BOUNDED DILATION MAPS OF HYPERCUBES INTO CAYLEY GRAPHS ON THE SYMMETRIC GROUP

by

Zevi Miller	Dan Pritikin	I.H. Sudborough*
Dept. of Mathematics and Statistics	Dept. of Mathematics and Statistics	Computer Science Program
Miami University	Miami University	University of Texas - Dallas
Oxford, Ohio 45056	Oxford, Ohio 45056	Richardson, Texas 75083

*This author's research was done during Spring semester 1991 as a visiting professor in the Department of Mathematics and Statistics at Miami University.

Abstract

Let G and H be graphs with $|V(G)| \leq |V(H)|$. If $f:V(G)\rightarrow V(H)$ is a one to one map, we let dilation(f) be the maximum of $dist_H(f(x),f(y))$ over all edges xy in G. The construction of maps from G to H of small dilation is motivated by the problem of designing small slowdown simulations on H of algorithms that were originally designed for the network G.

Let S(n), the star network of dimension n, be the graph whose vertices are the elements of the symmetric group of degree n, two vertices x and y being adjacent if $x\circ(1,i) = y$ for some i. That is, xy is an edge if x and y are related by a transposition involving some fixed symbol (which we take to be 1). Also let P(n), the pancake network of dimension n, be the graph whose vertices are the elements of the symmetric group of degree n, two vertices x and y being adjacent if one can be

obtained from the other by reversing some prefix. That is, xy is an edge if x and y are related by $x \circ (1,i)(2,i-1)\cdots(\lfloor i/2 \rfloor,\lceil i/2 \rceil) = y$. The star network (introduced in [AHK]) has nice symmetry properties, and its degree and diameter are sublogarithmic as functions of the number of vertices, making it compare favorably with the hypercube network. These advantages of S(n) motivate the study of how well it can simulate other parallel computation networks, in particular, the hypercube.

The concern of this paper is to construct low dilation maps of hypercube networks into star or pancake networks. Typically in such problems there is a tradeoff between keeping the dilation small and simulating a large hypercube. Our main result shows that at the cost of O(1) dilation as $n \to \infty$, one can embed a hypercube of near optimum dimension into the star or pancake networks S(n) or P(n) .

More precisely, letting Q(d) be the hypercube of dimension d, our main theorem is stated below. For simplicity we state it only in the special case when the star network dimension is a power of 2. A more general result (applying to star networks of arbitrary dimension) is obtained by a simple interpolation.

Theorem: Let n be a power of 2. Then there exist maps $Q(d) \to S(n)$ and $Q(d) \to P(n)$, each of dilation O(1) as $n \to \infty$, provided $d \leq n\log_2(n) - (\frac{3}{2} + o(1))n$.

1. Introduction

For a graph G, let V(G) and E(G) denote its vertex set and edge set respectively. All graphs in this paper are undirected, loopless, with no multiple edges.

Let Q(k) denote the graph of the k-dimensional hypercube. V(Q(k)) is the set of all binary k-tuples (typically expressed here as binary strings of length k), and E(Q(k)) is the set of pairs of points in Q(k) which differ in exactly one coordinate/bit. For strings $A_1, A_2, ..., A_n$ (binary or otherwise), let $A_1; A_2; ...; A_n$ denote the larger string formed by concatenating the strings together from left to right in that order, its length being the sum of the lengths of the A_i's. Let $\{0,1\}^*$ be the set of finite binary strings. Define the map Integer:$\{0,1\}^* \to \mathbb{Z}^+ \cup \{0\}$ which sends binary strings (interpreted as integers base 2) to the integers associated with them, e.g. Integer(011) = 3.

For a given ordered set of n symbols $S = \{s_1, s_2, ..., s_n\}$, let Perm(S) denote the set of permutations on that set, i.e. the set of all bijections $f:S \to S$. Define the function Sign:Perm(S)$\to \mathbb{Z}_2$ by letting Sign(f) = 0 when f is an even permutation, and Sign(f) = 1 when f is an odd permutation. For i≠j, let (i,j) denote the "transposition" f given by $f(x) = \begin{cases} s_i & \text{if } x = s_j \\ s_j & \text{if } x = s_i \\ x & \text{otherwise} \end{cases}$

