More Mechanisms for Generating Power-Law Distributions

Principles of Complex Systems
CSYS/MATH 300, Fall, 2010

Prof. Peter Dodds

Department of Mathematics & Statistics
Center for Complex Systems
Vermont Advanced Computing Center
University of Vermont

Outline

Outline

Optimization

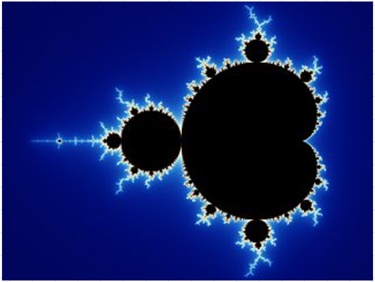
- Minimal Cost
- Mandelbrot vs. Simon
- Assumptions
- Model
- Analysis
- Extra
- And the winner is...?

Robustness

- HOT theory
- Self-Organized Criticality
- COLD theory
- Network robustness

References

Benoît Mandelbrot (⊞)



Nassim Taleb's tribute:

Benoit Mandelbrot, 1924-2010

A Greek among Romans

- Mandelbrot = father of fractals
- Mandelbrot = almond bread
- Bonus Mandelbrot set action: here (⊞).

Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

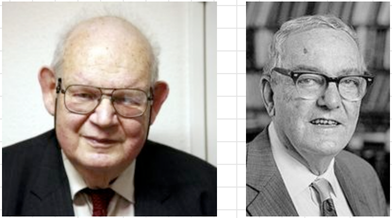
Another approach

Benoît Mandelbrot

- Derived Zipf's law through optimization [14]
- Idea: Language is efficient
- Communicate as much information as possible for as little cost
- Need measures of information (H) and average cost (C)...
- Language evolves to maximize H/C, the amount of information per average cost.
- Equivalently: minimize C/H.
- Recurring theme: what role does optimization play in complex systems?

Outline

Not everyone is happy...



Mandelbrot vs. Simon:

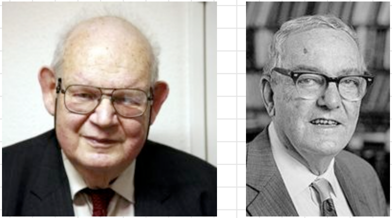
- Mandelbrot (1953): “An Informational Theory of the Statistical Structure of Languages”[14]
- Simon (1955): “On a class of skew distribution functions”[20]
- Mandelbrot (1959): “A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon”[15]
- Simon (1960): “Some further notes on a class of skew distribution functions”[21]

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): “An Informational Theory of the Statistical Structure of Languages”[14]
- Simon (1955): “On a class of skew distribution functions”[20]
- Mandelbrot (1959): “A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon”[15]
- Simon (1960): “Some further notes on a class of skew distribution functions”[21]

Not everyone is happy...



Mandelbrot vs. Simon:

- Mandelbrot (1953): “An Informational Theory of the Statistical Structure of Languages” [14]
- Simon (1955): “On a class of skew distribution functions” [20]
- Mandelbrot (1959): “A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon” [15]
- Simon (1960): “Some further notes on a class of skew distribution functions” [21]

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): "An Informational Theory of the Statistical Structure of Languages"[14]
- Simon (1955): "On a class of skew distribution functions"[20]
- Mandelbrot (1959): "A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon"[15]
- Simon (1960): "Some further notes on a class of skew distribution functions"[21]

Not everyone is happy...

