• Supplementary Material •

Tracing Multi-Scale Mechanisms of Drug Disposition in Normal and Diseased Livers

Sunwoo Park, Sean H.J. Kim, Glen E.P. Ropella, Michael S. Roberts, and C. Anthony Hunt

Validation and comparison of outflow profiles. ISL outflow profiles for four simulated cationic drugs in identically parameterized NORMAL ISLs were validated earlier against in situ referent profiles (Yan et al., 2008a,b,c). Disposition of DILTIAZEMS in the same NORMAL ISLs and in a related, DISEASED CCl4 ISLs were similarly validated using the same SM. The referent livers for the DISEASED CCl4 ISLs came from rats chronically treated with CCl4. The original citation (Hung et al., 2002b) contains histopathology details. For this report, the diltiazem outflow profiles from those same CCl4-treated livers were again matched by DILTIAZEM outflow profiles, but from a somewhat different, DISEASED CCl4 ISL: the match provided by the DISEASED CCl4 ISLs in this report were improved over that reported earlier in (Park et al., 2009) by adjusting eleven rather than nine key parameters. In addition, diltiazem outflow profiles from alcohol–treated livers were matched by DILTIAZEM outflow profiles from new, DISEASED ALC ISLs. The referent livers of the DISEASED ALC ISLs came from rats chronically treated with ethanol (Hung et al., 2002a). For both sets of experiments, SUCROSE was used as an EXTRACELLULAR marker. Outflow profiles from DISEASED CCl4 ISLs and DISEASED ALC ISLs were accepted initially as valid when SM > 0.8. The iterative refinement protocol described in (Park et al., 2009) was used to further improve the parameterization so that SM > 0.9 were achieved.

Similarity Measure. An ISL outflow profile was accepted as valid—as being indistinguishable experimentally from a profile obtained from a repeat wet-lab experiment—when SM > 0.8. Once that was achieved, it was increased to SM > 0.9. ISL outflow profiles were compared with referent profiles using the quantitative SM used previously (Hunt et al., 2006; Yan et al., 2008a,b; Park et al., 2009). It is the fraction of collected COMPOUNDS that lies within a band that was a prespecified, scaled factor of referent outflow values. We used

\[
SM_1(p_s, p_r, s, e, k) = \frac{\sum_{i=s}^{e} C(p'_r(i, k) \leq p_s(i) \leq p'_u(i, k))}{(e - s + 1)}
\]

where \(p_s\): simulated hepatic disposition outflow profile; \(p_r\): in situ hepatic disposition outflow profile; \(s, e \in Z^+\): start and end simulation cycle number; \(m\): sample mean of \(p_r\); \(\gamma_r = (p_r - \mu) / \mu\); \(k \in R^+\): scaling factor of the \(\pm k \cdot \sigma(\gamma_r)\) band; \(p'_r(i, k) = p_r(i)(1 - k \cdot \sigma(\gamma_r))\): lower bound of the band; \(p'_u(i, k) = p_r(i)(1 + k \cdot \sigma(\gamma_r))\): upper bound of the band; \(\sigma(\gamma_r)\): standard deviation of \(\gamma_r\); \(C(\text{cond}) = 1\) if the \(\text{cond}\) is true, otherwise, 0; a counting function \(\{F, T\} \rightarrow \{0, 1\}\); and \(Z\) is a set of integer values, \(R\) is of the real numbers \((0, \infty)\); \(k = 0.5, 0.75, \text{and} 1.0\) were used. Both raw and smoothed ISL profiles were scored.