,

and for i≠1 let t_i denote the particular transposition (1,i). Thus the transposition (i,j) is the permutation on S which interchanges i and j, leaving all elements of S-{i,j} fixed, while t_i is the special transposition which interchanges 1 and i. We define S(n), the vertex-labelled star network graph of dimension n on the symbol set S, as follows (equivalent to the definition given in [AHK], where star graphs were first introduced). V(S(n)) = Perm(S), and two vertices/permutations f, g ∈ Perm(S) are defined to be adjacent in S(n) if there exists an i≠1 for which $g(x) = f(t_i(x))$ [and consequently $f(x) = g(t_i(x))$] for all x. Observe that S(n) has n! vertices, is (n–1)-regular, and is bipartite with bipartition $(\mathrm{Sign}^{-1}(0),\mathrm{Sign}^{-1}(1))$. Figures 1a and 1b illustrate S(3) and S(4), where it is understood in Figure 1b that four edges **a**,**b**,**c**,**d** are not explicitly shown, but their end vertices are as indicated.

(Put Figure 1 here)

Let G and H be graphs, with |V(G)| ≤ |V(H)|, and f:V(G)→V(H) a one-to-one map. We define the dilation of f, written dilation(f), to be the maximum of $\mathrm{dist}_H(f(x),f(y))$ over all edges xy of G. Thus if dilation(f) = d, then pairs of adjacent points in G are sent to pairs of points in H at distance at most d, and we call such a map a dilation d embedding of G into H. We will write G⊆H to mean that G is a subgraph of H. We denote by G^d the "d'th power" of G, namely, the graph having the same vertex set as G in which two vertices are joined by an edge if their distance in G is at least 1 and at most d. Thus G can be embedded with dilation d into H if and only if $G \subseteq H^d$. We write $S^d(n)$ to denote $(S(n))^d$. Note that $S^1(n) = S(n)$ and that if d is at least the diameter of S(n), then $S^d(n)$ is a complete graph, so that every graph on n! or fewer vertices embeds with dilation d into S(n) for d that large.

The problem of minimizing dilation in graph embeddings has a large literature, motivated in part by the need to find good simulations of algorithms designed "originally" for some network G but requiring an implementation on some other network H. Each communication between adjacent processors x and y in the network G becomes, under the simulation, a communication between corresponding processors f(x) and f(y) in H that are at distance possibly as large as dilation(f). The simulation thus experiences a possible slowdown proportional to a factor of dilation(f), since we take the communication time between two vertices in a network to be proportional to the distance between them. We mention the papers [BCHLR], [MS] and the text [Lei] as just a few examples of the extensive literature on this topic. In much of this literature the study of simulations includes the analysis of both dilation and routing issues. In this paper we address only the dilation issue, so for our purposes the smaller the dilation of an embedding the better is the associated simulation.

The graphs of main concern in this paper (in particular Q(k) and S(n)) are instances of the more general Cayley graph construction. Indeed, let Γ be a group and A a generating set of Γ closed under inverses, i.e. satisfying $A = A^{-1}$. Then we define the Cayley graph on Γ, denoted Cay(Γ,A), as having Γ as its vertex set, with two vertices $g,h \in \Gamma$ joined by an edge in Γ if and only if g = hs for some $s \in S$. Note that the hypercube $Q(k) = Cay((\mathbb{Z}_2)^k, A)$, where A is the standard basis in $(\mathbb{Z}_2)^k$. Also $S(n) = Cay(\Sigma_n, A)$ where Σ_n is the symmetric group of degree n, and A is the set of transpositions $\{(1,i): 2 \le i \le n\}$. In [AK2] a related Cayley graph, the pancake graph P(n), was studied. Again the underlying group is Σ_n, and the generating set A is the set of "prefix reversals" ρ_i, $2 \le i \le n$, where ρ_i is the permutation $(1,i)(2,i-1)\cdots(\lfloor i/2 \rfloor, \lceil i/2 \rceil)$. A third Cayley graph on the symmetric group of degree n is TranS(n) = Cay(Σ_n,A), called the transposition graph, where A is the set of all tranpositions $\{(i,j): 1 \le i,j \le n\}$.

