Gossip Latin Square and The Meet-All Gossipers Problem

Nethanel Gelernter and Amir Herzberg

Department of Computer Science
Bar Ilan University
Ramat Gan, Israel 52900
Email: firstname.lastname@gmail.com

Abstract

Given a network of $n = 2^k$ gossipers, we want to schedule a cyclic calendar of meetings between all of them, such that: **(1)** each gossiper communicates (gossips) only once a day, with one other gossiper, **(2)** in every $(n-1)$ consecutive days, each gossiper meets all other gossipers, and **(3)** every gossip, initiated by any gossiper, will reach all gossipers within $k = \log(n)$ days.

In this paper we study the above stated *meet-all gossipers problem*, by defining and constructing the *Gossip Latin Square* (GLS), a combinatorial structure which solves the problem.

Keywords:
Latin square, gossip

1. Introduction

The gossip problem [3, 4] is one of the most studied problems in distributed computing. The original problem discusses how can n *gossipers*, each knowing some *rumor*, spread all rumors between them, where in each *round* (step), every gossiper can exchange rumors with only one other gossiper.

The problem has a lot of versions, which differ by topologies or by restrictions on the communication (e.g., instead of pairwise communication as in phone calls, group communication as in conference calls [6]), and many solutions, both randomized and deterministic.

The two main efficiency measures for solutions to the gossiping problem are: (1) How many phone calls are needed for spreading all the gossips? (2) How many rounds are needed?

In the abstract above, we defined the *meet-all gossipers problem* which is corresponding to the second question, as the minimal time to broadcast one gossip between $n = 2^k$ gossipers is k [2]. The difference between the gossipers problem and the *meet-all gossipers problem*, is in the second requirement: every gossiper must meet all the other $n-1$ gossipers in every $n-1$ consecutive days. In the rest of this paper we present a deterministic solution for the problem.

Beyond the theoretical interest, the problem is relevant to communication QoS: relying on other gossipers, we ensure the minimal delay, but even without them (on load, or when some of the gossipers are not entirely reliable) the maximal delay is $n-1$, as every gossiper meets all the others every consecutive $n-1$ days. This might be relevant also for communication mechanisms where there are messages that can be sent through other parties and secret messages that cannot. The deterministic routing that involves different paths and intermediate relays, can be used as a constant rate communication protocol with minimal delay for anonymous communication goals [8].

In the second section we formalize the *meet-all gossipers problem* and define *gossip Latin square (GLS)*, a combinatorial structure that solves the problem. In the third section we present a construction for GLS, and prove its correctness. In the last section we conclude and briefly discuss generalizations of the problem.

2. Definitions

2.1. The meet-all gossipers problem

To formally define the *meet-all gossipers problem*, we first define a *meeting schedule function*, which defines which pair of gossipers meet at any given day.

Definition 1. *Consider a finite set H (of gossipers). A* meeting schedule *is a function $Meet : \mathbb{N} \times H \to H$. We say that h* meets *$Meet_d(h)$ in day d.*

We next formalize requirements (1) and (2) in the abstract. A meeting schedule $Meet$ that satisfies these requirements, is said to be *perfectly fair*.

Definition 2. *Meeting function $Meet : \mathbb{N} \times H \to H$ is* perfectly fair *if it satisfies the following two requirements:*

Pairwise daily meetings *For every $d \in \mathbb{N}$ and $h \in H$ holds (1) $Meet_d(h) \neq h$ and (2) $h = Meet_d(Meet_d(h))$.*

Meet all *For every $d \in \mathbb{N}$ and $h \in H$, $\{Meet_{d+i}(h)\}_{0 \leq i < n-1} = H \backslash \{h\}$.*

To define the third requirement, we have to define the set of *recipients*, denoted $R^{Meet}_{d,m}(h)$, of a rumor initiated by $h \in H$ on day d and propagated m days. We first define the *recipient* relation between a pair of gossipers, h and h'. Informally, gossiper h' is a (d, m)-*recipient* from h, if there is a sequence of meetings between gossipers, beginning from day d, such that a rumor initiated on day d by h, will reach h' within m days, via the meetings sequence.

