James V. Beck, Filippo de Monte, et al. – December 5, 2011

X22B10T0 = X22B(*t*1)0T0 problem

fX22B10T0, fvX22B10T0

Heat conduction functions for the X22B10T0 case.

Syntax (Matlab)

```
Td = fX22B10T0(xd, td)
Td = fX22B10T0(xd, td, n)
Td = fvX22B10T0(xd, td)
Td = fvX22B10T0(xd, td, n)
```

Description

fX22B10T0 (*xd*, *td*) provides the dimensionless temperature distribution *Td* at a given dimensionless location *xd* from the heated surface, between 0 and 1, and at a given dimensionless time *td* with a default accuracy of 10^{-6} for a 1D Cartesian finite slab subject to a time-independent surface heat flux at one side (step change) and thermally insulated at the opposite side.

fvX22B10T0 (*xd*, *td*) provides the dimensionless temperature distribution *Td* in a matrix form for the same problem when *xd* and *td* are vectors defining the dimensionless locations and times of interest, respectively. If *xd* and *td* are vectors, length(*xd*) = *n* and length(*td*) = *m*, where [*m* ,*n*] = size(*Td*). The default accuracy is of 10^{-6}.

fX22B10T0 (*xd*, *td*, *n*) and fvX22B10T0 (*xd*, *td*, *n*) give the dimensionless temperature distribution *Td* for the same problem with an accuracy of 10^{-n} (*n* = 1, 2, …)

n = integer (1, 2, …, 10, …) for solution accuracy; *n* = 15 gives an accuracy of one part in 10^{15} (machine accuracy)

Examples

Example 1

```
Td=fX22B10T0(0,.1,2)

Td =

   0.356829372437085
```

Example 2

```
Td=fX22B10T0(0,.1,15)

Td =

    0.356826246008654
```

Example 3

```
>> n=15

n =

    15

>> xd=[0.1 0.2 0.3]

xd =

   0.100000000000000   0.200000000000000   0.300000000000000

>> td=[0.1 0.2 0.3]

td =

   0.100000000000000   0.200000000000000   0.300000000000000

>> Td=fvX22B10T0(xd,td,n)

Td =

   0.265710758078575   0.411546515326888   0.528355069707451
   0.191930076891576   0.330554257779286   0.444845453330602
   0.134245086447475   0.261793430683321   0.372166721740129
```

Example 4

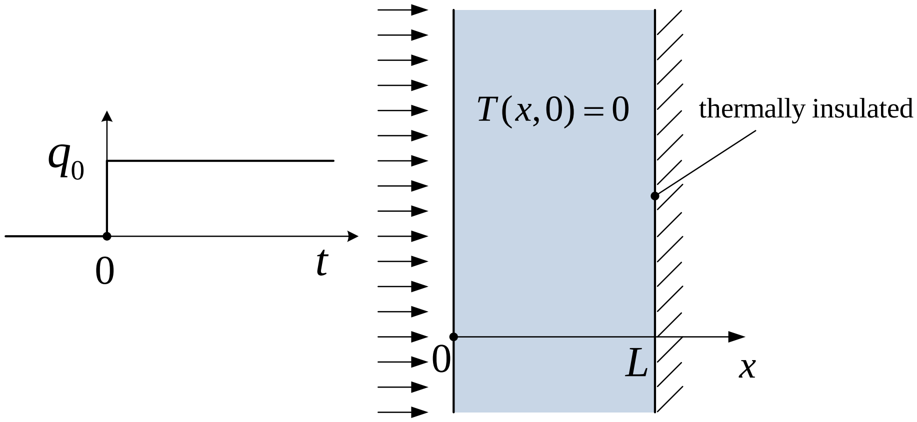
```
>> n=15

n =

    15

>> xd=[0.1 0.5 0.7]'

xd =

   0.100000000000000
   0.500000000000000
   0.700000000000000

>> td=[0.01 0.2]'

td =

   0.010000000000000
   0.200000000000000

>> Td=fvX22B10T0(xd,td,n)

Td =

   0.039928245674849   0.411546515326888
   0.000014352414313   0.158352196668220
   0.000000019773817   0.094884894165447
```