In this paper we investigate the problem of constructing bounded dilation embeddings of G into H when G = Q(k) and H = S(n) or P(n) for some k and n. The network S(n) was introduced in [AHK], where it was convincingly argued that S(n) has a number of desirable characteristics. The symmetry in S(n) resulting from its Cayley graph structure can be used to good effect in analyzing fault tolerance and broadcasting in S(n). The diameter of S(n) was shown to be $\left\lfloor \frac{3(n-1)}{2} \right\rfloor$ in [AK2]. Since S(n) is regular of degree n-1 and has n! vertices, we see that S(n) achieves diameter and degree which are sublogarithmic as functions of the number of vertices. By contrast, the hypercube Q(n), being regular of degree n on 2^n vertices and having diameter n, has degree and diameter which are logarithmic as functions of the number of vertices. Thus from the standpoint of degree-diameter values S(n) is clearly superior to Q(n) as an interconnection network. We also know [AK1] that S(n) has connectivity n-1 , which is of course the maximum it could be since S(n) is (n-1)-regular. This gives S(n) good fault tolerance properties. The problem of broadcasting a piece of information from one vertex in S(n) to all others was studied in [AHK], where an optimal O(nlog(n)) parallel time algorithm was developed. As observed in [AHK], the idea of this algorithm can be used to construct a dilation 1, one-to-one embedding of the complete binary tree of depth O(nlog(n)) into S(n). For all these reasons it seems important to develop good simulations on S(n) of algorithms designed for (or at least currently implemented on) various well known and commonly used networks. Among the latter is the hypercube, and this is the motivation for our work.

As is true of S(n), the network P(n) has sublogarithmic degree n-1 and has sublogarithmic diameter O(n). Concerning embeddings, it was shown that there exists a dilation 1 embedding of the depth O(nlog(n)) complete binary tree into P(n). Our results on dilation of embeddings of Q(k) into S(n) carry over to yield similar results on embeddings of Q(k) into P(n), as will be seen at the end of the paper.

There are a variety of ways to denote permutations on $S = \{s_1, s_2, ..., s_n\}$, i.e. to denote the vertices of S(n). One way is to list the images $f(s_1)\ f(s_2)\ ...\ f(s_n)$ as a string of symbols, and another way is to list the pre-images $f^{-1}(s_1)\ f^{-1}(s_2)\ ...\ f^{-1}(s_n)$ as a string of symbols. Using image-strings to represent vertices, two strings f and g are adjacent in S(n) when for some $i \neq 1$ it is the case that g is obtainable from f by interchanging the symbols in the first and *i*th positions of the string f. Using pre-image strings to represent vertices, two strings f and g are adjacent in S(n) when for some $i \neq 1$ it is the case that g is obtainable from f by interchanging the positions of symbols s_1 and s_i. Because of these differences in how to switch the positions of two symbols in a string to get an adjacent string in S(n), when using image-strings to denote vertices we say that S(n) is positionally represented on the symbols $s_1 s_2 ... s_n$ [position 1 being a special position], and when using pre-image strings to denote vertices, we say that S(n) is s_1-represented on the symbols $s_1 s_2 ... s_n$ [symbol s_1 being a special symbol]. For example, when S(4) is positionally represented on the symbols AXYZ, the string YXZA represents the permutation f for which f(A) = Y, f(X) = X, f(Y) = Z and f(Z) = A. The vertices adjacent to YXZA in S(4) are represented as XYZA, ZXYA and AXZY, i.e. the strings obtainable from YXZA by interchanging the symbol in its first position (which happens to by Y) with any other symbol. By contrast, when S(4) is A-represented on the symbols AXYZ, the string YXZA represents the permutation f for which f(Y) = A, f(X) = X, f(Z) = Y and f(A) = Z. The vertices adjacent to YXZA in S(4) are represented as AXZY, YAZX and YXAZ, i.e. the strings obtainable from YXZA by interchanging the symbol A with any other symbol. Figures 1b and 1c illustrate S(4) when positionally represented on the symbols AXYZ and when A-represented on the symbols AXYZ. Vertices that correspond visually in the two figures may or may not be labelled with the same symbol string, but do in fact correspond to the same permutation, viewing permutations as functions.

In [MPS] we constructed embeddings of Q(d) into S(n), with dilation kept very small (with values 1,2, or 3) at the cost of having a fairly large "expansion", i.e. ratio $\frac{|V(S(n))|}{|V(Q(d))|}$. Such embeddings can yield simulations with very little slowdown, but if the ratio is too large then we have succeeded in simulating only a small hypercube relative to the size of the host star network. Hence we may prefer to sacrifice some dilation if we can get in exchange a smaller expansion ratio for the same size star network. This suggests the natural question of studying the tradeoff between dilation and expansion. That is, we can ask how small an expansion we can achieve while keeping the dilation "reasonably" small, and what kinds of lower bounds on dilation are implied as the expansion gets progressively smaller. In this paper we address the first part of this question by studying how small the expansion can be made when the dilation is bounded by a constant. The problem of simulating Q(d) by TranS(n) has already appeared in Leighton's text [Lei], and we hope that the methods of this paper will help in this problem.

