Optimal prefix codes for some families of two-dimensional geometric distributions

Frédérique Bassino  U. de Marne-la-Vallée, France
Julien Clément  GREYC, U. de Caen, France
Gadiel Seroussi  MSRI, Berkeley, USA
Alfredo Viola  U. de la República, Uruguay

DCC, March 2006
Codes for geometric distributions

- Assume a sequence $X_1, X_2, X_3, \ldots$ of i.i.d. random variables obeying a geometric distribution (GD)

$$P(X = i) = (1 - q)q^i, \quad i \in \mathbb{Z}_{\geq 0}, \quad 0 < q < 1$$

- The $m$-th order Golomb code [Golomb 1966], denoted $G_m$, $m \geq 0$, encodes $i \geq 0$ as

$$G_m(i) = U(\lfloor i/m \rfloor) \ Q_m(i \mod m),$$

$U(j) = 0^j1$ (unary code), $Q_m = \text{optimal binary code}$ for a uniform distribution on $m$ symbols (codelengths $\lfloor \log_2 m \rfloor$ and $\lceil \log_2 m \rceil$).

$G_m = Q_m \bullet U \quad \text{concatenation}$
Codes for geometric distributions

- Assume a sequence $X_1, X_2, X_3, \ldots$ of i.i.d. random variables obeying a geometric distribution (GD)

$$P(X = i) = (1 - q)q^i, \quad i \in \mathbb{Z}_{\geq 0}, \quad 0 < q < 1$$

- The $m$-th order Golomb code [Golomb 1966], denoted $G_m$, $m \geq 0$, encodes $i \geq 0$ as

$$G_m(i) = U(\lfloor i/m \rfloor) \quad Q_m(i \mod m),$$

$U(j) = 0^j1$ (unary code), $Q_m$ = optimal binary code for a uniform distribution on $m$ symbols (codelengths $\lfloor \log_2 m \rfloor$ and $\lceil \log_2 m \rceil$).

$$G_m = Q_m \cdot U \quad \text{concatenation}$$

- $G_m$ is an optimal prefix code for GDs with $q$ satisfying $m = 1 + \lceil \log(1+q)/\log(1/q) \rceil$ [Gallager & van Voorhis (1973)].

- Geometric and related distributions occur in many practical applications, e.g., run-lengths, image prediction residuals (two-sided GDs or TSGDs, defined on $\mathbb{Z}$).

- Optimal prefix codes for TSGDs characterized in [Merhav, Seroussi & Weinberger (2000)].
Properties of known optimal codes for GDs and TSGDs

Optimal codes for GDs and TSGDs satisfy the following properties:

- There is a value $q_0 > 0$ such that all distributions with $q < q_0$ have $G_1 = U$ as optimal prefix code (e.g., $q_0 = \frac{\sqrt{5} - 1}{2} \approx 0.618$ for GDs).
- The *width* of the tree is bounded: For a given value of $q$, the number of nodes at any level of the optimal code tree is *uniformly* bounded (e.g., by $2m$ in $G_m$).
Symbol by symbol coding

Pros and cons:

- Low complexity and bounded (low) latency.
- In the case of Golomb (and related) codes: a simple calculation on the symbol yields the codeword, no code tables needed.
- The cost of simplicity is potentially high redundancy, especially in low-entropy regimes (code rate $\geq 1$ bit/symbol). Arithmetic coding-based schemes can achieve vanishing redundancy rate.

Symbol by symbol coding based on Golomb and related codes is used in practical applications: e.g., NASA/Rice, FELICS, LOCO-I/JPEG-LS.

Possible compromise: Use prefix code for blocks of $d$ symbols, $d \geq 2$.
- Can lower redundancy rate while maintaining simplicity
Two-dimensional geometric distributions

We study *prefix codes* for pairs \((X, Y)\) over \(\mathbb{Z}_{\geq 0} \times \mathbb{Z}_{\geq 0}\), distributed according to the *two-dimensional geometric distribution (TDGD)*

\[
P((X, Y) = (i, j)) = (1 - q)2q^{i+j}, \quad i, j \geq 0, \quad 0 < q < 1.
\]