Heat Conduction Toolbox – HC Toolbox

X22B10T0

James V. Beck, Filippo de Monte, et al. – December 5, 2011

Schematic



Nomenclature

- k thermal conductivity ($W/(m\ °C)$)
- L slab thickness (m)
- q_0 surface heat flux (W/m^2)
- t time (s)
- T temperature ($°C$)
- x Cartesian space coordinate (m)
- α thermal diffusivity (m^2/s)

Governing equations

$$\frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \qquad (0 < x < L;\ t > 0)$$

$$-k\left(\frac{\partial T}{\partial x}\right)_{x=0} = q_0 \qquad (t > 0)$$

$$\left(\frac{\partial T}{\partial x}\right)_{x=L} = 0 \qquad (t > 0)$$

$$T(x,0) = 0 \qquad (0 < x < L)$$

- **Exact short-time solution ([1, p. 112, Eq. (4), for the X22B01T0 case], [2, p. 196, Eq. (6.52b) with *m* = 0 and Eq. (6.53c)])**

$$
\begin{aligned}
T(x,t) &= 2\frac{q_0\sqrt{\alpha t}}{k}\,\text{ierfc}\left(\frac{x}{2\sqrt{\alpha t}}\right) \\
&\quad + 2\frac{q_0 L}{k}\left(\frac{\alpha t}{L^2}\right)^{1/2}\sum_{m=1}^{\infty}\left[\text{ierfc}\left(\frac{2mL+x}{2\sqrt{\alpha t}}\right)+\text{ierfc}\left(\frac{2mL-x}{2\sqrt{\alpha t}}\right)\right]
\end{aligned}
\qquad (0 \le x \le L\,;\; t \ge 0)
$$

where the first term on the RHS is the well-known 1D Cartesian semi-infinite solution of the X20B1T0 problem [1, p. 75, Eq. (6)]. In addition, $\text{ierfc}(z)$ is the complementary error function integral defined as [2, p. 498, Eq. (E.9a)]

$$
\text{ierfc}(z) = \int_z^{\infty} \text{erfc}(t)\,dt
$$

The relationship between the $\text{ierfc}(z)$ function and the complementary error function $\text{erfc}(z)$ returned by the Matlab function `erfc` is [2, p. 501, Eq. (E.14a)]

$$
\text{ierfc}(z) = \frac{e^{-z^2}}{\sqrt{\pi}} - z\,\text{erfc}(z)
$$

The short-time solution comes from the application of Laplace transform to the governing equations. It is valid at any time but it is computationally convenient at short times.
If the time t at a given location x is less than the 1D deviation time $t^{(dev)}$ [3, p. 5935, Eq. (19)], that is,

$$
t^{(dev)} = \frac{0.1}{n\alpha}(2L-x)^2 \qquad (n = 1,2,\ \dots,\ 15),
$$

we can consider only the first term in the above exact short-time solution with errors less than 10^{-n}. (Note that n = 2 is for visual comparison, while n = 15 is for verification purposes of large numerical codes.) Then, we have

$$
T(x,t) \approx 2\frac{q_0\sqrt{\alpha t}}{k}\,\text{ierfc}\left(\frac{x}{2\sqrt{\alpha t}}\right)
$$

This indicate that, at short times (less than the deviation time listed before), the thermal deviation effects due to the homogeneous 'inactive' boundary condition at $x = L$ are negligible (less than 10^{-n}) and the 1D finite slab can be considered as 1D semi-infinite along x and subject to a time-independent surface heat flux.