Drug input and dosage time management. ISL experiments followed the same dosing protocol used in situ (Yan et al., 2008a,b; Park et al., 2009). As illustrated in Fig. 2, a bolus dose of SCUROSE and/or DILTIAZEM was injected into a simulated catheter that feeds into PV. COMPOUNDS were collected as they entered CV, simulating collection by a fraction collector.
Hung et al. used the sum of two inverse Gaussian density functions (requiring five parameters) to simulate compound dilution and dispersion within catheters and perfusion tubing before and after the lobular level (Hung et al., 2002a). They fitted the density functions to averaged outflow profiles obtained following their standard experimental protocol when the liver was replaced by a shunt. They used that dosing function to correct outflow profiles prior to PK analysis. We obtained identically shaped dosing curves using the three-parameter density function, $d(t)$; we parameterized $d(t)$ to provide quantitative control of COMPOUND input into PV and to simulate all influences on diltiazem in situ prior to reaching PV and after exiting CV. Park et al. (2009) proved an example of a parameterized $d(t)$ in their Fig. 2.

$$d(t) = a * \frac{b^c * t^{c-1} * e^{-b*t}}{(c - 1)!}$$

$a$, $b$, and $c$ determine the dose input function’s amplitude, location, and shape; $t$ is time and $e$ is an exponential function. The injection model $D(a,b,c)$ uses $d(t)$. Different values of $a$, $b$, and $c$ can be used when catheter, perfusion and/or collection details change, and when different pathological liver states alter the path from injection to PV or following CV (Park et al., 2009). Different dosage injection models $D(5000,1,2)$ and $D(5000,2,2)$ were used for the validation of NOMRAL and both DISEASED ISLs, respectively. Different injection models were needed because diltiazem was detected earlier in the collected outflow perfusate from both types of diseased relative to normal livers.

ISL profiles were also compared to that of the existing mathematical models – two-phased stochastic liver model (TPSLM). We used Hollenbeck’s (1998) implementation of de Hoog’s (1982) numerical inverse Laplace transform algorithm to compute numerical inverse Laplace transforms of the TPSLM (Hung et al., 2001, 2002a)[3, 15]. TPSLM predicts hepatic disposition of an in situ rat liver using Laplace transform and its inverse form. It uses the following equation to predict hepatic disposition of COMPOUNDS.

$$C_{drug}(t) = \frac{Dose}{Q} L^{-1}\{\hat{f}_{cath}(s) \hat{f}_{y,w}(s)\}$$

$Dose$ is the injected drug bolus and $Q$ is the perfusion rate. $L^{-1}[\cdot]$ denotes the inverse Laplace transform.

$\hat{f}_{cath}(s)$ is the Laplace transform of the transit time density function of U-14C sucrose molecules across the liver.

$$\hat{f}_{cath}(s) = \hat{f}_B(s + k_{in}(1 - \hat{f}_y(s)))$$

$k_{in}$ is the influx rate constant from a sinusoid into to hepatocytes.

$\hat{f}_y(s)$ is the Laplace transform of the transit time density function of the nonpermeating indicator.

$$\hat{f}_B(s) = p \hat{f}_1(s) + (1 - p) \hat{f}_2(s)$$

$\hat{f}_1$ and $\hat{f}_2$ are Laplace transforms of the inverse Gaussian density function with $MT_1$ and $CV_1^2$, and $MT_2$ and $CV_2^2$, respectively. $MT_1$ and $CV_1^2$ are mean and standard deviation of $\hat{f}_1(t)$. $MT_2$ and $CV_2^2$ are mean and standard deviation of $\hat{f}_2(t)$. 

2
\[
\hat{f}_{y,w}(s) = \frac{k_{in,w}/v_{c,w}}{k_{in,w}/v_{c,w} + s}
\]

\( \hat{f}_{y,w}(s) \) is the Laplace transform of the transit time density function of water \( \hat{f}_w(s) \).

\( k_{in,w} \) is permeation constant. \( v_{c,w} \) is the normalized water volume, \( V_y/V_C \).

\[
\hat{f}_1(s) = \exp \left( \frac{1}{CV_1^2} - \left( \frac{MT_1}{CV_1^2/2} \left( s + \frac{1}{2MT_1CV_1^2} \right) \right)^{1/2} \right)
\]

\[
\hat{f}_2(s) = \exp \left( \frac{1}{CV_2^2} - \left( \frac{MT_2}{CV_2^2/2} \left( s + \frac{1}{2MT_2CV_2^2} \right) \right)^{1/2} \right)
\]

**Table 1.** *RefModel* is a two-phase stochastic PK liver model. Listed are the parameters and values used to fit the referent PK data.

