Homework #2

Caroline Hughes

February 22, 2018

Table of Contents

1 Crank-Nicolson

In general, a hybrid implicit-explicit (IMEX) method using central differences can be formulated as shown in Equation 1, where implicit method has weight α.

$$u_j^{n+1} - u_j^n = \lambda\alpha\left(u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}\right) + \lambda(1-\alpha)\left(u_{j-1}^n - 2u_j^n + u_{j+1}^n\right) \tag{1}$$

When $\alpha = 1/2$, this is called a Crank-Nicolson scheme and is second-order in both time and space. The truncation error τ_n is given by Equation 2.

$$\tau_n = u_j^{n+1} - u_j^n - \lambda\alpha\left(u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}\right) - \lambda(1-\alpha)\left(u_{j-1}^n - 2u_j^n + u_{j+1}^n\right) \tag{2}$$

To prove that this method is $\mathcal{O}\left(k^2 + h^2\right)$ for $\alpha = 1/2$, I will consider the point centered at $u_j^{n+1/2}$. The Taylor series approximation of the function at $u_j^{n+1/2\pm1/2}$ is given in Equation 3a. Equations 3b and 3c show the two-dimensional Taylor series expansions around $u_{j\pm1}^{n+1/2\pm1/2}$. I chose here to express forward and backwards spatial approximations separately to avoid confusion with signs.

$$u_j^{n+1/2\pm1/2} = u \pm (k/2)u_t + \frac{(k/2)^2}{2!}u_{tt} \pm \frac{(k/2)^3}{3!}u_{ttt} + \mathcal{O}\left(k^4\right) \tag{3a}$$

$$\begin{aligned} u_{j+1}^{n+1/2\pm1/2} &= u \pm (k/2)u_t + hu_x + \frac{(k/2)^2}{2!}u_{tt} \pm (k/2)hu_{tx} + \frac{h^2}{2!}u_{xx} \\ &\pm \frac{(k/2)^3}{3!}u_{ttt} + \frac{(k/2)^2h}{2!}u_{ttx} \pm \frac{(k/2)h^2}{2!}u_{txx} + \frac{h^3}{3!}u_{xxx} + \mathcal{O}\left(k^4 + h^4\right) \end{aligned} \tag{3b}$$

$$\begin{aligned} u_{j-1}^{n+1/2\pm1/2} &= u \pm (k/2)u_t - hu_x + \frac{(k/2)^2}{2!}u_{tt} \mp (k/2)hu_{tx} + \frac{h^2}{2!}u_{xx} \\ &\pm \frac{(k/2)^3}{3!}u_{ttt} - \frac{(k/2)^2h}{2!}u_{ttx} \pm \frac{(k/2)h^2}{2!}u_{txx} - \frac{h^3}{3!}u_{xxx} + \mathcal{O}\left(k^4 + h^4\right) \end{aligned} \tag{3c}$$

Substituting Equations 3a to 3c into Equation 2,

$$\tau_n = \frac{1}{k}\left\{ \left(u + (k/2)u_t + \frac{(k/2)^2}{2!}u_{tt} + \frac{(k/2)^3}{3!}u_{ttt} + \mathcal{O}(k^4) \right) - \left(u - (k/2)u_t + \frac{(k/2)^2}{2!}u_{tt} - \frac{(k/2)^3}{3!}u_{ttt} + \mathcal{O}(k^4) \right) \right\}$$

$$- \frac{\alpha}{h^2}\left\{ \left(u + (k/2)u_t - hu_x + \frac{(k/2)^2}{2!}u_{tt} - (k/2)hu_{tx} + \frac{h^2}{2!}u_{xx} + \frac{(k/2)^3}{3!}u_{ttt} - \frac{(k/2)^2 h}{2!}u_{ttx} + \frac{(k/2)h^2}{2!}u_{txx} - \frac{h^3}{3!}u_{xxx} + \mathcal{O}\left(k^4 + h^4\right) \right) \right.$$
$$-2\left(u + (k/2)u_t + \frac{(k/2)^2}{2!}u_{tt} + \frac{(k/2)^3}{3!}u_{ttt} + \mathcal{O}\left(k^4\right) \right)$$
$$\left. + \left(u + (k/2)u_t + hu_x + \frac{(k/2)^2}{2!}u_{tt} + (k/2)hu_{tx} + \frac{h^2}{2!}u_{xx} + \frac{(k/2)^3}{3!}u_{ttt} + \frac{(k/2)^2 h}{2!}u_{ttx} + \frac{(k/2)h^2}{2!}u_{txx} + \frac{h^3}{3!}u_{xxx} + \mathcal{O}\left(k^4 + h^4\right) \right) \right\}$$

$$- \frac{(1-\alpha)}{h^2}\left\{ \left(u - (k/2)u_t - hu_x + \frac{(k/2)^2}{2!}u_{tt} + (k/2)hu_{tx} + \frac{h^2}{2!}u_{xx} - \frac{(k/2)^3}{3!}u_{ttt} - \frac{(k/2)^2 h}{2!}u_{ttx} - \frac{(k/2)h^2}{2!}u_{txx} - \frac{h^3}{3!}u_{xxx} + \mathcal{O}\left(k^4 + h^4\right) \right) \right.$$
$$-2\left(u - (k/2)u_t + \frac{(k/2)^2}{2!}u_{tt} - \frac{(k/2)^3}{3!}u_{ttt} + \mathcal{O}\left(k^4\right) \right)$$
$$\left. + \left(u - (k/2)u_t + hu_x + \frac{(k/2)^2}{2!}u_{tt} - (k/2)hu_{tx} + \frac{h^2}{2!}u_{xx} - \frac{(k/2)^3}{3!}u_{ttt} + \frac{(k/2)^2 h}{2!}u_{ttx} - \frac{(k/2)h^2}{2!}u_{txx} + \frac{h^3}{3!}u_{xxx} + \mathcal{O}\left(k^4 + h^4\right) \right) \right\}$$

Simplified, this becomes

$$\tau_n = \left(u_t + \frac{(k/2)^2}{3}u_{ttt} + \mathcal{O}\left(k^3\right) \right) - \alpha\left(u_{xx} + (k/2)u_{txx} + \mathcal{O}\left(k^4 + h^2\right) \right)$$
$$- (1-\alpha)\left(u_{xx} - (k/2)u_{txx} + \mathcal{O}\left(k^4 + h^2\right) \right)$$

$$= \underbrace{u_t - u_{xx}}_{0} - (1-2\alpha)(k/2)u_{txx} + \mathcal{O}\left(k^2 + h^2\right)$$

$$= \begin{cases} \mathcal{O}(k^2 + h^2), & \alpha = 1/2 \\ \mathcal{O}(k + h^2), & \text{otherwise} \end{cases}$$

Setting $\alpha = 1/2$ eliminates the $\mathcal{O}\left(k\right)$ term and elevates the method to $\mathcal{O}\left(k^2 + h^2\right)$. If $\alpha \neq 1/2$, this Ok term remains, leaving the method $\mathcal{O}\left(k + h^2\right)$.