- **Exact large-time solution ([2, p. 205, Eqs. (6.87) and (6.95)], [4, p. 2559, Eq. (25a)])**

$$T(x,t) = \frac{q_0 L}{k}\left[\frac{\alpha t}{L^2} + \underbrace{\frac{1}{2}\left(\frac{x}{L}\right)^2 - \frac{x}{L} + \frac{1}{3}}_{steady\ state\ component} - \underbrace{2\sum_{m=1}^{\infty}\frac{1}{(m\pi)^2}\cos\left(m\pi\frac{x}{L}\right)e^{-(m\pi)^2\frac{\alpha t}{L^2}}}_{complementary\ transient\ component}\right]$$

$$(0 \le x \le L\,;\ t \ge 0)$$

The large-time solution comes from the application of separation-of-variables (SOV) method to the governing equations. It is valid at any time but it is computationally convenient at large times.
If the time t is greater than a characteristic time $t^{(c)}$, that is,

$$t^{(c)} = \frac{n \ln 10}{\pi^2}\frac{L^2}{\alpha} \qquad (n = 1,2, \ldots, 15),$$

we can consider at any location x only one term in the above exact large-time solution with errors less than 10^{-n}. Then, we have

$$T(x,t) \approx \frac{q_0 L}{k}\left[\frac{\alpha t}{L^2} + \frac{1}{2}\left(\frac{x}{L}\right)^2 - \frac{x}{L} + \frac{1}{3} - \frac{2}{\pi^2}\cos\left(\pi\frac{x}{L}\right)e^{-\pi^2\frac{\alpha t}{L^2}}\right]$$

Dimensionless quantities

The Π theorem states that, with four basic dimensions, mass, [M], length, [L], time, [t] and temperature, [T], a reduction of up to four may be hoped for in the number of the variables (seven) involved in the X22B10T0 problem. Therefore, we have a total of three dimensionless groups

$$T_d = \frac{T}{q_0 L / k}, \qquad x_d = \frac{x}{L}, \qquad t_d = \frac{\alpha t}{L^2} = Fo,$$

where $x_d \in [0,1]$ and Fo is the well-known Fourier number.

Computation of the dimensionless temperature solution at any location and time

$$T_d(x_d, t_d) \approx \begin{cases} 2\sqrt{t_d}\,\mathrm{ierfc}\left(\dfrac{x_d}{2\sqrt{t_d}}\right) & \text{for} \quad 0 \le t_d < t_d^{(p)} \\ t_d + \dfrac{x_d^2}{2} - x_d + \dfrac{1}{3} - 2\displaystyle\sum_{m=1}^{M} \dfrac{1}{(m\pi)^2} \cos(m\pi x_d) e^{-(m\pi)^2 t_d} & \text{for} \quad t_d^{(p)} \le t_d \le t_d^{(c)} \\ t_d + \dfrac{x_d^2}{2} - x_d + \dfrac{1}{3} - \dfrac{2}{\pi^2} \cos(\pi x_d) e^{-\pi^2 t_d} & \text{for} \quad t_d > t_d^{(c)} \end{cases}$$

where

- $t_d^{(p)}$ is the dimensionless partitioning time. In this case, it is exactly the same as the 1D deviation time defined before. In dimensionless form, we have

  $$t_d^{(p)} = \frac{0.1}{n}(2 - x_d)^2 \qquad (n = 1,2, \ldots, 15)$$

  For n = 15, we have a machine accuracy, i.e. 10^{-15}, but the user can choose whichever accuracy s/he likes (10^{-n}).

- M is the maximum number of terms in the summation given by [2, p. 153, Subsection 5.2.1]

$$M = \mathrm{ceil}\left(\frac{n\ln 10}{\pi^2 t_d}\right)^{1/2} \qquad (n = 1,2, \ldots, 15)$$

where the function "ceil(A)" rounds the number A to the nearest integer greater than or equal to A. For n = 15, we have a machine accuracy but the user can choose the accuracy desired (10^{-n}). The tail S_M of the summation (2nd expression of $T_d(x_d, t_d)$) is given by [2, p. 153, Eq. (5.13)]

$$S_M = \frac{1}{M\pi^{3/2}}\mathrm{ierfc}\left(M\pi t_d^{1/2}\right)$$

It is always less than 10^{-n} [5].