Definition 3. *Consider set of n gossipers H, and two gossipers $h, h' \in H$. We say that h' is a* (d, m)-recipient *from h according to the meeting schedule $Meet : \mathbb{N} \times H \to H$,* if and only if *there is a sequence of $y < m$ pairs $\{(p_t, q_t) | p_t \in H,\ 0 \leq q_t < q_{t+1} < m\}_{t=0}^{y}$ such that*

1. *$p_0 = Meet_{d+q_0}(h)$.*
2. *For every $0 < t \leq y$, $p_t = Meet_{d+q_t}(p_{t-1})$.*

3. $h' = p_y$.

Let $R^{Meet}_{d,m}(h)$ denote the set of all gossipers h', s.t. h' is a (d,m)-recipient *from h according to $Meet$. We call $R^{Meet}_{d,m}(h)$ the* (d,m)-recipients set *of h according to $Meet$.*

We can now formally define a *meet-all gossipers problem schedule*, i.e., a schedule that satisfies all three requirements in the abstract; such a meeting schedule is said to be *perfectly fair and round-optimal*.

Definition 4. *Meeting schedule function $Meet : \mathbb{N} \times H \to H$ ($|H| = 2^k$) is* perfectly fair *and* round-optimal*, if it is perfectly fair (Def. 2) and also satisfies the following (*Connectivity*) requirement: For every $d \in \mathbb{N}$ and $h \in H$, $R^{Meet}_{d,k}(h) = H \backslash \{h\}$.*

2.2. Schedule matrix

We next present a *schedule matrix*, a square matrix whose bottom row is the set of gossipers H, and whose content defines the meetings schedule.

Definition 5. *Let M be an $n \times n$ matrix. We say that M is a* schedule matrix *if*

1. *The last row (index $n-1$), called also the* headline row*, contains n distinct elements denoted $H = \{h_i\}_{i=0}^{n-1}$ (gossipers).*
2. *For $0 \leq i < n$, $0 \leq j < m$, $M_{i,j}$ appears in the headline row.*

We now define a mapping from a given schedule matrix M, to a meeting schedule function.

Definition 6. *Let M be an $n \times n$ schedule matrix, and let H be its headline row. The* meeting schedule *of M is denoted $Meet^M : \mathbb{N} \times H \to H$, and defined as: $Meet^M_d(h_j) = M_{d \bmod (n-1),j}$.*

We denote the (d,m)-recipients set of $Meet^M$ by $R^M_{d,m}$.

2.3. Gossip Latin Square (GLS)

A *Latin square (LS)* [7] is a $n \times n$ matrix whose rows and columns are permutations over n distinct elements. We now show that every Latin square is also a *schedule matrix*.

Lemma 7. *Every $n \times n$ Latin square matrix, is also a schedule matrix.*

Proof The *headline row* contains n distinct elements, and every other row is a permutation over the elements of the *headline row*. ■

We now define GLS, a Latin square that maps to meeting schedule function that solves the *meet-all gossipers problem*.

Definition 8. *Given $n = 2^k$, a matrix $M^{n \times n}$ is* Gossip Latin Square (GLS) *of order k,* if and only if

ROW INDEX																
0:	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
1:	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3
2:	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1
3:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
4:	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8
5:	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4
6:	5	4	7	6	1	0	3	**2**	13	12	15	14	9	8	11	10
7:	10	11	**8**	9	14	15	12	**13**	2	3	0	1	6	**7**	4	5
8:	13	12	**15**	14	9	**8**	11	**10**	**5**	4	**7**	6	1	**0**	3	2
9:	**6**	7	**4**	5	2	**3**	0	**1**	**14**	15	**12**	13	10	**11**	8	**9**
10:	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12
11:	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6
12:	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11
13:	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13
14:	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14
Headline-row:	0	1	2	3	4	5	6	**7**	8	9	10	11	12	13	14	15

Figure 1: Example to GLS of order 4. The boxed numbers above the headline row, represent all the destinations that 7 (boxed in the headline row) can reach from any day d, such that $d = 6 \bmod 15$ within 4 days. We can see that all the other 15 numbers are boxed.

