The following is the analytical solution for the diffusion in composite spherical geometry: (For more details refer to "Lithium Intercalation in Core-Shell Materials–Theoretical Analysis" )

```
> restart:with(plots):Digits:=15:with(RootFinding):with(IntegrationTools):
>
> #User defined parameters:
```

Description of parameters: beta^2=D2/D1, alpha=R1/R2, kappa=c1*/c2*,delta=current density

```
> constants:=[beta=2,alpha=0.5,gamma=1000,kappa=2,delta=-.25];
```

$$constants := [\beta = 2, \alpha = 0.5, \gamma = 1000, \kappa = 2, \delta = -0.25]$$

Number of eigenvalues to be used (Higher the better!!!)

```
> NN:=20:
>
> #solution:
> param1:=[theta[n]=alpha*lambda[n]*beta,phi[n]=lambda[n]*(alpha-1)];
```

$$param1 := [\theta_n = \alpha\,\lambda_n\,\beta, \phi_n = \lambda_n\,(\alpha - 1)]$$

```
> param2:=[theta[n]=alpha*lambda*beta,phi[n]=lambda*(alpha-1)];
```

$$param2 := [\theta_n = \alpha\,\lambda\,\beta, \phi_n = \lambda\,(\alpha - 1)]$$

Eigenvalue Equation

```
> eigeneqn:=subs(param2,(1/gamma+beta^2*alpha^2*lambda^2+1/gamma*alpha*lambda^2-alpha*kappa+beta^2*alpha)*sin(theta[n])*sin(phi[n])+(-beta^2*alpha^2*lambda+beta^2*alpha*lambda-alpha*lambda*kappa+1/gamma*lambda-1/gamma*lambda*alpha)*cos(phi[n])*sin(theta[n])+(beta*alpha^2*lambda*kappa-1/gamma*beta*lambda*alpha-1/gamma*beta*lambda^3*alpha^2)*sin(phi[n])*cos(theta[n])+(beta*alpha^2*lambda^2*kappa+1/gamma*beta*lambda^2*alpha^2-1/gamma*beta*lambda^2*alpha)*cos(theta[n])*cos(phi[n]));
```

$$eigeneqn := \left(\frac{1}{\gamma} + \beta^2\,\alpha^2\,\lambda^2 + \frac{\alpha\,\lambda^2}{\gamma} - \alpha\,\kappa + \beta^2\,\alpha\right)\sin(\alpha\,\lambda\,\beta)\sin(\lambda\,(\alpha - 1)) + \left(-\beta^2\,\alpha^2\,\lambda + \beta^2\,\alpha\,\lambda - \alpha\,\lambda\,\kappa + \frac{\lambda}{\gamma} - \frac{\lambda\,\alpha}{\gamma}\right)\cos(\lambda\,(\alpha - 1))\sin(\alpha\,\lambda\,\beta) + \left(\beta\,\alpha^2\,\lambda\,\kappa - \frac{\beta\,\lambda\,\alpha}{\gamma} - \frac{\beta\,\lambda^3\,\alpha^2}{\gamma}\right)\sin(\lambda\,(\alpha - 1))\cos(\alpha\,\lambda\,\beta) + \left(\beta\,\alpha^2\,\lambda^2\,\kappa + \frac{\beta\,\lambda^2\,\alpha^2}{\gamma} - \frac{\beta\,\lambda^2\,\alpha}{\gamma}\right)\cos(\alpha\,\lambda\,\beta)\cos(\lambda\,(\alpha - 1))$$

Function to return Eigenequation (necessary for using Nextzero function of Maple) The following eqution is hardcoded (copy pasted from above), this is the only way I can make the nextzero work.

```
> KKK:=lambda->(1/gamma+beta^2*alpha^2*lambda^2+1/gamma*alpha*lambda^2-alpha*kappa+beta^2*alpha)*sin(alpha*lambda*beta)*sin(lambda*(alpha-1))+(-beta^2*alpha^2*lambda+beta^2*alpha*lambda-alpha*lambda*k
```

```
appa+1/gamma*lambda-1/gamma*lambda*alpha)*cos(lambda*(alpha-1))*si
n(alpha*lambda*beta)+(beta*alpha^2*lambda*kappa-1/gamma*beta*lambd
a*alpha-1/gamma*beta*lambda^3*alpha^2)*sin(lambda*(alpha-1))*cos(a
lpha*lambda*beta)+(beta*alpha^2*lambda^2*kappa+1/gamma*beta*lambda
^2*alpha^2-1/gamma*beta*lambda^2*alpha)*cos(alpha*lambda*beta)*cos
(lambda*(alpha-1));
```