<table>
<thead>
<tr>
<th>Category</th>
<th>Parameter</th>
<th>Normal</th>
<th>Diseased(_{\text{CCI4}})</th>
<th>Diseased(_{\text{MC}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed Parameters</td>
<td>( Q )</td>
<td>2( a )</td>
<td>2( a )</td>
<td>2( a )</td>
</tr>
<tr>
<td></td>
<td>( \text{Dose} )</td>
<td>15( a )</td>
<td>15( a )</td>
<td>15( a )</td>
</tr>
<tr>
<td></td>
<td>( k_{in} )</td>
<td>0.35( b )</td>
<td>0.35( b )</td>
<td>0.35( b )</td>
</tr>
<tr>
<td></td>
<td>( k_{eff} )</td>
<td>0.021( b )</td>
<td>0.021( b )</td>
<td>0.021( b )</td>
</tr>
<tr>
<td></td>
<td>( K_S )</td>
<td>16.67</td>
<td>10.59</td>
<td>23.56</td>
</tr>
<tr>
<td></td>
<td>( K_R )</td>
<td>0.95</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td></td>
<td>( CL_{int} )</td>
<td>67.41( c )</td>
<td>45.99( c )</td>
<td>77.25( c )</td>
</tr>
<tr>
<td></td>
<td>( CL_{pt} )</td>
<td>25.35( c )</td>
<td>25.35( c )</td>
<td>25.35( c )</td>
</tr>
<tr>
<td></td>
<td>( PS )</td>
<td>90.54( c )</td>
<td>46.44( c )</td>
<td>83.76( c )</td>
</tr>
<tr>
<td></td>
<td>( V_B )</td>
<td>0.49( d )</td>
<td>0.57( d )</td>
<td>0.53( d )</td>
</tr>
<tr>
<td></td>
<td>( V_C )</td>
<td>1.30( d )</td>
<td>1.34( d )</td>
<td>1.38( d )</td>
</tr>
<tr>
<td>Derived Parameters</td>
<td>( k_{out,w} )</td>
<td>( k_{out,w} )</td>
<td>( v_{c,w} )</td>
<td>( k_{in,w} )</td>
</tr>
<tr>
<td></td>
<td>( k_{in,w} )</td>
<td>( k_{in,w} )</td>
<td>( k_{out,w} )</td>
<td>( k_{in,w} )</td>
</tr>
<tr>
<td></td>
<td>( k_c )</td>
<td>( CL_{int} / V_C )</td>
<td>( CL_{int} / V_C )</td>
<td>( CL_{int} / V_C )</td>
</tr>
<tr>
<td></td>
<td>( v_{c,w} )</td>
<td>5 (( V_C / V_B ))</td>
<td>5 (( V_C / V_B ))</td>
<td>5 (( V_C / V_B ))</td>
</tr>
<tr>
<td></td>
<td>( k_{in,w} )</td>
<td>0.00815 (( CL_{pt} / V_B ))</td>
<td>0.01250 (( CL_{pt} / V_B ))</td>
<td>0.00695 (( CL_{pt} / V_B ))</td>
</tr>
<tr>
<td>Estimated Parameters</td>
<td>( MT_1 )</td>
<td>11.975</td>
<td>10.075</td>
<td>9.875</td>
</tr>
<tr>
<td></td>
<td>( MT_2 )</td>
<td>67.575</td>
<td>108.575</td>
<td>65.575</td>
</tr>
<tr>
<td></td>
<td>( CV_1 )</td>
<td>0.282</td>
<td>0.252</td>
<td>0.252</td>
</tr>
<tr>
<td></td>
<td>( CV_2 )</td>
<td>0.671</td>
<td>0.571</td>
<td>0.771</td>
</tr>
<tr>
<td></td>
<td>( p )</td>
<td>0.675</td>
<td>0.645</td>
<td>0.645</td>
</tr>
</tbody>
</table>

