More Mechanisms for Generating Power-Law Distributions

Principles of Complex Systems
Course CSYS/MATH 300, Fall, 2009

Prof. Peter Dodds

Dept. of Mathematics & Statistics
Center for Complex Systems :: Vermont Advanced Computing Center
University of Vermont

Outline

Outline

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoit Mandelbrot

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Derived Zipf's law through optimization [11]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and cost (C)...
- Minimize C/H by varying word frequency
- Recurring theme: what role does optimization play in complex systems?

Outline

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): “An Informational Theory of the Statistical Structure of Languages”[11]
- Simon (1955): “On a class of skew distribution functions”[14]
- Mandelbrot (1959): “A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon”
- Simon (1960): “Some further notes on a class of skew distribution functions”

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): "An Informational Theory of the Statistical Structure of Languages" [11]
- Simon (1955): "On a class of skew distribution functions" [14]
- Mandelbrot (1959): "A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon"
- Simon (1960): "Some further notes on a class of skew distribution functions"

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): "An Informational Theory of the Statistical Structure of Languages" [11]
- Simon (1955): "On a class of skew distribution functions" [14]
- Mandelbrot (1959): "A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon"
- Simon (1960): "Some further notes on a class of skew distribution functions"

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): "An Informational Theory of the Statistical Structure of Languages" [11]
- Simon (1955): "On a class of skew distribution functions" [14]
- Mandelbrot (1959): "A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon"
- Simon (1960): "Some further notes on a class of skew distribution functions"

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): “An Informational Theory of the Statistical Structure of Languages” [11]
- Simon (1955): “On a class of skew distribution functions” [14]
- Mandelbrot (1959): “A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon”
- Simon (1960): “Some further notes on a class of skew distribution functions”

Not everyone is happy... (cont.)

Mandelbrot vs. Simon:

- Mandelbrot (1961): “Final note on a class of skew distribution functions: analysis and critique of a model due to H.A. Simon”
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot”
- Mandelbrot (1961): “Post scriptum to ‘final note’”
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum”

Not everyone is happy... (cont.)

Mandelbrot vs. Simon:

- Mandelbrot (1961): “Final note on a class of skew distribution functions: analysis and critique of a model due to H.A. Simon”
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot”
- Mandelbrot (1961): “Post scriptum to ‘final note’”
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum”

Not everyone is happy... (cont.)

Mandelbrot vs. Simon:

- Mandelbrot (1961): “Final note on a class of skew distribution functions: analysis and critique of a model due to H.A. Simon”
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot”
- Mandelbrot (1961): “Post scriptum to ‘final note”’
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum”

Not everyone is happy... (cont.)

Mandelbrot vs. Simon:

- Mandelbrot (1961): "Final note on a class of skew distribution functions: analysis and critique of a model due to H.A. Simon"
- Simon (1961): "Reply to 'final note' by Benoit Mandelbrot"
- Mandelbrot (1961): "Post scriptum to 'final note'"
- Simon (1961): "Reply to Dr. Mandelbrot's post scriptum"

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

"You can't do this to me, I WENT TO COLLEGE!"

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

"You can't do this to me, I WENT TO COLLEGE!" "You weak minded fool!"

Not everyone is happy... (cont.)

Mandelbrot:

"We shall restate in detail our 1959 objections to Simon's 1955 model for the Pareto-Yule-Zipf distribution. Our objections are valid quite irrespectively of the sign of p-1, so that most of Simon's (1960) reply was irrelevant."

Simon:

"Dr. Mandelbrot has proposed a new set of objections to my 1955 models of the Yule distribution. Like his earlier objections, these are invalid."

Plankton:

"You can't do this to me, I WENT TO COLLEGE!" "You weak minded fool!" "That's it Mister! You just lost your brain privileges," etc.

Outline

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Mandelbrot's Assumptions

- Language contains n words: $w_1, w_2, \ldots, w_n$.
- ith word appears with probability p_i
- Words appear randomly according to this distribution (obviously not true...)
- Words = composition of letters is important
- Alphabet contains m letters
- Words are ordered by length (shortest first)

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Word Cost

- Length of word (plus a space)
- Word length was irrelevant for Simon's method

Objection

- Real words don't use all letter sequences

Objections to Objection

- Maybe real words roughly follow this pattern (?)
- Words can be encoded this way
- Na na na-na naaaaa...

Zipfarama via Optimization

Binary alphabet plus a space symbol

i	1	2	3	4	5	6	7	8
word	1	10	11	100	101	110	111	1000
length	1	2	2	3	3	3	3	4
$1 + \ln_2 i$	1	2	2.58	3	3.32	3.58	3.81	4

- Word length of 2^kth word: $= k + 1$
- Word length of ith word $\simeq 1 + \log_2 i$
- For an alphabet with m letters,
  word length of ith word $\simeq 1 + \log_m i$.

Zipfarama via Optimization

Binary alphabet plus a space symbol

i	1	2	3	4	5	6	7	8
word	1	10	11	100	101	110	111	1000
length	1	2	2	3	3	3	3	4
$1 + \ln_2 i$	1	2	2.58	3	3.32	3.58	3.81	4

- Word length of 2^kth word: $= k + 1$
- Word length of ith word $\simeq 1 + \log_2 i$
- For an alphabet with m letters, word length of ith word $\simeq 1 + \log_m i$.

Zipfarama via Optimization

Binary alphabet plus a space symbol

i	1	2	3	4	5	6	7	8
word	1	10	11	100	101	110	111	1000
length	1	2	2	3	3	3	3	4
$1 + \ln_2 i$	1	2	2.58	3	3.32	3.58	3.81	4

- Word length of 2^kth word: $= k + 1 = 1 + \log_2 2^k$
- Word length of ith word $\simeq 1 + \log_2 i$
- For an alphabet with m letters,
  word length of ith word $\simeq 1 + \log_m i$.

Zipfarama via Optimization

Binary alphabet plus a space symbol

i	1	2	3	4	5	6	7	8
word	1	10	11	100	101	110	111	1000
length	1	2	2	3	3	3	3	4
$1 + \ln_2 i$	1	2	2.58	3	3.32	3.58	3.81	4

- Word length of 2^kth word: $= k + 1 = 1 + \log_2 2^k$
- Word length of ith word $\simeq 1 + \log_2 i$
- For an alphabet with m letters, word length of ith word $\simeq 1 + \log_m i$.

Zipfarama via Optimization

Binary alphabet plus a space symbol

i	1	2	3	4	5	6	7	8
word	1	10	11	100	101	110	111	1000
length	1	2	2	3	3	3	3	4
$1 + \ln_2 i$	1	2	2.58	3	3.32	3.58	3.81	4

- Word length of 2^kth word: $= k + 1 = 1 + \log_2 2^k$
- Word length of ith word $\simeq 1 + \log_2 i$
- For an alphabet with m letters, word length of ith word $\simeq 1 + \log_m i$.

