Pulling self-avoiding walks from a surface.

Tony Guttmann

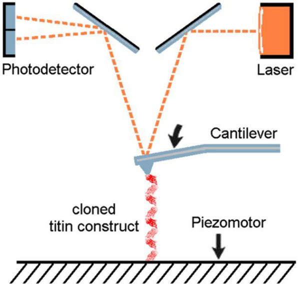
ARC Centre of Excellence for Mathematics and Statistics of Complex Systems
Department of Mathematics and Statistics
The University of Melbourne, Australia

ANZAMP Conference, November 2013

Joint work with Iwan Jensen (Melbourne)
and Stu Whittington (Toronto)

INTRODUCTION

- Techniques such as AFM allow adsorbed polymer molecules to be pulled off a surface. Need theories of adsorbed polymers subject to a force.



PREVIOUS WORK

- Earlier work focussed on random, directed and partially directed walk models. We consider the more realistic SAW model.
- Recently, vR & W established the existence of a phase boundary between an adsorbed phase and a ballistic phase when the force is applied normal to the surface.
- We give the first proof that this phase transition is first-order.

PREVIOUS WORK

- Earlier work focussed on random, directed and partially directed walk models. We consider the more realistic SAW model.
- Recently, vR & W established the existence of a phase boundary between an adsorbed phase and a ballistic phase when the force is applied normal to the surface.
- We give the first proof that this phase transition is first-order.

Previous work

- Earlier work focussed on random, directed and partially directed walk models. We consider the more realistic SAW model.
- Recently, vR & W established the existence of a phase boundary between an adsorbed phase and a ballistic phase when the force is applied normal to the surface.
- We give the first proof that this phase transition is first-order.

OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

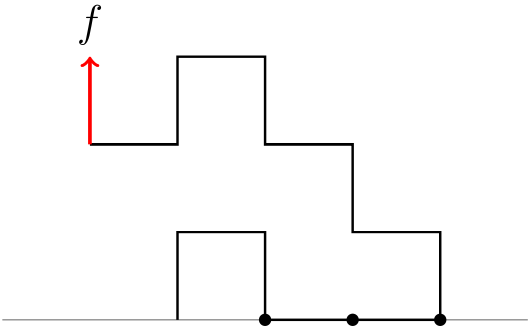
OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

OUTLINE OF THIS WORK

- We use exact enumeration and series analysis techniques to identify this phase boundary for SAWs on the square lattice.
- We give precise estimates of various critical points.
- And various critical exponents.
- A combination of three ingredients
- Ne rigorous results.
- Faster algorithms giving extended series data
- New numerical techniques.

14 STEPS, 3 CONTACTS, END-POINT AT HEIGHT 2 $\Longrightarrow$ $x^{14}a^3y^2$.



Notation and definitions

- Square lattice: vertex coordinates (x_i, y_i), $i = 0, 1, 2, \ldots n$.
- c_n is the number of n-step SAWs.
- $$\lim_{n\to\infty} n^{-1} \log c_n = \log \mu$$
  exists (HM54), where μ is the *growth constant* of SAWs.
- A *positive walk* is a SAW that starts at the origin and has $y_i \geq 0$ for all $0 \leq i \leq n$. Cardinality c_n^+.
- $$\lim_{n\to\infty} n^{-1} \log c_n^+ = \log \mu. \quad \text{(W75)}$$
- Vertices of a positive walk with $y_i = 0$ are *visits* to the surface, (by convention we exclude the vertex at the origin).

NOTATION AND DEFINITIONS

- Square lattice: vertex coordinates (x_i, y_i), $i = 0, 1, 2, \ldots n$.
- c_n is the number of n-step SAWs.
- $$\lim_{n\to\infty} n^{-1} \log c_n = \log \mu$$

  exists (HM54), where μ is the *growth constant* of SAWs.
- A *positive walk* is a SAW that starts at the origin and has $y_i \geq 0$ for all $0 \leq i \leq n$. Cardinality c_n^+.
- $$\lim_{n\to\infty} n^{-1} \log c_n^+ = \log \mu. \quad \text{(W75)}$$
- Vertices of a positive walk with $y_i = 0$ are *visits* to the surface, (by convention we exclude the vertex at the origin).

NOTATION AND DEFINITIONS

- Square lattice: vertex coordinates (x_i, y_i), $i = 0, 1, 2, \ldots n$.
- c_n is the number of n-step SAWs.
- $$\lim_{n\to\infty} n^{-1} \log c_n = \log \mu$$

  exists (HM54), where μ is the *growth constant* of SAWs.
- A *positive walk* is a SAW that starts at the origin and has $y_i \geq 0$ for all $0 \leq i \leq n$. Cardinality c_n^+.
- $$\lim_{n\to\infty} n^{-1} \log c_n^+ = \log \mu. \quad \text{(W75)}$$
- Vertices of a positive walk with $y_i = 0$ are *visits* to the surface, (by convention we exclude the vertex at the origin).

Notation and definitions

- Square lattice: vertex coordinates (x_i, y_i), $i = 0, 1, 2, \ldots n$.
- c_n is the number of n-step SAWs.
- $$\lim_{n\to\infty} n^{-1} \log c_n = \log \mu$$
  exists (HM54), where μ is the *growth constant* of SAWs.
- A *positive walk* is a SAW that starts at the origin and has $y_i \geq 0$ for all $0 \leq i \leq n$. Cardinality c_n^+.
- $$\lim_{n\to\infty} n^{-1} \log c_n^+ = \log \mu. \quad \text{(W75)}$$
- Vertices of a positive walk with $y_i = 0$ are *visits* to the surface, (by convention we exclude the vertex at the origin).

NOTATION AND DEFINITIONS

- Square lattice: vertex coordinates (x_i, y_i), $i = 0, 1, 2, \ldots n$.
- c_n is the number of n-step SAWs.
- $$\lim_{n \to \infty} n^{-1} \log c_n = \log \mu$$

  exists (HM54), where μ is the *growth constant* of SAWs.
- A *positive walk* is a SAW that starts at the origin and has $y_i \geq 0$ for all $0 \leq i \leq n$. Cardinality c_n^+.
- $$\lim_{n \to \infty} n^{-1} \log c_n^+ = \log \mu. \quad \text{(W75)}$$

- Vertices of a positive walk with $y_i = 0$ are *visits* to the surface, (by convention we exclude the vertex at the origin).

NOTATION AND DEFINITIONS

- Square lattice: vertex coordinates (x_i, y_i), $i = 0, 1, 2, \ldots n$.
- c_n is the number of n-step SAWs.
- $$\lim_{n\to\infty} n^{-1} \log c_n = \log \mu$$

  exists (HM54), where μ is the *growth constant* of SAWs.
- A *positive walk* is a SAW that starts at the origin and has $y_i \geq 0$ for all $0 \leq i \leq n$. Cardinality c_n^+.
- $$\lim_{n\to\infty} n^{-1} \log c_n^+ = \log \mu. \ \ \text{(W75)}$$
- Vertices of a positive walk with $y_i = 0$ are *visits* to the surface, (by convention we exclude the vertex at the origin).