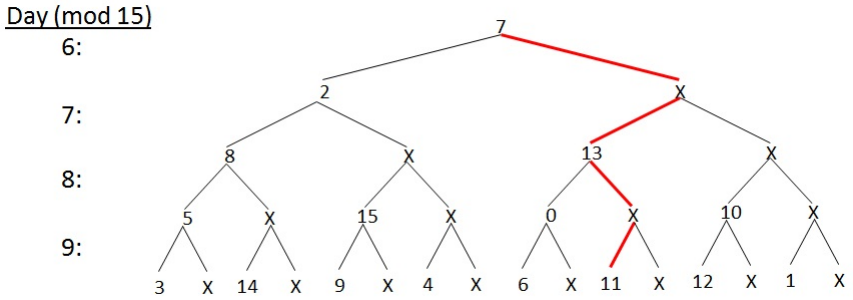

Figure 2: The routing tree of 7 from day $d = 6 \bmod 15$ according to the GLS in Figure 1. The X symbol signifies holding of the message. The bold lines are example to the routing procedure if 7 wants to send a message to 11: 7 holds the message one day and then forwards it to 13. 13 holds the message one more day, and forwards the message to 11 on day $d + 3$.

1. *M is a Latin square.*
2. *For $0 \leq i \leq n-2$, $0 \leq j \neq l \leq n-1$, $M_{i,j} = h_l$* if and only if *$M_{i,l} = h_j$.*
3. *For every $d \in \mathbb{N}$, $0 \leq j \leq n-1$, $R^M_{d,k}(h_j) = \{h_0, h_1, ..., h_{n-1}\} \backslash \{h_j\}$.*

Figure 1 contains example for GLS. The boxed values in Figure 1 and Figure 2 demonstrate, how the third requirement of Definition 8, is satisfied.

The following theorem shows that the meeting schedule function of a GLS solves the *meet-all gossipers problem.*

Theorem 9. *Given $n = 2^k$, a matrix $M^{n \times n}$ is a GLS* if and only if *$Meet^M$ is perfectly fair and round optimal (Definition 4).*

Proof $(\rightarrow)$: If M is a GLS, then $Meet^M$ satisfies the requirements of the *meet-all gossipers problem.* From Lemma 7, because M is LS, M is a *schedule matrix.* The *pairwise* requirement is satisfied by the second requirement of Definition 8. The *meet-all* requirement is satisfied because M is a Latin square, and therefore

for each column j, the first $n-1$ rows contains all the other elements of the headline row. Hence, $Meet^M$ is *perfectly fair*. The *connectivity* requirement, is satisfied directly by the third requirement of Definition 8.

($\leftarrow$) : If $Meet^M$ satisfies the requirements of the *meet-all gossipers problem* (i.e., is *perfectly fair* and *round-optimal*), then M is a GLS. The headline row of M contains n different elements. Because $Meet^M$ satisfies the *pairwise* requirement, every other row contains all the elements of the headline row: let $0 \leq i \leq n-2$ be some row, and h_j be an element of the headline row, then if $Meet_i^M(h_j) = h_l$, then h_j must appear in $M_{i,l}$. Therefore, every M's row contains all the elements of the headline row.

The *meet-all* requirement of $Meet^M$ implies directly that every M's column is permutation over the headline row's elements. Hence M is a Latin square (requirement 1 of Definition 8).

The second and the third requirements of GLS are satisfied directly by Definition 6 and the fact that $Meet^M$ satisfies the *pairwise* and *meet-all* requirements (Definition 2). ■

3. Construction of GLS

In this section we present an efficient construction of GLS, that is based on maximal Fibonacci LFSR.

3.1. Maximal Fibonacci Linear Feedback Shift Register (LFSR)

Fibonacci *LFSR* [5] is a register that contains k bits, indexed from 1 to k, named the *state* of the LFSR [1], such that shift operation removes the rightmost bit, pushes to the right the rest of the bits and creates a new leftmost bit by XORing some of the current LFSR's state bits in predefined indices.

A *Maximal LFSR* is a LFSR that given initial non-zero state, its shift operations produce a cyclic sequence of all the possible $2^k - 1$ non-zero states.