Outline

Zipfarama via Optimization

Total Cost C

- Cost of the ith word: $C_i \simeq 1 + \log_m i$
- Cost of the ith word plus space: $C_i \simeq 1 + \log_m(i+1)$
- Subtract fixed cost: $C'_i = C_i - 1 \simeq \log_m(i+1)$
- Simplify base of logarithm:

$$C'_i \simeq \log_m(i+1) = \frac{\log_e(i+1)}{\log_e m}$$

- Total Cost:

$$C \sim \sum_{i=1}^{n} p_i C'_i \propto \sum_{i=1}^{n} p_i \ln(i+1)$$

Zipfarama via Optimization

Total Cost C

- Cost of the ith word: $C_i \simeq 1 + \log_m i$
- Cost of the ith word plus space: $C_i \simeq 1 + \log_m(i+1)$
- Subtract fixed cost: $C'_i = C_i - 1 \simeq \log_m(i+1)$
- Simplify base of logarithm:

$$C'_i \simeq \log_m(i+1) = \frac{\log_e(i+1)}{\log_e m}$$

- Total Cost:

$$C \sim \sum_{i=1}^{n} p_i C'_i \propto \sum_{i=1}^{n} p_i \ln(i+1)$$

Zipfarama via Optimization

Total Cost C

- Cost of the ith word: $C_i \simeq 1 + \log_m i$
- Cost of the ith word plus space: $C_i \simeq 1 + \log_m(i+1)$
- Subtract fixed cost: $C'_i = C_i - 1 \simeq \log_m(i+1)$
- Simplify base of logarithm:

$$C'_i \simeq \log_m(i+1) = \frac{\log_e(i+1)}{\log_e m}$$

- Total Cost:

$$C \sim \sum_{i=1}^{n} p_i C'_i \propto \sum_{i=1}^{n} p_i \ln(i+1)$$

Zipfarama via Optimization

Total Cost C

- Cost of the ith word: $C_i \simeq 1 + \log_m i$
- Cost of the ith word plus space: $C_i \simeq 1 + \log_m(i + 1)$
- Subtract fixed cost: $C'_i = C_i - 1 \simeq \log_m(i + 1)$
- Simplify base of logarithm:

$$C'_i \simeq \log_m(i + 1) = \frac{\log_e(i + 1)}{\log_e m}$$

- Total Cost:

$$C \sim \sum_{i=1}^{n} p_i C'_i \propto \sum_{i=1}^{n} p_i \ln(i + 1)$$

Zipfarama via Optimization

Total Cost C

- Cost of the ith word: $C_i \simeq 1 + \log_m i$
- Cost of the ith word plus space: $C_i \simeq 1 + \log_m(i+1)$
- Subtract fixed cost: $C'_i = C_i - 1 \simeq \log_m(i+1)$
- Simplify base of logarithm:

$$C'_i \simeq \log_m(i+1) = \frac{\log_e(i+1)}{\log_e m} \propto \ln(i+1)$$

- Total Cost:

$$C \sim \sum_{i=1}^{n} p_i C'_i \propto \sum_{i=1}^{n} p_i \ln(i+1)$$

Zipfarama via Optimization

Total Cost C

- Cost of the ith word: $C_i \simeq 1 + \log_m i$
- Cost of the ith word plus space: $C_i \simeq 1 + \log_m(i + 1)$
- Subtract fixed cost: $C'_i = C_i - 1 \simeq \log_m(i + 1)$
- Simplify base of logarithm:

$$C'_i \simeq \log_m(i + 1) = \frac{\log_e(i + 1)}{\log_e m} \propto \ln(i + 1)$$

- Total Cost:

$$C \sim \sum_{i=1}^{n} p_i C'_i \propto \sum_{i=1}^{n} p_i \ln(i + 1)$$

Zipfarama via Optimization

Information Measure

- Use Shannon's Entropy (or Uncertainty):

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- (allegedly) von Neumann suggested 'entropy'...
- Proportional to average number of bits needed to encode each 'word' based on frequency of occurrence
- $-\log_2 p_i = \log_2 1/p_i$ = minimum number of bits needed to distinguish event i from all others
- If $p_i = 1/2$, need only 1 bit ($log_2 1/p_i = 1$)
- If $p_i = 1/64$, need 6 bits ($log_2 1/p_i = 6$)

Zipfarama via Optimization

Information Measure

- Use Shannon's Entropy (or Uncertainty):

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- (allegedly) von Neumann suggested 'entropy'...
- Proportional to average number of bits needed to encode each 'word' based on frequency of occurrence
- $-\log_2 p_i = \log_2 1/p_i$ = minimum number of bits needed to distinguish event i from all others
- If $p_i = 1/2$, need only 1 bit ($log_2 1/p_i = 1$)
- If $p_i = 1/64$, need 6 bits ($log_2 1/p_i = 6$)

Zipfarama via Optimization

Information Measure

- Use Shannon's Entropy (or Uncertainty):

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- (allegedly) von Neumann suggested 'entropy'...
- Proportional to average number of bits needed to encode each 'word' based on frequency of occurrence
- $-\log_2 p_i = \log_2 1/p_i$ = minimum number of bits needed to distinguish event i from all others
- If $p_i = 1/2$, need only 1 bit ($log_2 1/p_i = 1$)
- If $p_i = 1/64$, need 6 bits ($log_2 1/p_i = 6$)

Zipfarama via Optimization

Information Measure

- Use Shannon's Entropy (or Uncertainty):

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- (allegedly) von Neumann suggested 'entropy'...
- Proportional to average number of bits needed to encode each 'word' based on frequency of occurrence
- $-\log_2 p_i = \log_2 1/p_i$ = minimum number of bits needed to distinguish event i from all others
- If $p_i = 1/2$, need only 1 bit ($log_2 1/p_i = 1$)
- If $p_i = 1/64$, need 6 bits ($log_2 1/p_i = 6$)

Zipfarama via Optimization

Information Measure

- Use Shannon's Entropy (or Uncertainty):

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- (allegedly) von Neumann suggested 'entropy'...
- Proportional to average number of bits needed to encode each 'word' based on frequency of occurrence
- $-\log_2 p_i = \log_2 1/p_i$ = minimum number of bits needed to distinguish event i from all others
- If $p_i = 1/2$, need only 1 bit ($log_2 1/p_i = 1$)
- If $p_i = 1/64$, need 6 bits ($log_2 1/p_i = 6$)

Zipfarama via Optimization

Information Measure

- Use Shannon's Entropy (or Uncertainty):

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

- (allegedly) von Neumann suggested 'entropy'...
- Proportional to average number of bits needed to encode each 'word' based on frequency of occurrence
- $-\log_2 p_i = \log_2 1/p_i$ = minimum number of bits needed to distinguish event i from all others
- If $p_i = 1/2$, need only 1 bit ($log_2 1/p_i = 1$)
- If $p_i = 1/64$, need 6 bits ($log_2 1/p_i = 6$)

Zipfarama via Optimization

Information Measure

- Use a slightly simpler form:

$$H = -\sum_{i=1}^{n} p_i \log_e p_i / \log_e 2 = -g \sum_{i=1}^{n} p_i \ln p_i$$

where $g = 1/\ln 2$

Zipfarama via Optimization

Information Measure

- Use a slightly simpler form:

$$H = -\sum_{i=1}^{n} p_i \log_e p_i / \log_e 2 = -g \sum_{i=1}^{n} p_i \ln p_i$$

where $g = 1/\ln 2$

Zipfarama via Optimization

- Minimize

$$F(p_1, p_2, \ldots, p_n) = C/H$$

subject to constraint

$$\sum_{i=1}^{n} p_i = 1$$

- Tension:
(1) Shorter words are cheaper

- (Good) question: how much does choice of C/H as function to minimize affect things?