Mandelbrot vs. Simon:

- Mandelbrot (1953): “An Informational Theory of the Statistical Structure of Languages”[14]
- Simon (1955): “On a class of skew distribution functions”[20]
- Mandelbrot (1959): “A note on a class of skew distribution function: analysis and critique of a paper by H.A. Simon”[15]
- Simon (1960): “Some further notes on a class of skew distribution functions”[21]

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” [17]
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot” [23]
- Mandelbrot (1961): “Post scriptum to ‘final note’” [17]
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum” [22]

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” [17]
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot” [23]
- Mandelbrot (1961): “Post scriptum to ‘final note”’ [17]
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum” [22]

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” [17]
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot” [23]
- Mandelbrot (1961): “Post scriptum to ‘final note”’ [17]
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum” [22]

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” [17]
- Simon (1961): “Reply to ‘final note’ by Benoit Mandelbrot” [23]
- Mandelbrot (1961): “Post scriptum to ‘final note”’ [17]
- Simon (1961): “Reply to Dr. Mandelbrot’s post scriptum” [22]

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.” [16]

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.” [23]

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.”[16]

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.”[23]

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.” [16]

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.” [23]

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.”[16]

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.”[23]

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."[16]

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."[23]

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."[16]

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."[23]

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."[16]

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."[23]

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.”[16]

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.”[23]

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."[16]

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."[23]

Plankton:

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

Outline

Optimization
- Minimal Cost
- Mandelbrot vs. Simon
- Assumptions
- Model
- Analysis
- Extra
- And the winner is...?

Robustness
- HOT theory
- Self-Organized Criticality
- COLD theory
- Network robustness

References

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

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

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)

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 from assignment 5 (⊞)

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

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

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 $n \rightarrow \infty$, 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 $n \rightarrow \infty$, 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 $n \rightarrow \infty$, 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

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

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 [6, 8]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [10]
3. D'Souza et al., 2007:
   Scale-free networks [9]

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 [6, 8]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [10]
3. D'Souza et al., 2007:
   Scale-free networks [9]

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 [6, 8]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [10]
3. D'Souza et al., 2007:
   Scale-free networks [9]

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 [6, 8]
2. Ferrer i Cancho and Solé, 2002:
   Zipf's Principle of Least Effort [10]
3. D'Souza et al., 2007:
   Scale-free networks [9]

More

Other mechanisms:

Much argument about whether or not monkeys typing could produce Zipf's law... (Miller, 1957)[18]

Outline

Optimization

- Minimal Cost
- Mandelbrot vs. Simon
- Assumptions
- Model
- Analysis
- Extra
- And the winner is...?

Robustness

- HOT theory
- Self-Organized Criticality
- COLD theory
- Network robustness

References

Others are also not happy

Krugman and Simon

- "The Self-Organizing Economy" (Paul Krugman, 1995) [12]
- 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) [12]
- 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)[12]
- 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) [12]
- 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

Optimization

- Minimal Cost
- Mandelbrot vs. Simon
- Assumptions
- Model
- Analysis
- Extra
- And the winner is...?

Robustness

- HOT theory
- Self-Organized Criticality
- COLD theory
- Network robustness

References

So who's right?

Empirical Tests of Zipf's Law Mechanism in Open Source Linux Distribution

T. Maillart,[1] D. Sornette,[1] S. Spaeth,[2] and G. von Krogh[2]
[1] *Chair of Entrepreneurial Risks, Department of Management, Technology and Economics, ETH Zurich, CH-8001 Zurich, Switzerland*
[2] *Chair of Strategic Management and Innovation, Department of Management, Technology and Economics, ETH Zurich, CH-8001 Zurich, Switzerland*

(Received 30 June 2008; published 19 November 2008)

Zipf's power law is a ubiquitous empirical regularity found in many systems, thought to result from proportional growth. Here, we establish empirically the usually assumed ingredients of stochastic growth models that have been previously conjectured to be at the origin of Zipf's law. We use exceptionally detailed data on the evolution of open source software projects in Linux distributions, which offer a remarkable example of a growing complex self-organizing adaptive system, exhibiting Zipf's law over four full decades.

So who’s right?