- $t_d^{(c)}$ is the dimensionless characteristic time defined before. In dimensionless form, we have

$$t_d^{(c)} = \frac{n\ln 10}{\pi^2} \qquad (n = 1,2, \ldots, 15)$$

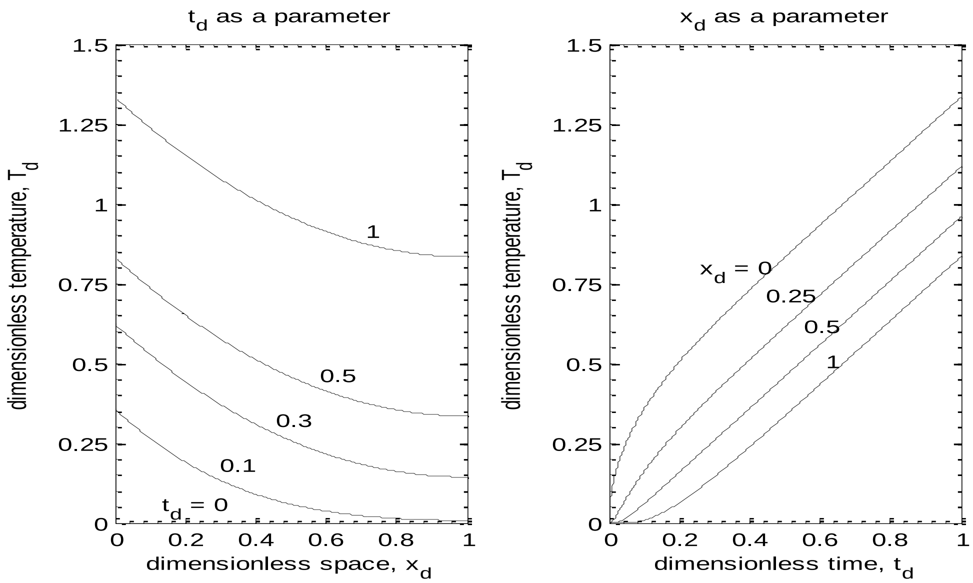
- **Matlab function:** `fX22B10T0.m`

```
% fX22B10T0.m function
% Revision History
% 11 12 11 written by James V. Beck and Filippo de Monte
% calling sequence:
% none called
function Td=fX22B10T0(xd,td,n)
if      td==0
        Td=0;
elseif 0<td<(0.1/n)*(2-xd)^2;
        arg=xd/sqrt(4*td);
        ierfc_arg=(1/sqrt(pi))*exp(-arg^2)-arg*erfc(arg); %
ierfc function
        Td=2*sqrt(td)*ierfc_arg; % X20B1T0
else
        M=round(sqrt(n*log(10)/(td*pi^2))); %maximum number of
terms
        Td=td+((1/3)-xd+xd^2/2); % Start X22B10T0 case
        for m=1:M
            beta=m*pi; % m-th eigenvalue
            Xf=cos(beta*xd); % m-th eigenfunction
            Td=Td-2*exp(-beta^2*td)*Xf/beta^2;
        end % for m
end % if
```