Zipfarama via Optimization

- Minimize

$$F(p_1, p_2, \ldots, p_n) = C/H$$

subject to constraint

$$\sum_{i=1}^{n} p_i = 1$$

- Tension:
  (1) Shorter words are cheaper

- (Good) question: how much does choice of C/H as function to minimize affect things?

Zipfarama via Optimization

- Minimize
$$F(p_1, p_2, \ldots, p_n) = C/H$$
subject to constraint
$$\sum_{i=1}^{n} p_i = 1$$
- Tension:
(1) Shorter words are cheaper
(2) Longer words are more informative (rarer)
- (Good) question: how much does choice of C/H as function to minimize affect things?

Zipfarama via Optimization

- Minimize

$$F(p_1, p_2, \dots, p_n) = C/H$$

subject to constraint

$$\sum_{i=1}^{n} p_i = 1$$

- Tension:
  (1) Shorter words are cheaper
  (2) Longer words are more informative (rarer)
- (Good) question: how much does choice of C/H as function to minimize affect things?

Outline

Zipfarama via Optimization

Time for Lagrange Multipliers:

- Minimize

$$\Psi(p_1, p_2, \ldots, p_n) =$$
$$F(p_1, p_2, \ldots, p_n) + \lambda G(p_1, p_2, \ldots, p_n)$$

where

$$F(p_1, p_2, \ldots, p_n) = \frac{C}{H} = \frac{\sum_{i=1}^{n} p_i \ln(i+1)}{-g \sum_{i=1}^{n} p_i \ln p_i}$$

and the constraint function is

$$G(p_1, p_2, \ldots, p_n) = \sum_{i=1}^{n} p_i - 1 = 0$$

Zipfarama via Optimization

Time for Lagrange Multipliers:

- Minimize

$$\Psi(p_1, p_2, \ldots, p_n) =$$

$$F(p_1, p_2, \ldots, p_n) + \lambda G(p_1, p_2, \ldots, p_n)$$

where

$$F(p_1, p_2, \ldots, p_n) = \frac{C}{H} = \frac{\sum_{i=1}^{n} p_i \ln(i+1)}{-g \sum_{i=1}^{n} p_i \ln p_i}$$

and the constraint function is

$$G(p_1, p_2, \ldots, p_n) = \sum_{i=1}^{n} p_i - 1 = 0$$

Zipfarama via Optimization

Time for Lagrange Multipliers:

- Minimize

$$\Psi(p_1, p_2, \ldots, p_n) =$$

$$F(p_1, p_2, \ldots, p_n) + \lambda G(p_1, p_2, \ldots, p_n)$$

where

$$F(p_1, p_2, \ldots, p_n) = \frac{C}{H} = \frac{\sum_{i=1}^{n} p_i \ln(i+1)}{-g \sum_{i=1}^{n} p_i \ln p_i}$$

and the constraint function is

$$G(p_1, p_2, \ldots, p_n) = \sum_{i=1}^{n} p_i - 1 = 0$$

Insert question 4, assignment 2 (⊞)

Zipfarama via Optimization

Some mild suffering leads to:

- $$p_j = e^{-1-\lambda H^2/gC}(j+1)^{-H/gC} \propto (j+1)^{-H/gC}$$
- A power law appears [applause]: $\alpha = H/gC$
- Next: sneakily deduce λ in terms of g, C, and H.
- Find
$$p_j = (j+1)^{-H/gC}$$

Zipfarama via Optimization

Some mild suffering leads to:

- $$p_j = e^{-1-\lambda H^2/gC}(j+1)^{-H/gC} \propto (j+1)^{-H/gC}$$
- A power law appears [applause]: $\alpha = H/gC$
- Next: sneakily deduce λ in terms of g, C, and H.
- Find

$$p_j = (j+1)^{-H/gC}$$

Zipfarama via Optimization

Some mild suffering leads to:

- $$p_j = e^{-1-\lambda H^2/gC}(j+1)^{-H/gC} \propto (j+1)^{-H/gC}$$
- A power law appears [applause]: $\alpha = H/gC$
- Next: sneakily deduce λ in terms of g, C, and H.
- Find
$$p_j = (j+1)^{-H/gC}$$

Zipfarama via Optimization

Some mild suffering leads to:

- $$p_j = e^{-1-\lambda H^2/gC}(j+1)^{-H/gC} \propto (j+1)^{-H/gC}$$
- A power law appears [applause]: $\boxed{\alpha = H/gC}$
- Next: sneakily deduce λ in terms of g, C, and H.
- Find
$$p_j = (j+1)^{-H/gC}$$

Zipfarama via Optimization

Some mild suffering leads to:

- $$p_j = e^{-1-\lambda H^2/gC}(j+1)^{-H/gC} \propto (j+1)^{-H/gC}$$
- A power law appears [applause]: $\boxed{\alpha = H/gC}$
- Next: sneakily deduce λ in terms of g, C, and H.
- Find
$$p_j = (j+1)^{-H/gC}$$

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \to \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j + 1 \to j + a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \rightarrow \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j + 1 \rightarrow j + a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \to \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j + 1 \to j + a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \to \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j + 1 \to j + a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \rightarrow \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j+1 \rightarrow j+a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \to \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j + 1 \to j + a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

Finding the exponent

- Now use the normalization constraint:

$$1 = \sum_{j=1}^{n} p_j = \sum_{j=1}^{n} (j+1)^{-H/gC} = \sum_{j=1}^{n} (j+1)^{-\alpha}$$

- As $n \to \infty$, we end up with $\zeta(H/gC) = 2$ where ζ is the Riemann Zeta Function
- Gives $\alpha \simeq 1.73$ (> 1, too high)
- If cost function changes ($j + 1 \to j + a$) then exponent is tunable
- Increase a, decrease α

Zipfarama via Optimization

All told:

- Reasonable approach: Optimization is at work in evolutionary processes
- But optimization can involve many incommensurate elements: monetary cost, robustness, happiness,...
- Mandelbrot's argument is not super convincing
- Exponent depends too much on a loose definition of cost

Zipfarama via Optimization

All told:

- Reasonable approach: Optimization is at work in evolutionary processes
- But optimization can involve many incommensurate elements: monetary cost, robustness, happiness,...
- Mandelbrot's argument is not super convincing
- Exponent depends too much on a loose definition of cost

Zipfarama via Optimization

All told:

- Reasonable approach: Optimization is at work in evolutionary processes
- But optimization can involve many incommensurate elements: monetary cost, robustness, happiness,...
- Mandelbrot's argument is not super convincing
- Exponent depends too much on a loose definition of cost

Zipfarama via Optimization

All told:

- Reasonable approach: Optimization is at work in evolutionary processes
- But optimization can involve many incommensurate elements: monetary cost, robustness, happiness,...
- Mandelbrot's argument is not super convincing
- Exponent depends too much on a loose definition of cost

Reconciling Mandelbrot and Simon

- Mixture of local optimization and randomness
- Numerous efforts...