- An interesting, open, mathematical problem (in addition to practical motivation).
  - Can we characterize optimal prefix codes, at least for some families of TDGDs? (Optimal codes prefix are known for few families of distributions over countable alphabets.)
  - Are their combinatorial properties significantly different from those of the known codes?
- We characterize optimal prefix codes for TDGDs with
  
  \[
  q = \frac{1}{2^k}, \quad k > 1 \quad (q \xrightarrow{k \to \infty} 0), \quad \text{and}
  \]
  
  \[
  q = \frac{1}{\sqrt{k}}, \quad k \geq 1 \quad (q \xrightarrow{k \to \infty} 1).
  \]

Two families of codes giving good coverage of the range \(0 < q < 1\).
Tools

• The alphabet of the TDGD source is

\[ A = \mathbb{Z}_{\geq 0} \times \mathbb{Z}_{\geq 0} = \{(0, 0), (0, 1), (1, 0), (0, 2), (1, 1), (2, 0), \ldots\}. \]

• The signature of \((a, b) \in A\) is \(s(a, b) = a + b\). **Signature determines probability.**

  ■ We will work over the *multiset* of signatures

\[ \overline{A} = \{0, 1, 1, 2, 2, \ldots, f, f, \ldots, f, \ldots\}. \]

• We seek to build an infinite binary tree \(T\), with each \(x \in A\) associated with a leaf of depth \(L_T(x)\) of \(T\), such that the expected cost

\[ c(T) = \sum_{x \in A} q^{s(x)} L_T(x) \]

is minimized \((\propto \text{average code length})\).

• The *weight* of a sub-tree \(T' \subseteq T\) is \(w(T') = \sum_{x \in T'} q^{s(x)} \) (\(\propto \text{probability of the leaves of } T'\)).
The method of Gallager and van Voorhis (1973)

- A method to construct infinite prefix codes and establish their optimality
  - Define a countable sequence of *reduced alphabets* (multisets) \((S_f)^\infty_{f=-1}\), where
    \[
    S_f = \{ 0, 1, 1, \ldots, f, f, \ldots, f, \text{ all signatures } \leq f, \text{ finite partition of the signatures } > f \},
    \]
  - Associate each *virtual symbol* \(R_i\) with the sum of weights of the signatures it contains.
  - Verify that after a finite number of merging steps of the Huffman algorithm, starting with \(S_f\), one gets \(S_{f'}\) with \(f' < f\).
  - Apply the Huffman algorithm to the “core” reduced alphabet \(S_{-1}\).

- “Bottom-up” procedure establishes optimality; “top down” unrolling describes the code.

- *All the effort and any ingenuity go into “guessing” the structure of the reduced sources. No general method for doing that.*
• Trees and weights

\[ T = \text{a tree with symbolic labels (weights) associated to leaves}; \quad g = \text{a scalar} \]
\[ \Rightarrow \quad gT = T \quad \text{with all labels multiplied by} \quad g. \]

• \( C_m \) = complete binary tree of depth \( m \) and \( 2^m \) leaves labeled \( q^0 \) (or \( 1 \)).

• For fixed \( k \), the *infinite* tree \( \mathcal{L}_q^k \) is defined by the following grammar:

\[
\begin{align*}
\mathcal{L}_q^0 & \rightarrow q\mathcal{L}_q^k, \\
\mathcal{L}_q^m & \rightarrow \mathcal{L}_q^{m-1} C_{m-1}, \quad \text{for} \quad 0 < m \leq k.
\end{align*}
\]

- \( \mathcal{L}_q^k \) has \( 2^k - 1 \) leaves of weight \( q^f \) at depth \( (f+1)k \) for all \( f \geq 0 \).
Proposition 1 (Part I: Top of the tree).

Assume $0 \leq f < 2^{k-1}$, and write $f = 2^i + j - 1$, $0 \leq j \leq 2^i - 1$. All signatures $f$ in the optimal tree are distributed in at most two levels as follows:

$$q^f \cdot \left[ \begin{array}{c} \binom{2^i - j - 1}{1} \\ \mathcal{R}_f \\ \binom{1}{1} \end{array} \right] \cdot \left( \begin{array}{c} \binom{1}{1} \\ \binom{j}{1} \end{array} \right)$$

Here, $\mathcal{R}_f$ represents a tree containing all signatures $> f$. 

Optimal code tree for the case $q = 2^{-k}$, $k > 1$ (I)
Proposition 1 (Part II: Bottom of the tree).