Maximal Fibonacci LFSR of k bits is created by a primitive polynomial $P(x) = 1 + \sum_{i=1}^{k} c_i \cdot x^i$ of degree k over $\mathbb{Z}_2$ ($c_k = 1$). The new leftmost bit (index 1) is calculated as a XOR of the bits in the indices set $\{i-1 | c_i = 1\}$.

*3.2. The GLS Construction (*GLSC*)*

We now present a simple construction of a GLS of order k that uses Fibonacci LFSR. GLSC chooses the headline rows such that $h_j = j$ for every $0 \leq j \leq n-1$.

Definition 10. *Let $n = 2^k$. The* GLS Construction (GLSC) *of a GLS of order k, $M^{n \times n}$, contains three steps:*

1. *Take a maximal Fibonacci LFSR of k bits with some non-zero initial state, and by running $n-2$ shift operations, create a vector V of length $n-1$, containing all the consecutive non-zero k-bits binary numbers.*

[1] We deal only with Fibonacci LFSR, hence, we often omit 'Fibonacci' and simply write LFSR.

2. *For* $0 \leq j \leq n-1$, *let* $M_{n-1,j} = j$.
3. *For* $0 \leq i \leq n-2$, $0 \leq j \leq n-1$ $M_{i,j} := V[i] \oplus j$.

Lemma 11. *The matrix* $M^{n \times n}$ *outputted by GLSC is a Latin square.*

Proof Let $H = \{0, 1, ..., n-1\}$. For every $h \in H$, for $0 \leq i \leq n-2$, the ith row of M, contains the permutation $\pi_i(h) = h \oplus V[i]$. For the last row (index $n-1$), $\pi_{n-1}(h) = h$.

For $0 \leq j \leq n-1$, the jth column is the permutation

$$\tau_j(h) = \begin{cases} j & \text{if } h = n-1 \\ j \oplus V[h] & \text{else} \end{cases}$$

τ_j is a permutation over H because $j = j \oplus 0^k$, and $\{V[h] | 0 \leq h \leq n-2\} \cup \{0^k\} = \{0,1\}^k$. ■

Lemma 12. *In every matrix* $M^{n \times n}$ *outputted by GLSC, for* $0 \leq i \leq n-2$ *and* $0 \leq j \neq l \leq n-1$*: (1)* $M_{i,j} \neq j$. *(2)* $M_{i,j} = l \Rightarrow M_{i,l} = j$.

Proof The first requirement is satisfied because M is LS (Lemma 11), and $M_{n-1,j} = j$.

The second requirement holds because by the GLS construction (Definition 10):

1. $l = M_{i,j} = V[i] \oplus j$.
2. $M_{i,l} = V[i] \oplus l$.

Substitution of (1) in (2) brings the desired result. ■

Lemma 13. *A sequence of* k *consecutive non-zero states of* k*-bits maximal Fibonacci LFSR forms a linear base for* $\{0,1\}^k$.

Proof Let $P(x) = 1 + \sum_{i=1}^{k} c_i \cdot x^i$ ($c_k = 1$) be the primitive polynomial used by the LFSR, and let $\{s_i\}_{i=0}^{2^k-2}$ be a cyclic sequence of the LFSR's non-zero states.

For simplicity, we omit the $\bmod(2^k - 1)$ from the states subscript until the end of the proof. The state s_j of the LFSR is a linear combination (the XOR operation is the same as addition in $\mathbb{Z}_2$) of the states $S = \{s_{j-i} | c_i = 1\}$; i.e., $s_j = \bigoplus_{s \in S} s$.

Because $c_k = 1$, $s_{j-k} \in S$.

Therefore, given a sequence of states $\{s_i\}_{i=j-k}^{j-1}$ that is a linear base for $\{0,1\}^k$, the sequence $\{s_i\}_{i=j-k+1}^{j}$ is also a linear base of $\{0,1\}^k$ (because the state s_{j-k} that was omitted from the base, can be reproduced by $s_j \oplus \big(\bigoplus_{s \in S - \{s_{j-k}\}} s \big)$).