2 Spherical Coordinates

2.1 Spherical Laplacian Derivation

2.1.1 Derivation of the Laplacian Operator in Spherical Coordinates

$$\nabla^2 u(x,y,z) = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) \tag{4}$$

$$\boxed{\frac{\partial u}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial u}{\partial r} \right) + \frac{1}{r^2 \sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta \frac{\partial u}{\partial \theta} \right) + \frac{1}{r^2 \sin^2\theta}\frac{\partial^2 u}{\partial \phi^2}} \tag{5}$$

Assuming angular symmetry, $u(r,\theta,\phi,t) \to u(r,t)$ and Equation 5 can be rewritten as

$$\frac{\partial u}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial u}{\partial r}\right) = \frac{\partial^2 u}{\partial r^2} + \frac{2}{r}\frac{\partial u}{\partial r} \tag{6}$$

To discretize the Laplacian, I will select central differences $u_{rr} = D_+D_-$ and $u_r = D_0$ and substitute $r = jh$.

$$\begin{aligned} \frac{u_j^{n+1} - u_j^n}{k} &= \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} + \frac{\cancel{2}}{jh}\frac{u_{j+1}^n - u_{j-1}^n}{\cancel{2}h} \\ u_j^{n+1} &= u_j^n + \lambda\left[\left(\frac{j-1}{j}\right)u_{j-1}^n - 2u_j^n + \left(\frac{j+1}{j}\right)u_{j+1}^n\right] \end{aligned} \tag{7}$$

2.2 Surface Boundary Condition

The spherical Laplacian derived in Section 2.1 can be solved numerically to find the solution to the heat equation in problems such as that formulated in Equation 8.

$$\begin{cases} u_t = \nabla^2 u, & t \in [0,1], \quad r \in (0,1) \\ u(1,t) = 0, & t \in [0,1] \\ u(r,0) = 1, & r \in (0,1) \end{cases} \tag{8}$$

Cartesian Discretization Physically, this situation might arise when a uniformly-heated ball is submerged in an ice bath, setting the outer boundary condition to $u(r=1,\phi,\theta,t) = 0$. As the ball cools, the heat will decrease isotropically toward the center of the ball. I will exploit this symmetry to simplify the three-dimensional geometry to be one-dimensional such that $u(\boldsymbol{r},t) = u(r,\phi,\theta) \to u(r,t)$. Further recognizing that this problem now has identical geometry to that of a 1D rod with the domain of the sphere's diameter, characterized by the following differential equation:

$$\begin{cases} u_t = u_{xx}, & t \in [0,1], \quad x \in (0,1) \\ u(-1,t) = u(1,t) = 0, & t \in [0,1] \\ u(x,0) = 1, & r \in (-1,1) \end{cases}$$

This now resembles the heat equation solved for Homework #1, so I will skip the derivation of the solution process. I solved for u_j^{n+1} explicitly using Equations 9a and 9b:

$$\boldsymbol{u}^{n+1} = A\boldsymbol{u}^n \tag{9a}$$

$$A = \begin{pmatrix} 1-2\lambda & \lambda & & & \\ & \ddots & \ddots & & \\ & \lambda & 1-2\lambda & \lambda & \\ & & & \ddots & \ddots \\ & & & \lambda & 1-2\lambda \end{pmatrix} \tag{9b}$$

Spherical Discretization A problem arises in Equation 7 as $r \to 0$ and Equation 6 blows up. This is reflected in the difference scheme around $j = 0$. We can address this by examining this limit analytically using l'Hôpital's rule:

$$\lim_{r\to 0}\nabla^2 u(r) = \left.\frac{\partial^2 u}{\partial r^2}\right|_{r=0} + 2\lim_{r\to 0}\frac{1}{r}\frac{\partial u}{\partial r} \overset{\text{l'H}}{=} \left.\frac{\partial^2 u}{\partial r^2}\right|_{r=0} + 2\lim_{r\to 0}\frac{\partial^2 u}{\partial r^2} = 3\frac{\partial^2 u}{\partial r^2}$$

Now, at $j = 0$, I will discretize $u_t = 3u_{rr}$:

$$\left.\frac{du}{dr}\right|_{r=0} = 0\frac{u^n_{j+1} - u^n_{j-1}}{2h} = 0 \quad\Longrightarrow\quad u^n_{j-1} = u^n_{j1}$$

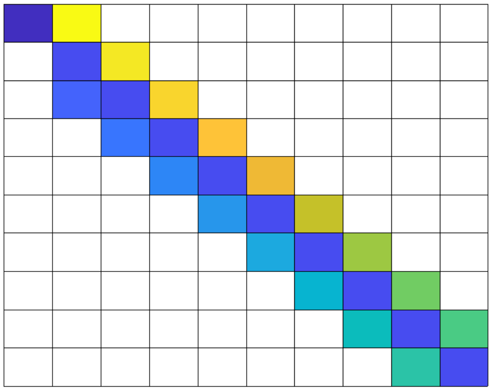
$$u^{n+1}_j = u^n_j + 3\left(\frac{u_{-1} - 2u_0 + u_1}{h^2}\right)$$

$$\boldsymbol{u}^{n+1}_j = (I - \lambda A)\,\boldsymbol{u}^n_j \tag{10}$$

$$A = \left(\begin{array}{cccccccc}
-6 & 6 & & & & & & \\
 & -2 & 2 & & & & & \\
 & 1/2 & -2 & 3/2 & & & & \\
 & & 2/3 & -2 & 4/3 & & & \\
 & & & \ddots & \ddots & \ddots & & \\
 & & & & (j-1)/j & -2 & (j+1)/j & \\
 & & & & & \ddots & \ddots & \ddots \\
 & & & & & (H-3)/(H-2) & -2 & (H-1)/(H-2) \\
 & & & & & & (H-2)/(H-1) & -2
\end{array}\right) \tag{11}$$

