

## **Drug Metabolism and Disposition**

Article # DMD-AR-2022-000831

### **Predicting impact of food and feeding time on oral absorption of drugs with a rat continuous intestinal absorption model**

Casey Radice, Ken Korzekwa, and Swati Nagar

Department of Pharmaceutical Sciences, Temple University School of Pharmacy, Philadelphia PA 19140, USA

### **Supplementary Material**

**Table S1.** Measurement of pH in fasted and fed rat intestine by segment

Segment Name	Fasted group				Fed group			
	Rat 34 <sup>3</sup>	Rat 36 <sup>3</sup>	Rat 37 <sup>3</sup>	Rat 38 <sup>3</sup>	Rat 25 <sup>2</sup>	Rat 26 <sup>2</sup>	Rat 33 <sup>3</sup>	Rat 35 <sup>3</sup>
Duodenum	6.6	6.4	6.4	6.3 <sup>1</sup>	6.1	6.0	5.9	6.2
Jejunum 1	6.5	6.4	6.3	6.5 <sup>1</sup>	6.0	6.0	5.7	6.1
Jejunum 2	6.8	6.5	6.5	6.6 <sup>1</sup>	6.3	6.3	6.1	6.2
Jejunum 3	6.5	6.5	6.5	6.6 <sup>1</sup>	6.5	6.7	6.5	6.5
Jejunum 4	6.7	6.6	6.7	6.7	6.7	7.0	6.9	6.9
Jejunum 5	7.1	7.8	7.3	7.4	7.0	7.1	6.7	7.0
Ileum	7.5	7.8	7.7	7.4	7.6	7.8	7.6	7.8
Cecum	6.9	6.9	6.6	6.3	5.8	5.6	5.5	5.8
Proximal Colon	6.7	6.5	6.4	6.1	5.6	5.5	5.4	5.8
Distal Colon	6.2	5.9	6.0	6.4	5.8	6.3	5.8	5.9

<sup>1</sup>(n=1), <sup>2</sup>(n=2), and <sup>3</sup>(n=3), where n is the number of pH readings taken per segment

**Table S2.** Gastrointestinal motility measurements in 12-hour fasted and fed rats with charcoal.

Fasted (n=15)		Fed (n=15)	
Average time post-dose (hr)	Average distance traveled (m)	Average time post-dose (hr)	Average distance traveled (m)
0.17	0.147	0.19	0.019
0.36	0.224	0.34	0.299
0.43	0.631	0.50	0.464
0.78	0.229	0.74	0.636
1.03	0.857	1.03	0.637
1.27	0.870	1.23	0.755
1.69	0.896	1.66	0.899
2.01	1.138	2.05	1.001
3.03	1.235	3.00	1.245

**Table S3.** Calculated gastrointestinal velocity in 12-hour fasted and fed rats.

Fed		Fasted		Average	
Distance (m)	Velocity (m/hr)	Distance (m)	Velocity (m/hr)	Average time (hr)	Velocity (m/hr)
0.241	1.442	0.389	1.854	0.324	1.648
0.55	0.330	0.744	0.382	0.746	0.356
0.819	0.355	0.997	0.285	1.527	0.320
1.12	0.257	1.190	0.073	2.521	0.161

## Mathematica code for AML Fasted-2hr FT group

```
In[222]:= ClearAll["Global`*"];  
In[223]:= Needs["NDSolve`FEM`"]  
In[224]:= cacosf = 1.2;  
  
Drug specific parameters  
  
In[225]:= {drug, f, papp0, fumics, dose1, vol0,  
           dose2, inftime, ph0, pkaa, pkab, ma, mb, acidflag, baseflag} =  
           {"AML-fast 2hr", 0.5, 22.3, 0.5, 1.5, 1., 0.6, 0.01, 7.4, 1., 9.1, 4., 4., 0., 1.};  
  
Drug dose in mg  
  
Fraction unbound in microsomes at 1 mg/mL  
  
CACO2 Papp in m/hr  
  
In[226]:= papp0 = papp0 36  $\times$  10-6.;  
In[227]:= papp0 = papp0 cacosf;  
  
Vol in m3, time in hr  
  
In[228]:= vol0 = vol0 10-6;  
  
Concentration in Dosing solution mg/m3  
In[229]:= C0 = dose1 / vol0;
```

---

### Rat Physiology

#### Radii (m)

duodenum, J1, J2, J3, J4, J5, ileum, proximal colon, and distal colon

circumference (cm) from in-house experimental \*scaleradii = radii (m) [scaleradii to make circumference in cm to radii in meters]

```
In[230]:= scaleradii = 0.01 / (2  $\pi$ )
```

```
Out[230]= 0.00159155
```

---

```
In[231]:= r1 = 0.79 scaleradii;
r2 = 0.92 scaleradii;
r3 = 0.92 scaleradii;
r4 = 0.92 scaleradii;
r5 = 0.92 scaleradii;
r6 = 0.91 scaleradii;
r7 = 0.89 scaleradii;
r8 = 1.17 scaleradii;
r9 = 1.17 scaleradii;
```

---

### Distances (m)

Distances measured in-house in meters. Distance is where the segment ends (cumulative). Ex: jej2 = duo + jej1+ jej2

```
In[240]:= d0 = 0.0005;
duo = 0.1005;
jej1 = 0.3125;
jej2 = 0.5255;
jej3 = 0.7375;
jej4 = 0.9495;
jej5 = 1.1615;
ile = 1.1915;
proxcol = 1.2715;
distcol = 1.4305;
```

---

### Cross sectional area ( $m^2$ )

From radii listed above

```
In[250]:= xa1 = π r1^2;
xa2 = π r2^2;
xa3 = π r3^2;
xa4 = π r4^2;
xa5 = π r5^2;
xa6 = π r6^2;
xa7 = π r7^2;
xa8 = π r8^2;
xa9 = π r9^2;

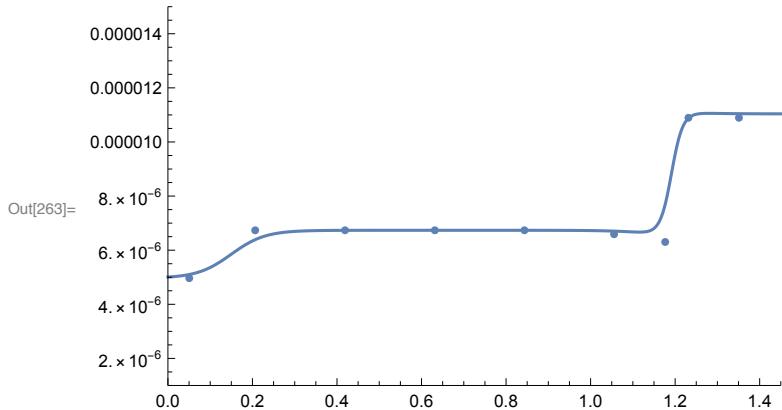
In[259]:= xat =
{{(duo - d0) / 2 + d0, xa1}, {(jej1 - duo) / 2 + duo, xa2}, {((jej2 - jej1) / 2 + jej1, xa3),
{((jej3 - jej2) / 2 + jej2, xa4), {((jej4 - jej3) / 2 + jej3, xa5),
{((jej5 - jej4) / 2 + jej4, xa6), {((ile - jej5) / 2 + jej5, xa7),
{((proxcol - ile) / 2 + ile, xa8), {((distcol - proxcol) / 2 + proxcol, xa9)}};

In[260]:= plot2 = ListPlot[xat];
```

```
In[261]:= xa[x] = xa1 + (xa2 - xa1) / (1 + e-25(x-0.15)) +
  (xa7 - xa6) / (1 + e-20(x-jej5)) + ((xa8 - xa7)) / (1 + e-75(x-ile));
```

```
In[262]:= plot1 =
  Plot[Evaluate[xa[x]], {x, 0, 10}, PlotRange -> {{0, 1.45}, {0.000001, 0.000015}}];
```

```
In[263]:= Show[plot1, plot2]
```