1. Carlson and Doyle, 1999:
   Highly Optimized Tolerance
   (HOT)—Evolved/Engineered Robustness [5]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [8]
3. D'Souza et al., 2007:
   Scale-free networks [7]

More

Reconciling Mandelbrot and Simon

- Mixture of local optimization and randomness
- Numerous efforts...

1. Carlson and Doyle, 1999:
   Highly Optimized Tolerance
   (HOT)—Evolved/Engineered Robustness [5]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [8]
3. D'Souza et al., 2007:
   Scale-free networks [7]

More

Reconciling Mandelbrot and Simon

- Mixture of local optimization and randomness
- Numerous efforts...

1. Carlson and Doyle, 1999:
   Highly Optimized Tolerance
   (HOT)—Evolved/Engineered Robustness [5]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [8]
3. D'Souza et al., 2007:
   Scale-free networks [7]

More

Reconciling Mandelbrot and Simon

- Mixture of local optimization and randomness
- Numerous efforts...

1. Carlson and Doyle, 1999:
   Highly Optimized Tolerance
   (HOT)—Evolved/Engineered Robustness [5]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [8]
3. D'Souza et al., 2007:
   Scale-free networks [7]

More

Other mechanisms:

Much argument about whether or not monkeys typing could produce Zipf's law... (Miller, 1957) [12]

Outline

Others are also not happy

Krugman and Simon

- "The Self-Organizing Economy" (Paul Krugman, 1995) [10]
- Krugman touts Zipf's law for cities, Simon's model
- "Déjà vu, Mr. Krugman" (Berry, 1999)
- Substantial work done by Urban Geographers

Others are also not happy

Krugman and Simon

- "The Self-Organizing Economy" (Paul Krugman, 1995) [10]
- Krugman touts Zipf's law for cities, Simon's model
- "Déjà vu, Mr. Krugman" (Berry, 1999)
- Substantial work done by Urban Geographers

Others are also not happy

Krugman and Simon

- "The Self-Organizing Economy" (Paul Krugman, 1995) [10]
- Krugman touts Zipf's law for cities, Simon's model
- "Déjà vu, Mr. Krugman" (Berry, 1999)
- Substantial work done by Urban Geographers

Others are also not happy

Krugman and Simon

- “The Self-Organizing Economy” (Paul Krugman, 1995) [10]
- Krugman touts Zipf’s law for cities, Simon’s model
- “Déjà vu, Mr. Krugman” (Berry, 1999)
- Substantial work done by Urban Geographers

Who needs a hug?

From Berry [4]

- Déjà vu, Mr. Krugman. Been there, done that. The Simon-Ijiri model was introduced to geographers in 1958 as an explanation of city size distributions, the first of many such contributions dealing with the steady states of random growth processes, ...
- But then, I suppose, even if Krugman had known about these studies, they would have been discounted because they were not written by professional economists or published in one of the top five journals in economics!

Who needs a hug?

From Berry [4]

- Déjà vu, Mr. Krugman. Been there, done that. The Simon-Ijiri model was introduced to geographers in 1958 as an explanation of city size distributions, the first of many such contributions dealing with the steady states of random growth processes, ...
- But then, I suppose, even if Krugman had known about these studies, they would have been discounted because they were not written by professional economists or published in one of the top five journals in economics!

Who needs a hug?

From Berry [4]

- ... [Krugman] needs to exercise some humility, for his world view is circumscribed by folkways that militate against recognition and acknowledgment of scholarship beyond his disciplinary frontier.
- Urban geographers, thank heavens, are not so afflicted.

Who needs a hug?

From Berry [4]

- ... [Krugman] needs to exercise some humility, for his world view is circumscribed by folkways that militate against recognition and acknowledgment of scholarship beyond his disciplinary frontier.
- Urban geographers, thank heavens, are not so afflicted.

Outline

Robustness

- Many complex systems are prone to cascading catastrophic failure:
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure:
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure:
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure:
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure:
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure: exciting!!!
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure: exciting!!!
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure: exciting!!!
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness (not as exciting but important...)
- Robustness and Failure may be a power-law story...

Robustness

- Many complex systems are prone to cascading catastrophic failure: exciting!!!
  - Blackouts
  - Disease outbreaks
  - Wildfires
  - Earthquakes
- But complex systems also show persistent robustness (not as exciting but important...)
- Robustness and Failure may be a power-law story...

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: 'Highly Optimized Tolerance' (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: 'Highly Optimized Tolerance' (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: 'Highly Optimized Tolerance' (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: 'Highly Optimized Tolerance' (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: 'Highly Optimized Tolerance' (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: 'Highly Optimized Tolerance' (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

- System robustness may result from
  1. Evolutionary processes
  2. Engineering/Design
- Idea: Explore systems optimized to perform under uncertain conditions.
- The handle: ‘Highly Optimized Tolerance’ (HOT) [5, 6, 15]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle

Robustness

Features of HOT systems: [6]

- High performance and robustness
- Designed/evolved to handle known stochastic environmental variability
- Fragile in the face of unpredicted environmental signals
- Highly specialized, low entropy configurations
- Power-law distributions appear (of course...)

Robustness

Features of HOT systems: [6]

- High performance and robustness
- Designed/evolved to handle known stochastic environmental variability
- Fragile in the face of unpredicted environmental signals
- Highly specialized, low entropy configurations
- Power-law distributions appear (of course...)

Robustness

Features of HOT systems: [6]

- High performance and robustness
- Designed/evolved to handle known stochastic environmental variability
- Fragile in the face of unpredicted environmental signals
- Highly specialized, low entropy configurations
- Power-law distributions appear (of course...)

Robustness

Features of HOT systems: [6]

- High performance and robustness
- Designed/evolved to handle known stochastic environmental variability
- Fragile in the face of unpredicted environmental signals
- Highly specialized, low entropy configurations
- Power-law distributions appear (of course...)

Robustness

Features of HOT systems: [6]

- High performance and robustness
- Designed/evolved to handle known stochastic environmental variability
- Fragile in the face of unpredicted environmental signals
- Highly specialized, low entropy configurations
- Power-law distributions appear (of course...)

Robustness

Features of HOT systems: [6]

- High performance and robustness
- Designed/evolved to handle known stochastic environmental variability
- Fragile in the face of unpredicted environmental signals
- Highly specialized, low entropy configurations
- Power-law distributions appear (of course...)