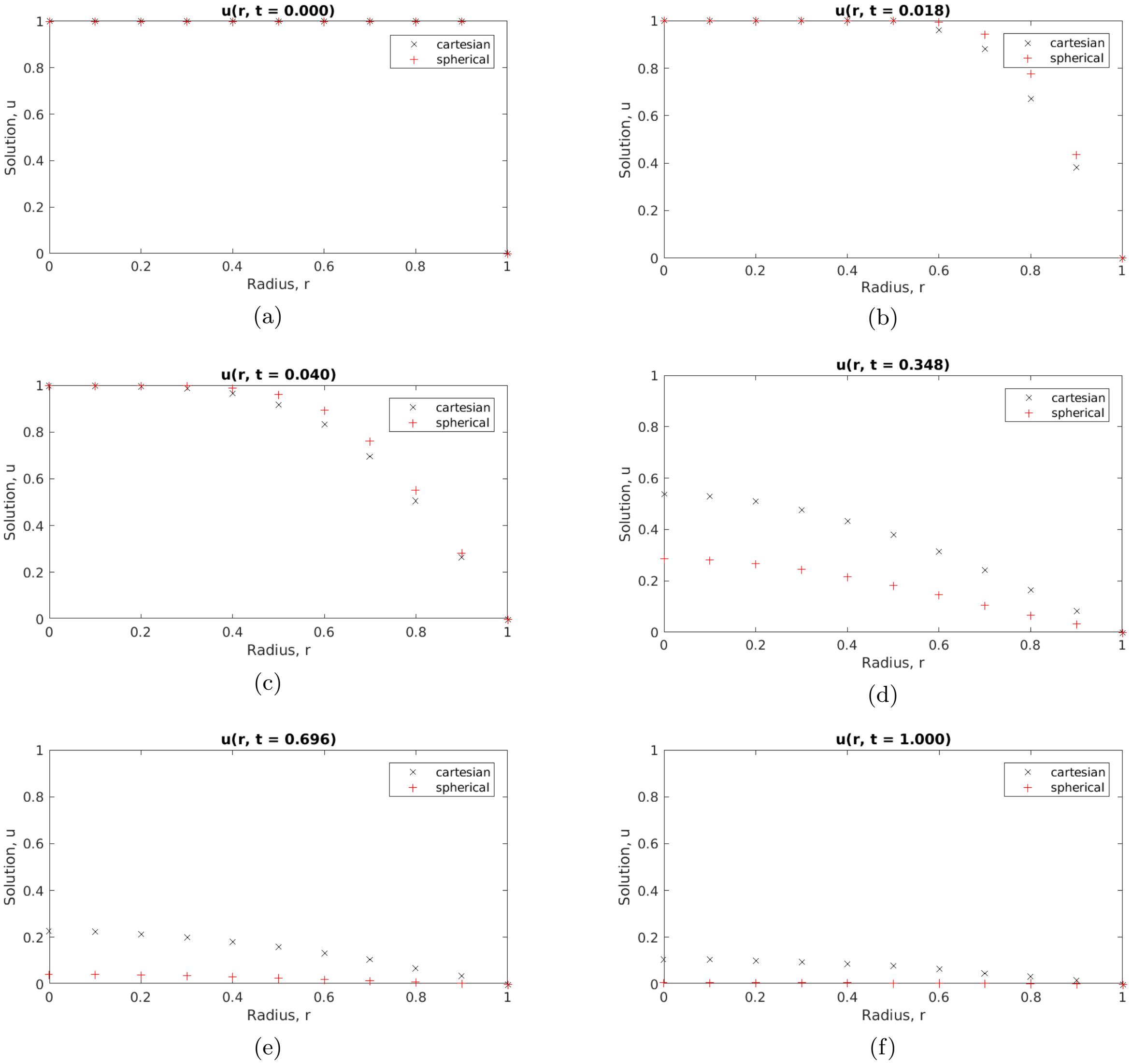
Graphically, Equation 11 can be represented as Fig. 1. From both the equation and the figure, another problem becomes apparent: the value of the center point at $j = 0$ depends on the value of the point at $j = 1$, but the point at $j = 1$ does not get information from the center point.

Figure 1: Visual representation of the A matrix applied to the function of $\boldsymbol{u}(x, t = nk)$ to iterate to find the solution at the following time step



The breakdown in communication at the $j = 1$ point causes this point to be more heavily influenced by the $j = 2$ point. Therefore, this point will be able to "pull" on the center point without experiencing any pull in return. This is apparent in Fig. 2, which shows the solution found using both cartesian and spherical difference schemes for several time steps on the solution time domain. The cartesian method solves over the entire sphere diameter; however, Fig. 2 shows only the solution for $r \in [0, 1]$ for easier comparison with the spherical scheme. This problem is further exacerbated in Section 2.3 when the solution at the center point is fixed to $u(0, t) = 1$.

Figure 2: The solution to Equation 8 at six different times on the interval $0 \leq t \leq 1$ found using both cartesian and spherical methods as discussed in Section 2.2



(a)

(b)

(c)

(d)

(e)

(f)