---

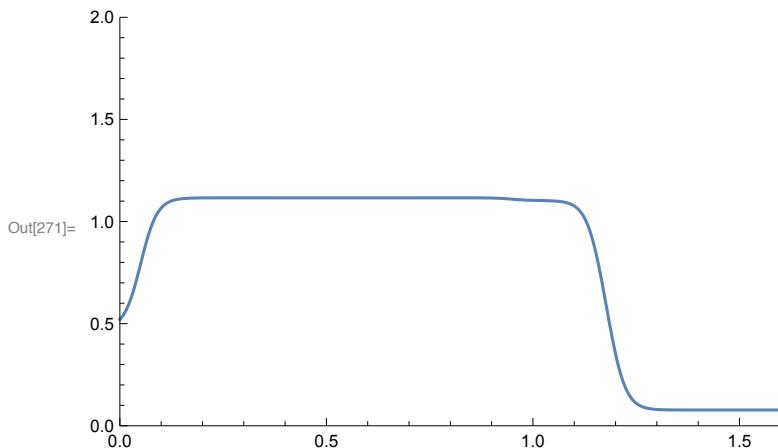
```
In[264]:= dxa[x] = D[xa[x], x];
```

SA per length (or circumference) in m^2/m

```
In[265]:= sa1 = 0.47242;
sa2 = 1.116;
sa3 = 1.103;
sa4 = 0.615;
sa5 = 0.0772;
```

```
In[270]:= sa[x] = sa1 + (sa2 - sa1) / (1 + e-50(x-(d0+duo)/2)) - (sa2 - sa3) / (1 + e-50(x-jej4)) -
  (sa3 - sa4) / (1 + e-50(x-jej5)) - (sa4 - sa5) / (1 + e-50(x-ile));
```

```
In[271]:= Plot[Evaluate[sa[x]], {x, 0, 1.6}, PlotRange -> {{0, 1.6}, {0, 2}}]
```




---

microvilli factor as a function of x

```
In[272]:= mf1 = 9.2;
mf2 = 14.1;
mf3 = 15.7;
mf4 = 6.6;

In[276]:= mf[x] = mf1 + (mf2 - mf1) / (1 + e^-50(x - (d0+duo)/2)) -
(mf2 - mf3) / (1 + e^-50(x - jej^3)) - (mf3 - mf4) / (1 + e^-25(x - ile));

In[277]:= Plot[Evaluate[mf[x]], {x, 0, 1.6}, PlotRange -> {{0, 1.6}, {0, 16}}]
```

Out[277]=

### Cross sectional surface areas of cells ( $m^2$ )

Cell: SA of the small intestine/ colon was multiplied by the diameter of an enterocyte ( $20 \mu m$ ) [ $20 \mu m = 0.000020$  meters]

Membrane: SA of the small intestine/ colon was multiplied by width of the plasma membrane ( $35 \text{\AA}$ ) [ $35 \text{\AA} = 0.000000035$  meters]

Microvilli are only present on apical surface of epithelial cells, so only apical membrane includes villi and microvilli

Lipids: SA of the small intestine/ colon was multiplied by width of cytosolic lipids (assuming they make up 7% of enterocyte volume)[

```
In[278]:= xacell[x] = sa[x] / mf[x] 0.000020;
xamem[x] = sa[x] 0.000000035;
xalip[x] = sa[x] / mf[x] 0.0000015;
```

### Axial Velocity m/hr

Velocity (in-house)

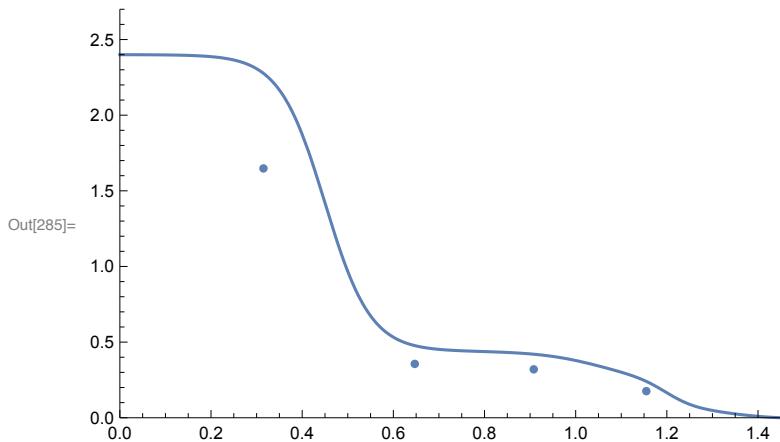
```
In[281]:= veltab = {{0.315, 1.648}, {0.647, 0.356}, {0.908, 0.320}, {1.155, 0.176}};
```

```
In[282]:= plot3 = ListPlot[veltab, PlotRange -> {{0, 1.45}, {0, 2.7}}];
```

```
In[283]:= vel[x] = 2.4 - (2.4 - .440) / (1 + e^-20(x -.45)) -
(0.340 - 0.15) / (1 + e^-15(x - 1.05)) - (0.2) / (1 + e^-30(x - 1.2)) - (0.0575) / (1 + e^-20(x - 1.35));
```

```
In[284]:= plot4 = Plot[Evaluate[vel[x]], {x, 0, 1.45}], PlotRange -> {{0, 1.45}, {0, 3}}];
```

In[285]:= **Show[plot3, plot4]**

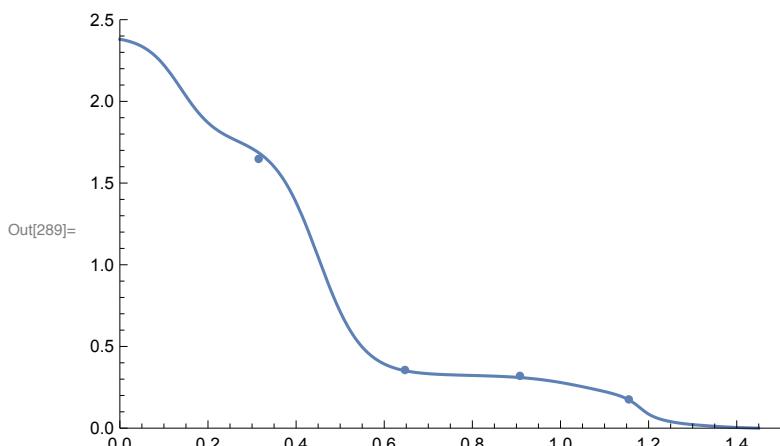


In[286]:= **vel1 = vel[x] /. x → 0;**

In[287]:= **vel2[x] = (vel[x]) (xa1 / xa[x]);**

In[288]:= **plot5 = Plot[Evaluate[vel2[x], {x, 0, 1.45}], PlotRange → {{0, 1.5}, {0, 2.5}}];**