PARTITION FUNCTION AND VARIABLES a AND y

- The walk has *height* h if $y_n = h$.
- $c_n^+(v, h)$ is card. of positive walks of n steps, v visits, height h.
- The partition function is

$$C_n(a, y) = \sum_{v,h} c_n^+(v, h) a^v y^h. \tag{1}$$

- ϵ is the energy associated with a visit and f is the force applied normally at the last vertex,

$$a = \exp[-\epsilon/k_B T] \quad \text{and} \quad y = \exp[f/k_B T] \tag{2}$$

- No force: $y = 1$ and the partition function is $C_n(a, 1)$,
- No surface interaction: $a = 1$. The partition function is $C_n(1, y)$.

PARTITION FUNCTION AND VARIABLES a AND y

- The walk has *height* h if $y_n = h$.
- $c_n^+(v,h)$ is card. of positive walks of n steps, v visits, height h.
- The partition function is

$$C_n(a,y) = \sum_{v,h} c_n^+(v,h) a^v y^h. \tag{1}$$

- ϵ is the energy associated with a visit and f is the force applied normally at the last vertex,

$$a = \exp[-\epsilon/k_B T] \quad \text{and} \quad y = \exp[f/k_B T] \tag{2}$$

- No force: $y = 1$ and the partition function is $C_n(a,1)$,
- No surface interaction: $a = 1$. The partition function is $C_n(1,y)$.

PARTITION FUNCTION AND VARIABLES a AND y

- The walk has *height* h if $y_n = h$.
- $c_n^+(v, h)$ is card. of positive walks of n steps, v visits, height h.
- The partition function is

$$C_n(a, y) = \sum_{v,h} c_n^+(v, h) a^v y^h. \qquad (1)$$

- ϵ is the energy associated with a visit and f is the force applied normally at the last vertex,

$$a = \exp[-\epsilon/k_B T] \quad \text{and} \quad y = \exp[f/k_B T] \qquad (2)$$

- No force: $y = 1$ and the partition function is $C_n(a, 1)$,
- No surface interaction: $a = 1$. The partition function is $C_n(1, y)$.

PARTITION FUNCTION AND VARIABLES a AND y

- The walk has *height* h if $y_n = h$.
- $c_n^+(v, h)$ is card. of positive walks of n steps, v visits, height h.
- The partition function is

$$C_n(a, y) = \sum_{v,h} c_n^+(v, h) a^v y^h. \quad (1)$$

- ϵ is the energy associated with a visit and f is the force applied normally at the last vertex,

$$a = \exp[-\epsilon/k_B T] \quad \text{and} \quad y = \exp[f/k_B T] \quad (2)$$

- No force: $y = 1$ and the partition function is $C_n(a, 1)$,
- No surface interaction: $a = 1$. The partition function is $C_n(1, y)$.

Partition function and variables a and y

- The walk has *height* h if $y_n = h$.
- $c_n^+(v,h)$ is card. of positive walks of n steps, v visits, height h.
- The partition function is

$$C_n(a,y) = \sum_{v,h} c_n^+(v,h) a^v y^h. \tag{1}$$

- ϵ is the energy associated with a visit and f is the force applied normally at the last vertex,

$$a = \exp[-\epsilon/k_B T] \quad \text{and} \quad y = \exp[f/k_B T] \tag{2}$$

- No force: $y = 1$ and the partition function is $C_n(a,1)$,
- No surface interaction: $a = 1$. The partition function is $C_n(1,y)$.

PARTITION FUNCTION AND VARIABLES a AND y

- The walk has *height* h if $y_n = h$.
- $c_n^+(v, h)$ is card. of positive walks of n steps, v visits, height h.
- The partition function is

$$C_n(a, y) = \sum_{v,h} c_n^+(v, h) a^v y^h. \tag{1}$$

- ϵ is the energy associated with a visit and f is the force applied normally at the last vertex,

$$a = \exp[-\epsilon/k_B T] \quad \text{and} \quad y = \exp[f/k_B T] \tag{2}$$

- No force: $y = 1$ and the partition function is $C_n(a, 1)$,
- No surface interaction: $a = 1$. The partition function is $C_n(1, y)$.

EMBARRASINGLY FEW RIGOROUS RESULTS

- HTW1982 proved (no force)

$$\lim_{n\to\infty} n^{-1} \log C_n(a, 1) \equiv \kappa(a)$$

exists. $\kappa(a)$ is a convex function of $\log a$.

- There exists $a = a_c^o > 1$ such that $\kappa(a) = \log \mu$ for $a \leq a_c^o$. $\kappa(a)$ is strictly monotone increasing for $a > a_c^o$.
- So $\kappa(a)$ is non-analytic at $a = a_c^o$.
- For $a < a_c^o$, $\langle v \rangle = o(n)$. For $a > a_c^o$, (the adsorbed phase,)

$$\lim_{n\to\infty} \frac{\langle v \rangle}{n} > 0$$

EMBARRASINGLY FEW RIGOROUS RESULTS

- HTW1982 proved (no force)

$$\lim_{n\to\infty} n^{-1} \log C_n(a,1) \equiv \kappa(a)$$

exists. $\kappa(a)$ is a convex function of $\log a$.

- There exists $a = a_c^o > 1$ such that $\kappa(a) = \log \mu$ for $a \le a_c^o$. $\kappa(a)$ is strictly monotone increasing for $a > a_c^o$.
- So $\kappa(a)$ is non-analytic at $a = a_c^o$.
- For $a < a_c^o$, $\langle v \rangle = o(n)$. For $a > a_c^o$, (the adsorbed phase,)

$$\lim_{n\to\infty} \frac{\langle v \rangle}{n} > 0$$

EMBARRASINGLY FEW RIGOROUS RESULTS

- HTW1982 proved (no force)

$$\lim_{n\to\infty} n^{-1} \log C_n(a,1) \equiv \kappa(a)$$

exists. $\kappa(a)$ is a convex function of $\log a$.

- There exists $a = a_c^o > 1$ such that $\kappa(a) = \log \mu$ for $a \leq a_c^o$. $\kappa(a)$ is strictly monotone increasing for $a > a_c^o$.
- So $\kappa(a)$ is non-analytic at $a = a_c^o$.
- For $a < a_c^o$, $\langle v \rangle = o(n)$. For $a > a_c^o$, (the adsorbed phase,)

$$\lim_{n\to\infty} \frac{\langle v \rangle}{n} > 0$$

Embarrasingly few rigorous results

- HTW1982 proved (no force)

$$\lim_{n\to\infty} n^{-1} \log C_n(a,1) \equiv \kappa(a)$$

exists. $\kappa(a)$ is a convex function of $\log a$.

- There exists $a = a_c^o > 1$ such that $\kappa(a) = \log \mu$ for $a \le a_c^o$. $\kappa(a)$ is strictly monotone increasing for $a > a_c^o$.
- So $\kappa(a)$ is non-analytic at $a = a_c^o$.
- For $a < a_c^o$, $\langle v \rangle = o(n)$. For $a > a_c^o$, (the adsorbed phase,)

$$\lim_{n\to\infty} \frac{\langle v \rangle}{n} > 0$$

EMBARRASINGLY FEW RIGOROUS RESULTS

- Similarly
$$\lim_{n\to\infty} n^{-1} \log C_n(1,y) \equiv \lambda(y)$$
exists and $\lambda(y)$ is a convex function of $\log y$.
- There is a critical point $y_c^o \geq 1$ such that $\lambda(y) = \log \mu$ for $y \leq y_c^o$ and $\lambda(y)$ is strictly monotone increasing for $y > y_c^o$ (R09).
- At y_c^o : Transition from a free phase, $(\langle h \rangle = o(n))$ to a ballistic phase where
$$\lim_{n\to\infty} \frac{\langle h \rangle}{n} > 0.$$
- There are good reasons to believe that $y_c^o = 1$.