This paper is concerned with obtaining bounded dilation embeddings for which the expansion is asymptotically good. But for small values of n, Table 1 gives a summary of how large a hypercube can be embedded in S(n) with extremely small dilation, via the combined results from [MPS] and this paper. It is likely that some of these table entries can be improved, but the column headed by $\lfloor \log_2(n!) \rfloor$ indicates the dimension of the largest hypercube which can be embedded in S(n), even enduring an absurd dilation equal to the diameter of S(n). Therefore that column serves as a gauge for how low one can expect to keep the ratio $\frac{|V(S(n))|}{|V(Q(k))|}$.

The following notation will be used. For a vector z, let z_i denote its i'th coordinate. When T is a subset of V(S(n)) and x,yεT, we let $dist_T(x,y)$ denote the distance between x and y in the subgraph of S(n) induced by T. Finally we employ the usual O and o notation for growth of functions, where O(f(x)) (resp. o(f(x))) refers to any function g(x) satisfying $\frac{|g(x)|}{|f(x)|} \leq C$ for some positive constant C (respectively $\frac{|g(x)|}{|f(x)|} \rightarrow 0$) as x grows without bound. In particular, O(1) refers to a function g(x) with bounded absolute value while o(1) refers to a function g(x) which approaches 0 as x grows.

2. An Overview and Some Combinatorial Lemmas

We state here our main theorem for the case when n is a power of 2. Although our proof applies equally well for all n, we delay its most general statement here to avoid unnecessary complications. The general statement given later follows readily from the proof we present here, and is obtained from it by a simple interpolation. We also note that the theorem as stated concerns the existence of dilation O(1) embeddings Q(d)$\rightarrow$S(n), but the same proof (together with a simple observation to be made later) will show that it remains true when S(n) is replaced by P(n) or TranS(n).

Theorem 1: Let n be a power of 2. Then there is a one-to-one map Q(d)$\rightarrow$S(n) of dilation O(1) (as n$\rightarrow\infty$) provided $d \leq n\log_2(n) - (\frac{3}{2} + o(1))\, n$.

(All logarithms are hereafter taken base 2.)

To put this result in perspective, recall that the Stirling approximation gives

$$n\log(n) - (\log(e))n < \log(n!) < n\log(n) - (\log(e) - o(1))n.$$

Now the largest possible d for which a one-to-one map Q(d)$\rightarrow$S(n) (of any dilation) is possible is d=$\lfloor \log(n!) \rfloor$. Hence the biggest coefficient of n we can hope to achieve in a result such as Theorem 1 is -log(e), which is roughly -1.44. The first step in this direction, and indeed the starting point for our work, was the following theorem from [NSK].

<u>Theorem 2</u> [NSK]: If there exists an embedding $Q(d) \rightarrow S(n)$ of dilation ≤ 4, then there exists an embedding $Q(d + \lfloor \log(n+1) \rfloor) \rightarrow S(n+1)$ of dilation ≤ 4.

Starting with an embedding of Q(1) into S(2) of dilation at most 4 we can apply that result iteratively together with the identity $\sum_{t=1}^{p} t2^t = 2^{p+1}(p-1) + 2$ to deduce that there is a dilation 4 embedding of Q(d) into S(n), for n a power of 2, provided $d \leq (n+1)\log(n) - 2n + 2$. Hence our theorem may be viewed as a step in bringing the coefficient of n closer to the optimal $-\log(e) \approx -1.44$ than the -2 which follows from Theorem 2, while still keeping the dilation bounded by a constant. Indeed, while the $(n+1)\log(n) - 2n + 2$ result differs from the theoretical optimum by $(2 - \log(e) + o(1))n \approx (.56)n$, our main result differs from the optimum by $(\frac{3}{2} - \log(e) + o(1))n \approx (.06)n$, reducing the gap on the coefficient of n by about 89% while keeping the dilation bounded by a constant. It remains open whether we can in fact achieve a O(1) dilation embedding $Q(d) \rightarrow S(n)$ with $d = n\log(n) - (\log(e) + o(1))n$. We will also point out how the method of proof may be useful in other studies of the star network.

