Supplementary Information for

"Accurate particle-based simulation of adsorption, desorption, and partial transmission"

Steven S. Andrews

1. Contents

This document
2. Simulation emulator details
3. Irreversible adsorption relationship between κ' and P_a
4. Details for stochastic simulations shown in Figure 5A
5. Reversible adsorption relationships among κ', k', P_a, and P_d
6. Details for stochastic simulations shown in Figure 6
7. Details for stochastic simulations shown in Figure 7

Other files
SurfaceParam.h – C language header file for simulating molecule-surface interactions
SurfaceParam.c – C language code for simulating molecule-surface interactions
SurfaceParam_doc.pdf – documentation for the SurfaceParam library file

2. Simulation emulator details

I used a simulation emulator to derive the relationship between κ' and P_a for irreversible adsorption. Section 4.2 of the main text summarizes the emulator's design and use, the sections titled "Functions for investigating a partially adsorbing surface" of the SurfaceParam.c and SurfaceParam.h files present the actual C language code of the emulator, and the section with the same name of the SurfaceParam_doc.pdf file presents the code documentation. The emulator details are presented here, much of which is redundant with the other sources.

All emulator variables use reduced units, which are shown here with prime symbols. As described in the main text, I ran the emulator with 401 x' values that ranged from -6 to +10; I used an odd number points so that there would not be a tabulated point at $x' = 0$. These x' values were equally spaced with an interval of $\Delta x' = 0.04$. In addition, x' values were included at -0.0001 and +0.0001 to minimize numerical errors that arise from concentration profile discontinuities at $x' = 0$. So that concentration reflection would work accurately, the x' values were symmetric about 0 as much as possible (*i.e.* over the domain from -6 to +6). For the first emulator run, the concentration profile was initialized to 0 for all $x' < 0$ and 1 for all $x' > 0$. For the second run, the emulator initialized the concentration profile to 0 for all tabulated x' values. The emulator iterated Steps 1 and 2 of the simulation scheme many times.

Step 1, which is described mathematically by Eq. 7 of the main text, was carried out by integrating the product of the concentration profile and the Green's function for simple diffusion:

$$C^{(1)}(x) = \int_{-\infty}^{\infty} C(x')\,\mathrm{grn}(x,x')\,dx'$$
$$\mathrm{grn}(x,x') = G_s(x-x')$$
$$G_s(x) = \frac{1}{s\sqrt{2\pi}}\exp\left(-\frac{x^2}{2s^2}\right).$$

The emulator used the trapezoid rule for integration. In Step 1, the concentration profile was not analytically extended for x values outside of the tabulated range because Mathematica cannot integrate the product of an error function and this Green's function. Thus, for $x < -6$, the emulator assumed that $C(x) = 0$, which results in an integrated area of 0, and for $x > 10$, it assumed that $C(x) = 1$. Using this latter assumption, the integral result for the largest x-value concentration profile point was

$$\int_{x_{n-1}}^{\infty} 1\cdot\mathrm{grn}(x,x')\,dx' = \frac{1}{2}\left(1+\mathrm{erf}\,\frac{x-x_{n-1}}{s\sqrt{2}}\right).$$

For Step 2, which is described mathematically by Eqs. 8 and 9 of the main text, the emulator's first task was to integrate the concentration profile over $-\infty < x' < 0$. Where possible, it integrated this using the trapezoid rule. For the initial interval, which extended from $-\infty$ to the first tabulated point, x_0, it extrapolated the concentration profile by assuming that it is an error function (this uses Eq. 2.14 from Crank). The extrapolation function is

$$C(x,\Delta t) = \frac{C_0}{2}\left[1+\mathrm{erf}\,\frac{x}{s\sqrt{2}}\right],$$