**Units:**  
\( a \) \( \text{ml} \cdot \text{min}^{-1} \)  \( b \) \( \text{sec}^{-1} \)  \( c \) \( \text{ml} \cdot \text{min}^{-1} \cdot \text{g}^{-1} \)  \( d \) \( \text{ml} \cdot \text{g}^{-1} \)

**In Silico Liver Grounding.** In order to develop and begin validating concretized theories about the progression from normal to disease states and how hepatic features interact with compounds, we need the ability to simultaneously explore different regions of plausible mechanism space at different levels of detail, and relate results to wet-lab observations. To facilitate that process it must be easy to change mechanistic details at any level without having to invest significant time in ISL reengineering. We have discovered that the best way to achieve those objectives is to remove metric grounding from ISLs. Hunt et al. (2009) discuss the merits of doing so.
The units, dimensions, and/or objects to which a variable or model constituent refers establish groundings. Inductive ordinary differential equation models are typically grounded to metric spaces. So doing provides simple, interpretive mappings between output and parameter values and referent data. Because phenomena and generators are tightly coupled in such models, the distinction between phenomenon and generator is often small. However, metric grounding creates issues that must be addressed each time one needs to expand the model to include additional phenomena and when combining models to form a larger system. Adding a term to an equation, for example, requires defining its variables and premises to be quantitatively commensurate with everything else in the model. Such expansions can be challenging and even infeasible when knowledge is limited and uncertainty is high, which is the situation that we faced. A model synthesized from components all grounded to the same metric spaces is itself grounded to the Cartesian composite of all those metric spaces. The solution is to remove metric grounding from the ISL and confine it to quantitative feature-to-feature and phenomena-to-phenomena mapping models.

The micromechanisms responsible for generation of hepatic disposition data during a perfused liver experiment do not interact according to any external measurement methods. Nor do they interact directly with the whole rat. They interact with the other components around them. Hepatic cells, for example, interact with each other and their local environment. They are independent of any measures used by an outside observer. From that fact, we inferred that the ISLs must employ similar internal organization, which in modeling terms, means each component is grounded to other components rather than to a metric imposed by an outside observer: they are relationally grounded.

**On Differences Between Traditional PK Models and Synthetic Analogues.** The above observations motivate comment about the differences between traditional, inductive, equation based PK models (left side of Fig. 1) and synthetic, internally grounded analogues like ISLs. In models grounded to metric spaces, parameters serve mostly to shift model behavior within a smooth region of the output metric space. In models grounded to hyperspaces or in those that are relationally grounded, like ISLs, parameters serve that same function. They also serve to discontinuously (even abruptly) shift the behavior of the model into an entirely different region of behavior space: they change the analogue’s dynamic phenotype. In metrically grounded models, the character of the model is bounded, whereas in relational or hyperspace grounded models, model character can change completely with a change in parameters. In the former case, parameters describe one, particular (though abstract) model type. In the latter case, parameters describe families of different yet related models. The DISEASED ISLs are examples. Relational grounding enables flexible, adaptable analogues, but requires a separate analogue-to-referent mapping model.

**Tracing and Computation Granularities.** When the granularity of the tracing is not the same as the very finest grain of the computation, then events below that tracing granularity are invisible. An example is that during a single simulation cycle, it is possible for a COMPOUND to take a path of (1) SS_A → SS_B → SS_C or (2) SS_A → SS_D → SS_A. In the first case, the COMPOUND traverses two SS nodes in a single cycle. This prevents a complete trace. We cannot detect that the COMPOUND visited SS_B. In the second case, the COMPOUND traverses back to its starting location, so potentially we might not get any tracing results for that COMPOUND. Additionally, if a large number of steps (e.g., 10 or 20 steps) had been executed per simulation cycle, we might, in theory, not get any tracing results for some COMPOUNDS because they would have reached the CV or been metabolized within a single simulation cycle. The current ISL solves the above problems by conducting profile validation in the cycle level (coarse-grain) and tracing in the step level (fine-grain). Although the simulation uses a logical concurrency model, which allows us to
consider steps as a number of parallel events in a single cycle, those events are serialized (by interleaving) within the cycle and we can index the trace by position in the sequence.