$$KKK := \lambda \to \left(\frac{1}{\gamma} + \beta^2 \alpha^2 \lambda^2 + \frac{\alpha \lambda^2}{\gamma} - \alpha \kappa + \beta^2 \alpha\right) \sin(\alpha \lambda \beta) \sin(\lambda (\alpha - 1)) + \left(-\beta^2 \alpha^2 \lambda + \beta^2 \alpha \lambda - \alpha \lambda \kappa + \frac{\lambda}{\gamma} - \frac{\lambda \alpha}{\gamma}\right) \cos(\lambda (\alpha - 1)) \sin(\alpha \lambda \beta) + \left(\beta \alpha^2 \lambda \kappa - \frac{\beta \lambda \alpha}{\gamma} - \frac{\beta \lambda^3 \alpha^2}{\gamma}\right) \sin(\lambda (\alpha - 1)) \cos(\alpha \lambda \beta) + \left(\beta \alpha^2 \lambda^2 \kappa + \frac{\beta \lambda^2 \alpha^2}{\gamma} - \frac{\beta \lambda^2 \alpha}{\gamma}\right) \cos(\alpha \lambda \beta) \cos(\lambda (\alpha - 1))$$

```
> ChiParam:=
[

chi[1] = cos(theta[n])*theta[n]-sin(theta[n]),
chi[2] = phi[n]*cos(phi[n])+(-alpha*lambda[n]^2-1)*sin(phi[n]),
chi[3] = -theta[n]+cos(theta[n])*sin(theta[n]),
chi[4] =
cos(phi[n])*sin(phi[n])*(-lambda[n]^2+1)+2*lambda[n]*cos(phi[n])^2
-phi[n]*lambda[n]^2+(-alpha-1)*lambda[n]

];
```

$$ChiParam := [\chi_1 = \cos(\theta_n)\,\theta_n - \sin(\theta_n),\ \chi_2 = \phi_n \cos(\phi_n) + (-\alpha \lambda_n^2 - 1) \sin(\phi_n),\ \chi_3 = -\theta_n + \cos(\theta_n) \sin(\theta_n),\ \chi_4 = \cos(\phi_n) \sin(\phi_n) (-\lambda_n^2 + 1) + 2 \lambda_n \cos(\phi_n)^2 - \phi_n \lambda_n^2 + (-\alpha - 1) \lambda_n]$$

```
>
> Aneqn:=A[n] =
2*chi[1]*delta*kappa*lambda[n]/(beta^3*chi[2]^2*chi[3]-kappa*chi[1
]^2*chi[4]);
```

$$Aneqn := A_n = \frac{2 \chi_1 \delta \kappa \lambda_n}{\beta^3 \chi_2^2 \chi_3 - \kappa \chi_1^2 \chi_4}$$

```
> OtherParam:=[

k[1] = -3*kappa*delta/(alpha^3*kappa-alpha^3+1),
k[2] = -3*delta/(alpha^3*kappa-alpha^3+1),
```

```
a[1] =
1/10*(-5*alpha^5*beta^2-3*alpha^5+(3*alpha^5*beta^2+5*alpha^5-15*a
lpha^3+10*alpha^2)*kappa+15*alpha^3-15*alpha^2+5*alpha^2*beta^2+3)
*delta*kappa/(alpha^3*kappa-alpha^3+1)^2+1/10*(10*alpha^4-10*alpha
)*delta*kappa/(alpha^3*kappa-alpha^3+1)^2/gamma,

a[2] =
-1/10/(alpha^6*kappa^2+(-2*alpha^6+2*alpha^3)*kappa+alpha^6-2*alph
a^3+1)*(10*kappa^2*alpha^5+((2*beta^2-30)*alpha^5+15*alpha^3)*kapp
a+18*alpha^5-15*alpha^3-3)*delta+1/(alpha^6*kappa^2+(-2*alpha^6+2*
alpha^3)*kappa+alpha^6-2*alpha^3+1)*kappa*alpha^4*delta/gamma
];
```