Hence, it is enough to show that there is some consecutive sequence of k LFSR states that spans $\{0,1\}^k$. We consider the sequence of k states starting from $1 \circ 0^{k-1}$. Putting the k states in a $k \times k$ matrix, gives a lower rectangle matrix, such that all the values on the diagonal are 1, which implies that these k states are a linear base for $\{0,1\}^k$. ■

Definition 14. *(Linear combination of linear base) Given a linear base* $S = \{S_0, S_1, ..., S_{k-1}\}$ *of* $\{0,1\}^k$*, and an element* $e \in \{0,1\}^k$*. We denote the linear representation of* e *as linear combination of the elements of* S *by* $e^S = (e_0^S, e_1^S, ..., e_{k-1}^S)$*, such that* $e_i^S \in \mathbb{Z}_2$*, and* $e = \sum_{i=0}^{k-1} e_i^S \cdot S_i$ *when the addition is over* $\mathbb{Z}_2$*, and therefore equals to* $\oplus_{i=0}^{k-1}(e_i^S \cdot S_i)$*.*

Lemma 15. *Let* $M^{n\times n}$ *be a matrix outputted by GLSC, let* V *be the LFSR states vector that was created in the first step of GLSC, and let some* $i \in \mathbb{N}$*. We denote* $S = \{V[i \bmod (n-1)], V[i+1 \bmod (n-1)], ..., V[i+k-1 \bmod (n-1)]\}$ *as the set of* k *consecutive (cyclically) elements of* V*, starting from index* $i \bmod (n-1)$*.*

Then S *is a linear base of* $\{0,1\}^k$*, and for every* $0 \le j, l \le n-1$*,* $0 \le m \le k-1$*,* $Meet_{i+m}^M(j) = l$ if and only if j^S *and* l^S *differ only in index* m*.*

Proof By Lemma 13, $S = \{V[i], V[i+1 \bmod (n-1)], ..., V[i+k-1 \bmod (n-1)]\}$ is a linear base of $\{0,1\}^k$.

By Definition 6, $Meet_{i+m}^M(j) = l$ if and only if $M_{i+m \bmod (n-1),j} = l$. According to GLSC:

$$l = Meet_{i+m}^M(j) = M_{i+m \bmod (n-1),j} = j \oplus V[i+m \bmod (n-1)] = j \oplus S_m \quad (1)$$

Therefore j^S and l^S differ only in index m (the difference between them is only by XORing one of the elements in S, S_m).

If j^S and l^S differ only in index m, then $j \oplus l = S_m$, and hence $l = j \oplus S_m$. From Equation (1) this implies that $l = Meet_{i+m}^M(j)$.

■

Lemma 16. *Given some* $n = 2^k$*, GLSC produces a matrix* $M^{n\times n}$*, such that for every* $i \in \mathbb{N}$*, and for every* $0 \le j \le n-1$*,* $R_{i,k}^M(j) = \{0,1,2,...,n-1\}\backslash\{j\}$*.*

Proof We show that for every $i \in \mathbb{N}$, and every two different $j \ne l \in H = \{0,1,...,n-1\}$, l is a $(i,k) - recipient$ from j. This is enough for concluding also that $j \notin R_{i,k}^M(j)$, because in k days, a gossip can be broadcast to maximum $2^k - 1$ gossipers [2].

We consider the representation of j and l as a linear combination of the base $S = \{V[i \bmod (n-1)], V[i+1 \bmod (n-1)], ..., V[i+k-1 \bmod (n-1)]\}$: j^S and l^S.

We define $Q = (q_0, q_1, ..., q_y)$, to be the sorted sequence of the elements in the set $\{t | 0 \le t \le k-1, j_t^S \ne l_t^S\}$. Namely, Q is the sequence of indices where j^S differs from l^S. Obviously $y < k$, Q is sorted and $q_y < k$, because Q is a subsequence of the indices $(0,1,...,k-1)$

For every q_t we define a corresponding p_t value, such that p_t^S is identical to l^S until index q_t (including), and identical to j^S from index q_t (excluding).

We define $P = (p_0, p_1, ..., p_y)$ to be the corresponding sequence to Q's elements. $P \subset H$.