In such situations, it is tempting to ground the model absolutely to, say, time in seconds. Note that for purposes of describing the traces, we do that. The interval between two adjacent cycles is divided by the number of steps. The quotient of the division is a new fine-grain time resolution for tracing. In the current ISL, time resolution in the cycle level is 0.5 SECONDS and the number of steps per cycle is 2. For these traces, time resolution in the step level then becomes 0.25 SECONDS (0.5/2). But, methodologically, it is important to note that this is a post-simulation analytic technique for which there is no referent counterpart.

**ISL Implementation and Execution.** Parallel executions were performed in different ways to improve performance compared to monotonic parallelism and sequential execution (Ropella et al., 2003). Each parallel mode was associated with one of the six ISL levels illustrated in manuscript Fig. 3 or an experimental requirement. Heterogeneity in parallel execution helped achieve improved performance along with efficient resource management. ISL parallel mode, illustrated in Fig. S1, was supported at group and experiment levels. Group Level Parallel mode enabled executing multiple experiments in parallel by segregating each and allowing each to run concurrently without interaction. Parallel batch processing and analysis of local execution results were performed using that mode. Experiment Level Parallel mode enabled executing single experiments in parallel as separate LOBULE Monte Carlo variants.

**Tools used.** We built the environment using Swarm 2.2 (www.swarm.org), MPICH (www-unix.mcs.anl.gov/mpi/mpich1/) 1.2, GCC (gcc.gnu.org) 4.1.1, OSCAR (oscar.openclustergroup.org) 5.0, and Fedora 5 (www.fedoraproject.org) on a small-scale, in-house, eight-node Beowulf cluster. Simulation and tracing results were analyzed using R (www.r-project.org) 2.7.1 and Matlab (www.mathworks.com) 7.14.

**Generating Raw Event Trace Data.** During the first phase, two types of raw tracing data were generated for SSs and METABOLITES. A trace data file was generated for each SS including PV and CV. It recorded the temporal order of spatiotemporal events experienced by all COMPOUNDS that resided within a particular SS. The collected trace data were grouped into three fields – simulation time, node, and compound. The simulation time field provided an execution step counter value and simulation TIME, using the fine-grain time resolution (0.25 SECONDS) for each traceable event. The graph node field provided SS identifier (ID), SS length in grid spaces, and number of COMPOUNDS. It was used mainly to compute a COMPOUND’S traverse path length.

The number of COMPOUNDS provided the population density of COMPOUNDS within a SS during a simulation cycle. The compound field provided spatiotemporal activities of COMPOUNDS within a SS. It grouped information into compound ID, compound type, layer, spatial location, bound/unbound, and number of compounds. Compound ID denoted the unique identifier of each COMPOUND, either DILTIAZEM or SUCROSE. Layer designated the COMPOUND’S “vertical” location within a SS: within Core or Spaces A, B, or C. Spatial location stated the COMPOUND’S coordinates within a space. The bound/unbound field recorded which component had bound a DILTIAZEM. It was set to -Binder-, -Enzyme-, +ECell+, +Hepatocyte+, or NA to indicate a DILTIAZEM’s location within the SSs in addition to its binding status. The -Binder- and -Enzyme-labels identified the compound as being bound to a BINDER in an ENDOTHELIAL CELL or to an ENZYME in a HEPATOCYTE; when unbound, the COMPOUND was labeled +ECell/+ or +Hepatocyte+. Bound/unbound was declared NA when the DILTIAZEM was located outside CELLS within any space. To trace METABOLIC events, each SS NODE also generated a tracing file that listed the COMPOUND’S ID and type along with the TIME METABOLISM occurred.
Additional Observations On Results