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good
- X has a characteristic size but Y does not

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good
- X has a characteristic size but Y does not

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good
- X has a characteristic size but Y does not

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good
- X has a characteristic size but Y does not

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good
- X has a characteristic size but Y does not

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good: Mild In, Wild Out
- X has a characteristic size but Y does not

Robustness

HOT combines things we've seen:

- Variable transformation
- Constrained optimization

- Need power law transformation between variables: $(Y = X^{-\alpha})$
- Recall PLIPLO is bad...
- MIWO is good: Mild In, Wild Out
- X has a characteristic size but Y does not

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location according to some distribution P_{ij}
- Fires spread from tree to tree (nearest neighbor only)
- Connected clusters of trees burn completely
- Empty sites block fire
- Best case scenario:
  Build firebreaks to maximize average # trees left intact

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Forest fire example: [6]

- Build a forest by adding one tree at a time
- Test D ways of adding one tree
- D = design parameter
- Average over P_{ij} = spark probability
- $D = 1$: random addition
- $D = N^2$: test all possibilities

Measure average area of forest left untouched

- $f(c)$ = distribution of fire sizes c (= cost)
- Yield = $Y = \rho - \langle f \rangle$

Robustness

Specifics:

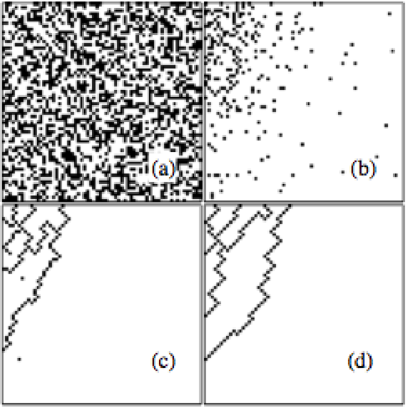
- $$P_{ij} = P_{i;a_x,b_x} P_{j;a_y,b_y}$$

  where

  $$P_{i;a,b} \propto e^{-[(i+a)/b]^2}$$

- In the original work, $b_y > b_x$
- Distribution has more width in y direction.

HOT Forests



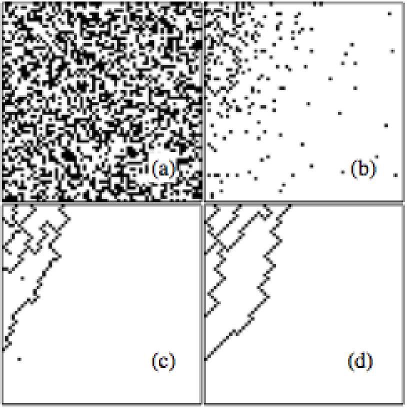
[6]

$N = 64$

(a) $D = 1$
(b) $D = 2$
(c) $D = N$
(d) $D = N^2$

P_{ij} has a Gaussian decay

HOT Forests



[6]

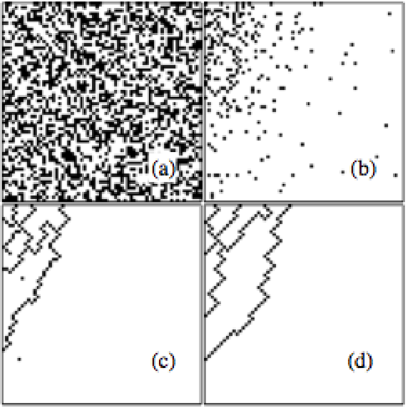
Optimized forests do well on average

$N = 64$

(a) $D = 1$
(b) $D = 2$
(c) $D = N$
(d) $D = N^2$

P_{ij} has a Gaussian decay

HOT Forests



[6]

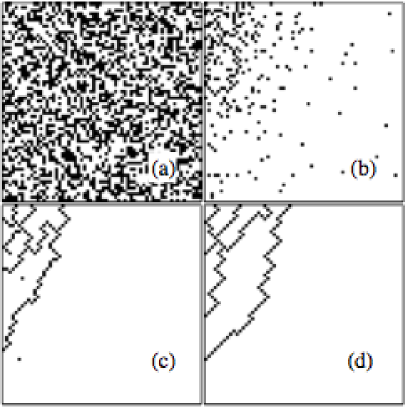
Optimized forests do well on average but rare extreme events occur

$N = 64$

(a) $D = 1$
(b) $D = 2$
(c) $D = N$
(d) $D = N^2$

P_{ij} has a Gaussian decay

HOT Forests



[6]

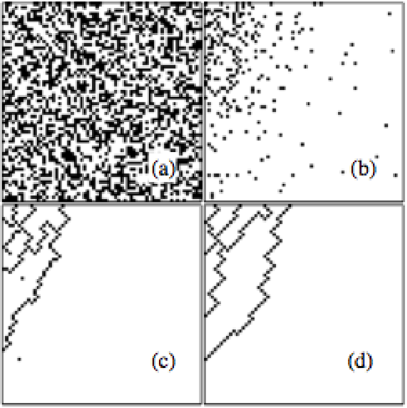
$N = 64$

(a) $D = 1$
(b) $D = 2$
(c) $D = N$
(d) $D = N^2$

P_{ij} has a Gaussian decay

Optimized forests do well on average (robustness) but rare extreme events occur

HOT Forests



[6]

$N = 64$

(a) $D = 1$
(b) $D = 2$
(c) $D = N$
(d) $D = N^2$

P_{ij} has a Gaussian decay

Optimized forests do well on average (robustness) but rare extreme events occur (fragility)

HOT Forests

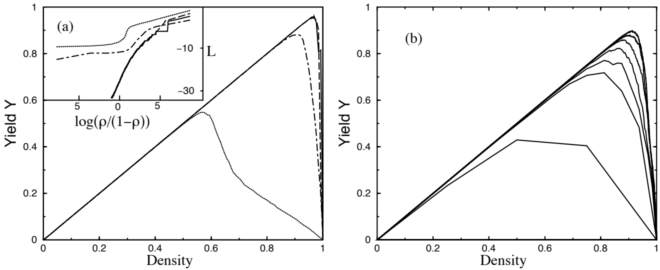


FIG. 2. Yield vs density $Y(\rho)$: (a) for design parameters $D = 1$ (dotted curve), 2 (dot-dashed), N (long dashed), and N^2 (solid) with $N = 64$, and (b) for $D = 2$ and $N = 2, 2^2, \ldots, 2^7$ running from the bottom to top curve. The results have been averaged over 100 runs. The inset to (a) illustrates corresponding loss functions $L = \log[\langle f \rangle/(1 - \langle f \rangle)]$, on a scale which more clearly differentiates between the curves.