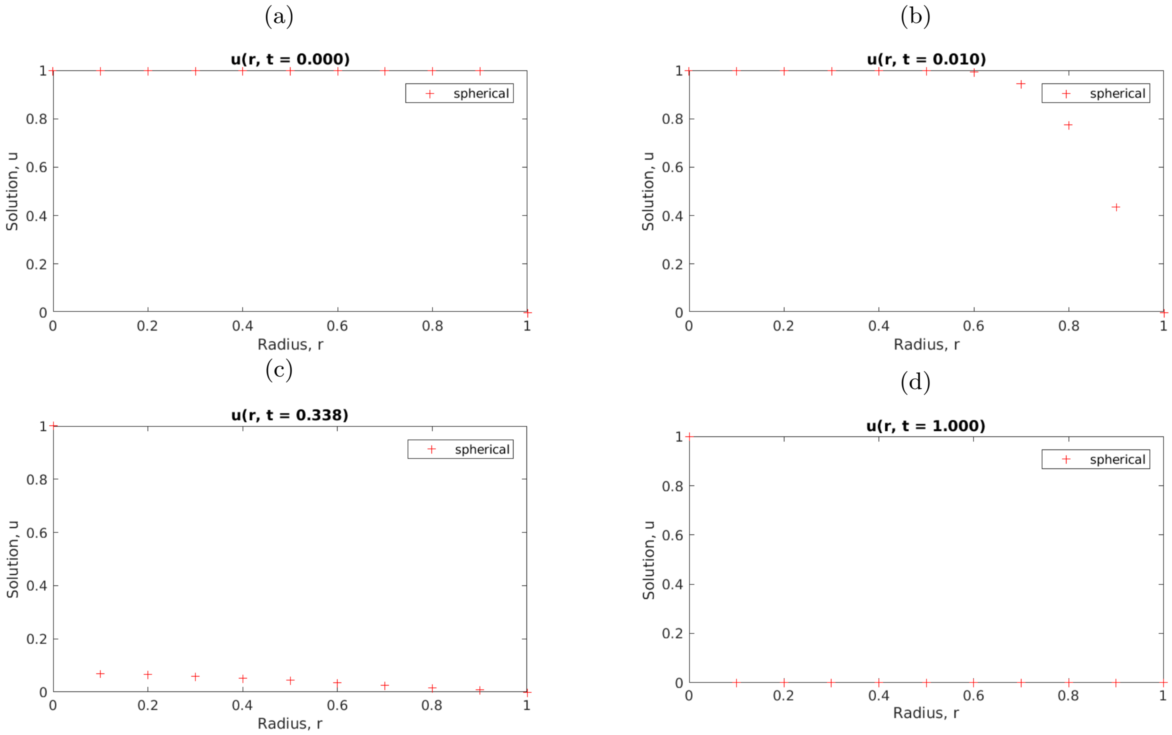
2.3 Center Condition

The phenomenon described in Section 2.2 is even more apparent for Equation 12, which now fixes the center point to $u(0, t) = 1$.

$$\begin{cases} u_t = \nabla^2 u, & t \in [0,1], \quad r \in (0,1) \\ u(0,t) = u(1,t) = 0, & t \in [0,1] \\ u(r,0) = 1, & r \in (0,1) \end{cases} \tag{12}$$

I adjusted the matrix to account for this new center condition and solved the spherical equation the same way I did in Section 2.2. Fig. 3 shows the solution plotted at four time steps using this method. Here, we can see that the second point does not communicate at all with the center point.

Figure 3: The solution to Equation 12 at four different times on the interval $0 \le t \le 1$ found using the spherical method discussed in Section 2.2


(a)
(b)
(c)
(d)

In the future, I would address this by reformulating the problem to solve for some $w(r,t) = u(r,t)/r$ so as not to have to deal with a $1/r$ term in the difference scheme.

2.4 Stability Requirement

Assessing a method's stability before even attempting to implement a numerical method is a huge time saver in the long run. I analyzed the stability of the method given in Equation 7, developed by von Neumann yet often attributed to Fourier [1]. We can consider a functional solution $\boldsymbol{u}(\xi)$ to the heat equation, with Fourier coefficients $\boldsymbol{u}_j^n$:

$$\boldsymbol{u}_j^n = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{\imath j \xi} \boldsymbol{u}(\xi) d\xi$$

The solution at each successive time step $\boldsymbol{u}_j^{n+1}$ can now be written:

$$\begin{aligned}
\boldsymbol{u}_j^{n+1} &= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\imath j\xi}\left(1+\lambda\left(e^{\imath\xi}-2+e^{-\imath\xi}\right)+\frac{\lambda}{j}\left(e^{\imath\xi}-e^{-\imath\xi}\right)\right)\boldsymbol{u}(\xi)d\xi \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\imath j\xi}\left(1+\lambda(2\imath)^2\left(\frac{e^{\imath\xi/2}-e^{-\imath\xi/2}}{2\imath}\right)^2+\frac{\lambda}{j}(2\imath)\left(\frac{e^{\imath\xi}-e^{-\imath\xi}}{2\imath}\right)\right)\boldsymbol{u}(\xi)d\xi \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\imath j\xi}\left(1-4\lambda\sin^2\left(\xi/2\right)+2\imath\frac{\lambda}{j}\sin(\xi)\right)\boldsymbol{u}(\xi)d\xi \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\imath j\xi}\rho(\xi)\boldsymbol{u}(\xi)d\xi
\end{aligned}$$

Applying Parseval's identity Section 2.4,

$$\sum_{j=-\infty}^{\infty}\left|u_j^{n+1}\right|^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left|\rho(\xi)\boldsymbol{u}(\xi)\right|^2 d\xi \le \left(\max|\rho(\xi)|\right)^2\frac{1}{2\pi}\int_{-\pi}^{\pi}|\boldsymbol{u}(\xi)|^2 d\xi = \left\|\rho(\xi)\right\|_\infty^2\sum_{j=-\infty}^{\infty}\left|u_j^n\right|^2$$

The amplification factor must be bounded such that $|\rho(\xi)| \le 1$.

$$\begin{aligned}
\rho(\xi) &= 1-4\lambda\sin^2(\xi/2)+2\imath\frac{\lambda}{j}\sin(\xi) \\
|\rho(\xi)|^2 &= \left(1-4\lambda\sin^2(\xi/2)\right)^2+4\frac{\lambda^2}{j^2}\sin^2(\xi) \\
&\le \left(1-4\lambda\sin^2(\xi/2)\right)^2+4\lambda^2\sin^2(\xi) \\
&= 1-8\lambda\sin^2(\xi/2)+16\lambda^2\sin^4(\xi/2)+8\lambda^2\sin^2(\xi/2)\cos^2(\xi/2) \\
&= 1-8\lambda\sin^2(\xi/2)\underbrace{\left[1-\cos^2(\xi/2)\right]}_{\sin^2(\xi/2)}+16\lambda^2\sin^4(\xi/2) \\
&= 1-\left(8\lambda-16\lambda^2\right)\sin^4(\xi/2) \le 1-8\lambda+16\lambda^2 \\
&= (1-4\lambda)^2
\end{aligned}$$

This requirement on the amplification factor can be applied to impose stability requirements on λ.

$$\rho(\xi) \ge 1-4\lambda \ge -1 \quad\Longrightarrow\quad \boxed{\lambda \le \frac{1}{2}}$$

This is the same as the stability requirement in cartesian coordinates! This is pretty exciting because it means that we can allow geometry to guide our choice coordinate system without fear of increased instability.