Assume \( f \geq 2^{k-1} \), and write \( f = 2^{k-1} - 1 + \ell(2^{k-1} - 1) + j \), \( 0 \leq j < 2^{k} - 1 \). Then, the signatures \( f \) are distributed in the optimal coding tree according to the five cases below:

(i) \( 0 \leq j < 2^{k-1} - 2 \):

\[
q^f \left[ \left( \begin{array}{c} \diamondsuit \\ \end{array} \right) \left( \begin{array}{c} \bullet \\ \end{array} \right) 2^{k-1} - j - 1 \right] = q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) \left( \begin{array}{c} \bullet \\ \end{array} \right) j
\]

(ii) \( j = 2^{k-1} - 2 \):

\[
q^f \left[ \left( \begin{array}{c} \diamondsuit \\ \end{array} \right) \ell \quad q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) \right] = q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right)
\]

(iii) \( 2^{k-1} - 2 < j < 2^{k-3} \):

\[
q^f \left[ \left( \begin{array}{c} \diamondsuit \\ \end{array} \right) \left( \begin{array}{c} \bullet \\ \end{array} \right) 3 \quad q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) j - 2^{k-1} + 1 \right] = q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) j - 2^{k-1} + 1
\]

(iv) \( j = 2^{k-3} \):

\[
q^f \left[ \left( \begin{array}{c} \diamondsuit \\ \end{array} \right) \ell \quad q_{L_q^{k-1}} \quad q_{C^{k-1}} \quad q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) 2^{k-1} - j \right] = q_{L_q^{k-1}} \quad q_{C^{k-1}} \quad q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) 2^{k-1} - j
\]

(v) \( j = 2^{k-2} \):

\[
q^f \left[ \left( \begin{array}{c} \diamondsuit \\ \end{array} \right) \ell \quad q_{L_q^{k-1}} q_{L_{q_{k-1}}} \quad q_{C^{k-1}} \quad q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) 2^{k-1} - j \right] = q_{L_q^{k-1}} q_{C^{k-1}} \quad q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) 2^{k-1} - j
\]

Here, \( \left( \begin{array}{c} \diamondsuit \\ \end{array} \right) \) stands for:

\[
q_{L_q^{k-1}} q_{C^{k-1}} \quad q_{R_j} \left( \begin{array}{c} \bullet \\ \end{array} \right) 2^{k-1}
\]
Example: \( q = 1/8 \)

Recursive structure allows for efficient encoding/decoding
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for $q = 2^{-k}$ has a component $\mathcal{C}^\ell$, with $\ell \approx f/2^{k-1}$ $\implies$ the width of the tree is unbounded for every $k$.

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for \( q = 2^{-k} \) has a component \((♣)\ell\), with \( \ell \approx f/2^{k-1} \implies \text{the width of the tree is unbounded for every } k \).

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When \( k \to \infty (q \to 0) \), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures \( f < 2^{k-1} \) does not change for \( k' > k \implies \text{there is a limiting code as } q \to 0 \).
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for $q = 2^{-k}$ has a component $(♣)^{ℓ}$, with $ℓ \approx f/2^{k-1}$ \implies the width of the tree is unbounded for every $k$.

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When $k \to \infty$ ($q \to 0$), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures $f < 2^{k-1}$ does not change for $k' > k \implies there is a limiting code as $q \to 0$.
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for $q = 2^{-k}$ has a component $(♣)^{\ell}$, with $\ell \approx f/2^{k-1}$ $\implies$ the width of the tree is unbounded for every $k$.

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When $k \to \infty$ ($q \to 0$), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures $f < 2^{k-1}$ does not change for $k' > k$ $\implies$ there is a limiting code as $q \to 0$.
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for $q = 2^{-k}$ has a component $(♣)^{\ell}$, with $\ell \approx f/2^{k-1}$ \implies the width of the tree is unbounded for every $k$.

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When $k \to \infty$ ($q \to 0$), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures $f < 2^{k-1}$ does not change for $k' > k$ \implies there is a limiting code as $q \to 0$. 
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for $q = 2^{-k}$ has a component $(♣)^{ℓ}$, with $ℓ \approx f/2^{k-1}$ $\implies$ the width of the tree is unbounded for every $k$.

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When $k \to \infty$ ($q \to 0$), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures $f < 2^{k-1}$ does not change for $k' > k$ $\implies$ there is a limiting code as $q \to 0$.
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for \( q = 2^{-k} \) has a component \((\spadesuit)^\ell\), with \( \ell \approx f/2^{k-1} \implies \text{the width of the tree is unbounded for every } k \).