Embarrasingly few rigorous results

- Similarly

$$\lim_{n\to\infty} n^{-1} \log C_n(1,y) \equiv \lambda(y)$$

exists and $\lambda(y)$ is a convex function of $\log y$.

- There is a critical point $y_c^o \geq 1$ such that $\lambda(y) = \log \mu$ for $y \leq y_c^o$ and $\lambda(y)$ is strictly monotone increasing for $y > y_c^o$ (R09).
- At y_c^o : Transition from a free phase, ($\langle h \rangle = o(n)$) to a ballistic phase where

$$\lim_{n\to\infty} \frac{\langle h \rangle}{n} > 0.$$

- There are good reasons to believe that $y_c^o = 1$.

Embarrasingly few rigorous results

- Similarly
$$\lim_{n\to\infty} n^{-1}\log C_n(1,y) \equiv \lambda(y)$$
exists and $\lambda(y)$ is a convex function of $\log y$.
- There is a critical point $y_c^o \geq 1$ such that $\lambda(y) = \log\mu$ for $y \leq y_c^o$ and $\lambda(y)$ is strictly monotone increasing for $y > y_c^o$ (R09).
- At y_c^o : Transition from a free phase, $(\langle h\rangle = o(n))$ to a ballistic phase where
$$\lim_{n\to\infty} \frac{\langle h\rangle}{n} > 0.$$
- There are good reasons to believe that $y_c^o = 1$.

EMBARRASINGLY FEW RIGOROUS RESULTS

- Similarly

$$\lim_{n\to\infty} n^{-1} \log C_n(1,y) \equiv \lambda(y)$$

exists and $\lambda(y)$ is a convex function of $\log y$.

- There is a critical point $y_c^o \geq 1$ such that $\lambda(y) = \log \mu$ for $y \leq y_c^o$ and $\lambda(y)$ is strictly monotone increasing for $y > y_c^o$ (R09).
- At y_c^o : Transition from a free phase, $(\langle h \rangle = o(n))$ to a ballistic phase where

$$\lim_{n\to\infty} \frac{\langle h \rangle}{n} > 0.$$

- There are good reasons to believe that $y_c^o = 1$.

MORE GENERAL MODEL

- For the two variable model, vRW2013 proved the existence of the free energy

$$\psi(a, y) = \lim_{n \to \infty} n^{-1} \log C_n(a, y).$$

- Further, $\psi(a, y)$ is a convex function of $\log a$ and $\log y$ and

$$\psi(a, y) = \max[\kappa(a), \lambda(y)].$$

- This implies that there is a *free phase* when $a < a_c^o$ and $y < y_c^o$ where $\langle v \rangle = o(n)$ and $\langle h \rangle = o(n)$, and a strictly monotone curve $y = y_c(a)$ through the point (a_c^o, y_c^o) separating two phases:
  1. an *adsorbed phase* when $a > a_c^o$ and $y < y_c(a)$, and
  2. a *ballistic phase* when $y > \max[y_c^o, y_c(a)]$.
- Moreover, $y_c(a)$ is asymptotic to $y = a$ as $a \to \infty$. (SQ).

MORE GENERAL MODEL

- For the two variable model, vRW2013 proved the existence of the free energy

$$\psi(a, y) = \lim_{n\to\infty} n^{-1} \log C_n(a, y).$$

- Further, $\psi(a, y)$ is a convex function of $\log a$ and $\log y$ and

$$\psi(a, y) = \max[\kappa(a), \lambda(y)].$$

- This implies that there is a *free phase* when $a < a_c^o$ and $y < y_c^o$ where $\langle v \rangle = o(n)$ and $\langle h \rangle = o(n)$, and a strictly monotone curve $y = y_c(a)$ through the point (a_c^o, y_c^o) separating two phases:
  1. an *adsorbed phase* when $a > a_c^o$ and $y < y_c(a)$, and
  2. a *ballistic phase* when $y > \max[y_c^o, y_c(a)]$.
- Moreover, $y_c(a)$ is asymptotic to $y = a$ as $a \to \infty$. (SQ).

More general model

- For the two variable model, vRW2013 proved the existence of the free energy

$$\psi(a,y) = \lim_{n\to\infty} n^{-1} \log C_n(a,y).$$

- Further, $\psi(a,y)$ is a convex function of $\log a$ and $\log y$ and

$$\psi(a,y) = \max[\kappa(a), \lambda(y)].$$

- This implies that there is a *free phase* when $a < a_c^o$ and $y < y_c^o$ where $\langle v \rangle = o(n)$ and $\langle h \rangle = o(n)$, and a strictly monotone curve $y = y_c(a)$ through the point (a_c^o, y_c^o) separating two phases:
  1. an *adsorbed phase* when $a > a_c^o$ and $y < y_c(a)$, and
  2. a *ballistic phase* when $y > \max[y_c^o, y_c(a)]$.
- Moreover, $y_c(a)$ is asymptotic to $y = a$ as $a \to \infty$. (SQ).

MORE GENERAL MODEL

- For the two variable model, vRW2013 proved the existence of the free energy

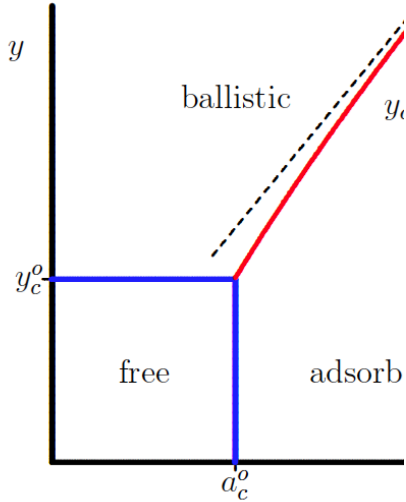
$$\psi(a, y) = \lim_{n\to\infty} n^{-1} \log C_n(a, y).$$

- Further, $\psi(a, y)$ is a convex function of $\log a$ and $\log y$ and

$$\psi(a, y) = \max[\kappa(a), \lambda(y)].$$

- This implies that there is a *free phase* when $a < a_c^o$ and $y < y_c^o$ where $\langle v \rangle = o(n)$ and $\langle h \rangle = o(n)$, and a strictly monotone curve $y = y_c(a)$ through the point (a_c^o, y_c^o) separating two phases:
  1. an *adsorbed phase* when $a > a_c^o$ and $y < y_c(a)$, and
  2. a *ballistic phase* when $y > \max[y_c^o, y_c(a)]$.
- Moreover, $y_c(a)$ is asymptotic to $y = a$ as $a \to \infty$. (SQ).

SCHEMATIC PHASE DIAGRAM. (VAN RENSBURG & WHITTINGTON)



IMPROVED ALGORITHM

- The algorithm is based on the CEG (1993) SAW algorithm.
- The TM algorithm keeps track of the way partially constructed SAWs are connected to the left of a cut-line.
- Recently Clisby and Jensen (2012) devised a more efficient implementation of the algorithm for SAPs.
- They kept track of how a partially constructed SAP must connect up to the right of the cut-line.
- Jensen recently extended this approach to SAWs.