3 Swimming Pool

$$\begin{cases} u_t = \alpha \nabla^2 u, & \forall\, t, \quad \text{on } \partial\Omega \\ \dfrac{du}{dn}\hat{\mathbf{n}} = 0, & \forall\, t, \quad \text{on } \partial\Omega \\ u(x, y, z = 1, t) = 1, & \forall\, t \in \text{day} \\ u(x, y, z = 1, t) = 0, & \forall\, t \in \text{night} \\ u(\boldsymbol{r}, 0) = 0, & \forall\, t \end{cases} \tag{13}$$

Numerical Solution using Alternating Direction Implicit

Equations 14a to 14c show the three steps of the alternating direction implicit (ADI) scheme, which includes intermediary solutions at non-physical points represented at $u_j^{n+1/3}$ and $u_j^{n+2/3}$. Respectively, these equations are implicit in x, explicit in z; implicit in y, explicit in x; and implicit in z, explicit in y.

$$\left(1 - \frac{\alpha k}{2}\delta_x^2\right)u_j^{n+1/3} = \left(1 + \frac{\alpha k}{2}\delta_z^2\right)u_j^n \tag{14a}$$

$$\left(1 - \frac{\alpha k}{2}\delta_y^2\right)u_j^{n+2/3} = \left(1 + \frac{\alpha k}{2}\delta_x^2\right)u_j^{n+1/3} \tag{14b}$$

$$\left(1 - \frac{\alpha k}{2}\delta_z^2\right)u_j^{n+1} = \left(1 + \frac{\alpha k}{2}\delta_y^2\right)u_j^{n+2/3} \tag{14c}$$

where k is the step size in time and α is the diffusion coefficient in water. Incorporating physical constants into this ADI scheme will be discussed further in Section 3.2.

Difference scheme in x and y The pool has the same Neumann insulating boundary condition (BC) one either wall; therefore, the solutions in x and y will have the same shape can be found using the same difference scheme. will address the insulating BCs by adding a ghost point directly outside of the matrix, outside u_0 and u_H at u_{-1} and u_{H+1}. Using central differences to approximate these Neumann BCs, it is possible to relate these ghost points to the points directly inside the matrix.

$$\left.\frac{du}{dn}\right|_{u_0} = \frac{u_1 - u_{-1}}{2h} \implies u_1 = u_{-1}$$

$$\left.\frac{du}{dn}\right|_{u_H} = \frac{u_{H+1} - u_{H-1}}{2h} \implies u_{H-1} = u_{H+1}$$

Substituting these relationships into the central difference scheme used to approximate δ^2[1],

$$\frac{u_{-1} - 2u_0 + u_1}{h^2} = \frac{2(-u_0 + u_1)}{h^2}$$

$$\frac{u_{H-1} - 2u_H + u_{H+1}}{h^2} = \frac{2(u_{H-1} - u_H)}{h^2}$$

In matrix form, δ_x^2 and δ_y^2 can be represented as $H(H+1) \times H(H+1)$ block matrix of $(H+1) \times (H+1)$ matrices given in Equation 16.

[1] I am intentionally vague on direction since this method applies to derivatives in both x and y. Also, since my stencils only incorporate one direction at a time, I am using j to represent steps in x, y, or z, depending on the context.

$$\delta_x^2 = \delta_y^2 = \frac{1}{h^2}\begin{pmatrix} -2 & 2 & & & & & \\ 1 & -2 & 1 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & 1 & -2 & 1 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & 1 & -2 & 1 \\ & & & & & 2 & -2 \end{pmatrix} \tag{15}$$

Difference scheme in z The same insulating BCs are imposed at the floor of the pool, so the same technique can be applied here. Now, instead of an insulating BC at the surface of the pool, we have a heat source from the sun incident on the water, such that $u_H = f(t)$, where f is a known function dependent on the time of day. This function will be discussed in detail in Section 3.1. For now, I will assume this is a step function, as specified in Equation 13.

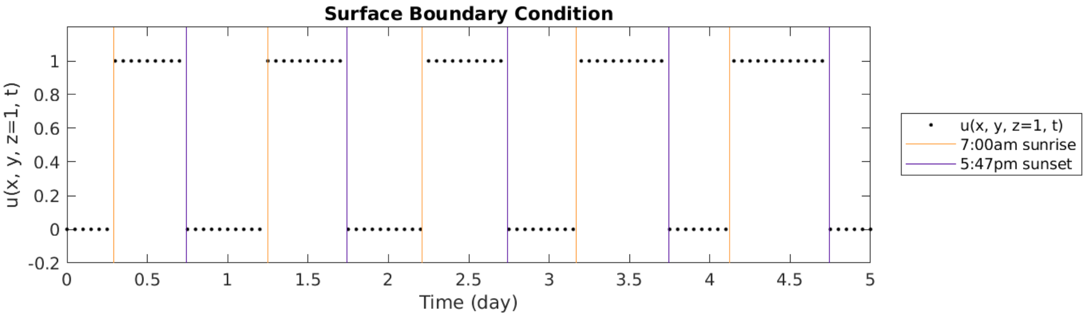
$$\frac{u_{H-2} - 2u_{H-1} + u_H}{h^2} = \frac{u_{H-2} - 2u_{H-1} + f(t)}{h^2}$$

Now, δ_z^2 can be represented as a $(H+1)^2 \times (H+1)^2$ block matrix of $H \times H$ matrices given in Equation 16.

$$\delta_z^2 \boldsymbol{u}_j^n = D\boldsymbol{u}_j^n + \boldsymbol{f}(t) \tag{16}$$

