Compressione di grafi di grandi dimensioni tramite sistemi di numerazione asimmetrici

Francesco Tomaselli

Laureando

Sebastiano Vigna

Relatore

Paolo Boldi

Correlatore

Università degli studi di Milano

- Codificatori entropici presentati da Jarek Duda;
- Compressione paragonabile a codifica aritmetica, con velocità dei codici di Huffman.
- Base di tutti i sistemi di compressione moderni: Zstd, JPEG XL, etc.

Obiettivi della tesi:

- Applicazione a grafi web e sociali di grandi dimensioni;
- Valutazione su compressione e velocità di accesso.

Sistemi di numerazione asimmetrici: idea

- Sequenza di simboli codificata in un singolo intero x;
- Codifica di s con probabilità p_s, crea nuovo stato $x' \approx x/p_s$;
- Numero di bit per x vicino all'entropia di Shannon;
- Operazioni di divisione intera e non con virgola mobile;
- Gestione overflow stato.

- Dato un grafo G con nodi numerati da 0 a $N-1$;
- Rappresentazione successori per gap:

$$succ(x) = 1, 2, 3, 7, 9, 11, \ldots$$
$$gap(x) = 1, (2-1), (3-2), (7-4), (9-7), (11-9), \ldots$$

- Modello ottimo per i gap di ogni nodo:

Sym	f_s
1	3
4	1
2	2
...	...

- Costo nascosto: i simboli vanno elencati esplicitamente.

.graph

Nodo	Outd.	Id mod.	# stati	Stati
1	5	1	1	1873215
2	2	2	1	4732153
3	130	3	16	1237953, 543843, ...
...	...	...	...	...

.model

Id mod.	# sim.	Simboli	Frequenze
1	3	1, 5, 3	211, 300, 513
2	1	1	1024
3	6	1, 2, 4, 9, ...	32, 84, 12, 4, ...
...	...	...	...

Osservazioni sull'approccio base:

- Spazio ottimo per *.graph*;
- Occupazione su disco dominata da *.model*.

Idee:

- Limitare il numero di modelli, rendendoli sub-ottimali;
- Rimuovere simboli rari dalle mappe di frequenza.

- Ordinamento dei modelli: elementi consecutivi $\rightarrow$ simboli simili;
- Partizionamento dei modelli ordinati;
- Unione dei codificatori nella stessa partizione.

Compressione grafi: esempio unione

Modelli disordinati

S	f_s
1	8
2	5
3	3

S	f_s
5	4
6	2
9	1

S	f_s
1	10
2	8
4	5

S	f_s
5	14
6	6
8	3

S	f_s
2	5
3	3
4	1

Ordinamento e partizione

S	f_s
1	8
2	5
3	3

S	f_s
1	10
2	8
4	5

S	f_s
2	5
3	3
4	1

S	f_s
5	4
6	2
9	1

S	f_s
5	14
6	6
8	3

Unione

S	f_s
1	23
2	16
3	3
4	6

S	f_s
5	18
6	8
8	3
9	1

- Unione dei gap rari in un simbolo speciale;
- Scrittura esplicita elementi rimossi;
- Tradeoff su spazio occupato da successori e modelli.