In[289]:= **Show[plot5, plot3]**



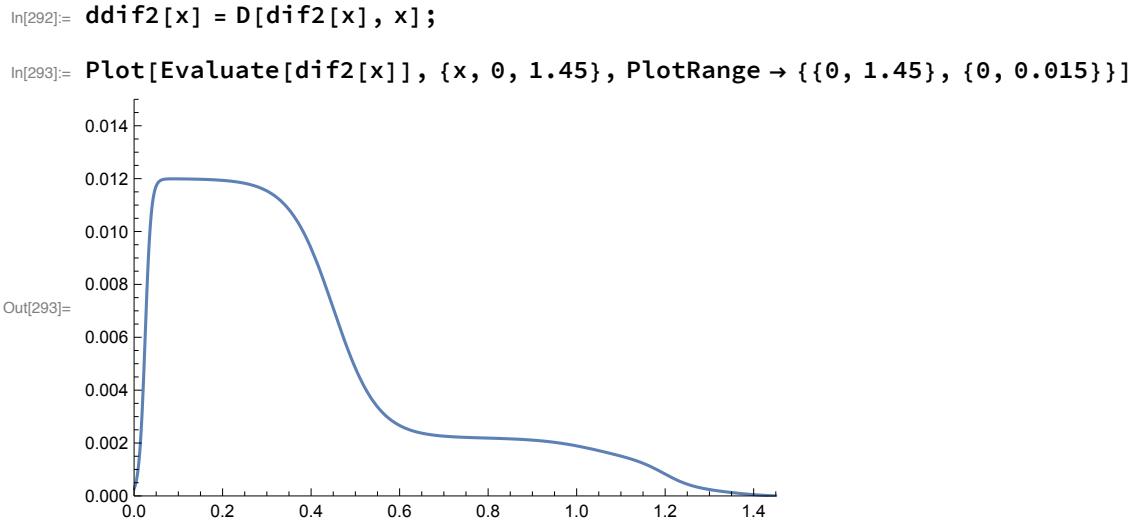
In[290]:= **dvel[x] = D[vel2[x], x];**

---

Axial Diffusion rate constant

effective diffusion in m^2/hr

In[291]:= **dif2[x] = 0.005 × (0.5 + 0.5 Tanh[75. (x - 0.025)]) vel[x];**



### pH Fasted and Fed

The pH was measured for fasted and fed animals in-house.

```
In[294]:= phfasttab = {{((d0 + duo) / 2), 6.5}, {((duo + jej1) / 2), 6.5},
{{(jej1 + jej2) / 2}, 6.6}, {{(jej2 + jej3) / 2}, 6.5},
{{{(jej3 + jej4) / 2}, 6.6}, {{{(jej4 + jej5) / 2}, 7.3}, {{{(jej5 + ile) / 2}, 7.7},
{{{(ile + proxcol) / 2}, 6.5}, {{{(proxcol + distcol) / 2}, 6.1}}};
```

```
In[295]:= plot6 = ListPlot[phfasttab, PlotRange → {{0, 1.45}, {5, 8}}];
```

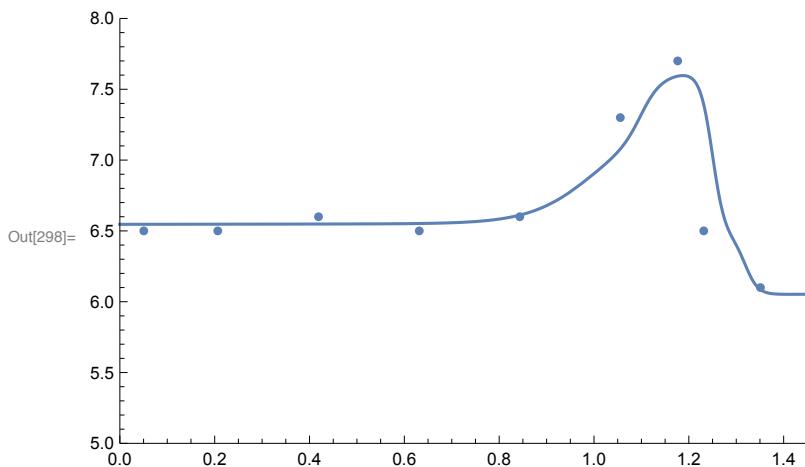
  

```
In[296]:= phfast[x] = phfasttab[[1, 2]] + (phfasttab[[5, 2] - phfasttab[[1, 2]]) / (1 + e^-2(x-jej3)) +
(phfasttab[[6, 2]] - phfasttab[[5, 2]]) / (1 + e^-15(x-1)) +
(phfasttab[[7, 2]] - phfasttab[[6, 2]]) / (1 + e^-50(x-1.1)) +
(phfasttab[[8, 2]] - phfasttab[[7, 2]]) / (1 + e^-75(x-1.25)) +
(phfasttab[[9, 2]] - phfasttab[[8, 2]]) / (1 + e^-75(x-1.32));
```

```
In[297]:= plot7 = Plot[Evaluate[phfast[x]], {x, 0, 1.45}, PlotRange → {{0, 1.45}, {5, 8}}];
```

In[298]:= Show[plot6, plot7]



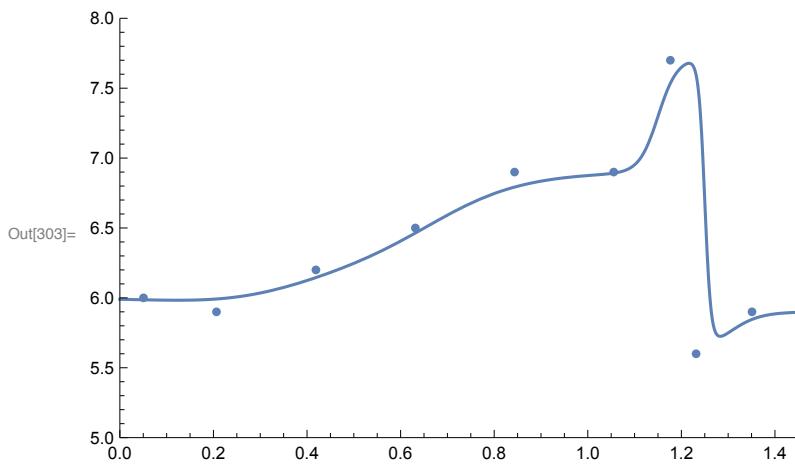
In[299]:= phfedtab = {{((d0 + duo) / 2), 6.0}, {((duo + jej1) / 2), 5.9},  
 {{{(jej1 + jej2) / 2}, 6.2}, {((jej2 + jej3) / 2), 6.5},  
 {{{(jej3 + jej4) / 2}, 6.9}, {((jej4 + jej5) / 2), 6.9}, {((jej5 + ile) / 2), 7.7},  
 {{{(ile + proxcol) / 2}, 5.6}, {((proxcol + distcol) / 2), 5.9}}};

In[300]:= plot8 = ListPlot[phfedtab, PlotRange -> {{0, 1.45}, {5, 8}}];

In[301]:= phfed[x] = phfedtab[[1, 2]] + (phfedtab[[2, 2]] - phfedtab[[1, 2]]) /  $(1 + e^{-10(x-1.15)}) +$   
 $(phfedtab[[3, 2]] - phfedtab[[2, 2]]) / (1 + e^{-10(x-0.37)}) +$   
 $(phfedtab[[5, 2]] - phfedtab[[3, 2]]) / (1 + e^{-10(x-0.67)}) +$   
 $(phfedtab[[7, 2]] - phfedtab[[6, 2]]) / (1 + e^{-50(x-1.15)}) +$   
 $(phfedtab[[8, 2]] - phfedtab[[7, 2]]) / (1 + e^{-150(x-1.25)}) +$   
 $(phfedtab[[9, 2]] - phfedtab[[8, 2]]) / (1 + e^{-30(x-1.3)});$

In[302]:= plot9 = Plot[Evaluate[phfed[x]], {x, 0, 1.45}, PlotRange -> {{0, 1.45}, {5, 8}}];

In[303]:= Show[plot8, plot9]




---

Papp(x) assuming fasted conditions

```
In[304]:= fn[x] = 1 / (1 + acidflag (10^(ma / 10 (phfast[x] - pkaa))) +
  baseflag (10^(mb / 10 (pkab - phfast[x]))));

In[305]:= fn0 = 1 / (1 + acidflag (10^(ma / 10 (ph0 - pkaa))) + baseflag (10^(mb / 10 (pkab - ph0))));

In[306]:= p = papp0 / fn0;

In[307]:= papp[x] = p fn[x];

In[308]:= Plot[Evaluate[papp[x]], {x, 0, 1.45}, PlotRange -> {{0, 1.5}, {0.0002, 0.0012}}]

Out[308]=
```