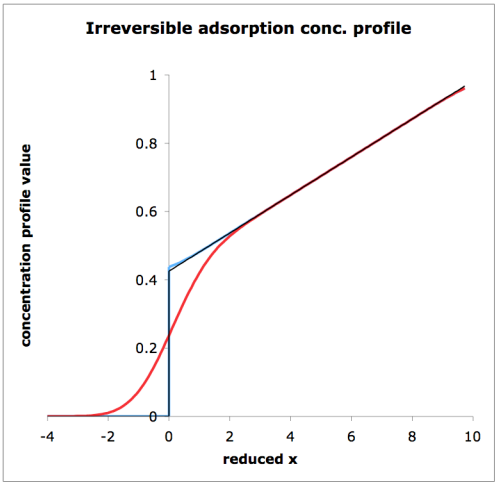
where C_0 is the concentration at $x' = 0$. The emulator used the first tabulated x' value for x and the first tabulated concentration profile value for $C(x,\Delta t)$ to compute C_0. Then, using this C_0 value, it used the integral of this extrapolation function,

$$\int_{-\infty}^{x_0} C(x,\Delta t)\,dx = \frac{C_0}{2}\left[s\sqrt{\frac{2}{\pi}}e^{-\frac{x_0^2}{2s^2}} + x_0\left(1+\mathrm{erf}\,\frac{x_0}{s\sqrt{2}}\right)\right],$$

to estimate the concentration profile area between $-\infty$ and the first tabulated point. Finally, for the final interval of the integral, which extends from $x' = -0.0001$ to 0, the emulator included the upper left triangle of this interval's trapezoid, but not the lower right triangle. Doing this ensured that two sequential adsorption/reflection steps would yield an area of 0.

After the concentration profile was integrated from -∞ to 0, the emulator completed Step 2 by reflecting and adsorbing concentration as appropriate. Adsorption consisted of simply adding P_a times the prior integral to C_a. Reflection consisted of adding $(1\text{-}P_a)C(\text{-}x')$ to $C(x)$ for each tabulated x' point, for x' between -6 and 0, as in Eq. 8, and then setting the concentration profile values for these negative x' values to 0. Because the tabulated x' points were symmetric about 0, no concentration was lost or gained in this reflection process.

The emulator iterated Steps 1 and 2 until ΔC_a, which is the amount of concentration that was adsorbed to the surface during one time step, changed by less than 0.01% between sequential iterations. Typical concentration profiles at this near-steady-state final result are shown in the following figure (which show the same data as Figure 4A of the main paper):



This figure was calculated for $P_a = 0.15$. The black line represents the theoretical concentration profile from Eq. 15 of the main text, the red line represents the emulator concentration profile after Step 1, and the blue line represents the emulator concentration profile after Step 2. Clearly, the simulated steady-state concentration profile after a complete sequence of steps is extremely close to the theoretical result. This similarity holds for all P_a values.

Once it finished iterating, the emulator fit a straight line, using a simple least squares metric, to the concentration profile between $x' = 3$ and 7 and it used Eq. 19 of the main text to find κ' from this fit. Finally, the emulator repeated this entire procedure, but with the concentration profile initialized to 0 for all tabulated x' values. It then calculated and reported the arithmetic mean of the two κ' results. The emulator calculated reduced adsorption coefficients for P_a values from 0 to 1 in steps of 0.05, for which the results are shown in Figure 5A and below.

3. Irreversible adsorption relationship between κ' and P_a

The following data were computed with the "simulation emulator," which is described above and in the main text. The complete code for the emulator is included in the SurfaceParam.c source code file.

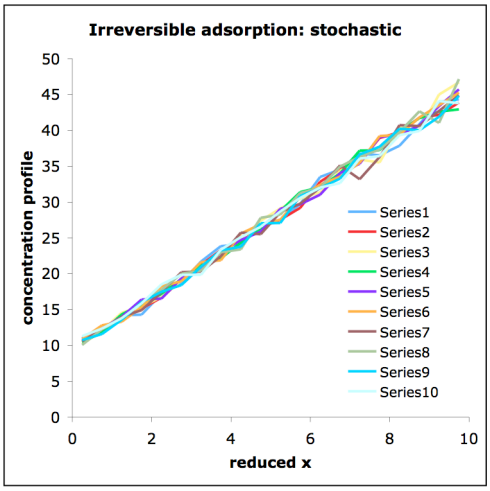
P_a	κ'
0	0
0.05	0.0205318
0.1	0.0422852
0.15	0.0653621
0.2	0.089876
0.25	0.115953
0.3	0.143735
0.35	0.173380
0.4	0.205067
0.45	0.238998
0.5	0.275401
0.55	0.314537
0.6	0.356702
0.65	0.402241
0.7	0.451546
0.75	0.505076
0.8	0.563366
0.85	0.627042
0.9	0.696848
0.95	0.773667
1	0.858559

