Distributed Blood Tissue Models Explained

Mathematically, a distributed model for capillary-tissue exchange is very similar to a compartmental model. A distributed model has a spatial dependence, usually given as the x-coordinate. Distributed models are partial differential equations (PDE's). They model the same processes as compartmental models plus advection and diffusion.
Let *V* be the volume, assumed constant, and *C* be the time and spatially dependent concentration of some material.
The governing equation is

$$d(V\cdot C)/dt = Sources - Sinks + V\cdot Diffusive\,Term.$$

The total derivative,

$$d(V\cdot C)/dt = \partial(V\cdot C)/\partial t + U\cdot\partial(V\cdot C)/\partial x.$$

The $\partial(V\cdot C)/\partial t$ is the local rate of change. The rate of change given by the advection of a gradient is $U\cdot\partial(V\cdot C)/\partial x$, where *U* is the velocity. (See any text on fluid dynamics, e.g. G.K. Batchelor, An Introduction to Fluid Dynamics, Cambridge University Press, 1970, page 74).

The advection term is subtracted from both sides yielding,

$$V\cdot\partial(C)/\partial t = -U\cdot V\cdot\partial C/\partial x + Sources - Sinks + V\cdot\partial(Dp\cdot\partial C/\partial x)/\partial x.$$

The velocity *U* is given by the flow divided by an *area*, and the volume, *V*, in the advective term is given as an *area* multiplying a length, designated *L:*

$$-U\cdot V = (-Flow/Area)\cdot(Area\cdot L) = -Flow\cdot L.$$

This yields

$$V\cdot\partial C/\partial t = -Flow\cdot L\cdot\partial C/\partial x + Sources - Sinks + V\cdot\partial(D_p\,\partial C/\partial x)/\partial x.$$

Assuming D_p is constant, and dividing both sides by *V*, yields

$$\partial C/\partial t = -(Flow\cdot L/V)\cdot\partial C/\partial x + Sources/V - Sinks/V + D_p\cdot\partial^2 C/\partial x^2.$$

The advection of the gradient is similar to the flow term in a compartmental model, *i.e.,*

$$dC/dt = (Flow/V)\cdot(C_{in} - C_{out}) = \frac{-Flow\cdot L}{V}\cdot\frac{(C_{out} - C_{in})}{L}.$$

The source and sink terms are the same for compartment and distributed models, e.g., consumption of material:

$$\partial C/\partial t = \frac{-G}{V}\cdot C;$$

and exchange between two compartments:

$$\partial C_1/\partial t = \frac{-PS_g}{V_1}\cdot(C_1 - C_2)$$

and

$$\partial C_2/\partial t = \frac{-PS_g}{V_2}\cdot(C_2 - C_1) \quad .$$

Putting it all together in a two region distributed model for the plasma (p) and interstitial fluid (isf) regions, yields

$$\frac{\partial C_p}{\partial t} = \frac{-(F_p \cdot L)}{V_p}\cdot\frac{\partial C_p}{\partial x} - \frac{PS_g}{V_p}\cdot(C_p - C_{isf}) - \frac{G_p}{V_p}\cdot C_p + D_p \cdot\frac{\partial^2 C_p}{\partial x^2}$$

and

$$\frac{\partial C_{isf}}{\partial t} = -\frac{PS_g}{V'_{isf}}\cdot(C_{isf} - C_p) - \frac{G_{isf}}{V'_{isf}}\cdot C_{isf} + D_{isf}\cdot\frac{\partial^2 C_{isf}}{\partial x^2}.$$

Another difference between distributed models and compartmental models is the introduction of boundary conditions.

For a model with an inflowing concentration on the left side, (flow is positive going to the right), the total flux boundary condition is

$$\frac{-F_p \cdot L}{V_p}\cdot(C_p - C_{\text{in}}) + D_p \cdot\frac{\partial C_p}{\partial x} = 0.$$

The rest of the boundary conditions are given as no flux conditions (gradient is zero at the boundaries):

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

Example: In JSim's mathematical modeling language (MML) the two region distributed model for plasma and interstitial fluid regions is written as follows:

```
import  nsrunit; unit conversion on;
math btex20_pde {
// INDEPENDENT VARIABLES
realDomain t sec ; t.min=0; t.max=30; t.delta=0.1;
realDomain x cm; real L=0.1 cm, Ngrid=61; x.min=0; x.max=L; x.ct=Ngrid;
private x.min, x.max, x.ct;
// PARAMETERS
real Fp     = 1 ml/(g*min),    // Plasma flow
     Vp     = 0.05 ml/g,       // Plasma volume
     Visfp  = 0.15 ml/g,       // ISF virtual volume
     PSg    = 1 ml/(g*min),    // Plasma-isf exchange coefficient
     Gp     = 0 ml/(g*min),    // plasma consumption coefficient
     Gisf   = 0 ml/(g*min),    // ISF consumption coefficient
     Dp     = 1.0e-5 cm^2/sec, // Plasma axial diffusion coefficient
     Disf   = 1.0e-6 cm^2/sec; // ISF axial diffusion coefficient
extern real Cin(t) mM;         // Plasma inflowing concentration
// VARIABLES
real Cp(t,x)    mM,            // Plasma concentration
     Cisf(t,x)  mM,            // ISF concentration
     Cout_pde(t)    mM;        // Plasma outflowing concentration
//   BOUNDARY CONDITIONS (Note total flux BC for inflowing region.)
when  (x=x.min) { (-Fp*L/Vp)*(Cp-Cin)+Dp*Cp:x = 0; Cisf:x = 0;  }
when  (x=x.max) { Cout_pde = Cp;         Cp:x = 0; Cisf:x = 0; }
//   INITIAL CONDITIONS
when (t=t.min) {                         Cp   = 0; Cisf   = 0; }
//   PARTIAL DIFFERENTIAL EQUATIONS
Cp:t   = -Fp*L/Vp*Cp:x -Gp/Vp*Cp+ Dp*Cp:x:x
        - PSg/Vp*(Cp-Cisf);
Cisf:t =               -Gisf/Visfp*Cisf + Disf*Cisf:x:x
        - PSg/Visfp*(Cisf-Cp) ;
}
```

Comparison with two compartment ODE model:

```
import  nsrunit; unit conversion on;
math comp2_ode {
// INDEPENDENT VARIABLE
realDomain t sec ; t.min=0; t.max=30; t.delta=0.1;
// PARAMETERS
real Fp     = 1 ml/(g*min),    // Plasma flow
     Vp     = 0.05 ml/g,       // Plasma volume
     Visfp  = 0.15 ml/g,       // ISF virtual volume
     PSg    = 1 ml/(g*min),    // Plasma-isf exchange coefficient
     Gp     = 0 ml/(g*min),    // plasma consumption coefficient
     Gisf   = 0 ml/(g*min);    // ISF consumption coefficient
     extern real Cin(t) mM     // Plasma inflowing concentration
// VARIABLES
real Cp(t)    mM,              // Plasma concentration
     Cisf(t)  mM,              // ISF concentration
     Cout_ode(t)    mM;        // Plasma outflowing concentration
//   INITIAL CONDITIONS
when (t=t.min) {                         Cp   = 0; Cisf   = 0; }
//   ORDINARY DIFFERENTIAL EQUATIONS
Cp:t   = (Fp/Vp)*(Cin-Cp) -Gp/Vp*Cp - PSg/Vp*(Cp-Cisf);
Cisf:t =                  -Gisf/Visfp*Cisf -PSg/Visfp*(Cisf-Cp) ;
Cout_ode = Cp;
}
```