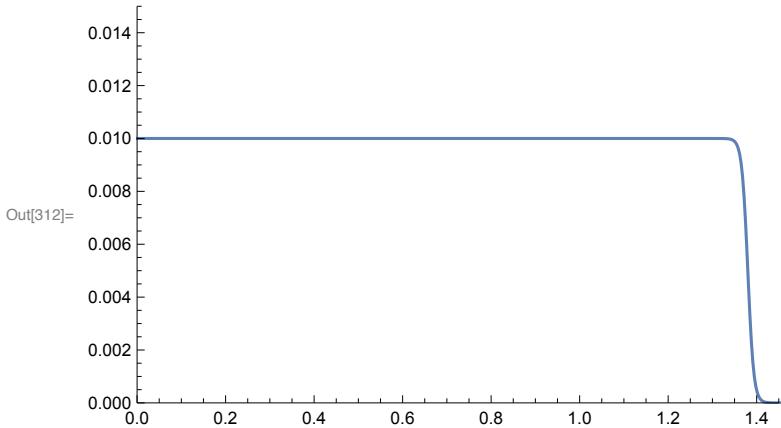
### Radial diffusion rate constant

```
In[309]:= ClearAll[difr1]

In[310]:= difrad = 0.01;

In[311]:= difr1[x] = 0.5 difrad (1 - Tanh[75 (x - 1.38)]);
```

```
In[312]:= Plot[Evaluate[difr1[x]], {x, 0, 10}, PlotRange -> {{0, 1.45}, {0, 0.015}}]
```

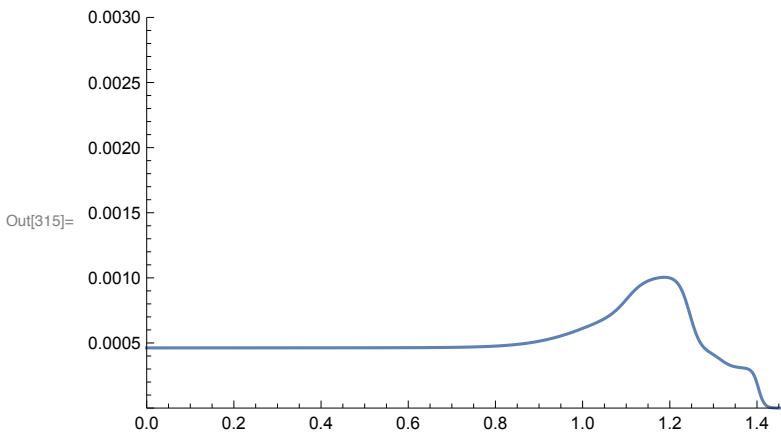


```
In[313]:= Evaluate[difr1[x] /. x -> {1.0, 1.45}]
```

```
Out[313]= {0.01, 2.75357 × 10-7}
```

```
In[314]:= papp2[x] = difr1[x] × papp[x] / (difr1[x] + papp[x]);
```

```
In[315]:= Plot[Evaluate[papp2[x]], {x, 0, 10}, PlotRange -> {{0, 1.45}, {0, 0.003}}]
```



## Drug Parameters

Partition coefficient based on 0.7 uL lipid per 1 mg microsomal protein

```
In[316]:= Kp = ((1 - fumics) / fumics) × (1 / 0.0007);
```

```
In[317]:= cli[x] = 4 papp[x] × sa[x];
clo[x] = cli[x] / Kp;
cli2[x] = 4 papp2[x] × sa[x];
clo2[x] = cli2[x] / Kp;
cli3[x] = 4 papp[x] 100 sa[x];
clo3[x] = cli3[x] / Kp;
```

## pmol/mg

```
In[323]:= pgpt2 = {{0.0505, 2.10}, {0.631, 3.20}, {1.178, 3.67}, {1.311, 2.32}};

In[324]:= pgpplot1 = ListPlot[pgpt2, PlotRange -> {{0, 8}, {0, 1.5}}, Frame -> True, FrameStyle ->
    Directive[Black, 14, Thickness[0.003]], LabelStyle -> (FontFamily -> "Arial"),
    FrameLabel -> {"distance (x), m", "P-gp, pmol/mg tissue"}];

In[325]:= pgp2[x] = 2.08 + (3.25 - 2.1) / (1 + e^-12(x-0.4)) +
    (3.69 - 3.20) / (1 + e^-10(x-0.9)) - (3.7 - 2.25) / (1 + e^-55(x-1.25));

In[326]:= pgpplot2 = Plot[Evaluate[pgp2[x]], {x, 0, 12}, PlotRange -> {{0, 1.5}, {0, 4}}];

In[327]:= Show[pgpplot2, pgpplot1]
```

Out[327]=

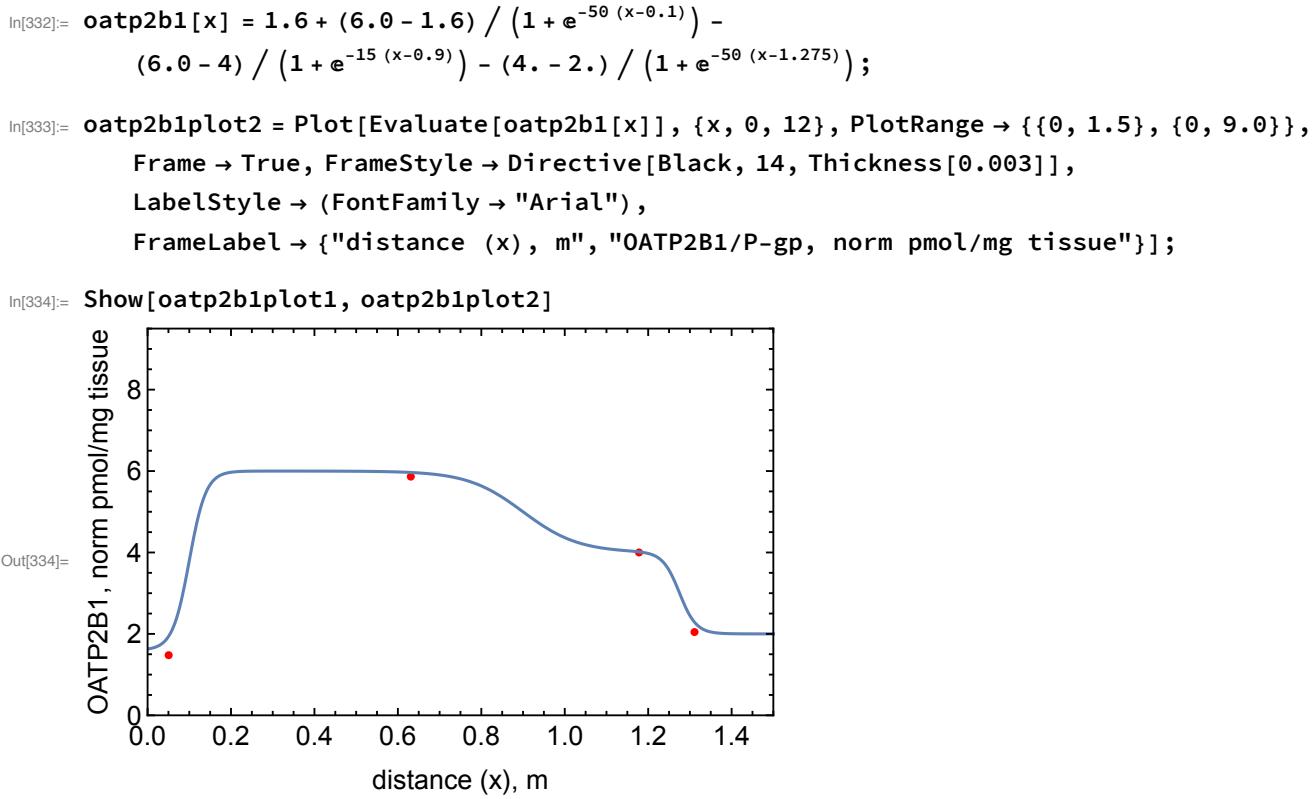
## Rat Oatp2b1 - MacLean 2010 mRNA data in Wistar rats

```
In[328]:= oatp2b1t = {{0.0505, 1.}, {0.631, 3.0}, {1.178, 4.}, {1.311, 9.}};

In[329]:= villinormt = {6.5 / 4.4, 8.6 / 4.4, 1, 1 / 4.4};

In[330]:= oatp2b1normt = Table[{oatp2b1t[[i, 1]], oatp2b1t[[i, 2]] * villinormt[[i]]}, {i, 1, 4}];

In[331]:= oatp2b1plot1 = ListPlot[oatp2b1normt, PlotRange -> {{0, 1.5}, {0, 9.5}}, Frame -> True,
    PlotStyle -> Red, FrameStyle -> Directive[Black, 14, Thickness[0.003]],
    LabelStyle -> (FontFamily -> "Arial"),
    FrameLabel -> {"distance (x), m", "OATP2B1, norm pmol/mg tissue"}];
```