$$OtherParam := \left[k_1 = -\frac{3\,\kappa\,\delta}{\alpha^3\,\kappa - \alpha^3 + 1},\ k_2 = -\frac{3\,\delta}{\alpha^3\,\kappa - \alpha^3 + 1},\ a_1 = \frac{(-5\,\alpha^5\,\beta^2 - 3\,\alpha^5 + (3\,\alpha^5\,\beta^2 + 5\,\alpha^5 - 15\,\alpha^3 + 10\,\alpha^2)\,\kappa + 15\,\alpha^3 - 15\,\alpha^2 + 5\,\alpha^2\,\beta^2 + 3)\,\delta\,\kappa}{10\,(\alpha^3\,\kappa - \alpha^3 + 1)^2} + \frac{(10\,\alpha^4 - 10\,\alpha)\,\delta\,\kappa}{10\,(\alpha^3\,\kappa - \alpha^3 + 1)^2\,\gamma},\ a_2 = -\frac{(10\,\kappa^2\,\alpha^5 + ((2\,\beta^2 - 30)\,\alpha^5 + 15\,\alpha^3)\,\kappa + 18\,\alpha^5 - 15\,\alpha^3 - 3)\,\delta}{10\,(\alpha^6\,\kappa^2 + (-2\,\alpha^6 + 2\,\alpha^3)\,\kappa + \alpha^6 - 2\,\alpha^3 + 1)} + \frac{\kappa\,\alpha^4\,\delta}{(\alpha^6\,\kappa^2 + (-2\,\alpha^6 + 2\,\alpha^3)\,\kappa + \alpha^6 - 2\,\alpha^3 + 1)\,\gamma}\right]$$

```
> ###################
```

Final Solution

x2 #part1=without infinite sum, part2 term in the infinite sum

```
>
> x1part1:=1/6*k[1]*beta^2*x^2+a[1]+k[1]*t;
```

$$x1part1 := \frac{1}{6}\,k_1\,\beta^2\,x^2 + a_1 + k_1\,t$$

```
> x1part2:=-sin(lambda[n]*beta*x)*chi[2]*A[n]*beta^2/lambda[n]/x*exp
  (-lambda[n]^2*t);
```

$$x1part2 := -\frac{\sin(\lambda_n\,\beta\,x)\,\chi_2\,A_n\,\beta^2\,\mathbf{e}^{(-\lambda_n^2\,t)}}{\lambda_n\,x}$$

```
>
```

x2 #part1=without infinite sum, part2 term in the infinite sum

```
> x2part1:=1/6*k[2]*x^2-(-delta-1/3*k[2])/x+a[2]+k[2]*t;
```

$$x2part1 := \frac{1}{6} k_2 x^2 - \frac{-\delta - \frac{1}{3} k_2}{x} + a_2 + k_2 t$$

```
> x2part2:=-(lambda[n]*cos(x*lambda[n]-lambda[n])+sin(x*lambda[n]-lambda[n]))/lambda[n]/x*chi[1]*A[n]*exp(-lambda[n]^2*t);
```

$$x2part2 := -\frac{(\lambda_n \cos(x \lambda_n - \lambda_n) + \sin(x \lambda_n - \lambda_n)) \chi_1 A_n \mathbf{e}^{(-\lambda_n^2 t)}}{\lambda_n x}$$

```
> ########################
>
```

# Calculation of x1 and x2 expressions using numerical eigenvalues

First Eigenvalue

```
> Lam[1]:=NextZero(subs(param1,constants,lambda[n]=lambda,eval(KKK)),0);
```

$$Lam_1 := 2.93420083104756$$

All eigenvalues till NN

```
> for i from 2 to NN do
Lam[i]:=NextZero(subs(constants,eval(KKK)),Lam[i-1]);od:
```

List of all the eigenvalues

```
> ListLambda:=[seq(lambda[i]=Lam[i],i=1..NN)]:
```

Check if the k1 k2 and a1 and a2 are calculated numerically

```
> evalf(subs(constants,OtherParam)):
```

Computation of Expressions for x1 and x2

```
> expr11:=0:expr22:=0:
for i from 1 to NN do

NumChiparam:=evalf(subs(param1,n=i,ListLambda,constants,ChiParam));
NumAn:=evalf(subs(param1,n=i,ListLambda,constants,NumChiparam,Aeqn));

expr11:=expr11+subs(param1,n=i,NumChiparam,ListLambda,NumAn,OtherParam,constants, x1part2):

expr22:=expr22+subs(param1,n=i,NumChiparam,ListLambda,NumAn,OtherParam,constants, x2part2):
```

```
   od:
> expr1:=expr11+subs(OtherParam,constants,x1part1):
> expr2:=expr22+subs(OtherParam,constants,x2part1):
> param1;
```