We begin with an overview of the proof technique in our theorem. The basic approach is inductive, a typical step being the assertion that if a map $\alpha: Q(d) \rightarrow S(n)$ exists, then a map $\beta: Q(d+b) \rightarrow S(n+c)$ exists, for suitable b and c, in which $\text{dilation}(\beta) \leq \max\{\text{dilation}(\alpha), M\}$ where M is an absolute constant. To accomplish this, we partition S(n+c) into $(n+1)(n+2)\cdots(n+c)$ "atoms", each atom consisting of all points in S(n+c) having a given string of length c (over an alphabet of size n+c) as suffix. The combinatorial construction which underlies our method is a way of labeling 2^b many such length c strings by vertices of Q(b) so that adjacent vertices of Q(b) label strings with O(1) disagreements in their coordinates as n and d become large. This labeling provides a natural Q(b) structure on suffixes which, when combined with the Q(d) structure already embedded inside each atom, yields a Q(d+b) structure in S(n+c) for which adjacent vertices of Q(d+b) are always at O(1) distance from each other in S(n+c). We remark that this general approach is a natural extension of the proof of Theorem 2 in [NSK], where that approach is employed for the special case c = 1. Allowing c to be arbitrarily large leads to the improved asymptotics discussed in the preceding paragraph.

We note that this approach sheds some light on the structure of star networks that goes beyond the connection with the embedding problem of this paper. Note that each of the $(n+1)(n+2)\cdots(n+c)$ atoms in S(n+c) is isomorphic in a natural way to S(n). Similarly Q(n+c) partitions into 2^c atoms, each specified by a binary length c suffix, and each isomorphic to Q(n) in a natural way. The fact that the distance in Q(n+c) between corresponding points in different hypercube atoms equals the number of coordinates in which their length c suffixes disagree, independent of c, is an important feature of the recursive structure of the hypercube. By contrast, there is no obvious relation connecting the distance in S(n+c) between corresponding points in different star atoms and the number of coordinates in which their length c suffixes disagree. Our later results give upper bounds on this distance in S(n+c) which depend only on the number of such disagreements and not on either n or c. Consequently this partition of S(n+c) together with these results could well be useful tools in inductively attacking other problems concerning distance in star networks by allowing inductive jumps of arbitrary dimension c from S(n) to S(n+c).

Let T(i,j) be the set of i-tuples over a j-letter alphabet, and let S(i,j) be the subset of T(i,j) consisting of i-tuples with distinct coordinates. The first step in the above program is the labelling of length c suffixes by hypercube vertices; in effect, the construction of a map $f:Q(b)\rightarrow S(c,n+c)$ in which adjacent vertices of Q(b) are mapped to points that are at distance bounded by an absolute constant as n, b, and c grow without bound. The construction is such that it is useful to express b, c, and n in terms of certain parameters. Given three positive integers k,p, and r satisfying $1\leq r\leq k-1$, we will take b = kp+r, and c = k, and $n=\lceil \frac{k+r}{k} 2^p \rceil -1$.

We now describe a one to one embedding $f: Q(kp+r) \rightarrow S(k, \lceil \frac{k+r}{k} 2^p \rceil + k-1)$. It will be proved later that adjacent pairs of points in the domain are sent to pairs with a constant bound on their number of coordinate disagreements. Because of the intricacy of the embedding, we break down its description into several steps.

Step 1: Rewrite the points of Q(kp+r) as strings of base 10 integers instead of as binary strings.

Let x be any vertex of Q(kp+r), represented in the usual way as a binary string $x = (x_1, x_2,..., x_{kp+r})$. We define a k-dimensional vector x' corresponding to x having base 10 integer coordinate values as follows. Let $x' = (c_1, c_2,..., c_k)$ where the c_i are base 10 integers given by $c_i = 1 + \text{Integer}(x_{(i-1)p+1}, x_{(i-1)p+2},..., x_{ip})$ for $1 \leq i \leq k-r$, and
$c_i = 1+ \text{Integer}(x_{(i-1)(p+1)-(k-r)+1}, x_{(i-1)(p+1)-(k-r)+2},..., x_{i(p+1)-(k-r)})$ for $k-r+1 \leq i \leq k$. We have thus partitioned the entries of x into k-r successive binary strings each having p successive coordinates, followed by r successive binary strings each having p+1 coordinates. We then let c_i be one plus the base 10 integer equivalent of the i'th binary string. As an example, with k=3, p=2, and r=2 the binary string x = (1,0,1,0,1,0,1,1) will now be represented by the base 10 equivalent x' = (1+2,1+5,1+3) = (3,6,4). Hence we have associated with each point x of Q(kp+r) a k-tuple x' of integers $(c_1, c_2,..., c_k)$ such that $1\leq c_i\leq 2^p$ for $1\leq i \leq k-r$ and $1\leq c_i\leq 2^{p+1}$ for $k-r+1\leq i \leq k$. For the rest of this section we assume that any point of Q(kp+r) is written in this base 10 way, and we drop the notation x', simply continuing to write x. For $1\leq i \leq k-r$ we call the variable c_i "small" and for $k-r+1 \leq i \leq k$ we call c_i "large" because the set of potential values is larger than for a small variable.