Solution Pulse in stomach w lag

Velocity leaving stomach (m/hr)

```
In[335]:= lag = 0.1;
ft = 2.;
pl0 = 2.5;
pl1 = 0.1;
pl2 = ft;
pl3 = 22;

In[341]:= vel0 = vel[x] /. x → 0;
volcor1 = 0.1 vol0 / (π r1^2 pl1 vel0);
volcor2 = 0.9 pl2 / 2.5 vol0 / (π r1^2 pl2 vel0);
volcor3 = 0.9 × (2.5 - pl2) / 2.5 vol0 / (π r1^2 pl3 vel0);

In[345]:= upulse1[t] = 0.5 (Tanh[300. (t - lag)] - Tanh[50. (t - (pl1 + lag))]);
upulse2[t] = 0.5 (Tanh[300. (t - lag)] - Tanh[50. (t - (pl2 + lag))]);
upulse3[t] = 0.5 (Tanh[50. (t - (lag + pl1))] - Tanh[1. (t - (pl2 + pl3 + lag))]);

In[348]:= pulse1[t] = C0 (volcor1 upulse1[t]);
pulse2[t] = C0 (volcor2 upulse2[t]);
pulse3[t] = C0 (volcor3 upulse3[t]);

In[351]:= c10 = Evaluate[pulse1[t] /. t → 0];
```

```
In[352]:= coordx =  
Join[0. + Range[0, 120] / 800, 0.15 + Range[1, 250] / 1000, 0.40 + Range[1, 200] / 1000,  
0.6 + Range[1, 45.0] / 300.0, 0.75 + Range[1, 350] / 500];  
  
In[353]:= Length[coordx]  
Out[353]= 966
```

---

## P-gp(2) and Oatp(1) parameters

```
In[354]:= km1 = 5000;  
vm1[x] = 0;  
  
In[356]:= km2 = 60 000;  
vm2[x] = 0;
```

---

## PDEs Solution Dose

```
In[358]:= soln = NDSolve[{  
    D[C1[t, x], t] == dif2[x] × D[C1[t, x], {x, 2}] +  
    (-vel2[x] + ddif2[x] + dif2[x] × dxa[x] / xa[x]) × D[C1[t, x], x] +  
    C1[t, x] × (-dvel[x] - (dxa[x] / xa[x]) × vel2[x]) - cli2[x] × C1[t, x] / xa[x] +  
    clo2[x] × C2[t, x] / xa[x] - vm1[x] × C1[t, x] / ((km1 + C1[t, x]) xa[x]) +  
    vm2[x] × C2[t, x] / ((km2 + C2[t, x]) xa[x]),  
    D[C2[t, x], t] == cli2[x] × C1[t, x] / xamem[x] -  
    (clo[x] + clo2[x]) × C2[t, x] / xamem[x] + cli[x] × C3[t, x] / xamem[x] -  
    vm2[x] × C2[t, x] / ((km2 + C2[t, x]) xamem[x]),  
    D[C3[t, x], t] == clo[x] × C2[t, x] / xacell[x] - cli[x] × C3[t, x] / xacell[x] -  
    cli3[x] × C3[t, x] / xacell[x] + clo3[x] × C4[t, x] / xacell[x] - (cli[x] / 2) ×  
    C3[t, x] / xacell[x] + vm1[x] × C1[t, x] / ((km1 + C1[t, x]) xacell[x]),  
    D[C4[t, x], t] == -clo3[x] × C4[t, x] / xalip[x] + cli3[x] × C3[t, x] / xalip[x],  
    {  
        C1[t, 0] == pulse1[t] + pulse2[t] + pulse3[t], C1[0, x] == c10,  
        C1[t, 1.45] == 0, C2[0, x] == 0, C2[t, 0] == 0, C2[t, 1.45] == 0, C3[0, x] == 0,  
        C3[t, 0] == 0, C3[t, 1.45] == 0, C4[t, 0] == 0, C4[0, x] == 0, C4[t, 1.45] == 0},  
    {C1, C2, C3, C4}, {t, 0, 24}, {x, 0, 1.45}, (*MaxStepSize→0.005,*),  
    (*Method→{"FiniteElement","MeshOptions"→{MaxCellMeasure→0.0005}}]*)  
    Method → {"PDEDiscritization" → {"MethodOfLines",  
        (*"DiscretizedMonitorVariables"→True,*)"SpatialDiscritization" →  
        {"TensorProductGrid", "Coordinates" → {coordx}, "DifferenceOrder" → 2}}}]
```

Out[358]=  $\left\{ \begin{array}{l} C1 \rightarrow \text{InterpolatingFunction} \left[ \begin{array}{c} \text{Domain: } \{0., 24.\}, \{0., 1.45\} \\ \text{Output: scalar} \end{array} \right], \\ \text{Data not in notebook. Store now } \end{array} \right\}$

$C2 \rightarrow \text{InterpolatingFunction} \left[ \begin{array}{c} \text{Domain: } \{0., 24.\}, \{0., 1.45\} \\ \text{Output: scalar} \end{array} \right],$

Data not in notebook. Store now

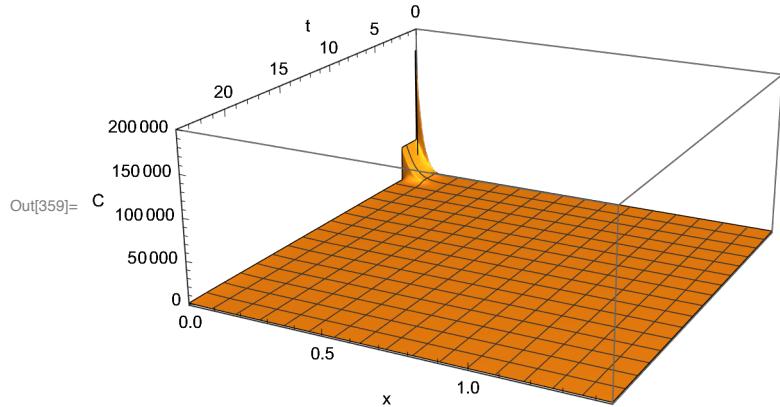
$C3 \rightarrow \text{InterpolatingFunction} \left[ \begin{array}{c} \text{Domain: } \{0., 24.\}, \{0., 1.45\} \\ \text{Output: scalar} \end{array} \right],$