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When \( k \to \infty \quad (q \to 0) \), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures \( f < 2^{k-1} \) does not change for \( k' > k \implies \text{there is a limiting code as } q \to 0 \).
Properties

Optimal prefix codes for TDGDs do not satisfy some of the basic structural properties of known optimal codes for GDs and TSGDs.

- Every slice of the (infinite) “bottom” part of the optimal tree for $q = 2^{-k}$ has a component $(♣)^{ℓ}$, with $ℓ \approx f/2^{k-1}$ $\implies$ the width of the tree is unbounded for every $k$.

- Still, the number of non-isomorphic infinite subtrees in each optimal tree is finite — the tree is rational.

- When $k \to \infty$ ($q \to 0$), we have an infinite sequence of distinct optimal codes. The code never fully stabilizes.

- However, the “top” of the tree with signatures $f < 2^{k-1}$ does not change for $k' > k$ $\implies$ there is a limiting code as $q \to 0$.
The case \( q = 1/\sqrt{2}, \ k \geq 1 \)

Define the tree \( q^{-2}T_q^2 \) as follows:

- The underlying tree of \( q^{-2}T_q^2 \) is the *concatenation* of two Golomb (unary) codes: \( G_1 \bullet G_1 \).
- *It defines an optimal prefix code for a TDGD with* \( q = 1/2 \ (k = 1) \) (as expected, since \( G_1 \) redundancy \( = 0 \) in this case).
Optimal prefix codes for \( q = 1/\sqrt{2} \)

Proposition 2. An optimal prefix code for a TDGD with parameter \( q = 1/\sqrt{2} \) is obtained by application of the Huffman algorithm to the reduced alphabet

\[
S_T = \left\{ q^{-2k}T^2_{q^k}, q^{-2k+1}T^2_{q^k}, q^{-2k+2}T^2_{q^k}, \ldots, q^{-k-1}T^2_{q^k} \right\} \quad \text{1 time} \quad \text{2 times} \quad \text{3 times} \quad \text{\( \ldots \) \( k \) times}
\]

\[
\bigcup \left\{ q^{-k}T^2_{q^k}, \ldots, q^{-3}T^2_{q^k}, q^{-2}T^2_{q^k} \right\} \quad \text{\( \ldots \) \( k-1 \) times} \quad \text{2 times} \quad \text{1 times}
\]

- Generalizes the case \( q = 1/2 \).
- The concatenation \( G_1 \cdot G_1 \) plays here the role that \( G_1 \) plays for Golomb codes: optimal tree consists of a “top” \( S_T \) from which many copies of \( G_1 \cdot G_1 \) “hang.”
- \( S_T \) is not quasi-uniform, but a prescribed sequence of pairings brings it to a quasi-uniform source.
Redundancy, in bits/integer symbol, for the optimal code for 2-blocks (empirical), the Golomb code, and the best code from the sequences $C_k$ (optimal for $q = 2^{-k}$) and $D_k$ (optimal for $q = 1/\sqrt{2}$).
Ongoing research

- Characterize optimal prefix codes for other (all?) values of $q$.
- Extend results to other GD-related distributions (e.g. TSGD).
- Extend to higher dimensions ($d > 2$). Some of the results extend easily.
- Develop fast adaptation and code selection strategies.
- The codes described are *rational*: they have a finite number of non-isomorphic infinite subtrees. In general, when is this the case?
Kraft functions and average code length

For a countable alphabet $\mathcal{A}$, a probability distribution $P(\cdot)$ on $\mathcal{A}$, a prefix code $C$ with codeword lengths $\{\ell_a\}_{a \in \mathcal{A}}$, define the Kraft function of $C$ as

$$F(z) = \sum_{a \in \mathcal{A}} P(a) z^{\ell_a}.$$ 

The recursive structure of the codes of Proposition 1 allows for explicit computation of their Kraft functions.

**Example:** For $q = 1/4$, we have

$$F_{1/4}(z) = \frac{9}{16} \left( z + \frac{qz}{1-q^3z^6} \left( 2z^2 + qz^2 \left( 3z + qz^2 \left( z + 3z^2 + \frac{3qz^3}{1-qz^2} \right) \right) \right) \right).$$

The average code length of $C$ is given by $c(C) = z \left. \frac{\partial}{\partial z} F(z) \right|_{z=1}$.

We use this tool to compute average code lengths and redundancies for our optimal prefix codes.