Step 2: Classify the "large" coordinate values for a given x using the vector Code(x).

We divide the interval $[2^p,2^{p+1}]$ into k subintervals as follows. Define integers $M_{t,k}$, $0\le t \le k$, by $M_{t,k} = 2^p + \lceil t\frac{2^p}{k}\rceil$. Note that the intervals $(M_{t,k}, M_{t+1,k}]$ are non-empty, and all have length at least the length of $(M_{k-1,k}, 2^{p+1}]$. In particular, we use the fact that if some n satisfies $M_{t,k}< n \le M_{t+1,k}$, then $n - M_{t,k} + M_{s,k}$ lies in the interval $(M_{s,k}, M_{s+1,k}]$.

Observe that the value of each coordinate c_i of $x\in Q(kp+r)$ can be in any one of the k-r+1 intervals $I_0=[1,M_{r,k})$, $I_1=[M_{r,k},M_{r+1,k})$, ...$I_{k-r-1}=[M_{k-2,k},M_{k-1,k})$,$I_{k-r}=[M_{k-1,k},2^{p+1}]$. All but the first of these intervals are ones in which only the large variables could take on values. We can then associate with each $x\in Q(kp+r)$ an r-dimensional vector Code(x) whose i'th coordinate indicates in which of the k-r+1 intervals the large coordinate c_{k-r+i} of x lies, $1 \le i \le r$. Specifically, we let $Code(x) = (i_1,i_2, ...,i_r)$, $0 \le i_j \le k-r$, when for each t, $1 \le t \le r$, the large coordinate c_{k-r+t} of x satisfies $c_{k-r+t}\in I_{i_t}$.

Step 3: Construct the map g: $Q(kp+r) \rightarrow T(k,\lceil \frac{k+r}{k} 2^p\rceil)$.

We define the map g: $Q(kp+r) \rightarrow T(k,\lceil \frac{k+r}{k} 2^p\rceil)$ in two stages. Starting with a given domain point $x\in Q(kp+r)$, in the first stage the coordinates of x are permuted among each other, and in the second the coordinate values of the resulting string are shifted so that their values lie in the interval $[1,\lceil \frac{k+r}{k} 2^p\rceil = M_{r,k}]$. The shift leaves the small coordinates fixed, and it translates a given large coordinate c_{k-r+t} by a constant which depends on t and the entry i_t of Code(x). To simplify the construction of g, in Step 3a we describe the permutation action, and then in Step 3b we describe the shift action. The reader may find Figure 2 helpful while digesting Step 3a, and Figure 3 helpful while digesting Step 3b.

Step 3a: Given $x\in Q(kp+r)$, construct the permutation action via the k-dimensional vector [V;r,k], where V = Code(x).

Let r and k be a pair of integers with $1\le r \le k-1$, and let $V = Code(x) = (i_1,i_2, ...,i_r)$, $0 \le i_j \le k-r$ for a point $x\in Q(kp+r)$. We will define a k-dimensional vector [V;r,k] whose entries will come from the symbol set $\{s_i: 1\le i \le k-r\}\cup\{z(m,d): 1\le m \le r, 0\le d \le k-r\}$. These symbols carry a meaning, which will be used in defining the shift action later, as follows. The entry s_i in a coordinate position t of [V;r,k] indicates that the small coordinate entry c_i of x will appear as the entry of the same position t in g(x), $1\le t \le k$. Similarly the symbol z(m,d) in a coordinate position t of [V;r,k] indicates that some shift of the large coordinate entry c_{k-r+m} of x will appear in position t of g(x), with the second index d indicating that the entry i_m of Code(x) satisfies $i_m=d$. The precise shift used and the explicit definitions of g(x) will be given in Step 3b below.

We begin by defining [V;r,k] when r=1 and $k\ge 2$ is arbitrary. The vector V has just the single entry i_1, with $0 \le i_1 \le k-1$, and we define [V;r,k] by cases based on the value of i_1. Let

$[V;r,k] = (s_1, s_2, ..., s_{k-1}, z(1,0))$ if $i_1 = 0$,
$= (z(1,1), s_2, ..., s_{k-1}, s_1)$ if $i_1 = 1$,
$= (s_1, z(1,2), ..., s_{k-1}, s_2)$ if $i_1 = 2$,
.
.
$= (s_1, s_2, ..., s_{k-2}, z(1,k-1), s_{k-1})$ if $i_1 = k-1$.