Data not in notebook. Store now

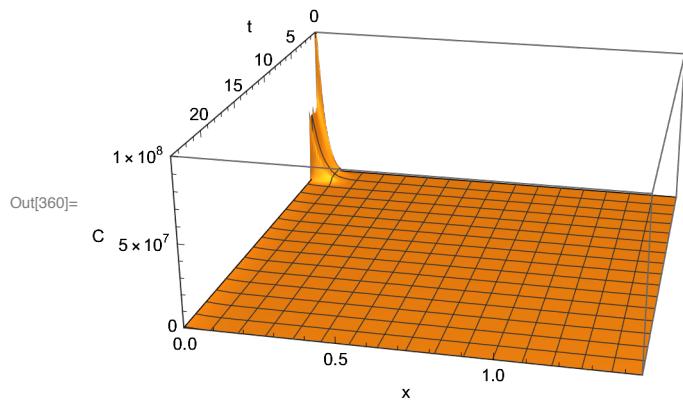
$C4 \rightarrow \text{InterpolatingFunction} \left[ \begin{array}{c} \text{Domain: } \{0., 24.\}, \{0., 1.45\} \\ \text{Output: scalar} \end{array} \right] \}$

Data not in notebook. Store now

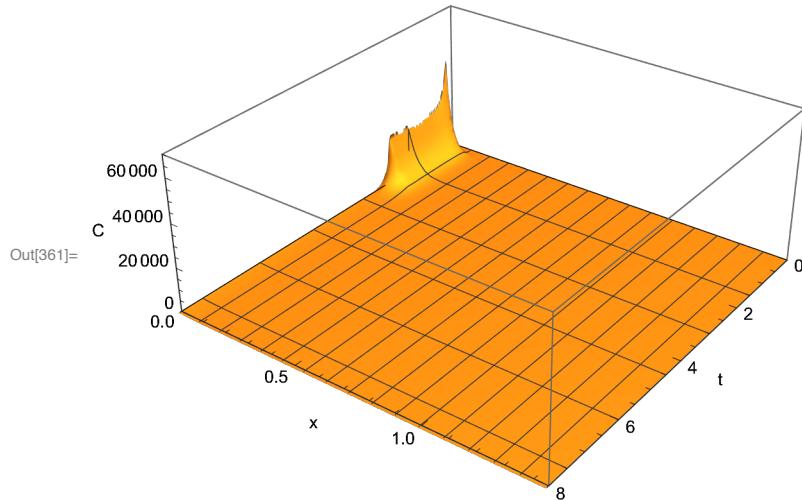
```
In[359]:= plot1 = Plot3D[C1[t, x] /. soln, {t, 0, 24},  
{x, 0, 1.45}, PlotRange -> {{0, 24}, {0, 1.45}, {-2000, 200000}},  
AxesLabel -> {"t", "x", "C"}, MaxRecursion -> 7]
```



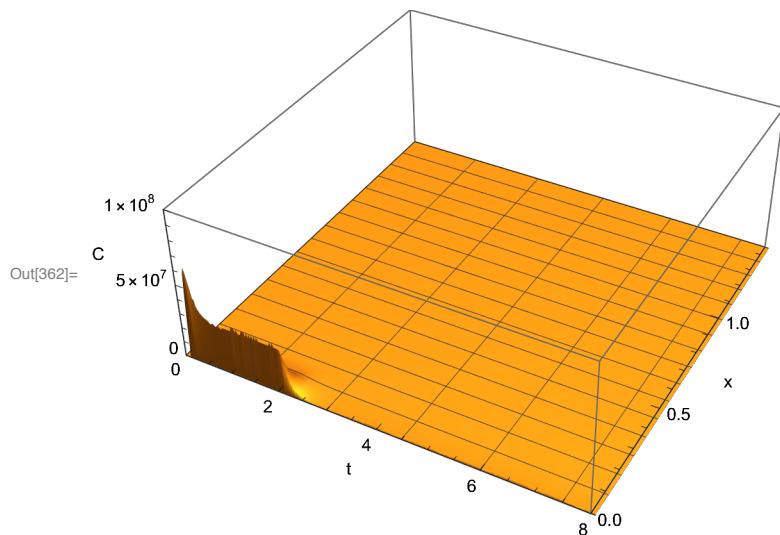
```
In[360]:= plot4 = Plot3D[C2[t, x] /. soln, {t, 0, 24},  
{x, 0, 1.45}, PlotRange -> {{0, 24}, {0, 1.45}, {-1000, 100000000}},  
AxesLabel -> {"t", "x", "C"}, MaxRecursion -> 7]
```



```
In[361]:= plot5 = Plot3D[C3[t, x] /. soln, {t, 0, 24},
{x, 0, 1.45}, PlotRange -> {{0, 8}, {0, 1.45}, {-100, 70000}},
AxesLabel -> {"t", "x", "C"}, MaxRecursion -> 7]
```



```
In[362]:= plot6 = Plot3D[C4[t, x] /. soln, {t, 0, 24},
{x, 0, 1.45}, PlotRange -> {{0, 8}, {0, 1.45}, {-100, 100000000}},
AxesLabel -> {"t", "x", "C"}, MaxRecursion -> 7]
```



```
In[363]:= amtout = NIntegrate[((cli[x]/2) * C3[t, x]) /. soln[[1]], {t, 0, 24},
{x, 0, 1.45}, AccuracyGoal -> 7, Method -> "AdaptiveQuasiMonteCarlo"]
```

Out[363]= 1.49727

```
In[364]:= residC1 = NIntegrate[(C1[24, x] * xa[x]) /. soln[[1]],
{x, 0, 1.45}, AccuracyGoal -> 7, Method -> "AdaptiveQuasiMonteCarlo"]
```

Out[364]= 0.000115234

```
In[365]:= residC2 = NIntegrate[(C2[24, x] xamem[x]) /. soln[[1]],
{x, 0, 1.45}, AccuracyGoal -> 7, Method -> "AdaptiveQuasiMonteCarlo"]
Out[365]= 0.0000587019

In[366]:= residC3 = NIntegrate[(C3[24, x] xacell[x]) /. soln[[1]],
{x, 0, 1.45}, AccuracyGoal -> 7, Method -> "AdaptiveQuasiMonteCarlo"]
Out[366]= 0.0000149563

In[367]:= residC4 = NIntegrate[(C4[24, x] xalip[x]) /. soln[[1]],
{x, 0, 1.45}, AccuracyGoal -> 7, Method -> "AdaptiveQuasiMonteCarlo"]
Out[367]= 0.00160359

In[368]:= total = amtout + residC1 + residC2 + residC3 + residC4
Out[368]= 1.49907

In[369]:= facalc = amtout / total
Out[369]= 0.998804

In[370]:= total / dose1
Out[370]= 0.999378
```

---

```
In[371]:= ClearAll[c5]
C4 is dA/Dt in mg/hr

In[372]:= c5[t_] = NIntegrate[((cli[x] / 2) C3[t, x]) /. soln[[1]],
{x, 0, 1.45}, AccuracyGoal -> 7, Method -> "AdaptiveQuasiMonteCarlo"];

In[373]:= c5t1 = Table[{t, If[c5[t] < 0, 0, c5[t]]}, {t, 0, 2, 0.05}];
c5t2 = Table[{t, If[c5[t] < 0, 0, c5[t]]}, {t, 3, 24, 1.}];
c5t = Join[c5t1, c5t2];

In[376]:= c5fun = Interpolation[c5t, InterpolationOrder -> 2];
```