Improved algorithm

- The algorithm is based on the CEG (1993) SAW algorithm.
- The TM algorithm keeps track of the way partially constructed SAWs are connected to the left of a cut-line.
- Recently Clisby and Jensen (2012) devised a more efficient implementation of the algorithm for SAPs.
- They kept track of how a partially constructed SAP must connect up to the right of the cut-line.
- Jensen recently extended this approach to SAWs.

IMPROVED ALGORITHM

- The algorithm is based on the CEG (1993) SAW algorithm.
- The TM algorithm keeps track of the way partially constructed SAWs are connected to the left of a cut-line.
- Recently Clisby and Jensen (2012) devised a more efficient implementation of the algorithm for SAPs.
- They kept track of how a partially constructed SAP must connect up to the right of the cut-line.
- Jensen recently extended this approach to SAWs.

Improved algorithm

- The algorithm is based on the CEG (1993) SAW algorithm.
- The TM algorithm keeps track of the way partially constructed SAWs are connected to the left of a cut-line.
- Recently Clisby and Jensen (2012) devised a more efficient implementation of the algorithm for SAPs.
- They kept track of how a partially constructed SAP must connect up to the right of the cut-line.
- Jensen recently extended this approach to SAWs.

Geometry of rectangle

- We count the number of walks in rectangles $W \times L$ unit cells.
- A spanning walk has length at least $W + L$ steps.
- We add contributions from all rectangles of width $W \leq W_{\max}$, and length $W \leq L \leq 2W_{\max} - W + 1$
- This gives the number of walks for an infinite lattice correctly up to length $N = 2W_{\max} + 1$.

GEOMETRY OF RECTANGLE

- We count the number of walks in rectangles $W \times L$ unit cells.
- A spanning walk has length at least $W + L$ steps.
- We add contributions from all rectangles of width $W \leq W_{\max}$, and length $W \leq L \leq 2W_{\max} - W + 1$
- This gives the number of walks for an infinite lattice correctly up to length $N = 2W_{\max} + 1$.

Geometry of rectangle

- We count the number of walks in rectangles $W \times L$ unit cells.
- A spanning walk has length at least $W + L$ steps.
- We add contributions from all rectangles of width $W \leq W_{\text{max}}$, and length $W \leq L \leq 2W_{\text{max}} - W + 1$
- This gives the number of walks for an infinite lattice correctly up to length $N = 2W_{\text{max}} + 1$.

Geometry of rectangle

- We count the number of walks in rectangles $W \times L$ unit cells.
- A spanning walk has length at least $W + L$ steps.
- We add contributions from all rectangles of width $W \leq W_{\text{max}}$, and length $W \leq L \leq 2W_{\text{max}} - W + 1$
- This gives the number of walks for an infinite lattice correctly up to length $N = 2W_{\text{max}} + 1$.

EXAMPLE OF A SAW IN A RECTANGLE

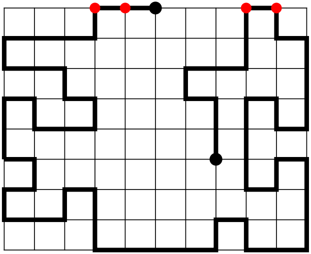


Figure: An example of a self-avoiding walk on a 10×8 rectangle. The walk is tethered to the surface, has the end-point at $h = 5$ and four vertices (other than the start-point) in the surface.

Basic idea: Any SAW has exactly two end-points.

OUTLINE OF THE ALGORITHM

Cutting the SAW by a vertical line (dashed), the SAW is broken into pieces to the left and right of the cut-line. On *either* side of the line are a set of arcs connecting two edges on the cut-line and at most two line pieces connected to the end-points of the SAW.

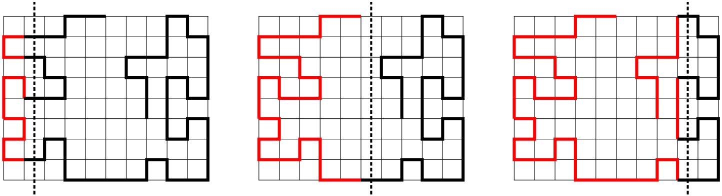


Figure: Examples of cut-lines through the SAW such that the signature of the incomplete section to the right of the cut-line (black lines) contains, respectively, two, one and no free edges.

MOVING THROUGH THE RECTANGLE–BUILDING THE TM

- At every stage a configuration of occupied edges along the cut-line can be described in two ways.
- The edges are connected forming either arcs or line pieces to the left or right of the cut-line.
- Moving the cut-line from left to right we can keep track of how the pieces are connected to the left (the past). This is the traditional TM.
- Tracking how edges can be connected to the right of the cut-line so as to form a valid SAW (the future), is the basis of the new algorithm.
- Looking at a given SAW and cut-line, the partial SAW to the right of this line consists of a number of arcs connecting two edges and at most two *free* edges which are not connected to any occupied edge on the current cut-line.

Moving through the rectangle–building the TM

- At every stage a configuration of occupied edges along the cut-line can be described in two ways.
- The edges are connected forming either arcs or line pieces to the left or right of the cut-line.
- Moving the cut-line from left to right we can keep track of how the pieces are connected to the left (the past). This is the traditional TM.
- Tracking how edges can be connected to the right of the cut-line so as to form a valid SAW (the future), is the basis of the new algorithm.
- Looking at a given SAW and cut-line, the partial SAW to the right of this line consists of a number of arcs connecting two edges and at most two *free* edges which are not connected to any occupied edge on the current cut-line.

Moving through the rectangle–building the TM

- At every stage a configuration of occupied edges along the cut-line can be described in two ways.
- The edges are connected forming either arcs or line pieces to the left or right of the cut-line.
- Moving the cut-line from left to right we can keep track of how the pieces are connected to the left (the past). This is the traditional TM.
- Tracking how edges can be connected to the right of the cut-line so as to form a valid SAW (the future), is the basis of the new algorithm.
- Looking at a given SAW and cut-line, the partial SAW to the right of this line consists of a number of arcs connecting two edges and at most two *free* edges which are not connected to any occupied edge on the current cut-line.

MOVING THROUGH THE RECTANGLE–BUILDING THE TM

- At every stage a configuration of occupied edges along the cut-line can be described in two ways.
- The edges are connected forming either arcs or line pieces to the left or right of the cut-line.
- Moving the cut-line from left to right we can keep track of how the pieces are connected to the left (the past). This is the traditional TM.
- Tracking how edges can be connected to the right of the cut-line so as to form a valid SAW (the future), is the basis of the new algorithm.
- Looking at a given SAW and cut-line, the partial SAW to the right of this line consists of a number of arcs connecting two edges and at most two *free* edges which are not connected to any occupied edge on the current cut-line.

MOVING THROUGH THE RECTANGLE–BUILDING THE TM

- At every stage a configuration of occupied edges along the cut-line can be described in two ways.
- The edges are connected forming either arcs or line pieces to the left or right of the cut-line.
- Moving the cut-line from left to right we can keep track of how the pieces are connected to the left (the past). This is the traditional TM.
- Tracking how edges can be connected to the right of the cut-line so as to form a valid SAW (the future), is the basis of the new algorithm.
- Looking at a given SAW and cut-line, the partial SAW to the right of this line consists of a number of arcs connecting two edges and at most two *free* edges which are not connected to any occupied edge on the current cut-line.