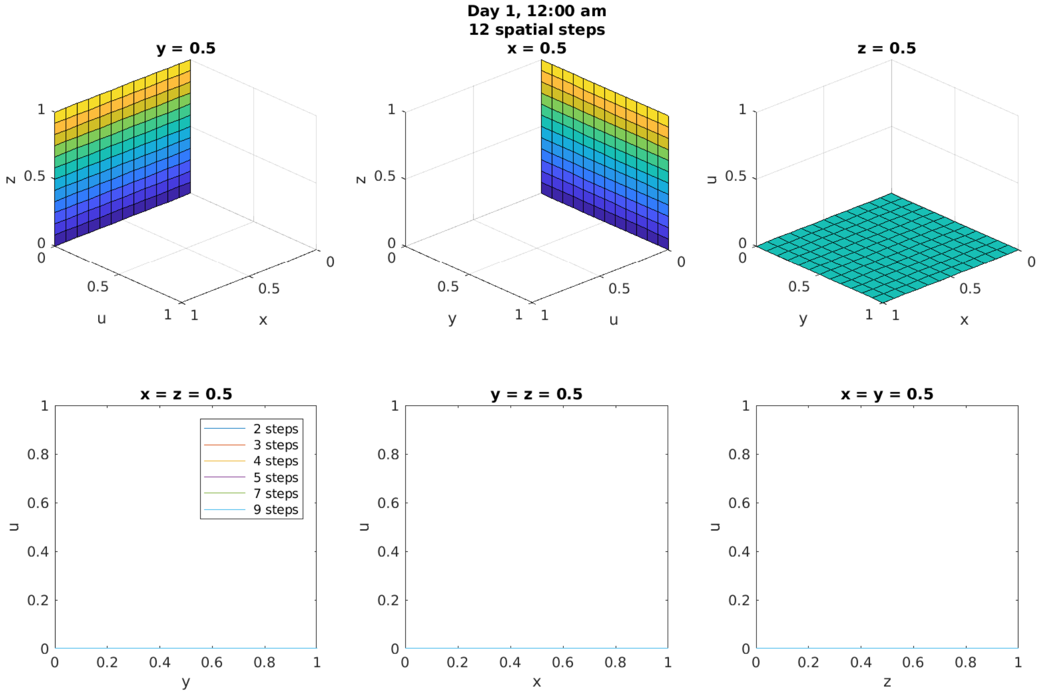
$$\text{where } D = \frac{1}{h^2}\begin{pmatrix} -2 & 2 & & & & & \\ 1 & -2 & 1 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & 1 & -2 & 1 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & 1 & -2 & 1 \\ & & & & & 1 & -2 \end{pmatrix}, \qquad \boldsymbol{f}(t) = \frac{1}{h^2}\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ f(t) \end{pmatrix}$$