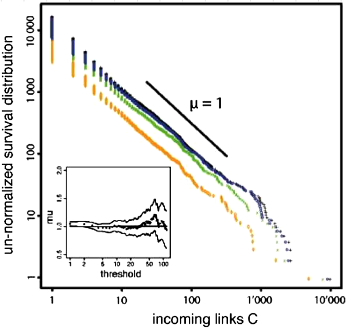


FIG. 1 (color online). (Color Online) Log-log plot of the number of packages in four Debian Linux Distributions with more than C in-directed links. The four Debian Linux Distributions are Woody (19.07.2002) (orange diamonds), Sarge (06.06.2005) (green crosses), Etch (15.08.2007) (blue circles), Lenny (15.12.2007) (black+’s). The inset shows the maximum likelihood estimate (MLE) of the exponent μ together with two boundaries defining its 95% confidence interval (approximately given by $1 \pm 2/\sqrt{n}$, where n is the number of data points using in the MLE), as a function of the lower threshold. The MLE has been modified from the standard Hill estimator to take into account the discreteness of C.

Maillart et al., PRL, 2008:
“Empirical Tests of Zipf’s Law Mechanism in Open Source Linux Distribution” [13]

So who’s right?

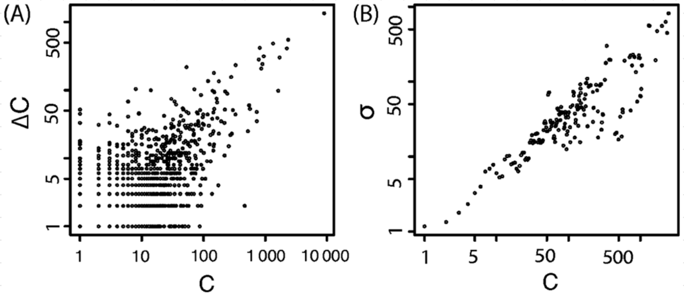


FIG. 2. Left panel: Plots of ΔC versus C from the Etch release (15.08.2007) to the latest Lenny version (05.05.2008) in double logarithmic scale. Only positive values are displayed. The linear regression $\Delta C = R \times C + C_0$ is significant at the 95% confidence level, with a small value $C_0 = 0.3$ at the origin and $R = 0.09$. Right panel: same as left panel for the standard deviation of ΔC.

- Rough, approximately linear relationship between C number of in-links and ΔC.

So who’s right?

Bornholdt and Ebel (PRE), 2001:
“World Wide Web scaling exponent from Simon’s 1955 model” [5].

- Show Simon’s model fares well.
- Recall ρ = probability new flavor appears.
- Alta Vista (⊞) crawls in approximately 6 month period in 1999 give $\rho \simeq 0.10$
- Leads to $\gamma = 1 + \frac{1}{1-\rho} \simeq 2.1$ for in-link distribution.
- Cite direct measurement of γ at the time: 2.1 ± 0.1 and 2.09 in two studies.

So who's right?

Bornholdt and Ebel (PRE), 2001:
"World Wide Web scaling exponent from Simon's 1955 model"[5].

- Show Simon's model fares well.
- Recall ρ = probability new flavor appears.
- Alta Vista (⊞) crawls in approximately 6 month period in 1999 give $\rho \simeq 0.10$
- Leads to $\gamma = 1 + \frac{1}{1-\rho} \simeq 2.1$ for in-link distribution.
- Cite direct measurement of γ at the time: 2.1 ± 0.1 and 2.09 in two studies.

So who's right?

Bornholdt and Ebel (PRE), 2001:
"World Wide Web scaling exponent from Simon's 1955 model"[5].

- Show Simon's model fares well.
- Recall ρ = probability new flavor appears.
- Alta Vista (⊞) crawls in approximately 6 month period in 1999 give $\rho \simeq 0.10$
- Leads to $\gamma = 1 + \frac{1}{1-\rho} \simeq 2.1$ for in-link distribution.
- Cite direct measurement of γ at the time: 2.1 ± 0.1 and 2.09 in two studies.

So who's right?

Bornholdt and Ebel (PRE), 2001:
"World Wide Web scaling exponent from Simon's 1955 model"[5].