Observe then that [V;r,k] in the case $i_1 > 0$ is obtained from [V;r,k] in the case $i_1 = 0$ by inserting the entry $z(1,i_1)$ in place of the entry s_{i_1}, inserting the entry s_{i_1} in place of the entry z(1,0), and leaving the remaining k-2 entries (from the case $i_1 = 0$) the same.

Proceeding inductively, suppose for a given pair of integers r and k with $1 \le r \le k-1$ that we have constructed the k-dimensional vector [W;r,k] for all suitable r-dimensional vectors W = Code(y), y∈Q(kp+r). Consider then the pair r+1 and k+1, and we will construct the (k+1)-dimensional vector [V;r,k] for all suitable (r+1)-dimensional vectors V = Code(x), x∈Q((k+1)p+r+1). Given such a V = $(i_1,i_2, ...,i_{r+1})$, $0 \le i_j \le (k+1)-(r+1) = k-r$, let V' = $(i_1,i_2, ...,i_r)$ be the r-dimensional vector obtained by delelting the last coordinate from V. As above, the definition of [V;r+1,k+1] is given by cases based on the value of the entry i_{r+1}. We begin, in analogy with the case r=1, by letting [V;r+1,k+1] be the concatenation

$$[V;r+1,k+1] = [V';r,k]; z(r+1,0) \quad \text{if } i_{r+1} = 0.$$

For brevity, set Z = [V';r,k]; z(r+1,0). Now suppose $i_{r+1} > 0$. Then since $1 \le i_{r+1} \le k-r$, we know that there is a coordinate entry $s_{i_{r+1}}$ in the vector [V';r,k] by Observation 1 later, and hence in the vector Z. Let c be the coordinate position of $s_{i_{r+1}}$ in Z. Then we let [V;r+1,k+1] be the vector obtained from Z by inserting the entry $z(r+1,i_{r+1})$ in place of the entry $s_{i_{r+1}}$, inserting the entry $s_{i_{r+1}}$ in place of the entry z(r+1,0), and leaving the remaining k-1 entries of Z the same.

As an example, in Figures 2A and 2B we show the construction of [V;1,3] and [V;2,4] for all suitable vectors V in the figures below in the format (Vector V)→(corresponding vector [V;r,k]), where r and k are appropriate for the given V.

(Put Figure 2 here.)

<u>Step 3b</u>: We now give the map g explicitly by interpreting the vector [V;r,k] along lines sketched at the start of Step 3a.

Let x∈Q(kp+r) be given in base 10 form x = $(c_1, c_2,..., c_k)$ from Step 1, together with vectors V = Code(x) and [V;r,k] constructed in Steps 2 and 3a. Then define the k-dimensional vector g(x) as follows.

a) If $[V;r,k]_i = s_m$, $1 \le m \le k-r$, then $g(x)_i = c_m$.
b) If $[V;r,k]_i = z(m,d)$, $1 \le m \le r$ and $0 \le d \le k-r$, then
  b1) If d = 0, then $g(x)_i = c_{k-r+m}$.
  b2) If d > 0, then $g(x)_i = c_{k-r+m} - M_{r+d-1,k} + M_{m-1,k}$.

We can view this function g(x) as follows.

a') If $[V;r,k]_i = s_m$, $1 \le m \le r$, then g inserts the small variable entry c_m as the value of $[g(x)]_i$.

b1') If $[V;r,k]_i = z(m,\alpha)$ with $\alpha = 0$, then g inserts the large variable entry c_{k-r+m} (which lies in I_0 because $\alpha = 0$) as the value of $[g(x)]_i$.

b2') If $[V;r,k]_i = z(m,\alpha)$ with $\alpha > 0$, then g linearly shifts c_{k-r+m} (which lies in I_α) so the result lies in the interval $[M_{m-1,k}, M_{m,k}]$ in the same relative position as it lies in I_α, and then inserts this translate as the value of $[g(x)]_i$.

In Figure 3 we illustrate the shift action of g described in b2) and b2') .

(Put Figure 3 here.)

Step 4: Construction of the map f: $Q(kp+r) \to S(k, \lceil \frac{k+r}{k} 2^p \rceil + k-1)$

Recall that k-tuples in S(k,m) must have k distinct coordinates, while an image g(x) might have redundancies among its coordinates. From the map g, we now construct the required map f by introducing k-1 new and distinct letters $N_2, N_3, ..., N_k$ that are used to eliminate these redundancies as follows. If the i'th entry of g(x) matches any earlier entry of g(x), replace it by the letter N_i to eliminate the redundancy. Formally, for each i let $f(x)_i = g(x)_i$ if $g(x)_i \ne g(x)_t$ for all $t < i$, and let $f(x)_i = N_i$ otherwise. This completes the definition of the map f.