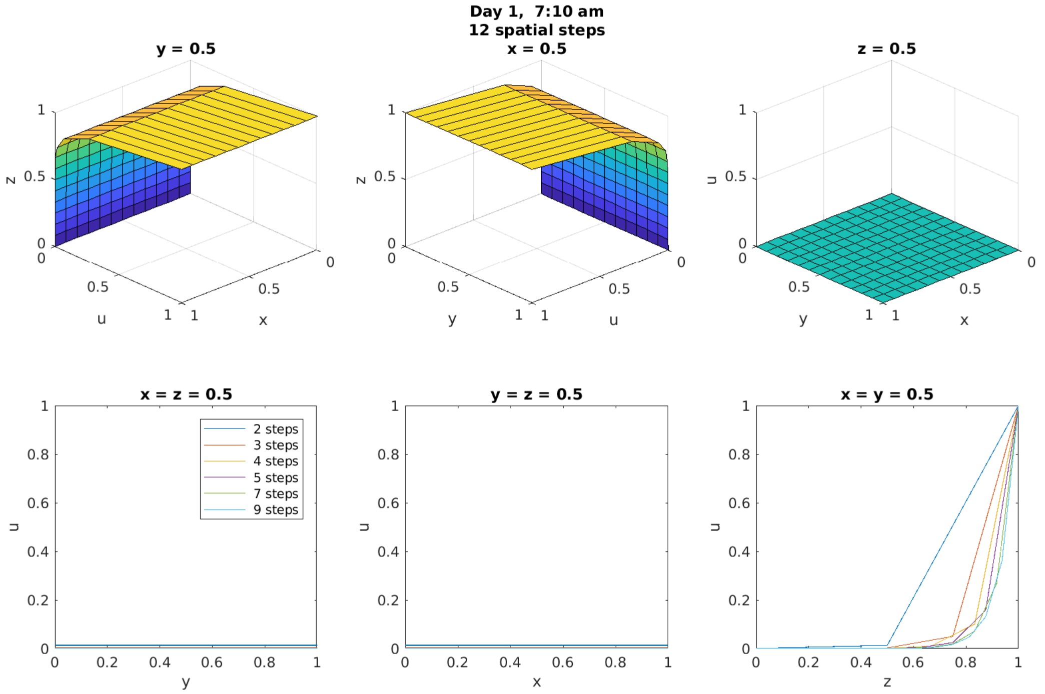
Figure 4

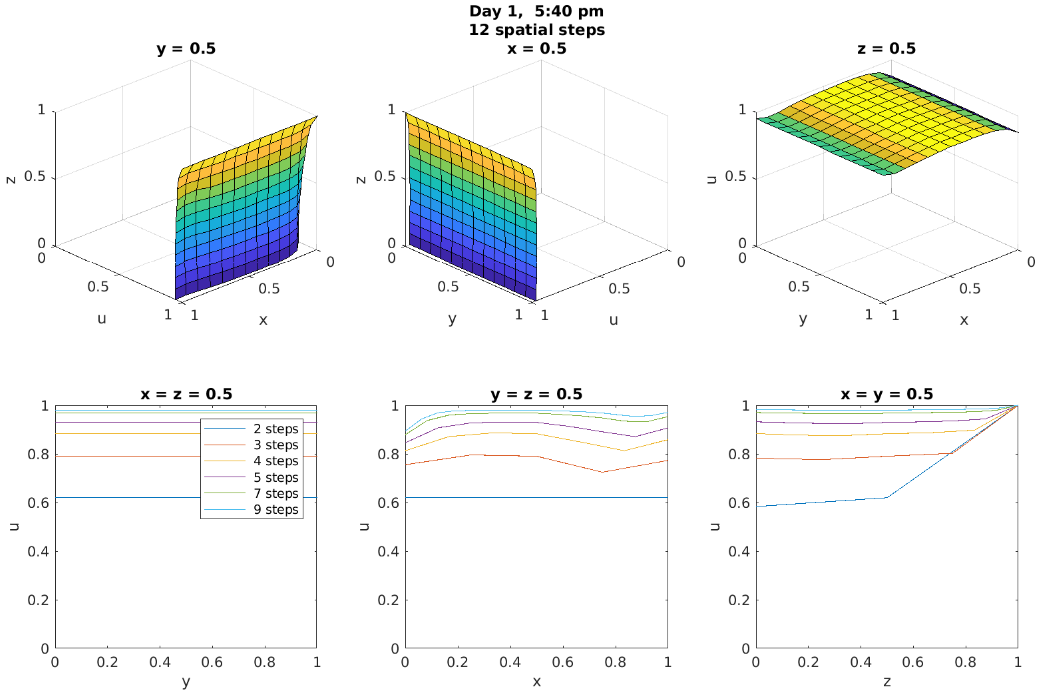


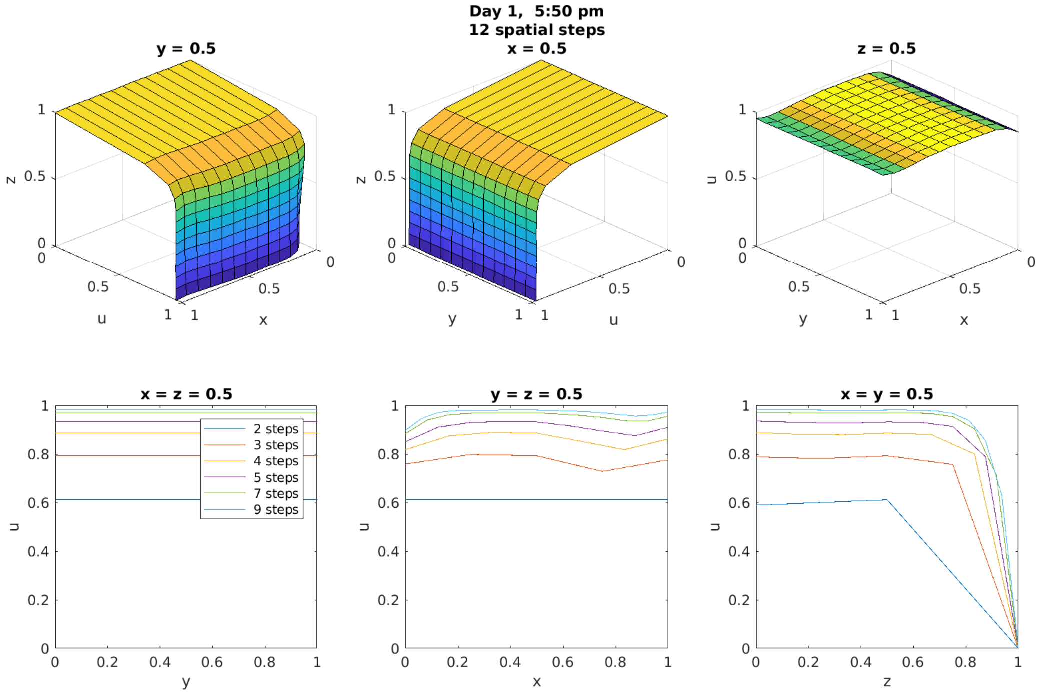
Equation 13 specifies a step function BC on the surface of the pool. On February 14, 2018, sunrise and sunset in Berkeley, California, occured at 7:00am and 5:47pm, respectively. Each day that follows, the sun

rises one day earlier and sets one day later, a phenomenon reflected in my boundary condition function `SurfaceStepBC`. The boundary condition over the five day period is plotted in Fig. 4.
Section 3.1 more accurately reflects the position of the sun.


Day 1, 12:00 am
12 spatial steps
y = 0.5
x = 0.5
z = 0.5
z
u
x
y
0
0.5
1
x = z = 0.5
y = z = 0.5
x = y = 0.5
2 steps
3 steps
4 steps
5 steps
7 steps
9 steps
0.2
0.4
0.6
0.8


Day 1, 7:10 am
12 spatial steps
y = 0.5
x = 0.5
z = 0.5
z
u
x
y
0
0.5
1
x = z = 0.5
y = z = 0.5
x = y = 0.5
2 steps
3 steps
4 steps
5 steps
7 steps
9 steps
0.2
0.4
0.6
0.8


Day 1, 5:40 pm
12 spatial steps
y = 0.5
x = 0.5
z = 0.5
x = z = 0.5
y = z = 0.5
x = y = 0.5
2 steps
3 steps
4 steps
5 steps
7 steps
9 steps
u
x
y
z


Day 1, 5:50 pm
12 spatial steps
y = 0.5
x = 0.5
z = 0.5
x = z = 0.5
y = z = 0.5
x = y = 0.5
2 steps
3 steps
4 steps
5 steps
7 steps
9 steps
u
x
y
z

3.1 Angle of the Sun

The angle of the sun in the sky depends on the time of day and year as well as position on Earth. I found an Excel spreadsheet distributed by National Oceanic and Atmospheric Administration (NOAA), which, given the latitude, longitude, time zone, and date, calculates comprehensive solar geometry data, including the elevation angle (as well as that corrected for atmospheric refraction – this is the value I used) over six minute time intervals. The default location in this spreadsheet is set to 40°N, -105°W – Boulder, CO, where NOAA is located and I went to undergrad (sko Buffs!).

To maintain consistency with my first solution, I again chose February, 14-18, 2018, as my date range. Fig. 6 shows the solar elevation angle as a function of time of day for each date in this range. I chose to superimpose this data in the plot to show that there is very little change in the elevation angle curve between each of the five days. Considering my time step size of 6 min, these changes will be very negligable over five days; but, to maintain generality for future calculations, I will use the boundary condition specific to each day.