- Show Simon's model fares well.
- Recall ρ = probability new flavor appears.
- Alta Vista (⊞) crawls in approximately 6 month period in 1999 give $\rho \simeq 0.10$
- Leads to $\gamma = 1 + \frac{1}{1-\rho} \simeq 2.1$ for in-link distribution.
- Cite direct measurement of γ at the time: 2.1 ± 0.1 and 2.09 in two studies.

So who's right?

Bornholdt and Ebel (PRE), 2001:
"World Wide Web scaling exponent from Simon's 1955 model"[5].

- Show Simon's model fares well.
- Recall ρ = probability new flavor appears.
- Alta Vista (⊞) crawls in approximately 6 month period in 1999 give $\rho \simeq 0.10$
- Leads to $\gamma = 1 + \frac{1}{1-\rho} \simeq 2.1$ for in-link distribution.
- Cite direct measurement of γ at the time: 2.1 ± 0.1 and 2.09 in two studies.

So who's right?

Bornholdt and Ebel (PRE), 2001:
"World Wide Web scaling exponent from Simon's 1955 model"[5].

- Show Simon's model fares well.
- Recall ρ = probability new flavor appears.
- Alta Vista (⊞) crawls in approximately 6 month period in 1999 give $\rho \simeq 0.10$
- Leads to $\gamma = 1 + \frac{1}{1-\rho} \simeq 2.1$ for in-link distribution.
- Cite direct measurement of γ at the time: 2.1 ± 0.1 and 2.09 in two studies.

So who’s right?

Nutshell:

- Simonish random ‘rich-get-richer’ models agree in detail with empirical observations.
- Power-lawfulness: Mandelbrot’s optimality is still apparent.
- Optimality arises for free in Random Competitive Replication models.

So who's right?

Nutshell:

- Simonish random 'rich-get-richer' models agree in detail with empirical observations.
- Power-lawfulness: Mandelbrot's optimality is still apparent.
- Optimality arises for free in Random Competitive Replication models.

So who’s right?

Nutshell:

- Simonish random ‘rich-get-richer’ models agree in detail with empirical observations.
- Power-lawfulness: Mandelbrot’s optimality is still apparent.
- Optimality arises for free in Random Competitive Replication models.

Outline

Optimization
- Minimal Cost
- Mandelbrot vs. Simon
- Assumptions
- Model
- Analysis
- Extra
- And the winner is...?

Robustness
- HOT theory
- Self-Organized Criticality
- COLD theory
- Network robustness

References

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) [6, 7, 8, 24]
- 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) [6, 7, 8, 24]
- 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) [6, 7, 8, 24]
- 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) [6, 7, 8, 24]
- 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) [6, 7, 8, 24]
- 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) [6, 7, 8, 24]
- 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) [6, 7, 8, 24]
- The catchphrase: Robust yet Fragile
- The people: Jean Carlson and John Doyle (⊞)

Robustness

Features of HOT systems: [7, 8]

- 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: [7, 8]

- 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: [7, 8]

- 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:[7, 8]

- 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: [7, 8]

- 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: [7, 8]

- 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:[7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example:[7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example: [7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example: [7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example: [7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example:[7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example:[7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example:[7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example:[7]

- Square $N \times N$ grid
- Sites contain a tree with probability ρ = density
- Sites are empty with probability $1 - \rho$
- Fires start at location (i, j) 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 given one spark

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example: [7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Forest fire example:[7]

- 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 = $Y = \rho - \langle c \rangle$

Robustness

Specifics:

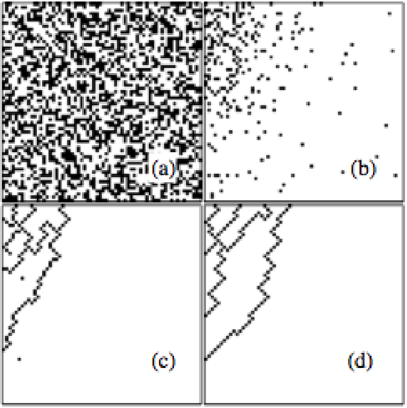
- $$P_{ij} = P_{i;a_x,b_x} P_{j;a_y,b_y}$$