[6]

HOT Forests

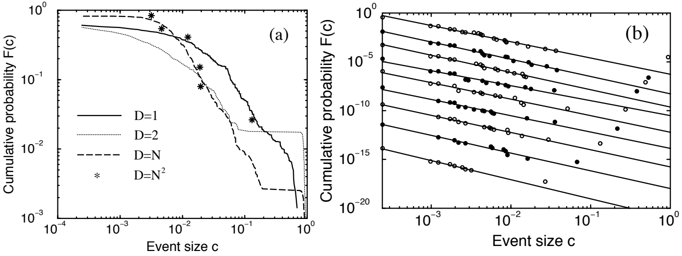


FIG. 3. Cumulative distributions of events $F(c)$: (a) at peak yield for $D = 1, 2, N$, and N^2 with $N = 64$, and (b) for $D = N^2$, and $N = 64$ at equal density increments of 0.1, ranging at $\rho = 0.1$ (bottom curve) to $\rho = 0.9$ (top curve).

[6]

Random Forests

$D = 1$: Random forests = Percolation [16]

- Randomly add trees
- Below critical density ρ_c, no fires take off
- Above critical density ρ_c, percolating cluster of trees burns
- Only at ρ_c, the critical density, is there a power-law distribution of tree cluster sizes
- Forest is random and featureless

Random Forests

$D = 1$: Random forests = Percolation [16]

- Randomly add trees
- Below critical density ρ_c, no fires take off
- Above critical density ρ_c, percolating cluster of trees burns
- Only at ρ_c, the critical density, is there a power-law distribution of tree cluster sizes
- Forest is random and featureless

Random Forests

$D = 1$: Random forests = Percolation [16]

- Randomly add trees
- Below critical density ρ_c, no fires take off
- Above critical density ρ_c, percolating cluster of trees burns
- Only at ρ_c, the critical density, is there a power-law distribution of tree cluster sizes
- Forest is random and featureless

Random Forests

$D = 1$: Random forests = Percolation [16]

- Randomly add trees
- Below critical density ρ_c, no fires take off
- Above critical density ρ_c, percolating cluster of trees burns
- Only at ρ_c, the critical density, is there a power-law distribution of tree cluster sizes
- Forest is random and featureless

Random Forests

$D = 1$: Random forests = Percolation [16]

- Randomly add trees
- Below critical density ρ_c, no fires take off
- Above critical density ρ_c, percolating cluster of trees burns
- Only at ρ_c, the critical density, is there a power-law distribution of tree cluster sizes
- Forest is random and featureless

HOT forests

HOT forests

- Highly structured
- Power law distribution of tree cluster sizes for $\rho > \rho_c$
- No specialness of ρ_c
- Forest states are tolerant
- Uncertainty is okay if well characterized
- If P_{ij} is characterized poorly, failure becomes highly likely

HOT forests

HOT forests

- Highly structured
- Power law distribution of tree cluster sizes for $\rho > \rho_c$
- No specialness of ρ_c
- Forest states are tolerant
- Uncertainty is okay if well characterized
- If P_{ij} is characterized poorly, failure becomes highly likely

HOT forests

HOT forests

- Highly structured
- Power law distribution of tree cluster sizes for $\rho > \rho_c$
- No specialness of ρ_c
- Forest states are tolerant
- Uncertainty is okay if well characterized
- If P_{ij} is characterized poorly, failure becomes highly likely

HOT forests

HOT forests

- Highly structured
- Power law distribution of tree cluster sizes for $\rho > \rho_c$
- No specialness of ρ_c
- Forest states are tolerant
- Uncertainty is okay if well characterized
- If P_{ij} is characterized poorly, failure becomes highly likely

HOT forests

HOT forests

- Highly structured
- Power law distribution of tree cluster sizes for $\rho > \rho_c$
- No specialness of ρ_c
- Forest states are tolerant
- Uncertainty is okay if well characterized
- If P_{ij} is characterized poorly, failure becomes highly likely

HOT forests

HOT forests

- Highly structured
- Power law distribution of tree cluster sizes for $\rho > \rho_c$
- No specialness of ρ_c
- Forest states are tolerant
- Uncertainty is okay if well characterized
- If P_{ij} is characterized poorly, failure becomes highly likely

HOT theory

The abstract story:

- Given $y_i = x_i^{-\alpha}$, $i = 1, \ldots, N_{\text{sites}}$
- Design system to minimize $\langle y \rangle$ subject to a constraint on the x_i
- Minimize cost:

$$C = \sum_{i=1}^{N_{\text{sites}}} Pr(y_i) y_i$$

Subject to $\sum_{i=1}^{N_{\text{sites}}} x_i = \text{constant}$

- Drag out the Lagrange Multipliers, battle away and find:

$$p_i \propto y_i^{-\gamma}$$

HOT theory

The abstract story:

- Given $y_i = x_i^{-\alpha}$, $i = 1, \ldots, N_{\text{sites}}$
- Design system to minimize $\langle y \rangle$ subject to a constraint on the x_i
- Minimize cost:

$$C = \sum_{i=1}^{N_{\text{sites}}} Pr(y_i) y_i$$

Subject to $\sum_{i=1}^{N_{\text{sites}}} x_i = \text{constant}$

- Drag out the Lagrange Multipliers, battle away and find:

$$p_i \propto y_i^{-\gamma}$$

HOT theory

The abstract story:

- Given $y_i = x_i^{-\alpha}$, $i = 1, \ldots, N_{\text{sites}}$
- Design system to minimize $\langle y \rangle$ subject to a constraint on the x_i
- Minimize cost:

$$C = \sum_{i=1}^{N_{\text{sites}}} Pr(y_i) y_i$$

Subject to $\sum_{i=1}^{N_{\text{sites}}} x_i = \text{constant}$

- Drag out the Lagrange Multipliers, battle away and find:

$$p_i \propto y_i^{-\gamma}$$

HOT theory

The abstract story:

- Given $y_i = x_i^{-\alpha}$, $i = 1, \ldots, N_{\text{sites}}$
- Design system to minimize $\langle y \rangle$ subject to a constraint on the x_i
- Minimize cost:

$$C = \sum_{i=1}^{N_{\text{sites}}} Pr(y_i) y_i$$

Subject to $\sum_{i=1}^{N_{\text{sites}}} x_i = \text{constant}$

- Drag out the Lagrange Multipliers, battle away and find:

$$p_i \propto y_i^{-\gamma}$$

HOT theory

The abstract story:

- Given $y_i = x_i^{-\alpha}$, $i = 1, \ldots, N_{\text{sites}}$
- Design system to minimize $\langle y \rangle$ subject to a constraint on the x_i
- Minimize cost:

$$C = \sum_{i=1}^{N_{\text{sites}}} Pr(y_i) y_i$$

Subject to $\sum_{i=1}^{N_{\text{sites}}} x_i = \text{constant}$

- Drag out the Lagrange Multipliers, battle away and find:

$$p_i \propto y_i^{-\gamma}$$

HOT: Optimal fire walls in *d* dimensions

Two costs:

1. Expected size of fire

$$C_{\text{fire}} \propto \sum_{i=1}^{N_{\text{sites}}} (p_i a_i) a_i = \sum_{i=1}^{N_{\text{sites}}} p_i a_i^2$$

- a_i = area of ith site's region
- p_i = avg. prob. of fire at site in ith site's region
- N_{sites} = total number of sites