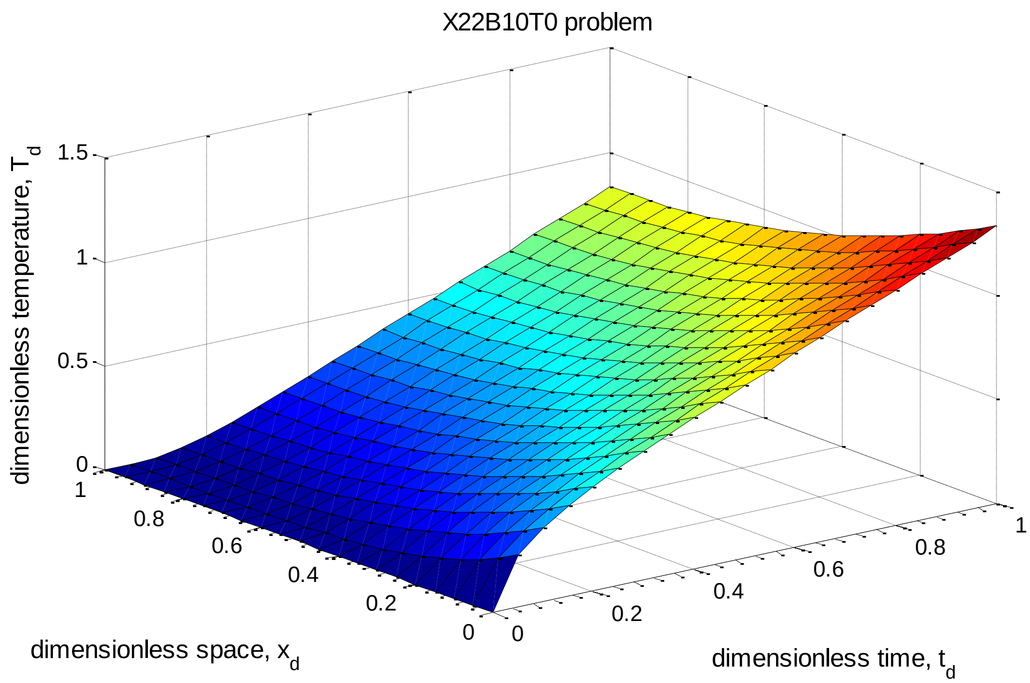
- **Matlab function:** `fvX22B10T0.m`

```
% fvX22B10T0.m function
% Revision History
% 11 16 11. Added option of xd and td being vectors by James
V. Beck
% 11 12 11 written by James V. Beck and Filippo de Monte
% calling sequence:
% none called
function Td=fvX22B10T0(xd,td,n)
sizex=length(xd);
xdv=xd;
sizet=length(td);
tdv=td; % end vector option
Td=zeros(sizex,sizet); % Preallocating Arrays for speed
for it=1:sizet
    td=tdv(it);
    for ix=1:sizex
        xd=xdv(ix);
        if      td==0
                Td(ix,it)=0;
        elseif 0<td<(0.1/n)*(2-xd)^2;
                arg=xd/sqrt(4*td);
                ierfc_arg=(1/sqrt(pi))*exp(-arg^2)-
                          arg*erfc(arg); % ierfc function
                Td(ix,it)=2*sqrt(td)*ierfc_arg; % X20B1T0
        else
                M=round(sqrt(n*log(10)/(td*pi^2))); % maximum
number of terms
                Td(ix,it)=td+((1/3)-xd+xd^2/2); % Start
X22B10T0 case
                for m=1:M
                    beta=m*pi; % m-th eigenvalue
                    Xf=cos(beta*xd); % m-th eigenfunction
                    Td(ix,it)=Td(ix,it)-2*exp(-
                                   beta^2*td)*Xf/beta^2;
                end % for m
        end % if
    end % ix
end % it
```

Dimensionless temperature values for various dimensionless times and distances. 2D Plot



- **Dimensionless temperature values for various dimensionless times and distances. 3D Plot**



Dimensionless temperature values for various dimensionless times and distances. Table