LABELING RULES

- Any configuration along the cut-line can thus be represented by a set of edge states $\{\sigma_i\}$, where

$$\sigma_i = \begin{cases} 0 & \text{empty edge,} \\ 1 & \text{lower edge,} \\ 2 & \text{upper edge.} \\ 3 & \text{free edge.} \end{cases}$$

  Reading from bottom to top, the signature S along the cut-lines of the SAW above are, respectively, $S = \{030010230\}$, $S = \{300000000\}$, and $S = \{102001002\}$.

- Since crossings are not permitted this encoding uniquely describes how the occupied edges are connected.

Labeling rules

- Any configuration along the cut-line can thus be represented by a set of edge states $\{\sigma_i\}$, where

$$\sigma_i = \begin{cases} 0 & \text{empty edge,} \\ 1 & \text{lower edge,} \\ 2 & \text{upper edge.} \\ 3 & \text{free edge.} \end{cases}$$

  Reading from bottom to top, the signature S along the cut-lines of the SAW above are, respectively, $S = \{030010230\}$, $S = \{300000000\}$, and $S = \{102001002\}$.
- Since crossings are not permitted this encoding uniquely describes how the occupied edges are connected.

Updating after a boundary move

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x,a)$ where x, a is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x,a) = G_{S'}(x,a) + x^k a^m G_S(x,a)$.
- Signatures are discarded after processing.

UPDATING AFTER A BOUNDARY MOVE

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x, a)$ where x, a is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x, a) = G_{S'}(x, a) + x^k a^m G_S(x, a)$.
- Signatures are discarded after processing.

Updating after a boundary move

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x, a)$ where x, a is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x, a) = G_{S'}(x, a) + x^k a^m G_S(x, a)$.
- Signatures are discarded after processing.

UPDATING AFTER A BOUNDARY MOVE

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x, a)$ where x, a is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x, a) = G_{S'}(x, a) + x^k a^m G_S(x, a)$.
- Signatures are discarded after processing.

UPDATING AFTER A BOUNDARY MOVE

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x, a)$ where x, a is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x, a) = G_{S'}(x, a) + x^k a^m G_S(x, a)$.
- Signatures are discarded after processing.

UPDATING AFTER A BOUNDARY MOVE

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x, a)$ where $x,\ a$ is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x, a) = G_{S'}(x, a) + x^k a^m G_S(x, a)$.
- Signatures are discarded after processing.

Updating after a boundary move

- The most efficient implementation of the algorithm involves moving the cut-line so as to build up the lattice vertex by vertex.
- The sum over all contributing graphs is calculated as the cut-line is moved through the lattice.
- For each configuration of edges we keep a generating function G_S for partial walks with signature S.
- Clearly, G_S is a polynomial $G_S(x,a)$ where $x,\ a$ is conjugate to the number of steps/surface vertices.
- Update: Each source signature S (before the boundary move) generates a few new target signatures S' as $k = 0, 1$ or 2 new edges are inserted with $m = 0$ or 1 surface visits.
- This leads to the update $G_{S'}(x,a) = G_{S'}(x,a) + x^k a^m G_S(x,a)$.
- Signatures are discarded after processing.

Modifications for this problem

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n + 1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

MODIFICATIONS FOR THIS PROBLEM

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n+1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

MODIFICATIONS FOR THIS PROBLEM

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n+1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

MODIFICATIONS FOR THIS PROBLEM

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n+1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

MODIFICATIONS FOR THIS PROBLEM

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n+1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

Modifications for this problem

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n+1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

MODIFICATIONS FOR THIS PROBLEM

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n + 1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

Modifications for this problem

- We force the SAW to have a free end at the top of the rectangle.
- We must consider all rectangles with $W \leq n + 1$.
- The number of signatures grows exponentially with W. Hence we must minimize the length of the cut-line for optimality.
- The rectangles are broken into two sub-sets, $L \geq W$ and $L < W$.
- For $L < W$ rectangles have start-point on the left-most border.
- To keep track of the height h, the end-point must be in a row h units from the surface, Then repeat for all h.
- We calculated the number of SAW up to length $n = 59$.
- Parallel calculations were performed using up to 16 processors, up to 40GB of memory and just under 6000 CPU hours

Analysis. No surface interaction, $a = 1$

- We have anaysed the series using differential approximants when $a = 1$, corresponding to no surface interaction.
- For $\log y < 0$, $\lambda(y) = \log \mu$ while for $\log y > 0$, $\max[\log \mu, \log y] \leq \lambda(y) \leq \log \mu + \log y$.

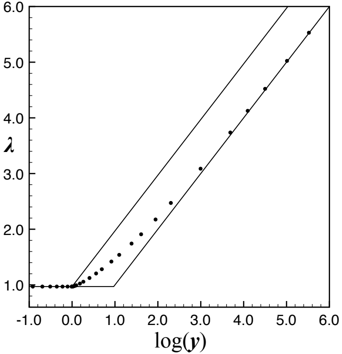


Figure: The free energy $\lambda(y)$, with bounds

Analysis. No surface interaction, $a = 1$

- We have anaysed the series using differential approximants when $a = 1$, corresponding to no surface interaction.
- For $\log y < 0$, $\lambda(y) = \log \mu$ while for $\log y > 0$, $\max[\log \mu, \log y] \leq \lambda(y) \leq \log \mu + \log y$.

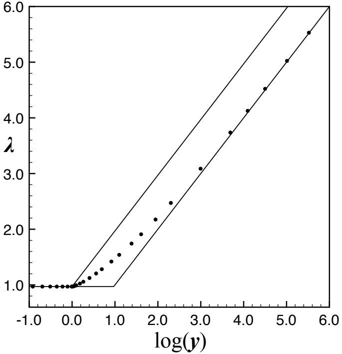


Figure: The free energy $\lambda(y)$, with bounds

NO SURFACE INTERACTION, $a = 1$ – MORE DETAILS

$$H(x,y) = \sum_n C_n(1,y)x^n = \sum_n e^{\lambda(y)n+o(n)}x^n$$

- $H(x,y)$ will be singular at $x = x_c(y) = \exp[-\lambda(y)]$, and near $x_c(y)$

$$H(x,y) \sim \frac{A}{[x_c(y) - x]^{\gamma(y)}}$$

where $\gamma(y)$ depends on y.