---

Apply to drug PK model for an IV dose.

```
In[377]:= ClearAll[modelpkiv1, modelpkiv2, modelpk01, modelIV, fitiv, dataIV];
data in hr, ng/mL - in house collection (n=3); dose - 2 mg/kg for a 250 g rat.

In[378]:= dataIV =
{{0.0933, 246.266}, {0.266, 140.66}, {0.511, 93.3}, {1., 70.7}, {2.010, 48.93},
{4., 33.4}, {6.02, 22.33}, {8.016, 14.01}, {10., 7.30}, {12., 4.045}};

In[379]:= nt = Length[dataIV];

In[380]:= colort = {Red, Blue, Orange, Green, Purple, Cyan, Brown};
```

```

In[381]:= k10init = 1;
k12init = 4;
k21init = 1.5;
V1init = 1000.;

dose in ng

In[385]:= doseiv = 1 000 000 dose2 ;

In[386]:= ClearAll[k12, k21, k10, V1, modelpkiv1, fitiv];

In[387]:= modelpkiv1[k12_?NumericQ, k21_?NumericQ,
k10_?NumericQ, V1_?NumericQ, te_?NumericQ] :=
(model1[k12, k21, k10, V1, te] =
(Xc[te] / V1) /. First[NDSolve[{Xc'[t] == -(k12 + k10) Xc[t] + k21 Xp1[t],
Xp1'[t] == k12 Xc[t] - k21 Xp1[t],
Xc[0] == doseiv,
Xp1[0] == 0},
{Xc, Xp1}, {t, 0, 35}, MaxSteps → 10 000, PrecisionGoal → ∞]]))

In[388]:= fitiv = NonlinearModelFit[dataIV, modelpkiv1[k12, k21, k10, V1, te],
{{k12, k12init}, {k21, k21init}, {k10, k10init}, {V1, V1init}},
{te}, PrecisionGoal → ∞, MaxIterations → 10 000, Weights → (1 / #2 &)]

Out[388]= FittedModel[modelpkiv1[3.79715, 1.6217, 0.915164, 1606.08, te]]

In[389]:= fitiv["ParameterTable"]

Out[389]= 

|     | Estimate | Standard Error | t-Statistic | P-Value                  |
|-----|----------|----------------|-------------|--------------------------|
| k12 | 3.79715  | 0.129563       | 29.3075     | $1.04592 \times 10^{-7}$ |
| k21 | 1.6217   | 0.0658335      | 24.6333     | $2.94405 \times 10^{-7}$ |
| k10 | 0.915164 | 0.0389048      | 23.5231     | $3.87289 \times 10^{-7}$ |
| V1  | 1606.08  | 60.5719        | 26.5152     | $1.89951 \times 10^{-7}$ |



In[390]:= fitiv["RSquared"]

Out[390]= 0.998333

In[391]:= fitiv["AICc"]

Out[391]= 68.5051

In[392]:= TableForm[fitiv["CorrelationMatrix"]]

Out[392]/TableForm=

$$\begin{matrix} 1. & -0.656648 & 0.630317 & -0.814385 \\ -0.656648 & 1. & -0.476754 & 0.521453 \\ 0.630317 & -0.476754 & 1. & -0.801226 \\ -0.814385 & 0.521453 & -0.801226 & 1. \end{matrix}$$


In[393]:= k12 = fitiv["BestFitParameters"][[1, 2]];

In[394]:= k21 = fitiv["BestFitParameters"][[2, 2]];

```

```

In[395]:= k10 = fitiv["BestFitParameters"][[3, 2]];
In[396]:= V1 = fitiv["BestFitParameters"][[4, 2]];
In[397]:= fitiv["FitResiduals"]
Out[397]= {-1.56628, 1.63144, 2.8512, 1.04516, -5.02387,
0.179371, 2.02377, 1.52494, -0.398783, -0.6839}

In[398]:= fitiv["PredictedResponse"]
Out[398]= {247.832, 139.029, 90.4488, 69.6548,
53.9539, 33.2206, 20.3062, 12.4851, 7.69878, 4.7289}

In[399]:= ClearAll[modelpkiv2];
modelpkiv2 = First[NDSolve[{Xc'[t] == -(k12 + k10) Xc[t] + k21 Xp1[t],
Xp1'[t] == k12 Xc[t] - k21 Xp1[t],
Xc[0] == doseiv,
Xp1[0] == 0},
{Xc, Xp1}, {t, 0, 35}, MaxSteps → 10000, PrecisionGoal → ∞]];

In[401]:= CL = k10 V1
Out[401]= 1469.82

In[402]:= beta = 1 / 2 ((k12 + k21 + k10) - ((k12 + k21 + k10)^2 - 4 k21 k10)^0.5)
Out[402]= 0.243685

In[403]:= plotpk1 = Plot[(Xc[t] / V1) /. modelpkiv2, {t, 0, 1.1 dataIV[[nt, 1]]},
PlotRange → {{0, 1.1 dataIV[[nt, 1]]}, {0, 500.}}, Frame → True, FrameStyle →
Directive[Black, 14, Thickness[0.003]], LabelStyle → (FontFamily → "Arial"),
FrameLabel → {"Time (hrs)", "AML Concentration (ng/mL)"}];

In[404]:= plotpk2 = ListPlot[dataIV, PlotRange → {{0, 24}, {0, 15.}}];

In[405]:= Show[plotpk1, plotpk2]

```

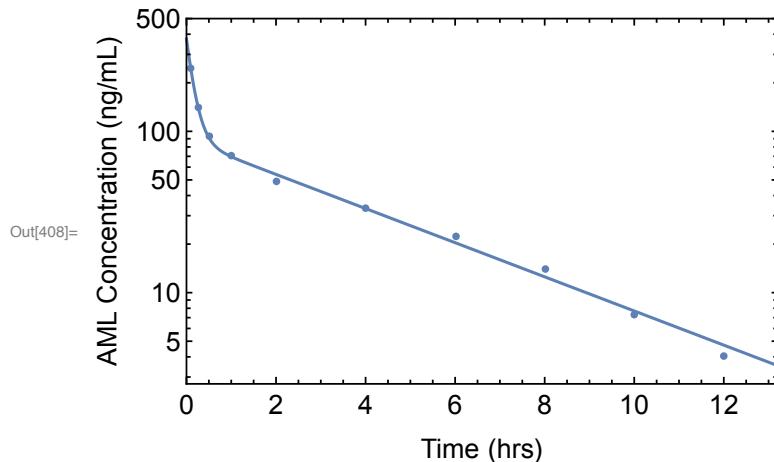
Out[405]=

Time (hrs)	AML Concentration (ng/mL)
0.0	250
0.5	140
1.0	80
2.0	50
4.0	25
6.0	15
8.0	10
10.0	5
12.0	2

```
In[406]:= plotpk3 = LogPlot[(Xc[t] / V1) /. modelpkiv2, {t, 0, 1.1 dataIV[[nt, 1]]},
  PlotRange -> {{0, 1.1 dataIV[[nt, 1]]}, {0, 500}}, Frame -> True, FrameStyle ->
  Directive[Black, 14, Thickness[0.003]], LabelStyle -> (FontFamily -> "Arial"),
  FrameLabel -> {"Time (hrs)", "AML Concentration (ng/mL)"}];

In[407]:= plotpk4 = ListLogPlot[dataIV];

In[408]:= Show[plotpk3, plotpk4]
```