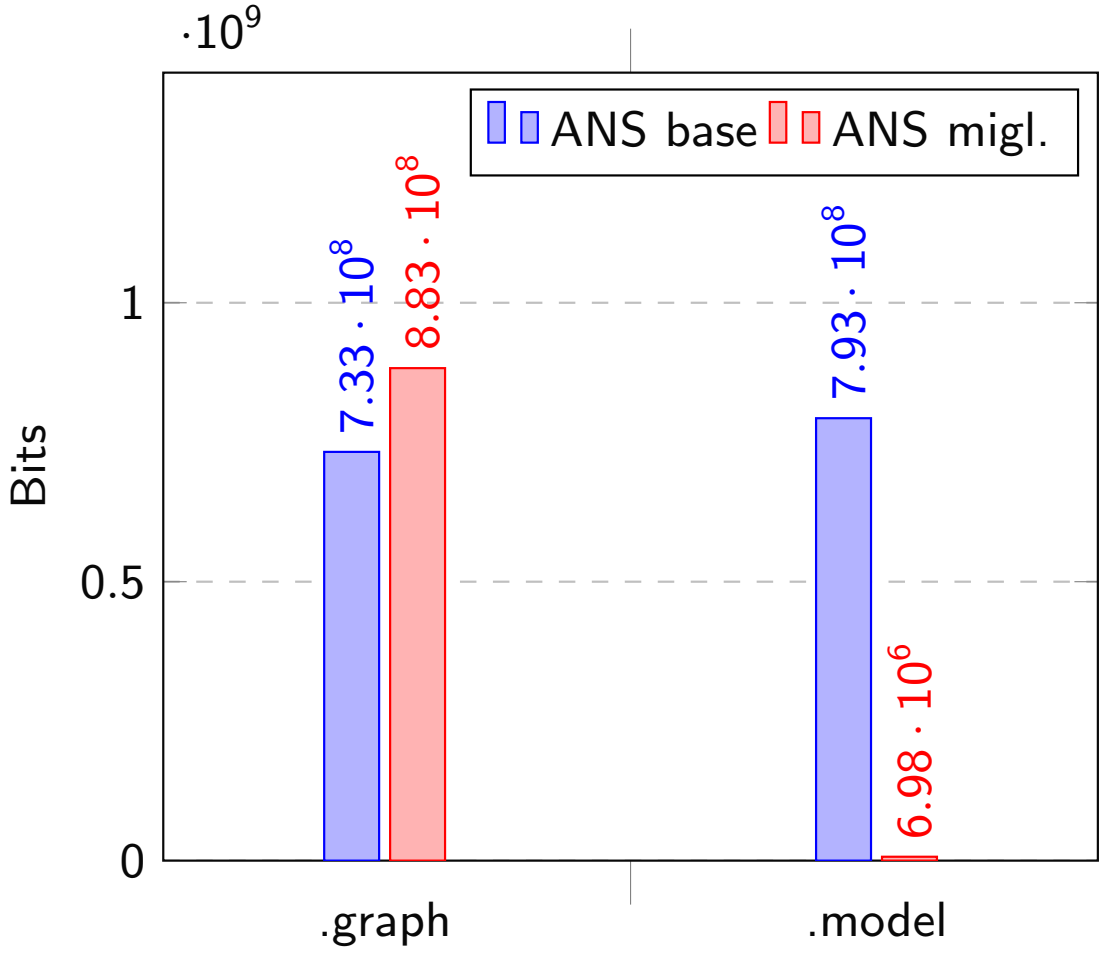
Sim.	Freq.
...	...
10	1
123	1
312	1

$\longrightarrow$

Sim.	Freq.
...	...
*	3

$$encode(state, 10) \longrightarrow encode(state, *), write(10)$$

Miglioramento di compressione su indochina-2014:



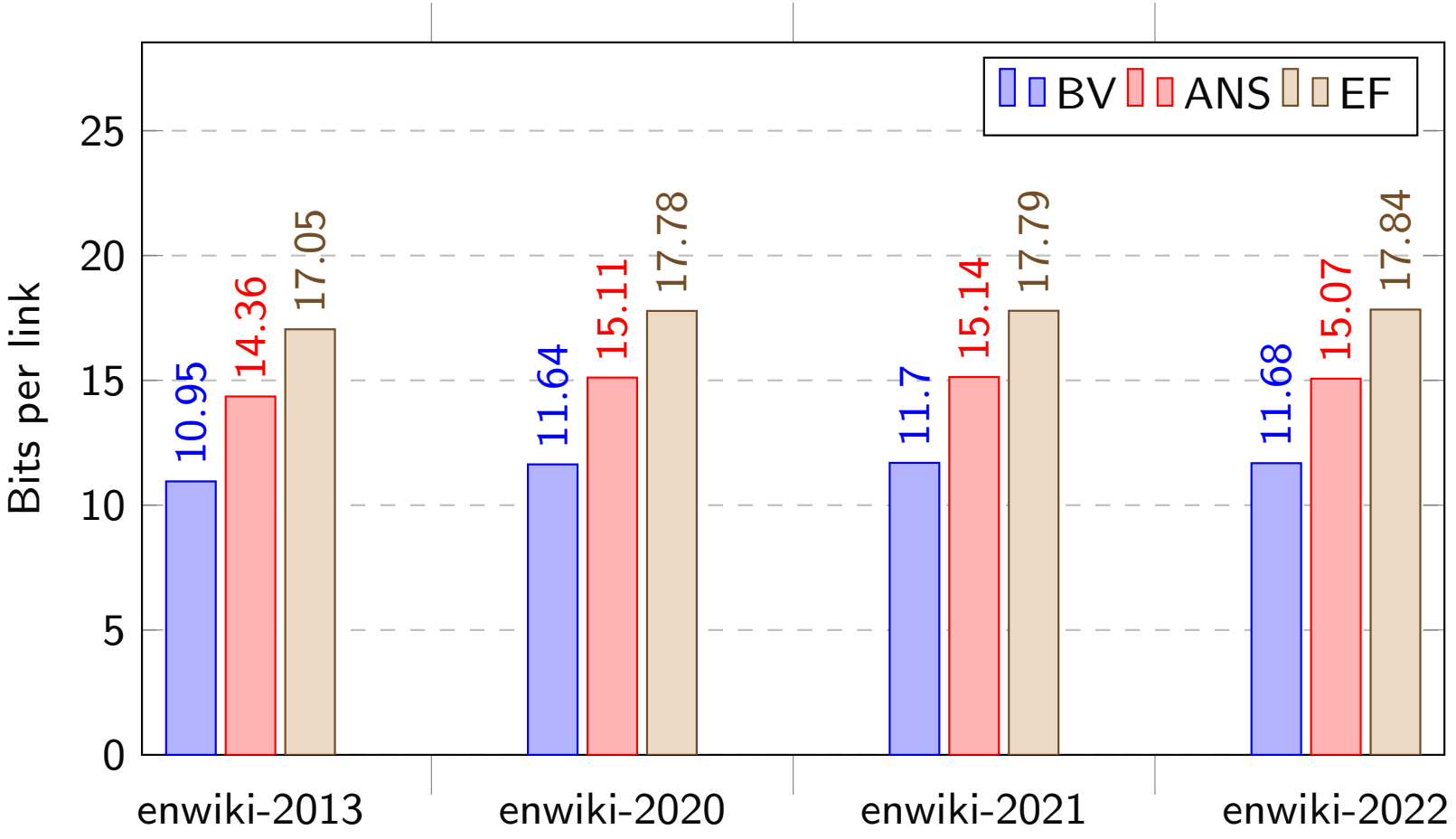
Valutazioni:

- Bit spesi per ogni arco;