NO SURFACE INTERACTION, $a = 1$ – MORE DETAILS

$$H(x,y) = \sum_n C_n(1,y)x^n = \sum_n e^{\lambda(y)n+o(n)}x^n$$

- $H(x,y)$ will be singular at $x = x_c(y) = \exp[-\lambda(y)]$, and near $x_c(y)$

$$H(x,y) \sim \frac{A}{[x_c(y) - x]^{\gamma(y)}}$$

where $\gamma(y)$ depends on y.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

No surface interaction, $a = 1$ – still more

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

NO SURFACE INTERACTION, $a = 1$ – STILL MORE

- At $y = 1$ the series is well behaved. The critical point $= 1/\mu$, the exponent is $\gamma_1 = 61/64$ (TASAW), as one expects.
- For $y < 1$, $x_c(y)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with y, settling at at $\gamma_{1,1} = -3/16 = -0.1875$.
- For $y > 1$, $x_c(y)$ mon. dec. as y inc. The sing. is a simple pole.
- The analysis is exquisitely sensitive to the value of y near $y = 1$. This gives us a method for confirming that $y_c = 1$.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of y_c until we get agreement with $1/\mu$.
- This turns out to be at $y_c = 0.9999995 \pm 0.0000005$.
- Now $y_c \geq 1$, plus the numerical result, suggests $y_c = 1$.
- In summary: For $y > y_c$, the exponent is 1. For $y = y_c$ it is $\gamma_1 = 61/64$, and is $\gamma_{1,1} = -3/16$ for $y < y_c$.

ANALYSIS. NO APPLIED FORCE, $y = 1$

$$K(x,a) = \sum_n C_n(a,1)x^n = \sum_n e^{\kappa(a)n+o(n)}x^n.$$

- $K(x,a)$ will be singular at $x = x_c(a) = \exp[-\kappa(a)]$ and near $x_c(a)$

$$K(x,a) \sim \frac{B}{[x_c(a) - x]^{\gamma(a)}}$$

where $\gamma(a)$ depends on a.

ANALYSIS. NO APPLIED FORCE, $y = 1$

$$K(x,a) = \sum_n C_n(a,1)x^n = \sum_n e^{\kappa(a)n+o(n)}x^n.$$

- $K(x,a)$ will be singular at $x = x_c(a) = \exp[-\kappa(a)]$ and near $x_c(a)$

$$K(x,a) \sim \frac{B}{[x_c(a) - x]^{\gamma(a)}}$$

where $\gamma(a)$ depends on a.

No applied force, $y = 1$ – more

- We have analysed the series using differential approximants when $y = 1$, corresponding to no applied force.
- For $\log a < 0$, $\kappa(a) = \log \mu$ while for $\log a > 0$ $\max[\log \mu, \log a] \leq \kappa(a) \leq \log \mu + \log a$.

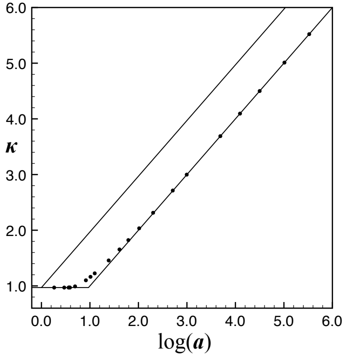


Figure: The free energy $\kappa(a)$, with bounds.

NO APPLIED FORCE, $y = 1$ – MORE

- We have analysed the series using differential approximants when $y = 1$, corresponding to no applied force.
- For $\log a < 0$, $\kappa(a) = \log \mu$ while for $\log a > 0$ $\max[\log \mu, \log a] \leq \kappa(a) \leq \log \mu + \log a$.

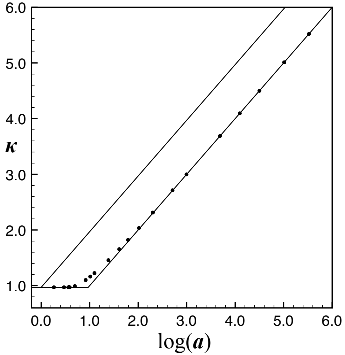


Figure: The free energy $\kappa(a)$, with bounds.

NO APPLIED FORCE, $y = 1$ – FIND a_c.

- The best exisiting estimate of a_c is $a_c = 1.77564$ (BGJ12).
- The series analysis is exquisitely sensitive to the value of a near a_c. This gives us a method for estimating a_c.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of a_c until we get agreement with $1/\mu$.
- This turns out to be at $a_c = 1.775615 \pm 0.000005$.
- At a_c the exponent is 1.4539.
- The exponent γ_1^{sp} for the special transition is conjectured to be $93/64 = 1.453125$, which is satisfyingly close.

NO APPLIED FORCE, $y = 1$ – FIND a_c.

- The best exisiting estimate of a_c is $a_c = 1.77564$ (BGJ12).
- The series analysis is exquisitely sensitive to the value of a near a_c. This gives us a method for estimating a_c.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of a_c until we get agreement with $1/\mu$.
- This turns out to be at $a_c = 1.775615 \pm 0.000005$.
- At a_c the exponent is 1.4539.
- The exponent γ_1^{sp} for the special transition is conjectured to be $93/64 = 1.453125$, which is satisfyingly close.

NO APPLIED FORCE, $y = 1$ – FIND a_c.

- The best exisiting estimate of a_c is $a_c = 1.77564$ (BGJ12).
- The series analysis is exquisitely sensitive to the value of a near a_c. This gives us a method for estimating a_c.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of a_c until we get agreement with $1/\mu$.
- This turns out to be at $a_c = 1.775615 \pm 0.000005$.
- At a_c the exponent is 1.4539.
- The exponent γ_1^{sp} for the special transition is conjectured to be $93/64 = 1.453125$, which is satisfyingly close.

NO APPLIED FORCE, $y = 1$ – FIND a_c.

- The best exisiting estimate of a_c is $a_c = 1.77564$ (BGJ12).
- The series analysis is exquisitely sensitive to the value of a near a_c. This gives us a method for estimating a_c.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of a_c until we get agreement with $1/\mu$.
- This turns out to be at $a_c = 1.775615 \pm 0.000005$.
- At a_c the exponent is 1.4539.
- The exponent γ_1^{sp} for the special transition is conjectured to be $93/64 = 1.453125$, which is satisfyingly close.

NO APPLIED FORCE, $y = 1$ – FIND a_c.

- The best exisiting estimate of a_c is $a_c = 1.77564$ (BGJ12).
- The series analysis is exquisitely sensitive to the value of a near a_c. This gives us a method for estimating a_c.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of a_c until we get agreement with $1/\mu$.
- This turns out to be at $a_c = 1.775615 \pm 0.000005$.
- At a_c the exponent is 1.4539.
- The exponent γ_1^{sp} for the special transition is conjectured to be $93/64 = 1.453125$, which is satisfyingly close.

NO APPLIED FORCE, $y = 1$ – FIND a_c.

- The best exisiting estimate of a_c is $a_c = 1.77564$ (BGJ12).
- The series analysis is exquisitely sensitive to the value of a near a_c. This gives us a method for estimating a_c.
- $1/\mu = 0.379052277751$, with uncertainty in the last digit. We vary our estimate of a_c until we get agreement with $1/\mu$.
- This turns out to be at $a_c = 1.775615 \pm 0.000005$.
- At a_c the exponent is 1.4539.
- The exponent γ_1^{sp} for the special transition is conjectured to be $93/64 = 1.453125$, which is satisfyingly close.

No applied force, $y = 1$ – still more

- For $a > a_c$, $x_c(a)$ is monotonically decreasing as a increases. The singularity is a simple pole.
- For $a < a_c$, $x_c(a)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with a, settling at at $\gamma_1 = 61/64$.
- Finally, with $y = 0$ and $a = a_c$ we expect loops at the special transition, with exponent $\gamma_{11}^{sp} = 13/16 = 0.8125$. Our estimate is 0.816 ± 0.006.
- In summary: For $a > a_c$, the exponent is 1. For $a = a_c$ it is $\gamma_1^{sp} = 93/64$, and is $\gamma_1 = 61/64$ for $a < a_c$.