  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



[7]

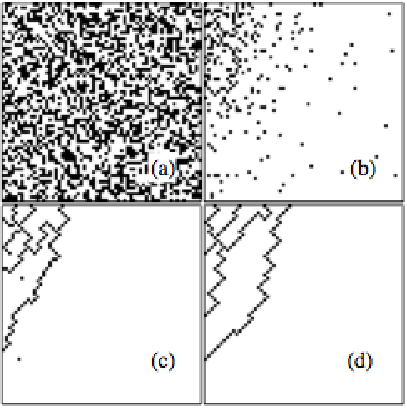
$N = 64$

(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
- But rare extreme events occur

HOT Forests



[7]

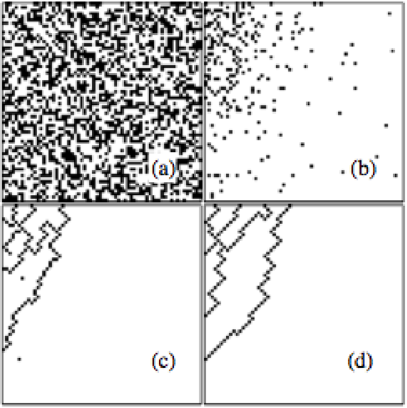
$N = 64$

(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
- But rare extreme events occur

HOT Forests



[7]

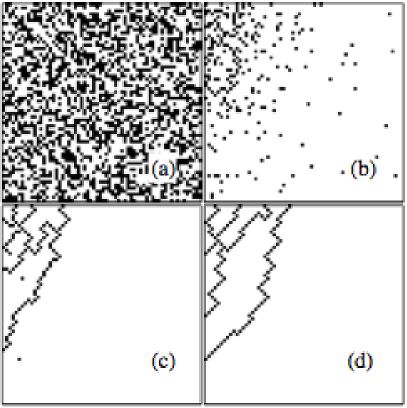
$N = 64$

(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
- But rare extreme events occur

HOT Forests



[7]

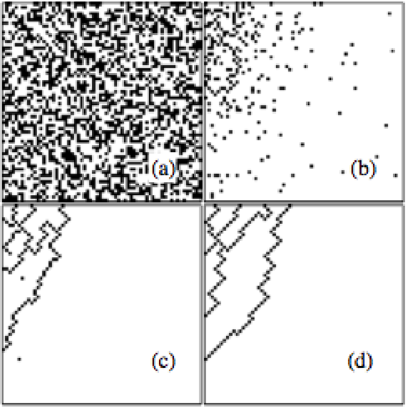
$N = 64$

(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



[7]

$N = 64$

(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

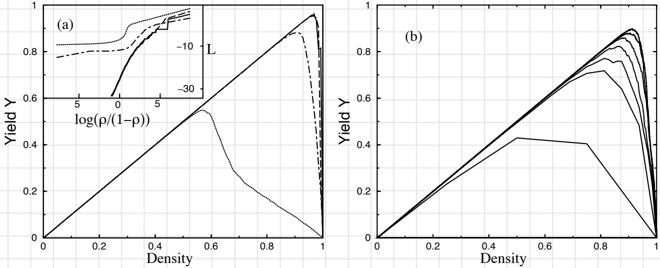


FIG. 2. Yield vs density $Y(\rho)$: (a) for design parameters $D = 1$ (dotted curve), 2 (dot-dashed), N (long dashed), and N^2 (solid) with $N = 64$, and (b) for $D = 2$ and $N = 2, 2^2, \ldots, 2^7$ running from the bottom to top curve. The results have been averaged over 100 runs. The inset to (a)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

[7]