2. Cost of building and maintaining firewalls

$$C_{\text{firewalls}} \propto \sum_{i=1}^{N_{\text{sites}}} a_i^{1/2}$$

- We are assuming isometry.
- In d dimensions, 1/2 is replaced by $(d-1)/d$

HOT: Optimal fire walls in d dimensions

Two costs:

1. Expected size of fire

$$C_{\text{fire}} \propto \sum_{i=1}^{N_{\text{sites}}} (p_i a_i) a_i = \sum_{i=1}^{N_{\text{sites}}} p_i a_i^2$$

- a_i = area of ith site's region
- p_i = avg. prob. of fire at site in ith site's region
- N_{sites} = total number of sites

2. Cost of building and maintaining firewalls

$$C_{\text{firewalls}} \propto \sum_{i=1}^{N_{\text{sites}}} a_i^{1/2}$$

- We are assuming isometry.
- In d dimensions, 1/2 is replaced by $(d-1)/d$

HOT: Optimal fire walls in d dimensions

Two costs:

1. Expected size of fire

$$C_{\text{fire}} \propto \sum_{i=1}^{N_{\text{sites}}} (p_i a_i) a_i = \sum_{i=1}^{N_{\text{sites}}} p_i a_i^2$$

- a_i = area of ith site's region
- p_i = avg. prob. of fire at site in ith site's region
- N_{sites} = total number of sites

2. Cost of building and maintaining firewalls

$$C_{\text{firewalls}} \propto \sum_{i=1}^{N_{\text{sites}}} a_i^{1/2}$$

- We are assuming isometry.
- In d dimensions, 1/2 is replaced by $(d-1)/d$

HOT: Optimal fire walls in d dimensions

Two costs:

1. Expected size of fire

$$C_{\text{fire}} \propto \sum_{i=1}^{N_{\text{sites}}} (p_i a_i) a_i = \sum_{i=1}^{N_{\text{sites}}} p_i a_i^2$$

- a_i = area of ith site's region
- p_i = avg. prob. of fire at site in ith site's region
- N_{sites} = total number of sites

2. Cost of building and maintaining firewalls

$$C_{\text{firewalls}} \propto \sum_{i=1}^{N_{\text{sites}}} a_i^{1/2}$$

- We are assuming isometry.
- In d dimensions, 1/2 is replaced by $(d-1)/d$

HOT: Optimal fire walls in d dimensions

Two costs:

1. Expected size of fire

$$C_{\text{fire}} \propto \sum_{i=1}^{N_{\text{sites}}} (p_i a_i) a_i = \sum_{i=1}^{N_{\text{sites}}} p_i a_i^2$$

- a_i = area of ith site's region
- p_i = avg. prob. of fire at site in ith site's region
- N_{sites} = total number of sites

2. Cost of building and maintaining firewalls

$$C_{\text{firewalls}} \propto \sum_{i=1}^{N_{\text{sites}}} a_i^{1/2}$$

- We are assuming isometry.
- In d dimensions, 1/2 is replaced by $(d-1)/d$

HOT: Optimal fire walls in d dimensions

Two costs:

1. Expected size of fire

$$C_{\text{fire}} \propto \sum_{i=1}^{N_{\text{sites}}} (p_i a_i) a_i = \sum_{i=1}^{N_{\text{sites}}} p_i a_i^2$$

- a_i = area of ith site's region
- p_i = avg. prob. of fire at site in ith site's region
- N_{sites} = total number of sites

2. Cost of building and maintaining firewalls

$$C_{\text{firewalls}} \propto \sum_{i=1}^{N_{\text{sites}}} a_i^{1/2}$$

- We are assuming isometry.
- In d dimensions, 1/2 is replaced by $(d-1)/d$

HOT theory

Third constraint:

- Total area is constrained:

$$\sum_{i=1}^{N_{\text{sites}}} \frac{1}{a_i} = N_{\text{regions}}$$

  where N_{regions} = number of cells.
- Can ignore in calculation...

HOT theory

Third constraint:

- Total area is constrained:

$$\sum_{i=1}^{N_{\text{sites}}} \frac{1}{a_i} = N_{\text{regions}}$$

  where N_{regions} = number of cells.
- Can ignore in calculation...

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(1+1/d)}$$
- $$\text{For } d = 2, \gamma = 3/2$$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(1+1/d)}$$
- $$\text{For } d = 2, \gamma = 3/2$$

HOT theory

- Minimize $C_{\rm fire}$ given $C_{\rm firewalls} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\rm fire} - \lambda C_{\rm firewalls}\right)$$
  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(1+1/d)}$$
- $$\text{For } d = 2, \gamma = 3/2$$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
$$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(1+1/d)}$$
- $$\text{For } d = 2, \gamma = 3/2$$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(1+1/d)}$$
- $$\text{For } d = 2, \gamma = 3/2$$

HOT theory

Summary of designed tolerance

- Build more firewalls in areas where sparks are likely
- Small connected regions in high-danger areas
- Large connected regions in low-danger areas
- Routinely see many small outbreaks (robust)
- Rarely see large outbreaks (fragile)
- Sensitive to changes in the environment (P_{ij})

HOT theory

Summary of designed tolerance

- Build more firewalls in areas where sparks are likely
- Small connected regions in high-danger areas
- Large connected regions in low-danger areas
- Routinely see many small outbreaks (robust)
- Rarely see large outbreaks (fragile)
- Sensitive to changes in the environment (P_{ij})

HOT theory

Summary of designed tolerance

- Build more firewalls in areas where sparks are likely
- Small connected regions in high-danger areas
- Large connected regions in low-danger areas
- Routinely see many small outbreaks (robust)
- Rarely see large outbreaks (fragile)
- Sensitive to changes in the environment (P_{ij})

HOT theory

Summary of designed tolerance

- Build more firewalls in areas where sparks are likely
- Small connected regions in high-danger areas
- Large connected regions in low-danger areas
- Routinely see many small outbreaks (robust)
- Rarely see large outbreaks (fragile)
- Sensitive to changes in the environment (P_{ij})

HOT theory

Summary of designed tolerance

- Build more firewalls in areas where sparks are likely
- Small connected regions in high-danger areas
- Large connected regions in low-danger areas
- Routinely see many small outbreaks (robust)
- Rarely see large outbreaks (fragile)
- Sensitive to changes in the environment (P_{ij})

HOT theory

Summary of designed tolerance

- Build more firewalls in areas where sparks are likely
- Small connected regions in high-danger areas
- Large connected regions in low-danger areas
- Routinely see many small outbreaks (robust)
- Rarely see large outbreaks (fragile)
- Sensitive to changes in the environment (P_{ij})

Outline

Avalanches on Sand and Rice

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at ‘critical states’
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise ‘for free’
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  “Self-organized criticality - an explanation of 1/f noise”
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at 'critical states'
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise 'for free'
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  "Self-organized criticality - an explanation of 1/f noise"
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at ‘critical states’
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise ‘for free’
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  “Self-organized criticality - an explanation of 1/f noise”
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at ‘critical states’
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise ‘for free’
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  “Self-organized criticality - an explanation of 1/f noise”
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at ‘critical states’
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise ‘for free’
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  “Self-organized criticality - an explanation of 1/f noise”
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at 'critical states'
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise 'for free'
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  "Self-organized criticality - an explanation of 1/f noise"
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

SOC theory

SOC = Self-Organized Criticality

- Idea: natural dissipative systems exist at 'critical states'
- Analogy: Ising model with temperature somehow self-tuning
- Power-law distributions of sizes and frequencies arise 'for free'
- Introduced in 1987 by Bak, Tang, and Weisenfeld [3, 2, 9]:
  "Self-organized criticality - an explanation of 1/f noise"
- Problem: Critical state is a very specific point
- Self-tuning not always possible
- Much criticism and arguing...