4. Details for stochastic simulations shown in Figure 5A

I performed the stochastic simulations shown in Figure 5A of the main paper using the Smoldyn program with the configuration file shown below. This file defines a system that is similar to the one that the simulation emulator investigated, although with a few changes. Here, the system is 2-dimensional, the *x*-axis boundaries extend from -1 to 20, and the *y*-axis extends from 0 to 10. It includes 2000 molecules. These molecules adsorb from the region $x > 0$ to a surface at $x = 0$ with adsorption coefficient `KAPPA` (set to 0.7 in the file shown below, but varied from 0 to 0.858558), and then desorb from this surface to the region with $x < 0$ with rate 1. This desorption has no affect on the molecular concentration on the $x > 0$ side of the surface and so adsorption is still effectively irreversible. Molecules that desorb to the $x < 0$ side of the surface then "jump" back to $x \sim 20$ using periodic boundaries. This replenishes the supply of molecules that can adsorb to the surface.

This file instructs Smoldyn to output the concentration profile as a histogram that has 20 bins in the *x* direction, from $x = 0$ to 10, so that each bin has an *x*-axis width of 0.5; each bin also extends from 0 to 10 on the *y*-axis. Because a single snapshot of these data would show significant stochastic noise, Smoldyn averages together 50 sequential histograms, each one taken 100 time steps after the previous one, before it outputs the results. Thus, effectively, each histogram represents $2000 \times 50 = 100{,}000$ molecules. Smoldyn outputs these 10 of these time-averaged histograms every 5000 time steps, from time 9900 to time 54,900. Data graphing, for which a typical example is shown below

($\kappa' = 0.2$ and $P_a = 0.3922$), showed that these histograms were indistinguishable from each other and thus represented systems that were very close to steady state.



Each time-averaged histogram was analyzed using Eq. 19 from the main text to find the calculated adsorption coefficient. These 10 results were then averaged to produce the data shown in Figure 5A. Error bars in Figure 5A show 1 standard deviation. Thus, in effect, each point in Figure 5A, represents 10 trials of 100,000 molecules each.

Configuration file for Figure 5A

```
# File to test molecule sticking
rate

define KAPPA 0.7

graphics opengl

dim 2
species A
max_mol 2500

difc all(solution) 0.5
color A(solution) 1 0 0
color A(up) 0 1 0

time_start 0
time_stop 55000
time_step 1

boundaries 0 -1 20
boundaries 1 0 10

max_surface 3

start_surface walls
action both all reflect
color both 0 0 0
max_panels rect 2
panel rect +1 -1 0 21
panel rect -1 -1 10 21
end_surface

start_surface stick
rate A fsoln up KAPPA
rate A up bsoln 1
color both 1 0.7 0
max_panels rect 1
panel rect +0 0 0 10
end_surface

start_surface jump
action front all jump
action back all reflect
color both 1 0.7 1
max_panels rect 2
panel rect +0 -1 0 10 r0
panel rect -0 20 0 10 r1
jump r0 front <-> r1 front
end_surface

mol 2000 A u u

output_files stdout
cmd i 9000 55000 5000 echo stdout "KAPPA "
cmd i 5000 55000 100 molcountspace
      A(solution) 0 0 10 20 0 10 50 stdout

end_file
```

5. Reversible adsorption relationships among κ', k', P_a, and P_d

The following table shows P_a as a function of κ' and k' for reversible adsorption. I calculated these values using Mathematica and Eq. 37 from the main text. To calculate P_d, combine the P_a result from below and Eq. 32 from the main text. The same data, but with more decimal places, are listed in the SurfaceParam.c source code file, in the function called `lookuprevads` and the array labeled `adsprobtable`. That function scales the κ' and k' values with the equation $\kappa' = x/(1-x)$, and the same for k', so that the tabulated data adequately cover the necessary parameter space.