**Validation of Disposition in NORMAL and DISEASED ISLs.** The following is an illustration of iteratively adjusting parameters of the validated NORMAL ISL toward those of DISEASEDALC ISLs. PERFUSATE flow through each SS and the extent of local movement bias were controlled by two parameters: CoreFlowRate and SinusoidTurbo. For simplicity, the value of the former was held constant for all LIVERS. SinusoidTurbo controls COMPOUND movement within EXTRAVASCULAR, EXTRACELLULAR spaces. Changes in compound properties and/or changes in extravascular space properties can influence compound movement within extravascular, extracellular spaces, so SinusoidTurbo was available to change if needed (which it was). We first needed to adjust the probabilistic movements of COMPOUNDS in DISEASEDALC so that the outflow fraction near the peak (< 15 SECONDS) was close to that of a NORMAL outflow profile, but lower after 15 SECONDS. To achieve the first, we tuned A2BJumpProb and B2AJumpProb to values smaller than those of the validated NORMAL ISLs. When using the same dosing function, the shape and height of an outflow profile around its peak were very sensitive to changes in those two parameters. We also tuned BinderPerCell for DISEASEDALC to be smaller than those of NORMAL ISLs. That adjustment also contributed to the placement of the outflow profile’s peak because it determined the population densities of BINDERS and ENZYMES in Spaces B and C. To achieve a lower outflow fraction after the peak, we increased B2CJumpProb but lowered C2BJumpProb relative to the values of NORMAL ISL. Consequently, more COMPOUNDS entered Space C in DISEASEDALC, but they were delayed in reaching CV. An increase in METABOLIC events in Space C was also an important factor contributing to a lowered outflow profile. Other adjustments that enabled achieving SM > 0.9 are diagrammed in manuscript Fig. 4 and listed in Table 1.

Hunt et al. (2006) discuss analyses of ISL parameter changes and their sensitivity along with the fact that the generative consequences of all ISL parameters are networked. Adjusting other parameters can often offset a change in an outflow profile caused by a small change in one parameter. Consequently, studies of sensitivity to individual parameters are less informative and less meaningful than are location changes in LOBULE parameter space. Individually, the parameter changes in manuscript Fig. 4 did not cause statistically distinguishable changes in outflow profiles. Nevertheless, someone experienced in observing different ISL outflow profiles may observe a perceptible change in outflow profile shape. In general, a 5% change in any one parameter will produce an imperceptible change in an outflow profile and no change in SM value. However, a 5% change in all parameters can cause a significant change in outflow profile. The changes in manuscript Fig. 4 averaged 31.9% for DISEASEDCCL4 ISLs and 18.1% for DISEASEDALC ISLs.

**Tracing COMPOUND Resident TIMES and METABOLIC Events.** One COMPOUND within one SS grid space in manuscript Fig. 3 can be viewed as mapping to a wet-lab lower limit of detection. For example, it may be viewed as the limit of detection of referent compound in a biopsy sample that has a volume 1/5,000th that of an average lobule. At that limit, some biopsy samples will test negative for compound, even though we are confident some is present. By analogy, an empty space within the ISL during simulation cycle maps to “no detectable drug.” Even though we can trace the change in location of a specific COMPOUND during execution, there is no mapping to corresponding changes in location for specific molecules. A COMPOUND maps to some number of actual molecules. From one simulation cycle to the next, that number of molecules is unchanged. However, the actual molecules to which a COMPOUND maps is not the same from one simulation cycle to another.

From raw COMPOUND tracing data, any number of derived measures can be obtained, and each enables viewing disposition from different perspectives. Each provides a somewhat different image of events occurring within ISLs during simulations. Some measures may be
useful in helping us think about ISLs (e.g., what change may be needed during parameter tuning to move closer to targeted phenomena). Others may be helpful in thinking about hepatic disposition. Still others may be helpful in thinking about different disease consequences and even disease progression.