No applied force, $y = 1$ – still more

- For $a > a_c$, $x_c(a)$ is monotonically decreasing as a increases. The singularity is a simple pole.
- For $a < a_c$, $x_c(a)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with a, settling at at $\gamma_1 = 61/64$.
- Finally, with $y = 0$ and $a = a_c$ we expect loops at the special transition, with exponent $\gamma_{11}^{sp} = 13/16 = 0.8125$. Our estimate is 0.816 ± 0.006.
- In summary: For $a > a_c$, the exponent is 1. For $a = a_c$ it is $\gamma_1^{sp} = 93/64$, and is $\gamma_1 = 61/64$ for $a < a_c$.

NO APPLIED FORCE, $y = 1$ – STILL MORE

- For $a > a_c$, $x_c(a)$ is monotonically decreasing as a increases. The singularity is a simple pole.
- For $a < a_c$, $x_c(a)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with a, settling at at $\gamma_1 = 61/64$.
- Finally, with $y = 0$ and $a = a_c$ we expect loops at the special transition, with exponent $\gamma_{11}^{sp} = 13/16 = 0.8125$. Our estimate is 0.816 ± 0.006.
- In summary: For $a > a_c$, the exponent is 1. For $a = a_c$ it is $\gamma_1^{sp} = 93/64$, and is $\gamma_1 = 61/64$ for $a < a_c$.

NO APPLIED FORCE, $y = 1$ – STILL MORE

- For $a > a_c$, $x_c(a)$ is monotonically decreasing as a increases. The singularity is a simple pole.
- For $a < a_c$, $x_c(a)$ remains unchanged at $1/\mu$, but the exponent estimates decrease rapidly with a, settling at at $\gamma_1 = 61/64$.
- Finally, with $y = 0$ and $a = a_c$ we expect loops at the special transition, with exponent $\gamma_{11}^{sp} = 13/16 = 0.8125$. Our estimate is 0.816 ± 0.006.
- In summary: For $a > a_c$, the exponent is 1. For $a = a_c$ it is $\gamma_1^{sp} = 93/64$, and is $\gamma_1 = 61/64$ for $a < a_c$.

PHASE DIAGRAM CALCULATION

- Recall that $\psi(a, y) = \kappa(a)$ throughout the adsorbed phase and $\psi(a, y) = \lambda(y)$ throughout the ballistic phase.
- The phase boundary is the locus of points where

$$\kappa(a) = \lambda(y)$$

.

- For a given a we calculated $\kappa(a)$ as above, then found the value of y s.t. $\lambda(y) = \kappa(a)$ by interpolation.
- More precisely, we calculated $y = f_1(x_c)$ by using the program Eureqa on our (y, x_c) data, and $a = f_2(x_c)$ from our (a, x_c) data.
- Thus for any x_c, we could calculate $(a(x_c), y(x_c))$.

Phase diagram calculation

- Recall that $\psi(a, y) = \kappa(a)$ throughout the adsorbed phase and $\psi(a, y) = \lambda(y)$ throughout the ballistic phase.
- The phase boundary is the locus of points where

$$\kappa(a) = \lambda(y)$$

.

- For a given a we calculated $\kappa(a)$ as above, then found the value of y s.t. $\lambda(y) = \kappa(a)$ by interpolation.
- More precisely, we calculated $y = f_1(x_c)$ by using the program Eureqa on our (y, x_c) data, and $a = f_2(x_c)$ from our (a, x_c) data.
- Thus for any x_c, we could calculate $(a(x_c), y(x_c))$.

Phase diagram calculation

- Recall that $\psi(a, y) = \kappa(a)$ throughout the adsorbed phase and $\psi(a, y) = \lambda(y)$ throughout the ballistic phase.
- The phase boundary is the locus of points where

$$\kappa(a) = \lambda(y)$$

.

- For a given a we calculated $\kappa(a)$ as above, then found the value of y s.t. $\lambda(y) = \kappa(a)$ by interpolation.
- More precisely, we calculated $y = f_1(x_c)$ by using the program Eureqa on our (y, x_c) data, and $a = f_2(x_c)$ from our (a, x_c) data.
- Thus for any x_c, we could calculate $(a(x_c), y(x_c))$.

Phase diagram calculation

- Recall that $\psi(a, y) = \kappa(a)$ throughout the adsorbed phase and $\psi(a, y) = \lambda(y)$ throughout the ballistic phase.
- The phase boundary is the locus of points where

$$\kappa(a) = \lambda(y)$$

.

- For a given a we calculated $\kappa(a)$ as above, then found the value of y s.t. $\lambda(y) = \kappa(a)$ by interpolation.
- More precisely, we calculated $y = f_1(x_c)$ by using the program Eureqa on our (y, x_c) data, and $a = f_2(x_c)$ from our (a, x_c) data.
- Thus for any x_c, we could calculate $(a(x_c), y(x_c))$.

PHASE DIAGRAM CALCULATION

- Recall that $\psi(a, y) = \kappa(a)$ throughout the adsorbed phase and $\psi(a, y) = \lambda(y)$ throughout the ballistic phase.
- The phase boundary is the locus of points where

$$\kappa(a) = \lambda(y)$$

.

- For a given a we calculated $\kappa(a)$ as above, then found the value of y s.t. $\lambda(y) = \kappa(a)$ by interpolation.
- More precisely, we calculated $y = f_1(x_c)$ by using the program Eureqa on our (y, x_c) data, and $a = f_2(x_c)$ from our (a, x_c) data.
- Thus for any x_c, we could calculate $(a(x_c), y(x_c))$.

PHASE DIAGRAM CALCULATION

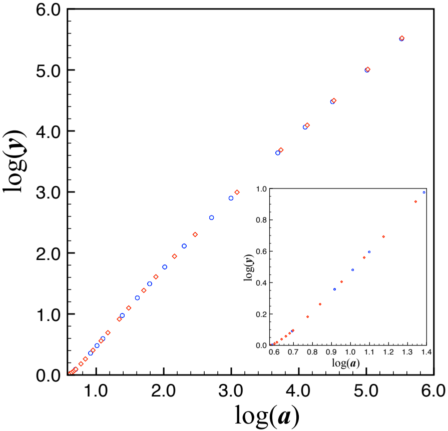
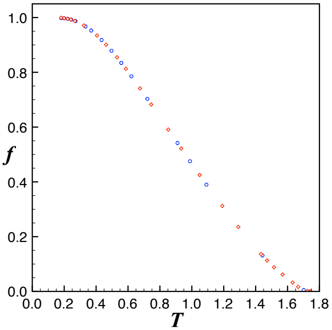


Figure: The phase boundary between the adsorbed and ballistic phases in the $(\log a, \log y)$-plane.

Physical variables

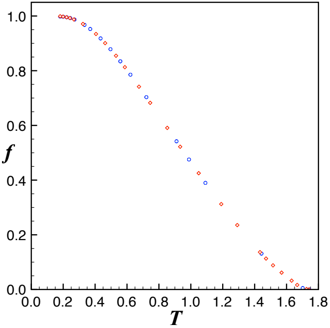
- We can switch to physical variables (force and temperature) using $a = \exp[-\epsilon/k_B T]$ and $y = \exp[f/k_B T]$.
- W.l.o.g we set $\epsilon = -1$ and $k_B = 1$.