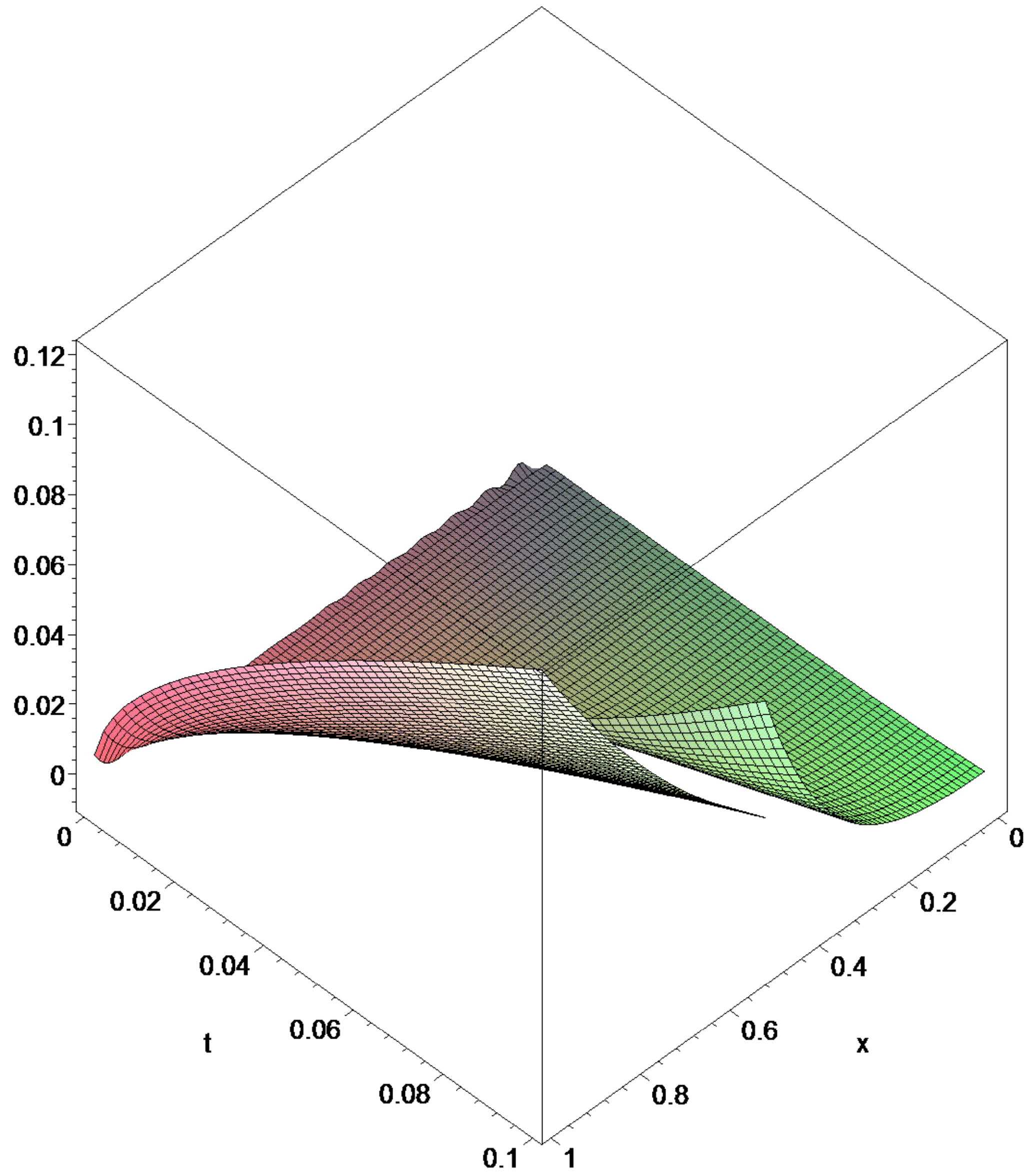
$[\theta_n = \alpha\,\lambda_n\,\beta, \phi_n = \lambda_n\,(\alpha - 1)]$

```
>
>
```

## Plot of curves

```
> p3:=plot(evalf(subs(t=0,expr1)),x=0..eval(alpha,constants)):p4:=plot(evalf(subs(t=0,expr2)),x=eval(alpha,constants)..1):display(p3,p4);

>
> q1:=plot3d(expr1,x=0..subs(constants,alpha),t=1/4000..0.1,axes=boxed):
> q2:=plot3d(expr2,x=subs(constants,alpha)..1,t=1/4000..0.1,axes=boxed):
> with(plots):
> display(q1,q2);
```




>

# Backcheck (Mass Conservation)

Integral of concentration over x int( x*c1(x),x=0..alpha)+int(x c2(x),x=alpha..1)

```
> IntC:=int(x^2*expr1,x=0..eval(alpha,constants))+int(x^2*expr2,x=eval(alpha,constants)..1):
```