		κ'																				
		0.01	**0.063**	**0.121**	**0.186**	**0.259**	**0.342**	**0.437**	**0.546**	**0.672**	**0.821**	**1**	**1.217**	**1.488**	**1.833**	**2.289**	**2.922**	**3.854**	**5.369**	**8.259**	**15.95**	**99**
	0.01	0.025	0.024	0.024	0.023	0.022	0.021	0.02	0.019	0.018	0.017	0.016	0.015	0.013	0.012	0.01	0.008	0.006	0.005	0.003	0.002	0
	0.063	0.147	0.143	0.139	0.135	0.13	0.126	0.12	0.115	0.109	0.102	0.095	0.087	0.079	0.069	0.06	0.05	0.039	0.029	0.019	0.01	0.002
	0.121	0.267	0.261	0.254	0.247	0.239	0.23	0.221	0.211	0.2	0.188	0.175	0.161	0.146	0.13	0.112	0.094	0.074	0.055	0.036	0.019	0.003
	0.186	0.387	0.378	0.368	0.358	0.347	0.335	0.322	0.308	0.293	0.276	0.258	0.238	0.216	0.193	0.168	0.141	0.112	0.083	0.055	0.029	0.005
	0.259	0.504	0.493	0.481	0.468	0.455	0.439	0.423	0.405	0.386	0.365	0.342	0.317	0.289	0.259	0.226	0.191	0.153	0.115	0.076	0.04	0.007
	0.342	0.621	0.607	0.593	0.578	0.562	0.544	0.525	0.504	0.481	0.456	0.428	0.398	0.364	0.328	0.288	0.244	0.197	0.149	0.1	0.053	0.009
	0.437	0.735	0.72	0.704	0.687	0.668	0.648	0.626	0.603	0.576	0.548	0.516	0.481	0.442	0.4	0.353	0.301	0.245	0.186	0.126	0.067	0.011
	0.546	0.846	0.83	0.813	0.794	0.774	0.752	0.728	0.702	0.673	0.641	0.606	0.567	0.524	0.475	0.422	0.363	0.298	0.228	0.155	0.083	0.014
	0.672	0.955	0.938	0.92	0.9	0.879	0.855	0.829	0.801	0.77	0.736	0.698	0.655	0.608	0.555	0.495	0.429	0.355	0.274	0.189	0.102	0.017
k'	**0.821**	1.062	1.044	1.025	1.004	0.982	0.957	0.93	0.901	0.868	0.832	0.792	0.746	0.696	0.638	0.573	0.5	0.418	0.326	0.227	0.123	0.021
	1	1.165	1.147	1.127	1.107	1.084	1.059	1.031	1.001	0.967	0.93	0.888	0.84	0.787	0.726	0.657	0.578	0.487	0.384	0.27	0.149	0.025
	1.217	1.264	1.247	1.227	1.207	1.184	1.159	1.131	1.1	1.066	1.028	0.986	0.937	0.882	0.819	0.746	0.662	0.563	0.45	0.321	0.179	0.031
	1.488	1.36	1.343	1.325	1.304	1.282	1.257	1.23	1.2	1.166	1.129	1.086	1.037	0.981	0.917	0.842	0.753	0.649	0.525	0.38	0.215	0.037
	1.833	1.453	1.436	1.419	1.399	1.378	1.354	1.328	1.299	1.267	1.23	1.188	1.14	1.085	1.02	0.944	0.853	0.744	0.611	0.451	0.26	0.046
	2.289	1.541	1.526	1.51	1.492	1.472	1.45	1.426	1.398	1.368	1.333	1.293	1.247	1.194	1.13	1.055	0.964	0.851	0.712	0.536	0.317	0.057
	2.922	1.625	1.612	1.597	1.581	1.563	1.544	1.522	1.497	1.469	1.437	1.4	1.357	1.307	1.247	1.175	1.085	0.973	0.829	0.64	0.39	0.072
	3.854	1.706	1.694	1.682	1.668	1.653	1.636	1.616	1.595	1.57	1.542	1.51	1.472	1.426	1.372	1.305	1.221	1.112	0.968	0.77	0.489	0.095
	5.369	1.782	1.773	1.763	1.752	1.739	1.725	1.71	1.692	1.672	1.649	1.622	1.59	1.551	1.505	1.446	1.372	1.273	1.136	0.937	0.628	0.13
	8.259	1.854	1.848	1.84	1.832	1.823	1.813	1.802	1.789	1.774	1.757	1.737	1.713	1.683	1.647	1.601	1.541	1.459	1.341	1.157	0.836	0.195
	15.95	1.923	1.919	1.915	1.91	1.905	1.899	1.893	1.885	1.877	1.866	1.854	1.84	1.822	1.8	1.771	1.733	1.679	1.597	1.458	1.176	0.353
	99	1.987	1.987	1.986	1.985	1.984	1.983	1.982	1.981	1.979	1.977	1.975	1.972	1.969	1.965	1.959	1.952	1.94	1.922	1.889	1.804	1.192

6. Details for stochastic simulations shown in Figure 6

I used the Smoldyn program and the configuration file listed below to calculate the simulation data shown in both panels of Figure 6. The important aspects of this configuration file are listed in the Figure caption, while other details are presented here.