**Tracing COMPOUND Path Lengths and Spatiotemporal Binding Patterns.** Below, all results are reported in the order NORMAL, DISEASED\textsubscript{CCl4} and, DISEASED\textsubscript{ALC}, when values for all three are provided, and DISEASED\textsubscript{CCl4} and DISEASED\textsubscript{ALC} when only DISEASED ISL values are provided. There are no wet-lab methods to measure which lobular subspaces (within a particular lobule) are visited by a compound during a single pass through the liver. We recorded each COMPOUND’s SINUSOID traverse path (in grid spaces) for 100 SECONDS after dosing: until it either exited the LOBULE, was METABOLIZED, or the run ended. Path lengths were divided into two types: complete and incomplete. In the above order, the mean percent of the dose that ended at CV was 47, 59, and 38%, whereas 23, 9, and 29% ended at a SS (it was METABOLIZED). Passage was still in progress when the run ended for 30, 32, and 33% of the DILTIAZEM dose. SUCROSE had shorter path lengths because it did not enter CELLS and so was more likely to reach CV before the run terminated. The average path lengths for DILTIAZEM, in the above order, were 64, 59, and 55, whereas for sucrose the averages were 79, 65, and 76. The shorter mean path for DISEASED ISLs shows that both DISEASE types made it easier for COMPOUNDS to move closer to CV as TIME advanced. It is evident from Fig. 6a & b inserts that DISEASED\textsubscript{CCl4} ISLs had a more narrowly distributed variety of path lengths. Note also that the DISEASED\textsubscript{CCl4} ISLs had significantly fewer of the shortest paths (0–25 grid spaces) than either NORMAL or DISEASED\textsubscript{ALC} ISLs.

The data in Fig. 6c & d show that the fraction of COMPOUNDS that was in a LOBULE at a particular TIME and was attached to a BINDER eventually reached a similar steady state ratio of about 0.8 in DISEASED\textsubscript{CCl4} and DISEASED\textsubscript{ALC} LOBULES. However, the relative fractions BOUND in ENDOTHELIAL (Space B) and HEPATOCYTE layers (Space C) were different. In the order presented above, means (and SD) for the fraction bound within all CELLS were 0.60 (0.19), 0.67 (0.20), and 0.62 (0.17). The fraction bound within ENDOTHELIAL CELLS in Space B was 0.53 (0.18), 0.49 (0.16), and 0.59 (0.17); the fraction bound within HEPATOCYTES in Space C was 0.07 (0.02), 0.17 (0.04), and 0.03 (0.01).

Consistent with the parsimony guideline, everything within or around hepatocytes that was capable of binding or sequestering diltiazem was conflated and represented using one INTRACELLULAR BINDER type. Only a small subset of that material includes enzymes that metabolize diltiazem. Because of how events were scheduled, it is possible for a DILTIAZEM to be released toward the end of one simulation step and—by chance—be bound again to another BINDER in that same CELL before being given an opportunity to move out of the CELL.
**Figure S1.** Lifecycle management of ISL experiments. There are eight stages. 1. An ISL experiment is configured with a LOBULE specification file that describes the structural topology of a LOBULE, an ISL parameter file that lists all parameters and their values, and a parameter-sweeping file that specifies a non-linear discrete region of ISL parameter space to be swept. A parameter sweeping space is a collection of ISL parameter files. They are dynamically constructed from ISL parameter and sweeping files by a parameter sweeper. The sweeping space is partitioned depending on the parallel mode selected. 2. The sweeping space is decomposed into a set of partition blocks in the Group Level Parallel mode, which is a coarse-grain parallelism. Parallel batch processing of multiple parameter files is performed in this mode. 3. A set of multiple Monte Carlo runs of a single parameter file is decomposed into a set of partition blocks in the Experiment Level Parallel mode, which is a fine-grain parallelism. 4. Partition blocks are dispatched to a set of computation nodes. 5. A simulator at each computation node runs concurrently using a parameter file or a collection of Monte Carlo runs depending on the parallel mode. 6. A posteriori analysis is conducted over the results produced concurrently by local analysis at each computation node. These two-phase analyses improve overall performance. All analytical results are stored within a shared file system. 7. If parameter sweeping is activated, ISL experiments are continued until all parameter sets in the sweeping space are consumed. 8. Otherwise, the experiment is stopped.
Figure S2. Multiscale COMPOUND tracing within an ISL during execution. There are nine stages. 1. All SS are decomposed into a set of partition blocks. 2. Each block is dispatched to a set of computation nodes. 3. Each computation node loads generated tracing data referenced by its SS identification. 4. The temporal changes within each SS are traced by analyzing the files. 5. The partial traverse path of each COMPOUND is built by reconstructing the changes in terms of SS visited by each COMPOUND. The path is constructed over only those SS in the block. 6. The full COMPOUND traverse path is computed by combining its partial (and local) traverse paths into a single global path. 7. Resident TIME is computed from the traverse path at all levels listed in Fig. 3. 8. ISL COMPONENTS that bind a COMPOUND are traced by analyzing the temporal changes within SS nodes. 9. The process stops once tracing resident TIMES is complete at every computation node.
Figure S3. Fraction of DILTIZEM dose that was BOUND or UNBOUND within different SS spaces in each zone. The selected SSs were the same as those in manuscript Fig. 4. A comparable size SS was selected from Zone 1 (three left panels), Zone 2 (three center panels), and Zone 3 (three right panels) from a NORMAL (three top panels), a DISEASED ALC (three middle panels), and a DISEASED CCl₄ ISL (three bottom panels). In each panel, the dose fraction (regardless of location or state) that is BOUND or UNBOUND in HEPATOCYTES (in Space C) or in ENDOTHELIAL CELLS (in Space B) at indicated times is plotted. The curves are approximate trend lines.