Robustness

HOT versus SOC

- Both produce power laws
- Optimization versus self-tuning
- HOT systems viable over a wide range of high densities
- SOC systems have one special density
- HOT systems produce specialized structures
- SOC systems produce generic structures

Robustness

HOT versus SOC

- Both produce power laws
- Optimization versus self-tuning
- HOT systems viable over a wide range of high densities
- SOC systems have one special density
- HOT systems produce specialized structures
- SOC systems produce generic structures

Robustness

HOT versus SOC

- Both produce power laws
- Optimization versus self-tuning
- HOT systems viable over a wide range of high densities
- SOC systems have one special density
- HOT systems produce specialized structures
- SOC systems produce generic structures

Robustness

HOT versus SOC

- Both produce power laws
- Optimization versus self-tuning
- HOT systems viable over a wide range of high densities
- SOC systems have one special density
- HOT systems produce specialized structures
- SOC systems produce generic structures

Robustness

HOT versus SOC

- Both produce power laws
- Optimization versus self-tuning
- HOT systems viable over a wide range of high densities
- SOC systems have one special density
- HOT systems produce specialized structures
- SOC systems produce generic structures

Robustness

HOT versus SOC

- Both produce power laws
- Optimization versus self-tuning
- HOT systems viable over a wide range of high densities
- SOC systems have one special density
- HOT systems produce specialized structures
- SOC systems produce generic structures

Outline

COLD forests

Avoidance of large-scale failures

- Constrained Optimization with Limited Deviations [13]
- Weight cost of larges losses more strongly
- Increases average cluster size of burned trees...
- ... but reduces chances of catastrophe
- Power law distribution of fire sizes is truncated

COLD forests

Avoidance of large-scale failures

- Constrained Optimization with Limited Deviations [13]
- Weight cost of larges losses more strongly
- Increases average cluster size of burned trees...
- ... but reduces chances of catastrophe
- Power law distribution of fire sizes is truncated

COLD forests

Avoidance of large-scale failures

- Constrained Optimization with Limited Deviations [13]
- Weight cost of larges losses more strongly
- Increases average cluster size of burned trees...
- ... but reduces chances of catastrophe
- Power law distribution of fire sizes is truncated

COLD forests

Avoidance of large-scale failures

- Constrained Optimization with Limited Deviations [13]
- Weight cost of larges losses more strongly
- Increases average cluster size of burned trees...
- ... but reduces chances of catastrophe
- Power law distribution of fire sizes is truncated

COLD forests

Avoidance of large-scale failures

- Constrained Optimization with Limited Deviations [13]
- Weight cost of larges losses more strongly
- Increases average cluster size of burned trees...
- ... but reduces chances of catastrophe
- Power law distribution of fire sizes is truncated

Cutoffs

Aside:

- Power law distributions often have an exponential cutoff

$$P(x) \sim x^{-\gamma} e^{-x/x_c}$$

where x_c is the approximate cutoff scale.

- May be stretched exponentials:

$$P(x) \sim x^{-\gamma} e^{-ax^{-\gamma+1}}$$

Cutoffs

Aside:

- Power law distributions often have an exponential cutoff

$$P(x) \sim x^{-\gamma} e^{-x/x_c}$$

where x_c is the approximate cutoff scale.

- May be stretched exponentials:

$$P(x) \sim x^{-\gamma} e^{-ax^{-\gamma+1}}$$

Outline

Robustness

And we've already seen this...

- network robustness.
- Albert et al., Nature, 2000:
  "Error and attack tolerance of complex networks" [1]
- Similar robust-yet-fragile story...
- See Networks Overview, Frame 57 (⊞)

References I

R. Albert, H. Jeong, and A.-L. Barabási.
Error and attack tolerance of complex networks.
Nature, 406:378–382, July 2000. pdf (⊞)

P. Bak.
How Nature Works: the Science of Self-Organized Criticality.
Springer-Verlag, New York, 1996.

P. Bak, C. Tang, and K. Wiesenfeld.
Self-organized criticality - an explanation of 1/f noise.
Phys. Rev. Lett., 59(4):381–384, 1987.

B. J. L. Berry.
Déjà vu, Mr. Krugman.
Urban Geography, 20:1–2, 1999. pdf (⊞)

References II

J. Carlson and J. Doyle.
Highly optimized tolerance: A mechanism for power laws in design systems.
Phys. Rev. E, 60(2):1412–1427, 1999. pdf (⊞)

J. Carlson and J. Doyle.
Highly optimized tolerance: Robustness and design in complex systems.
Phys. Rev. Lett., 84(11):2529–2532, 2000. pdf (⊞)

R. M. D'Souza, C. Borgs, J. T. Chayes, N. Berger, and R. D. Kleinberg.
Emergence of tempered preferential attachment from optimization.
Proc. Natl. Acad. Sci., 104:6112–6117, 2007. pdf (⊞)

References III

R. Ferrer i Cancho and R. V. Solé.
Zipf's law and random texts.
Advances in Complex Systems, 5(1):1–6, 2002.

H. J. Jensen.
Self-Organized Criticality: Emergent Complex Behavior in Physical and Biological Systems.
Cambridge Lecture Notes in Physics. Cambridge University Press, Cambridge, UK, 1998.

P. Krugman.
The self-organizing economy.
Blackwell Publishers, Cambridge, Massachusetts, 1995.

References IV

B. B. Mandelbrot.
An informational theory of the statistical structure of languages.
In W. Jackson, editor, *Communication Theory*, pages 486–502. Butterworth, Woburn, MA, 1953.

G. A. Miller.
Some effects of intermittent silence.
American Journal of Psychology, 70:311–314, 1957. pdf (⊞)

M. E. J. Newman, M. Girvan, and J. D. Farmer.
Optimal design, robustness, and risk aversion.
Phys. Rev. Lett., 89:028301, 2002.

H. A. Simon.
On a class of skew distribution functions.
Biometrika, 42:425–440, 1955. pdf (⊞)

References V

D. Sornette.
Critical Phenomena in Natural Sciences.
Springer-Verlag, Berlin, 2nd edition, 2003.

D. Stauffer and A. Aharony.
Introduction to Percolation Theory.
Taylor & Francis, Washington, D.C., Second edition, 1992.