HOT Forests

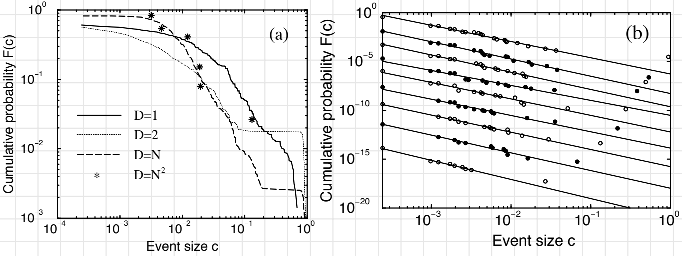


FIG. 3. Cumulative distributions of events $F(c)$: (a) at peak yield for $D = 1, 2, N$, and N^2 with $N = 64$, and (b) for $D = N^2$, and $N = 64$ at equal density increments of 0.1, ranging at $\rho = 0.1$ (bottom curve) to $\rho = 0.9$ (top curve).

[7]

Random Forests

$D = 1$: Random forests = Percolation [25]

- 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 [25]

- 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 [25]

- 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 [25]

- 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 [25]

- 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 nutshell:

- 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 nutshell:

- 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 nutshell:

- 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 nutshell:

- 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 nutshell:

- 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 nutshell:

- 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—Real data: [8]

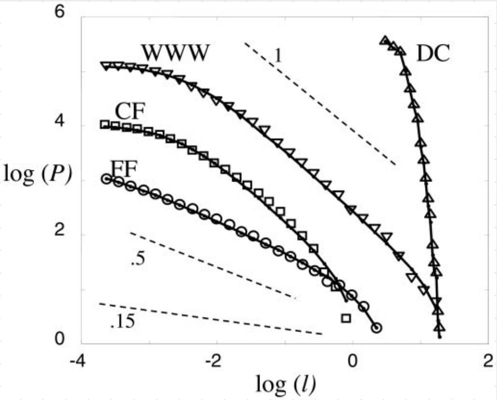


Fig. 1. Log–log (base 10) comparison of DC, WWW, CF, and FF data (symbols) with PLR models (solid lines) (for $\beta = 0, 0.9, 0.9, 1.85$, or $\alpha = 1/\beta = \infty$, 1.1,1.1, 0.054, respectively) and the SOC FF model ($\alpha = 0.15$, dashed). Reference lines of $\alpha = 0.5$, 1 (dashed) are included. The cumulative distributions of frequencies $\mathcal{P}(l \geq l_i)$ vs. l_i describe the areas burned in the largest 4,284 fires from 1986 to 1995 on all of the U.S. Fish and Wildlife Service Lands (FF) (17), the >10,000 largest California brushfires from 1878 to 1999 (CF) (18), 130,000 web file transfers at Boston University during 1994 and 1995 (WWW) (19), and code words from DC. The size units [1,000 km^2 (FF and CF), megabytes (WWW), and bytes (DC)] and the logarithmic decimation of the data are chosen for visualization.

HOT theory

The abstract story:

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

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

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

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

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

HOT theory

The abstract story:

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

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

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

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

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

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

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

HOT theory

HOT Theory—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

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

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

HOT Theory—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

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

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

HOT Theory—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

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

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

HOT Theory—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

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

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

HOT Theory—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

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

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

HOT Theory—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

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

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

HOT theory

Extra constraint:

- Total area is constrained:

$$\sum_{i=1}^{N_{\text{sites}}} a_i = N^2.$$

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

where N_{regions} = number of cells.

- Can ignore in calculation...

HOT theory

Extra constraint:

- Total area is constrained:

$$\sum_{i=1}^{N_{\rm sites}} a_i = N^2.$$

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

where $N_{\rm regions}$ = number of cells.

- Can ignore in calculation...