Integration of Flux up to that point

```
> IntF:=-subs(constants,delta*t);
```

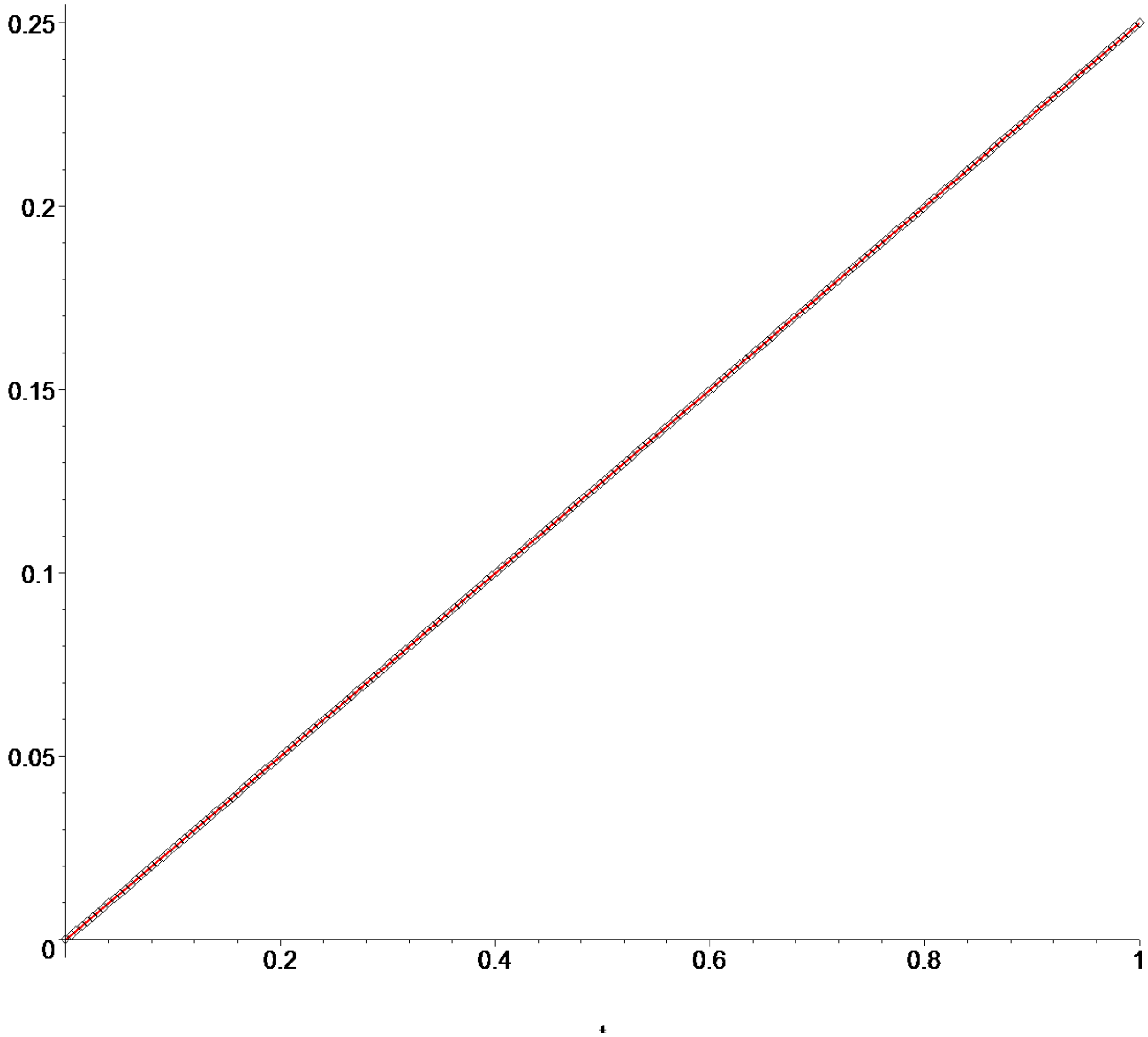
$$IntF := 0.25\ t$$

```
> pp1:=plot(IntC,t=0..1,style=point,color=black,symbolsize=20):
```

```
> pp2:=plot(IntF,t=0..1,thickness=3,color=red):
```

Plot of both integration (Both plots should be identical, if not increase the number of eigenvalues to be used)

```
> display(pp1,pp2);
```




```
>
>
>
>
>
```