- Velocità di accesso: nanosecondi per arco.

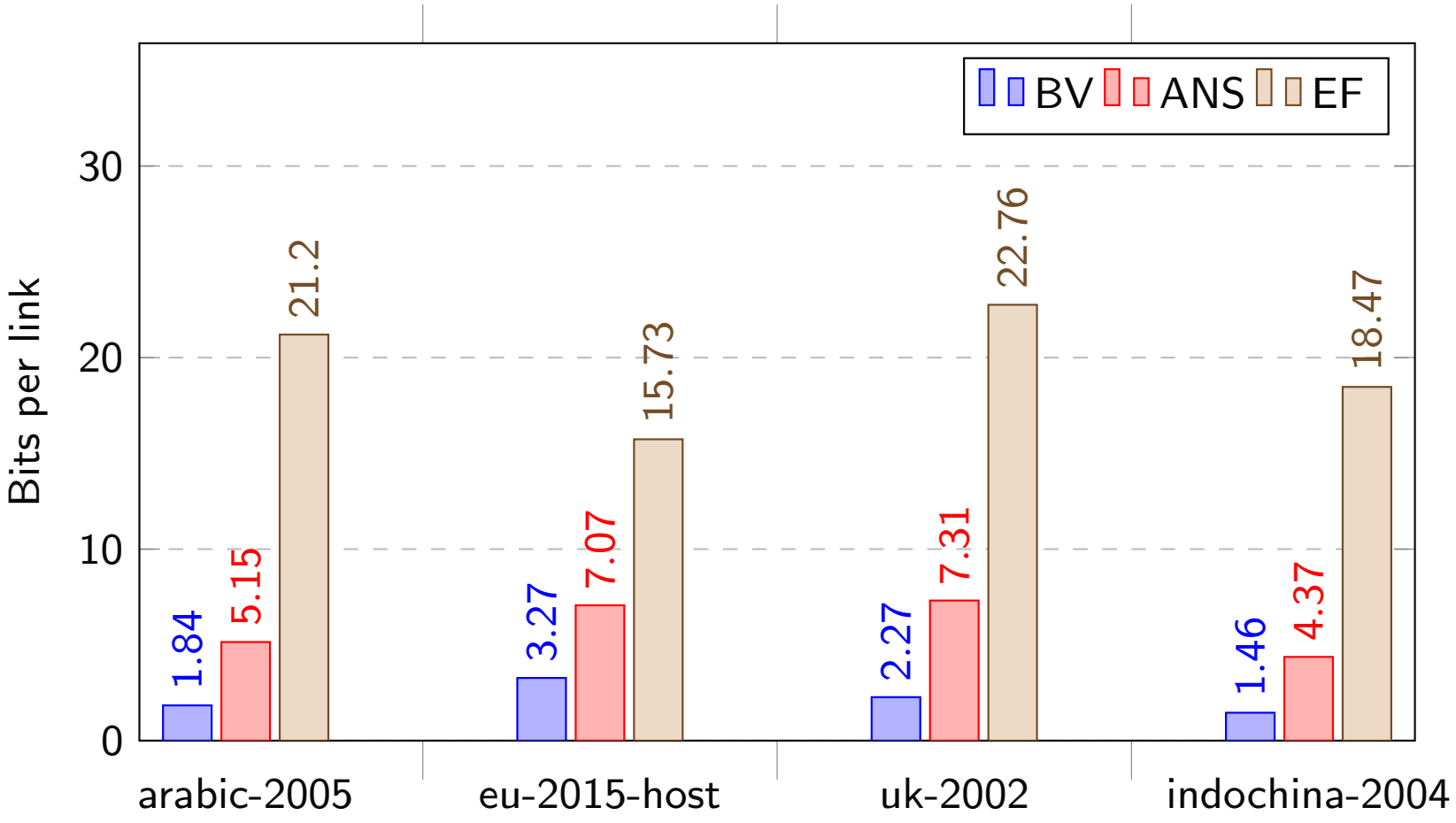
Tecniche a confronto:

- BV: tecnica presentata in Webgraph che utilizza codici istantanei, reference encoding e intervalisation;
- EF: liste monotone quasi-succinte di Elias-Fano.

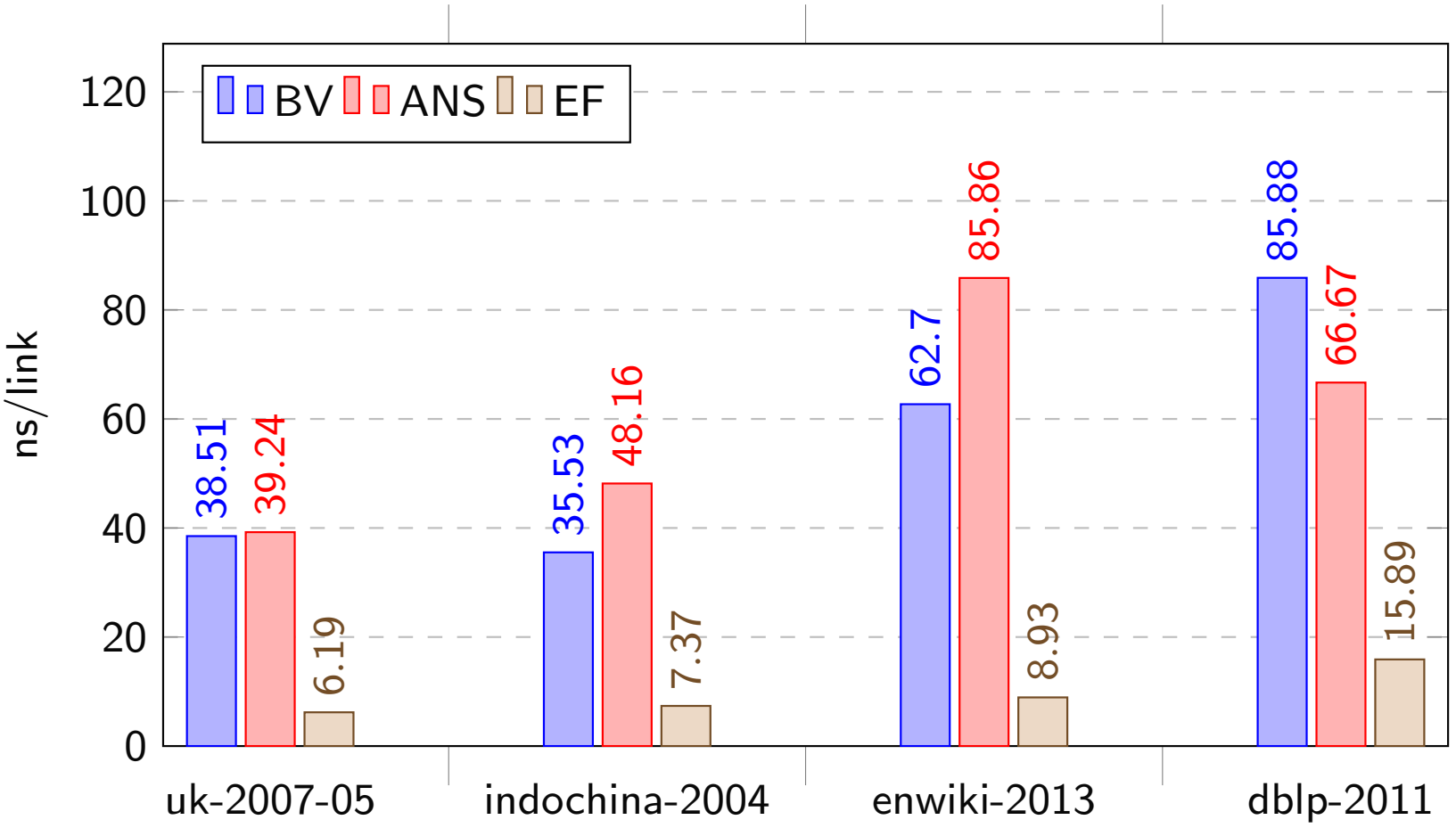
Compressione: grafi sociali



Compressione: grafi web



Velocità: nanosecondi per arco



- Nuova tecnica agnostica rispetto alla distribuzione dei gap;
- Compressione fino a ≈ 3.5 bit per link;
- Velocità paragonabile a sistemi esistenti.

Lavori futuri:

- Utilizzo delle proprietà empiriche dei grafi di interesse;
- Ordinamento e partizioni dei modelli con criteri differenti;
- Riduzione del numero di simboli tramite preprocessing.

Grazie dell'attenzione

Caso asimmetrico: simboli

- Simboli $S = \{s_1, \ldots s_n\}$ con frequenze $F = \{f_1, \ldots f_n\}$;
- Sia M la somma delle frequenze, $p_s = f_s/M$;
- Cumulativa per ogni simbolo $c_s = \sum_{t<s} f_t$;
- $n \in [0, M-1]$, simbolo associato $sym(n) = \max_{c_s \leq n} S$.

Esempio:

$$V = 1, 1, 1, 2, 2, 1, 2, 3, 1, 1, 3, 2, 1, 2, 3, 1$$

Sym	f_s	c_s
1	8	0
2	5	8
3	3	13

$$sym(6) = 1$$
$$sym(8) = 2$$
$$sym(15) = 3$$

Primitive di codifica e decodifica:

$$encode(state, s) = \lfloor state/f_s \rfloor * M + c_s + state \bmod f_s$$
$$decode(state) = \langle\ \lfloor (state - r)/M \rfloor * f_s - c_s + r,\ s\ \rangle$$

$r = state \bmod M$, $s = sym(r)$.

Osservazioni:

- $encode(state, s) \approx state/f_s * M \approx state/p_s$;
- Spazio ottimale per una determinata sorgente;
- Overflow stato.

Ordinamento per codici di Gray:

- $sym_1 = [0, 1, 2, 4]$, $sym_2 = [2, 4, 5]$, $sym_3 = [1, 2, 3]$;
- $gray_1 = 111010$, $gray_2 = 001011$, $gray_3 = 011100$;
- Risultato: sym_2, sym_3, sym_1;

Partizionamento euristico:

- Scelta numero massimo di simboli;
- Aggiunta a partizione attuale, se sforo, nuova partizione;
- Se risultano troppe partizioni, raddoppiamento soglia.

Scelta dei simboli in escape

- Calcolo statistiche per i gap di una partizione;
- Aggiungo progressivamente simboli meno frequenti all'insieme di escape;
- Scelgo threshold che minimizza:

$$\underbrace{E_{ans} + E_{esc}}_{\textit{codifica entropica}} + \underbrace{\textit{escape bits} + \textit{model bits}}_{\textit{info su escape e modelli}}$$

Esempio:

$$\textit{syms} = 1, 2, 5, 3, 13, 110, 45$$

$$S' = \emptyset \to \{45\} \to \{45, 110\} \to \{45, 110, 13\} \to \dots$$