- $f(T = 0) = 1$, and the slope at $T = 0$ is zero, (vRW2013). The curve is monotone decreasing, with no re-entrance.

Physical variables

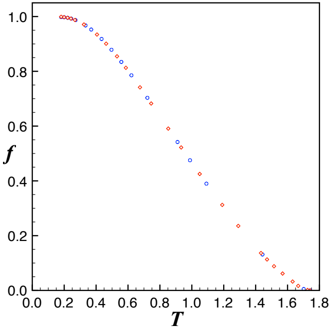
- We can switch to physical variables (force and temperature) using $a = \exp[-\epsilon/k_B T]$ and $y = \exp[f/k_B T]$.
- W.l.o.g we set $\epsilon = -1$ and $k_B = 1$.



- $f(T = 0) = 1$, and the slope at $T = 0$ is zero, (vRW2013). The curve is monotone decreasing, with no re-entrance.

Physical variables

- We can switch to physical variables (force and temperature) using $a = \exp[-\epsilon/k_B T]$ and $y = \exp[f/k_B T]$.
- W.l.o.g we set $\epsilon = -1$ and $k_B = 1$.



- $f(T = 0) = 1$, and the slope at $T = 0$ is zero, (vRW2013). The curve is monotone decreasing, with no re-entrance.

THE NATURE OF THE PHASE TRANSITION ON THE PHASE BOUNDARY.

- The phase transition from ballistic to adsorbed is first-order.
  Theorem The free energy $\psi(a, y)$ is not differentiable at the phase boundary between the ballistic and adsorbed phases, except perhaps at the triple point (a_c^o, y_c^o).
- At the phase boundary we find a double pole.
- That is, at any point (a_c, y_c), the series is singular at $x_c(a_c, y_c)$ with a double pole.
- The same situation is found with a Dyck path model and a Motzkin path model.

THE NATURE OF THE PHASE TRANSITION ON THE PHASE BOUNDARY.

- The phase transition from ballistic to adsorbed is first-order. **Theorem** The free energy $\psi(a, y)$ is not differentiable at the phase boundary between the ballistic and adsorbed phases, except perhaps at the triple point (a_c^o, y_c^o).
- At the phase boundary we find a double pole.
- That is, at any point (a_c, y_c), the series is singular at $x_c(a_c, y_c)$ with a double pole.
- The same situation is found with a Dyck path model and a Motzkin path model.

The nature of the phase transition on the phase boundary.

- The phase transition from ballistic to adsorbed is first-order. **Theorem** The free energy $\psi(a, y)$ is not differentiable at the phase boundary between the ballistic and adsorbed phases, except perhaps at the triple point (a_c^o, y_c^o).
- At the phase boundary we find a double pole.
- That is, at any point (a_c, y_c), the series is singular at $x_c(a_c, y_c)$ with a double pole.
- The same situation is found with a Dyck path model and a Motzkin path model.

THE NATURE OF THE PHASE TRANSITION ON THE PHASE BOUNDARY.

- The phase transition from ballistic to adsorbed is first-order. **Theorem** The free energy $\psi(a, y)$ is not differentiable at the phase boundary between the ballistic and adsorbed phases, except perhaps at the triple point (a_c^o, y_c^o).
- At the phase boundary we find a double pole.
- That is, at any point (a_c, y_c), the series is singular at $x_c(a_c, y_c)$ with a double pole.
- The same situation is found with a Dyck path model and a Motzkin path model.

THE NATURE OF THE PHASE TRANSITION ON THE PHASE BOUNDARY.

- The phase transition from ballistic to adsorbed is first-order. **Theorem** The free energy $\psi(a, y)$ is not differentiable at the phase boundary between the ballistic and adsorbed phases, except perhaps at the triple point (a_c^o, y_c^o).
- At the phase boundary we find a double pole.
- That is, at any point (a_c, y_c), the series is singular at $x_c(a_c, y_c)$ with a double pole.
- The same situation is found with a Dyck path model and a Motzkin path model.

CONCLUSION

- We have considered a SAW model of polymer adsorption at an impenetrable surface where
  1. the walk is terminally attached to the surface,
  2. the walk interacts with the surface, and
  3. a force applied normal to the surface at the last vertex of the walk.
- For the square lattice we have used series analysis techniques to investigate the phases and phase boundaries for the system.

CONCLUSION

- We have considered a SAW model of polymer adsorption at an impenetrable surface where
  1. the walk is terminally attached to the surface,
  2. the walk interacts with the surface, and
  3. a force applied normal to the surface at the last vertex of the walk.
- For the square lattice we have used series analysis techniques to investigate the phases and phase boundaries for the system.

Conclusion

- There are three phases,
- a free phase where the walk is desorbed but not ballistic,
- an adsorbed phase where the walk is adsorbed at the surface
- and a ballistic phase where the walk is desorbed but ballistic.
- We have located the phase boundaries and proved that the phase transition from the adsorbed to the ballistic phase is first order.
- In addition we have very precise values for the critical points for adsorption without a force and for the free to ballistic transition with no surface interaction.

Conclusion

- There are three phases,
- a free phase where the walk is desorbed but not ballistic,
- an adsorbed phase where the walk is adsorbed at the surface
- and a ballistic phase where the walk is desorbed but ballistic.
- We have located the phase boundaries and proved that the phase transition from the adsorbed to the ballistic phase is first order.
- In addition we have very precise values for the critical points for adsorption without a force and for the free to ballistic transition with no surface interaction.

Conclusion

- There are three phases,
- a free phase where the walk is desorbed but not ballistic,
- an adsorbed phase where the walk is adsorbed at the surface
- and a ballistic phase where the walk is desorbed but ballistic.
- We have located the phase boundaries and proved that the phase transition from the adsorbed to the ballistic phase is first order.
- In addition we have very precise values for the critical points for adsorption without a force and for the free to ballistic transition with no surface interaction.

CONCLUSION

- There are three phases,
- a free phase where the walk is desorbed but not ballistic,
- an adsorbed phase where the walk is adsorbed at the surface
- and a ballistic phase where the walk is desorbed but ballistic.
- We have located the phase boundaries and proved that the phase transition from the adsorbed to the ballistic phase is first order.
- In addition we have very precise values for the critical points for adsorption without a force and for the free to ballistic transition with no surface interaction.

CONCLUSION

- There are three phases,
- a free phase where the walk is desorbed but not ballistic,
- an adsorbed phase where the walk is adsorbed at the surface
- and a ballistic phase where the walk is desorbed but ballistic.
- We have located the phase boundaries and proved that the phase transition from the adsorbed to the ballistic phase is first order.
- In addition we have very precise values for the critical points for adsorption without a force and for the free to ballistic transition with no surface interaction.

CONCLUSION

- There are three phases,
- a free phase where the walk is desorbed but not ballistic,
- an adsorbed phase where the walk is adsorbed at the surface
- and a ballistic phase where the walk is desorbed but ballistic.
- We have located the phase boundaries and proved that the phase transition from the adsorbed to the ballistic phase is first order.
- In addition we have very precise values for the critical points for adsorption without a force and for the free to ballistic transition with no surface interaction.