The system is 3-dimensional, with x extending from -2 to 2 μm, y extending from 0 to 1 μm, and z extending from 0 to 1 μm. The outer bounds of this system are inert, so they reflect all molecules. There is also an active surface at $x = 0$. Molecules named A and A2 adsorb to this surface irreversibly with fast and slow rates, respectively, B and B2 adsorb reversibly with fast and slow rates, and C and C2 transmit reversibly with fast and slow rates. The simulation starts with 20,000 of each of these species, all of which are uniformly distributed throughout the region of the system with $x > 0$. All molecules diffuse with a 5 $\mu m^2/s$ diffusion coefficient. The simulation uses time steps of 0.001 s and runs for a total time of 0.1 s.

At every time step, Smoldyn saves numerical data for Figure 6A by counting the numbers of molecules of each species that are bound to the surface, as well as the numbers of molecules of each species in the region of the system with $x < 0$. Smoldyn also saves histograms of type A molecules at several time points, which are shown in Figure 6B. The histograms are not time-averaged (see above), but are single snapshots. Each histogram has 20 bins that are arrayed along the x-axis from $x = 0$ to 1 μm. These bins also extend from $y = 0$ to 1 μm and from $z = 0$ to 1 μm.

Configuration file for Figure 6

```
# File to test molecule sticking rate
# Figures 6A and 6B of adsorb paper
# units microns, seconds

define XLO -2
define XHI 2
define YLO 0
define YHI 1
define ZLO 0
define ZHI 1
define XSIZE 4
define YSIZE 1
define ZSIZE 1

graphics opengl
graphic_iter 10

dim 3
names A B C A2 B2 C2
max_mol 122000

difc all(solution) 5
# in 10 s, with D=5, diff. dist. is 10
      microns

color A(all) 1 0 0
color A2(all) 1 0 0
color B(all) 0 1 0
color B2(all) 0 1 0
color C(all) 0 0 1
color C2(all) 0 0 1
display_size all 2

time_start 0
time_stop 0.1
time_step 0.001
# with D=5 and Δt=0.001, rms step
      length is 0.1 micron

boundaries 0 XLO XHI
boundaries 1 YLO YHI
boundaries 2 ZLO ZHI

max_surface 2

start_surface
name walls
action both all reflect
color both 0 0 0
polygon both edge
max_panels rect 6
panel rect +0 XLO YLO ZLO YSIZE ZSIZE
panel rect -0 XHI YLO ZLO YSIZE ZSIZE
panel rect +1 XLO YLO ZLO XSIZE ZSIZE
```

```
panel rect -1 XLO YHI ZLO XSIZE ZSIZE
panel rect +2 XLO YLO ZLO XSIZE YSIZE
panel rect -2 XLO YLO ZHI XSIZE YSIZE
end_surface

start_surface
name stick
rate_internal A fsoln front 1
rate B fsoln front 85.8559
rate B front fsoln 276
rate_internal C fsoln bsoln 1
rate_internal C bsoln fsoln 1
rate_internal A2 fsoln front 0.1
rate B2 fsoln front 4.22852
rate B2 front fsoln 28
rate_internal C2 fsoln bsoln 0.1
rate_internal C2 bsoln fsoln 0.1
color both 0.5 0.5 0.5
max_panels rect 1
panel rect +0 0 YLO ZLO YSIZE ZSIZE
end_surface

mol 20000 A 0-XHI u u
mol 20000 B 0-XHI u u
mol 20000 C 0-XHI u u
mol 20000 A2 0-XHI u u
mol 20000 B2 0-XHI u u
mol 20000 C2 0-XHI u u
# 20000 molecules in 2x1x1 micron^3 is
      10000 per cubic micron

output_files FILEROOT1out.txt
      FILEROOT2out.txt
      FILEROOT3out.txt
cmd B molcountheader FILEROOT1out.txt
cmd B molcountheader FILEROOT2out.txt
cmd E molcountonsurf stick FILEROOT1out.txt
cmd E molcountinbox XLO 0 YLO YHI ZLO ZHI
      FILEROOT2out.txt

cmd i 0 0.1 0.0099 molcountspace A(fsoln)
      0 0 1 20 YLO YHI ZLO ZHI 0
      FILEROOT3out.txt
cmd @ 0.0009 molcountspace A(fsoln) 0 0 1
      20 YLO YHI ZLO ZHI 0 FILEROOT3out.txt

end_file
```