Figure 6

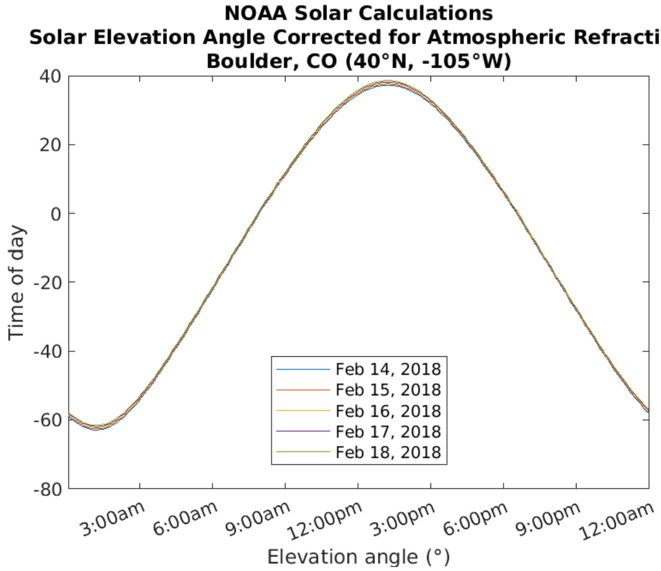


Figure 7

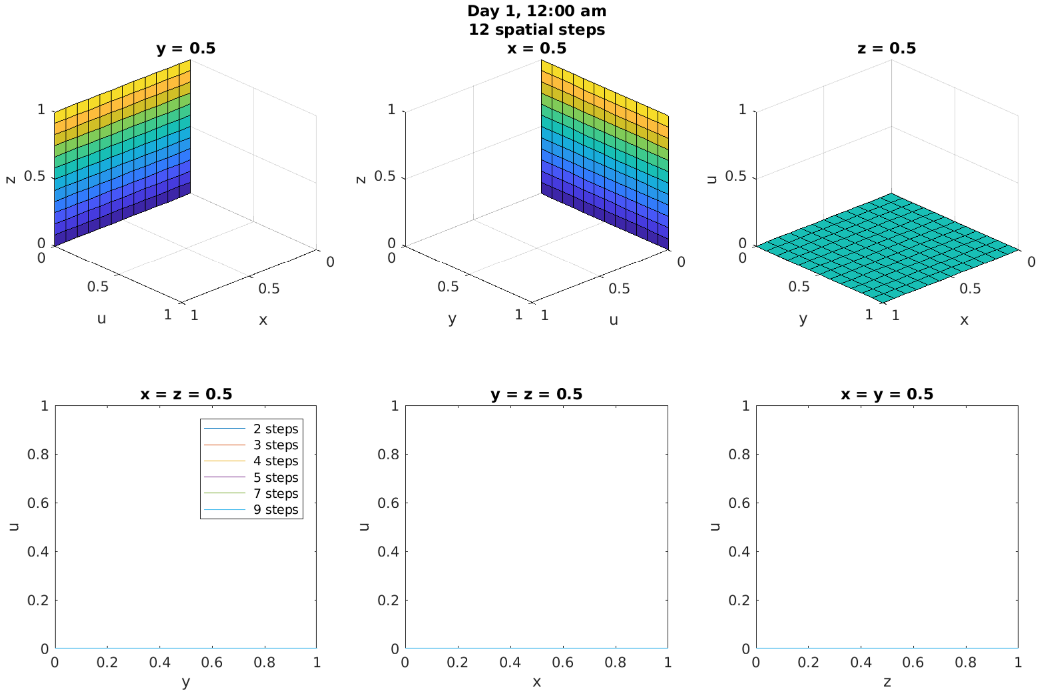
I will assume that only the z-component of the sunshine contributes to the surface boundary condition:

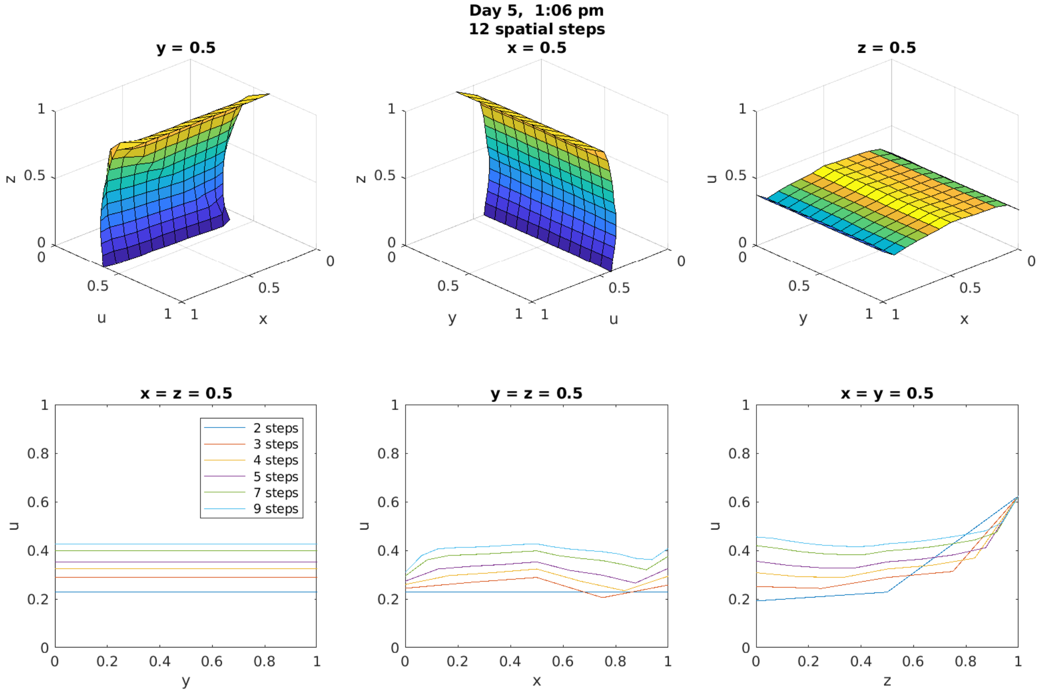
$$f(t) = \sin(\theta_{\mathrm{el}})$$

where θ_{el} is the elevation angle calculated by NOAA.

3.2 Physical constants

In addition to pulling actual physical data describing the sun's position as I did in Section 3.1, I can also adjust my solution method to include the thermal diffusivity of water by introducing a diffusion coefficient of $\alpha 0.143 \times 10^{-6}\,\mathrm{m}^2/\mathrm{s} = 0.012\,34\,\mathrm{m}^2/\mathrm{d}$.


Day 1, 12:00 am
12 spatial steps
y = 0.5
x = 0.5
z = 0.5
z
u
x
y
u
y
x
x = z = 0.5
y = z = 0.5
x = y = 0.5
2 steps
3 steps
4 steps
5 steps
7 steps
9 steps
u
y
x
z
0
0.2
0.4
0.6
0.8
1
0.5


Day 5, 1:06 pm
12 spatial steps
y = 0.5
x = 0.5
z = 0.5
z
u
x
y
u
y
x
x = z = 0.5
y = z = 0.5
x = y = 0.5
2 steps
3 steps
4 steps
5 steps
7 steps
9 steps
u
y
x
z
0
0.2
0.4
0.6
0.8
1
0.5

References

[1] David Kincaid and Ward Cheney. *Numerical Analysis: Mathematics of Scientific Computing*. 3rd. American Mathematical Society, 2002.