s simulation the residence time was $\tau_u = 200$ s and so the advection and sensible heat flux terms dominated the steady state result: a linearly sloping profile of -2°C/m. For the $U = 0.01$ m/s simulation the residence time was $\tau_u = 2000$ s, therefore the resulting curvy steady state profile was dominated by an advection-diffusion balance. Since we are interested in a flat profile that uses minimal water, we need an advection time scale that is just slightly smaller than the sensible heat flux time scale (i.e. hot water is pumped in faster than it cools).

To minimize the water use, one could use an inlet velocity corresponding to a residence time that is just a little less than the sensible heat flux time scale:

$$\tau_u \sim \tau_s = \frac{L}{U_0} \quad \rightarrow \quad U_0 \sim \frac{L}{\tau_s} \sim 0.004 \ \text{m/s}.$$

A inflow with this velocity could be turned off and on at a frequency corresponding to a period less than the diffusion time scale so that the thermal diffusion doesn't increase the spatial variance of the $T(x)$ profile like in Figure 0.1. Water can be saved by turning the faucet off and on with a period equivalent to roughly a quarter of the diffusion time scale:

$$U(t) = \frac{U_0}{2}(\sin \omega t + 1) \qquad \omega = \frac{4\pi}{\tau_d} \sim 0.016 \ \text{rad/s}.$$

This frequency corresponds to a period of about 7 minutes. The amount of water used per period of slowly turning the faucet on and off is

$$dL \cdot \int_0^{2\pi} U(t') \, dt' = \frac{U_0 dL}{2}\Big(1 - \frac{\cos \omega t}{\omega}\Big)_0^{2\pi} = \frac{U_0 dL}{\omega}(\omega\pi + \sin^2(\omega\pi)),$$

which is in this example:

$$\lim_{\omega\to\infty} dL \cdot \int_0^{2\pi} U(t') \, dt' = U_0 dL\pi = 0.004\text{m/s} \cdot 0.5\text{m} \cdot 2\text{m} \cdot \pi \cdot 1000 \text{ liters/m}^3 \approx 12.6 \text{ liters per oscillation}.$$

Note that the oscillations don't require more water than constantly running the mean flow $U_0/2$, and that the oscillations can be infinitely fast

$$\lim_{\omega\to\infty} dL \cdot \int_0^{2\pi} U(t') \, dt' = dL \cdot \int_0^{2\pi} \frac{U_0}{2} \, dt',$$

but for thermal perturbations that act on time scales faster than the diffusion time scale the "memory" of a thermal anomaly can be preserved. To show this, two simulations with the same mean $U_0/2 = 0.002$ m/s were integrated over the same amount of time, and one had the oscillating U described above. Figure 0.3 shows that the oscillating hot inflow simulation has a slightly higher temperature, and faster oscillations may prove better still.

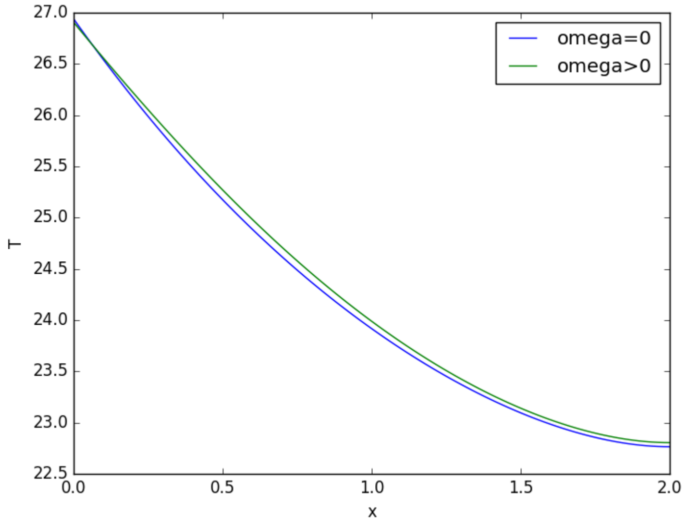

Figure 0.3: Comparison of mean temperature profiles $T(x)$ for oscillating $U(t)$ (green) and non-oscillating (blue) $U(t) = U_0/2$ for the same amount of water. The hot temperature is $T_H = 30°\mathrm{C}$ and the (cold) air temperature is $T_C = 20°\mathrm{C}$. Averaging was performed after initial transients decayed.