We now shows that the sequence $(p_t, q_t)_{t=1}^y$ satisfies the three requirements of Definition 3 for l is a $(i,k) - recipient$ from j:

1. $p_0 = Meet^M_{i+q_0}(j)$. By Lemma 15, it is enough to show that j^S and p_0^S differ only in index q_0. But q_0 is the first index where j^S differs from l^S, so by the choice of p_0, the first q_0 values in p_0^S are identical to l^S and therefore also to j^S. In indices higher than q_0, the values of j^S and p_0^S are identical by the choice of p_0. The only different index is therefore q_0, where p_0^S equals to l^S.
2. For every $0 < t \leq y$, $p_t = Meet^M_{i+q_t}(p_{t-1})$. Similarly to the previous requirement we show that p_t^S differs from p_{t-1}^S only in index q_t. For indices that are greater than q_t, both p_t^S and p_{t-1}^S are the same as j^S. For indices $\leq q_{t-1}$, both of them are the same as l^S. For indices between q_{t-1} and q_t (excluding), p_t^S and p_{t-1}^S are identical because j^S and l^S are identical in theses indices. Therefore the only difference between p_t^S and p_{t-1}^S is in index q_t is by the choice of q_t.
3. $l = p_y$. This is true, because for indices $\leq p_y$, by definition p_y^S is the same as l^S, and for the other indices ($> q_y$), j^S is the same as l^S.

■

Theorem 17. *GLSC produces a GLS.*

Proof Directly from Lemmas 11, 12 and 16. ■

3.3. Example

Figure 1 is a GLS that was outputted by GLSC, using a maximal Fibonacci LFSR that is based on the primitive polynomial $P(x) = x^4 + x + 1$, and with initial state 1000 (in decimal base 8). The vector V that was created in the first step of GLSC appears above the headline-row in the first column (XORed by 0).

4. Conclusions and Future Work

In this paper we presented the meet-all gossipers problem, defined Gossip Latin Square (GLS), and proved that GLS of order k represents a solution to the problem with 2^k gossipers. We then present a construction for GLS and prove its correctness.

We presented the *binary* version of the meet-all gossipers problem. The problem can be generalized by generalizing the *pairwise* requirement. We can take $n = m^k$ and require that every day there will be $\frac{n}{m}$ meetings, each one of m gossipers. To solve this problem, we can use $\mathbb{Z}_m$ and corresponding primitive polynomial to create $(\frac{n-1}{m-1} + 1) \times n$ matrix, such that except the headline row, every matrix cell contains $m - 1$ values.

The requirement can be further generalized by changing the number of gossipers in a meeting as a function of the day. As GLSC is similar to the hypercube graph construction [9], to solve this version of the problem, we offer to use the generalized hypercube [1].

5. Acknowledgements

We would like to thank Prof. David Peleg for his helpful comments. This research was supported by a grant from the Ministry of Science and Technology, Israel.

References

[1] LN Bhuyan and DP Agrawal. Generalized hypercube and hyperbus structures for a computer network. *IEEE Transactions on Computers*, pages 323–333, 1984.

[2] Michelangelo Grigni and David Peleg. Tight bounds on mimimum broadcast networks. *SIAM Journal on Discrete Mathematics*, 4(2):207–222, 1991.

[3] Sandra M Hedetniemi, Stephen T Hedetniemi, and Arthur L Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18(4):319–349, 1988.

[4] David Kempe and Jon Kleinberg. Protocols and impossibility results for gossip-based communication mechanisms. In *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, pages 471–480. IEEE, 2002.

[5] Andreas Klein. Linear feedback shift registers. In *Stream Ciphers*, pages 17–58. Springer, 2013.

[6] DJ Kleitman and JB Shearer. Further gossip problems. *Discrete Mathematics*, 30(2):151–156, 1980.

[7] Charles F Laywine and Gary L Mullen. *Discrete mathematics using Latin squares*. Wiley New York, 1998.

[8] A. Pfitzmann and M. Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. *URL: http://dud. inf. tu-dresden. de/literatur/Anon_Terminology_v0*, 34, 2010.

[9] Youcef Saad and Martin H Schultz. Topological properties of hypercubes. *IEEE Transactions on Computers*, 37(7):867–872, 1988.