7. Details for stochastic simulations shown in Figure 7

I used Smoldyn and the configuration files shown below to calculate the data shown in Figure 7. These configuration files define a system that is 3-dimensional and that extends from 0 to 5 μm on each axis. Each side of this cubical system is tiled with 25 panels, each 1 μm square. Their locations are defined in the second file, titled CubeSurface.txt, of which an excerpt is shown below. These panels absorb molecules according to absorption coefficients that are calculated from Eq. 54 in the main paper. These coefficients account for two molecular point sources: one which sources an average of 5 molecules per time step and is at location (1, 3, 2.5) and the other which sources an average of 10 molecules per time step and is at location (3.5, 0.5, 2.5). These sources are shown with large dots in Figure 7A. Simulated molecules diffuse with 1 $\mu m^2/s$ diffusion coefficient and the simulation uses time steps of 0.01 second.

At every time step, Smoldyn records a histogram for the numbers of molecules in each bin across a band of space. The histogram includes 20 bins along the x-axis, which extend from $x = 0$ to 5 μm. Each bin also extends from $y = 2.25$ to 2.75 μm and $z = 2.25$ to 2.75 μm. To reduce stochastic noise in the final data, 1000 of these histograms are averaged together before Smoldyn outputs a final time-averaged histogram. All time-averaged histograms recorded after time 20 s were indistinguishable from each other, implying that the system is essentially at steady state by 20 s. One of these time-averaged histograms (at time 50 s) is shown with dots in Figure 7B; I determined concentrations by dividing the histogram molecule counts by the bin volumes.

Configuration file for Figure 7

```
# File for figure 7
# units microns, seconds

define SOURCE1 A 5     1 3 2.5
define SOURCE2 A 10    3.5 0.5 2.5
define OUTFILE1 FILEROOTout1.txt
define OUTFILE2 FILEROOTout2.txt

graphics opengl
graphic_iter 10

dim 3
names A
max_mol 10000
boxsize 1

difc all(solution) 1

color A(all) 1 0 0

time_start 0
time_stop 100
time_step 0.01

boundaries 0 0 5
boundaries 1 0 5
boundaries 2 0 5

max_surface 1

start_surface
name walls
color both 0 0 0
polygon both edge
read_file CubeSurface.txt
unbounded_emitter front SOURCE1
unbounded_emitter front SOURCE2
end_surface

output_files OUTFILE1 OUTFILE2

cmd e pointsource SOURCE1
cmd e pointsource SOURCE2
cmd e molcountspace A 0  0 5  20  2.25
      2.75  2.25 2.75  1000 OUTFILE1
cmd a listmols OUTFILE2

end_file
```

Portion of CubeSurface.txt file

```
# surface of a cube, where the whole cube
# is size 5 x 5 x 5 and each side is
# divided into 25 square panels, each of
# size 1 x 1

max_panels rect 150
panel rect +0  0 0 0  1 1
panel rect +0  0 0 1  1 1
panel rect +0  0 0 2  1 1     x00
panel rect +0  0 0 3  1 1
panel rect +0  0 0 4  1 1
panel rect +0  0 1 0  1 1
panel rect +0  0 1 1  1 1
panel rect +0  0 1 2  1 1     x01
panel rect +0  0 1 3  1 1
panel rect +0  0 1 4  1 1
panel rect +0  0 2 0  1 1
panel rect +0  0 2 1  1 1
panel rect +0  0 2 2  1 1     x02

... 140 more lines ...

panel rect -2  4 3 5  1 1
panel rect -2  4 4 5  1 1

end_file
```