Oral PK inputs (5mg/kg dose-fasted 2hr), and in ng/ml and hr.

```
In[409]:= dataPO = {{0.21, 5.6}, {0.52, 11.8}, {1.01, 17.4}, {1.5, 28.6}, {2.51, 34.9},
  {4.01, 22.4}, {6.02, 16.4}, {8.0, 9.5}, {10., 7.9}, {24.03, 1.1}};

In[410]:= nt = Length[dataPO];
F=Fa*Fg*Fh

In[411]:= fgfh = f / facalc;
Drug VD in mL

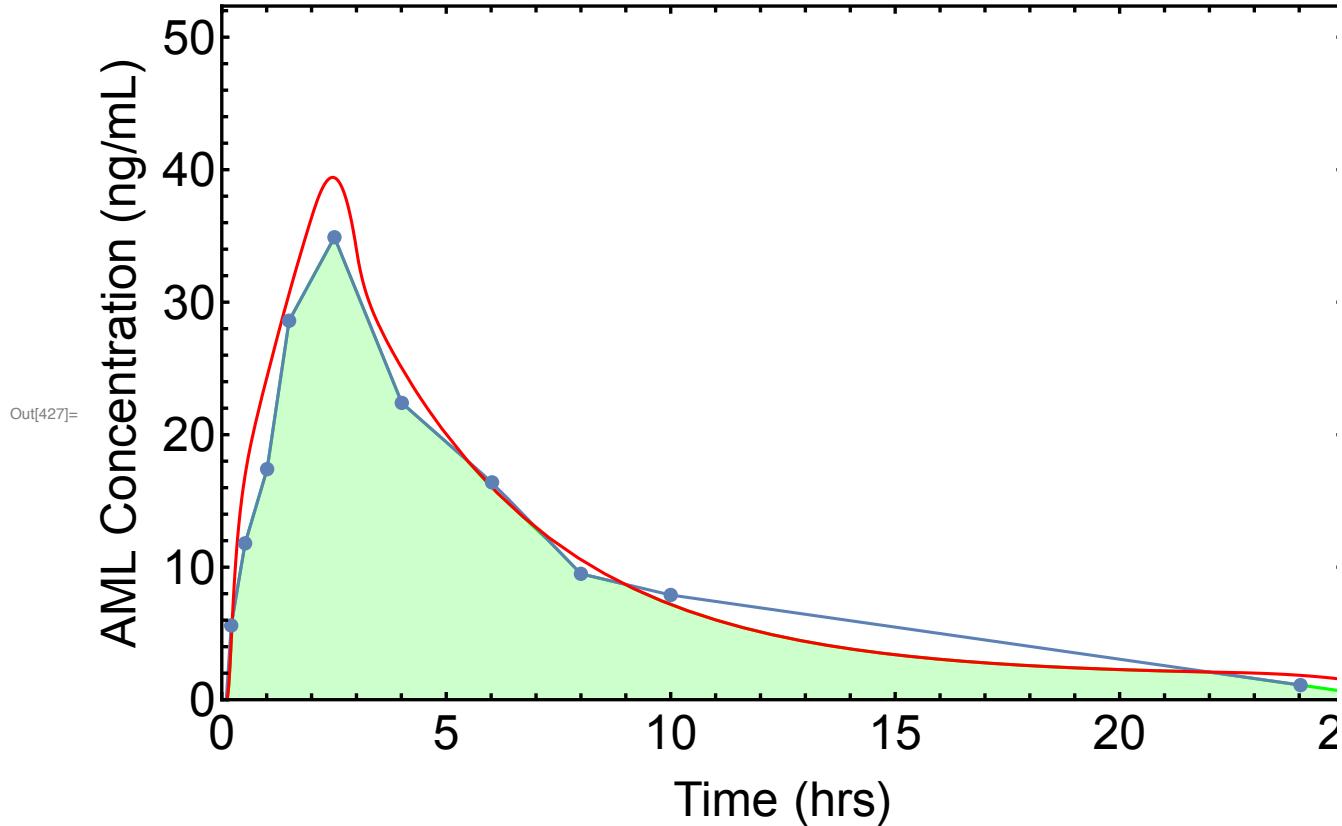
In[412]:= ClearAll[modelpk01];
modelpk01 =
First[NDSolve[{Xc'[t] == 1000000 fgfh c5fun[t] - (k12 + k10) Xc[t] + k21 Xp1[t],
  Xp1'[t] == k12 Xc[t] - k21 Xp1[t],
  Xc[0] == 0,
  Xp1[0] == 0},
  {Xc, Xp1}, {t, 0, 35}, MaxSteps -> 100000, PrecisionGoal -> \[Infinity]]]
```

Out[413]=  $\left\{ Xc \rightarrow \text{InterpolatingFunction}\left[\begin{array}{c} \text{+} \\ \text{L} \end{array} \right. \text{Domain: } \{\{0., 35.\}\} \text{, Output: scalar} \right],$

$Xp1 \rightarrow \text{InterpolatingFunction}\left[\begin{array}{c} \text{+} \\ \text{L} \end{array} \text{Domain: } \{\{0., 35.\}\} \text{, Output: scalar} \right]\}$

```
In[414]:= ClearAll[dataP02, interpP0, plotintP0, aucon];  
In[415]:= npo = Length[dataP0];  
In[416]:= auciv = NIntegrate[(Xc[t] / V1) /. modelpkiv2, {t, 0, 24}];  
In[417]:= dataP02 = Join[{{0, 0}, {Min[0.1, dataP0[[1, 1]] / 2], 0}}, dataP0];  
In[418]:= ymax = Max[Flatten[dataP0][[2 ;; 2 npo ;; 2]]];  
In[419]:= interpP0 = Interpolation[dataP02, InterpolationOrder -> 1, Method -> "Spline"];  
In[420]:= plotintP0 = Plot[Evaluate[interpP0[t]],  
{t, 0, dataP0[[npo, 1]]}, PlotRange -> {{0, dataP0[[npo, 1]]}, {0, 1.5 ymax}}];  
In[421]:= aucintP0n = NIntegrate[interpP0[t], {t, 0, dataP0[[npo, 1]]}];  
In[422]:= aucon = NIntegrate[Xc[t] / V1 /. modelpk01, {t, 0, dataP0[[npo, 1]]}];  
In[423]:= overlap = NIntegrate[Min[interpP0[t], (aucintP0n / aucon) ((Xc[t] / V1) /. modelpk01)],  
{t, 0, dataP0[[npo, 1]]}];  
In[424]:= eoc = overlap / aucintP0n  
Out[424]= 0.924138
```

```
In[425]:= plotoverlap = Plot[Min[interpP0[t], (aucintP0n / aucon) ((Xc[t] / V1) /. modelpk01)], {t, 0, 30}, PlotRange -> {{0, 25}, {0, 1.5 ymax}}, PlotStyle -> Green, Filling -> Axis, Frame -> True, FrameStyle -> Directive[Black, 25, Thickness[0.003]], LabelStyle -> (FontFamily -> "Arial"), FrameLabel -> {"Time (hrs)", "AML Concentration (ng/mL)"}];
plotnormsimP0 = Plot[Evaluate[(aucintP0n / aucon) ((Xc[t] / V1) /. modelpk01)], {t, 0, 30}, PlotRange -> {{0, 25}, {0, 1.5 Max[dataP02]}}, PlotStyle -> Red];
Show[plotoverlap, plotintP0, plotnormsimP0, ListPlot[dataP0]]
```



Calculate experimental oral AUC by extending the oral data to infinity (assume 1st order terminal elimination).

```
In[428]:= auctailo = dataP02[[npo + 1, 2]] / beta;
In[429]:= aucpo = aucintP0n + auctailo;
In[430]:= fcalc = aucpo dose2 / (auciv dose1);
```

Calculate Cmax and Tmax, exp and pred

```
In[431]:= ClearAll[cmax, intt];
In[432]:= intt = Table[{i, interpP0[i]}, {i, 0, dataP02[[npo + 1, 1]], 0.1}];
In[433]:= itt2 = Flatten[intt];
```

```
In[434]:= cmax = Max[itt2[[2 ;; Length[itt2] - 1 ;; 2]]]
Out[434]= 34.8376

In[435]:= tmax = Flatten[Position[intt, cmax]][[1]] / 10 - 0.1
Out[435]= 2.5

In[436]:= predt = Table[{i, Evaluate[(aucintP0n / aucon) ((Xc[i] / V1) /. modelpk01)]},
{i, 0, dataP02[[npo + 1, 1]], 0.1}];

In[437]:= predt2 = Flatten[predt];

In[438]:= cmaxpred = Max[predt2[[2 ;; Length[predt2] - 1 ;; 2]]]
Out[438]= 39.4054

In[439]:= tmaxpred = Flatten[Position[predt, cmaxpred]][[1]] / 10 - 0.1
Out[439]= 2.5
```

---