t_d	$x_d = 0$	$x_d = 0.25$	$x_d = 0.50$	$x_d = 0.75$	$x_d = 1.0$
0.00	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
0.01	0.11283792	0.00437714	0.00001435	0.00000000	0.00000000
0.02	0.15957691	0.02023475	0.00080165	0.00000841	0.00000004
0.03	0.19544100	0.03923836	0.00372182	0.00015023	0.00000484
0.04	0.22567583	0.05851013	0.00875429	0.00070229	0.00005741
0.05	0.25231325	0.07729750	0.01536594	0.00187860	0.00026934
0.06	0.27639532	0.09540469	0.02307429	0.00376392	0.00078555
0.07	0.29854108	0.11280721	0.03152836	0.00635986	0.00173473
0.08	0.31915391	0.12953703	0.04048632	0.00962972	0.00320662
0.09	0.33851415	0.14564372	0.04978409	0.01352296	0.00525097
0.10	0.35682625	0.16118032	0.05931089	0.01798635	0.00788529
0.11	0.37424506	0.17619814	0.06899204	0.02296853	0.01110419
0.12	0.39089169	0.19074503	0.07877719	0.02842164	0.01488727
0.13	0.40686340	0.20486485	0.08863241	0.03430182	0.01920511
0.14	0.42224011	0.21859759	0.09853486	0.04056907	0.02402351
0.15	0.43708878	0.23197957	0.10846913	0.04718709	0.02930629
0.16	0.45146649	0.24504372	0.11842483	0.05412294	0.03501718
0.17	0.46542247	0.25781991	0.12839499	0.06134676	0.04112089
0.18	0.47899972	0.27033519	0.13837488	0.06883148	0.04758385
0.19	0.49223611	0.28261407	0.14836133	0.07655259	0.05437457
0.20	0.50516519	0.29467879	0.15835220	0.08448788	0.06146375
0.25	0.56614563	0.35243165	0.20833595	0.12673502	0.10051579
0.30	0.62284151	0.40716475	0.25833370	0.17200191	0.14382443
0.35	0.67692827	0.46005430	0.30833338	0.21911236	0.18973830
0.40	0.72942308	0.51181837	0.35833334	0.26734830	0.23724357
0.45	0.78094613	0.56289533	0.40833333	0.31627134	0.28572053
0.50	0.83187595	0.61355281	0.45833333	0.36561386	0.33479071
0.55	0.88244361	0.66395420	0.50833333	0.41521247	0.38422306
0.60	0.93279016	0.71419925	0.55833333	0.46496742	0.43387651
0.65	0.98300172	0.76434885	0.60833333	0.51481782	0.48366494
0.70	1.03313089	0.81444018	0.65833333	0.56472648	0.53353578
0.75	1.08320974	0.86449594	0.70833333	0.61467073	0.58345693
0.80	1.13325788	0.91452998	0.75833333	0.66463669	0.63340879
0.85	1.18328727	0.96455076	0.80833333	0.71461591	0.68337940
0.90	1.23330521	1.01456345	0.85833333	0.76460322	0.73336146
0.95	1.28331616	1.06457119	0.90833333	0.81459547	0.78335050
1.00	1.33332285	1.11457592	0.95833333	0.86459074	0.83334381

References

1. Carslaw, H. S. and Jaeger, J. C., Conduction of Heat in Solids, Oxford University Press, 2nd Edition, 1959.
2. Cole, K. D., Beck, J. V., Haji-Sheikh, A., and Litkouhi, B., Heat Conduction Using Green's Functions, 2nd Edition, CRC Press, Taylor & Francis, 2011.
3. de Monte, F., Beck, J. V., and Amos, D. E., Diffusion of thermal disturbances in two-dimensional Cartesian transient heat conduction, Int. J. Heat Mass Transfer, Vol. 51, No. 25-26, pp. 5931-5941, December 2008.
4. Beck, J. V., Wright, N. T., and Haji-Sheikh, A., Transient power variation in surface conditions in heat conduction for plates, Int. J. Heat Mass Transfer, 51 (2008) 2553-2565.
5. de Monte, F., Beck, J. V., Tail for $S_M = \sum_{m=M}^{\infty} \frac{e^{-\beta_m^2 \tilde{t}}}{\beta_m^2}$ and $\sum_{m=M}^{\infty} \frac{e^{-\beta_m^2 \tilde{t}}}{\beta_m^4}$, $\beta_m = m\pi$, Unpublished Notes, December 2011.