Supplementary Discussion

Relating Differences in ISL Parameter Values to Wet-Lab Measures of Disease. Microsomal protein and cytoskeleton residue are attributes of homogenized tissue samples and have no ISL counterparts. It is noteworthy, however, that B2CJumpProb in Fig. 4 exhibits the identical opposite trend for DISEASED CCl₄ and DISEASED ALC ISLs. Permeability (called the permeability-surface [PS] product by Hung et al. (2002a,b)) is a derived measure of water’s ability to permeate lobular tissue. A2BJumpProb maps well to this measure. Number of fenestra counted fenestrae in comparable tissue sections. Fenestrae influence the ability of all material, especially larger material, to exit blood and access the space of Disse. The pattern of change in A2BJumpProb maps well to number of fenestra. In ISL Space B, an ENDOTHELIAL CELL is the lower limit or spatial resolution. Fenestrae are below that level of resolution and thus have no ISL counterpart. However, grid spaces not assigned to ENDOTHELIAL CELLS map to extracellular spaces, and so a...
subset maps to FENESTRAE. If fenestrae are influential, then one might expect $ECDensity$ to map inversely to number of fenestra (higher $ECDensity \rightarrow$ lower number of fenestra); but $ECDensity$ is lower for DISEASED$_{CCL4}$ ISLs. However, there may well be different ISL parameterizations (mechanistic hypotheses) that also validate, in which the outward $JumpProb$ parameter changes are lessened and $ECDensity$ is changed to compensate. Additional ISL changes could be considered given semi-quantitative wet-lab data against which to validate. For example, a parameter can be added to reduce the fraction of COMPOUND-accessible extracellular spaces in Spaces B and C in DISEASED relative to NORMAL ISLs.

The albumin space measure decreased for both diseased livers. It is a measure of the lobular volume accessible to albumin. We might expect LOBULAR resident TIMES to map somewhat to accessible space: if the spaces are smaller, then resident time should decrease, and it did. We see in Fig. 5(a) that the SUCROSE and DILTIAZEM dose fractions having longer resident times decreased in both DISEASED ISLs. Membrane thicknesses are direct measures of representative hepatocyte membranes in tissues sections. There is no ISL counterpart because the CELL is the limit of resolution. It is not clear if membrane thickness plays any significant role in influencing cellular entry and exit of diltiazem. The reduction in $HepDensity$ (and possibly $ECDensity$, also) maps directly to observed reductions in the number of functional hepatocytes in the CCl$_4$-treated livers.

Collagenization, which is increased in different ways in the two disease models, would be expected to make it harder for a compound to move “into” tissue spaces, and harder to return once there. In the DISEASED ISLs, movement between spaces, $A2BJumpProb$, etc. plus $SinusoidTurbo$ (for DISEASED$_{CCL4}$ ISL), are altered. Their complex interactions influenced resident TIMES (Fig. 5) and path lengths (Fig. 6), which are reflected in the details of locations within and between Spaces observed in Fig. 4.

**Supplementary References**