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$

  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d} a_i^{-1}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(2+1/d)}$$
- $$\text{For } d = 2, \gamma = 5/2$$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d} a_i^{-1}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(2+1/d)}$$
- For $d = 2, \gamma = 5/2$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
  $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d} a_i^{-1}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(2+1/d)}$$
- For $d = 2, \gamma = 5/2$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
-
$$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$
$$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d} a_i^{-1}\right)$$
-
$$p_i \propto a_i^{-\gamma} = a_i^{-(2+1/d)}$$
-
$$\text{For } d = 2, \gamma = 5/2$$

HOT theory

- Minimize C_{fire} given $C_{\text{firewalls}} = \text{constant}$.
- $$0 = \frac{\partial}{\partial a_j}\left(C_{\text{fire}} - \lambda C_{\text{firewalls}}\right)$$ $$\propto \frac{\partial}{\partial a_j}\left(\sum_{i=1}^{N} p_i a_i^2 - \lambda' a_i^{(d-1)/d} a_i^{-1}\right)$$
- $$p_i \propto a_i^{-\gamma} = a_i^{-(2+1/d)}$$
- $$\text{For } d = 2, \gamma = 5/2$$

HOT theory

Summary of designed tolerance [8]

- 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 [8]

- 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 [8]

- 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 [8]

- 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 [8]

- 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 [8]

- 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 of 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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, 11]:
  “Self-organized criticality - an explanation of 1/f noise” (PRL, 1987).
- 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

HOT theory—Summary of designed tolerance [8]

Table 1. Characteristics of SOC, HOT, and data

	Property	SOC	HOT and Data
1	Internal configuration	Generic, homogeneous, self-similar	Structured, heterogeneous, self-dissimilar
2	Robustness	Generic	Robust, yet fragile
3	Density and yield	Low	High
4	Max event size	Infinitesimal	Large
5	Large event shape	Fractal	Compact
6	Mechanism for power laws	Critical internal fluctuations	Robust performance
7	Exponent α	Small	Large
8	α vs. dimension d	$\alpha \approx (d-1)/10$	$\alpha \approx 1/d$
9	DDOFs	Small (1)	Large (∞)
10	Increase model resolution	No change	New structures, new sensitivities
11	Response to forcing	Homogeneous	Variable

Outline

COLD forests

Avoidance of large-scale failures

- Constrained Optimization with Limited Deviations [19]
- 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 [19]
- 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 [19]
- 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 [19]
- 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 [19]
- 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 Weibull distributions:

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

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 Weibull distributions:

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

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 67ish (⊞)

References I

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

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

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

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

References II

[5] S. Bornholdt and H. Ebel.
World Wide Web scaling exponent from Simon's 1955 model.
Phys. Rev. E, 64:035104(R), 2001. pdf (⊞)

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

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

[8] J. M. Carlson and J. Doyle.
Complexity and robustness.
Proc. Natl. Acad. Sci., 99:2538–2545, 2002. pdf (⊞)

References III

[9] 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 (⊞)

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

[11] 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.

References IV

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

[13] T. Maillart, D. Sornette, S. Spaeth, and G. von Krogh.
Empirical tests of Zipf's law mechanism in open source Linux distribution.
Phys. Rev. Lett., 101(21):218701, 2008. pdf (⊞)

[14] 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. pdf (⊞)

References V

[15] B. B. Mandelbrot.
A note on a class of skew distribution function. analysis and critique of a paper by H. A. Simon.
Information and Control, 2:90–99, 1959.

[16] B. B. Mandelbrot.
Final note on a class of skew distribution functions: analysis and critique of a model due to H. A. Simon.
Information and Control, 4:198–216, 1961.

[17] B. B. Mandelbrot.
Post scriptum to 'final note'.
Information and Control, 4:300–304, 1961.

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

References VI

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

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

[21] H. A. Simon.
Some further notes on a class of skew distribution functions.
Information and Control, 3:80–88, 1960.

[22] H. A. Simon.
Reply to Dr. Mandelbrot's post scriptum.
Information and Control, 4:305–308, 1961.

References VII

[23] H. A. Simon.
Reply to 'final note' by Benoît Mandelbrot.
Information and Control, 4:217–223, 1961.

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

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