We will need several simple observations about the above labeling scheme f. Their proofs are omitted since they are either transparent, or follow from a straightforward induction that follows the plan of the scheme; that is, treating first the case r = 1 and arbitrary k, and then the case r+1 and k+1 given the case r and k.

Observation 1: For any vector $V = (i_1, i_2, ..., i_r)$, the entries of the vector [V;r,k] are given by $\{s_i: 1 \le i \le k-r\} \cup \{z(t,i_t): 1 \le t \le r\}$.

Observation 2: Let $V = (i_1, i_2, ..., i_{r+1})$ and let $V' = (i_1, i_2, ..., i_r)$ be its length r prefix. Then [V;r+1,k+1] can be obtained from [V';r,k] by replacing the entry $s_{i_{r+1}}$ by $z(r+1, i_{r+1})$, and then appending $s_{i_{r+1}}$ as the (k+1)'st entry.

Observation 3: Each entry of g(x) is a linear shift of a unique entry of x, this entry being determined by [Code(x);r,k] (and therefore determined by Code(x)).

We begin an analysis of the labeling scheme f with the following lemma . We refer to the correspondence $V \to [V;r,k]$ constructed in step 3a as the map $C_{r,k}$.

Lemma 1: Let k and r be positive integers, $r < k$, $k \ge 2$. Then the map $C_{r,k}: V \to [V;r,k]$ constructed in Step 3a has the following properties.

(a) $C_{r,k}$ is a one to one map.

(b) If V_1 and V_2 are suitable k-dimensional vectors disagreeing in exactly one coordinate position, then $C_{r,k}(V_1)$ and $C_{r,k}(V_2)$ disagree in at most three coordinate positions.

<u>Proof</u>: In analogy with the construction of the map $C_{r,k}$ itself, we first prove the lemma for r = 1 and arbitrary k ≥ 2, and later we prove it for arbitrary r and k by induction.

When r = 1, V has a only a single integer entry lying between 0 and k-1. Thus for two such distinct vectors $V_1 = \{s\}$ and $V_2 = \{t\}$, the construction for r = 1 shows that $[V_1;r,k]$ and $[V_2;r,k]$ disagree in precisely coordinates s,t, and k. Parts a) and b) follow immediately.

Proceeding inductively, assume the lemma holds for map $C_{r,k}$ and consider the map $C_{r+1,k+1}$. Consider first part a), and let V and W be distinct (r+1)-dimensional vectors in the domain of $C_{r+1,k+1}$. Suppose first that $V_{r+1} \neq W_{r+1}$. Then since $[C_{r+1,k+1}(V)]_{k+1} = s_{V_{r+1}}$ if $V_{r+1} > 0$ and $[C_{r+1,k+1}(V)]_{k+1} = z(r+1,0)$ if $V_{r+1} = 0$ (with the same statement holding for W in place of V), it follows that $[C_{r+1,k+1}(V)]_{k+1} \neq [C_{r+1,k+1}(W)]_{k+1}$ and hence $[C_{r+1,k+1}(V)] \neq [C_{r+1,k+1}(W)]$. So assume $V_{r+1} = W_{r+1} = i$ for some i, $0 \leq i \leq k-r$. Then V and W have distinct length r prefixes, say V' and W' respectively. Let c_1 and c_2 be the coordinate positions of $C_{r,k}(V')$ and $C_{r,k}(W')$ respectively at which s_i can be found. By Observation 2, $C_{r+1,k+1}(V)$ (resp. $C_{r+1,k+1}(W)$) is obtained from $C_{r,k}(V')$ (resp. $C_{r,k}(W')$) by replacing the entry s_i in position c_1 (resp. c_2) by z(r+1,i), and then appending s_i as the (r+1)'st entry. Suppose first that $c_1 = c_2$. Since all disagreements between $C_{r,k}(V')$ and $C_{r,k}(W')$ occur at positions other than $c_1 = c_2$, such disagreements remain in those same positions in the vectors $C_{r+1,k+1}(V)$ and $C_{r+1,k+1}(W)$. Since by induction $C_{r,k}$ is one to one, there really are such disagreements, so $[C_{r+1,k+1}(V)]_{k+1} \neq [C_{r+1,k+1}(W)]_{k+1}$. So assume $c_1 \neq c_2$. Then z(r+1,i) is an entry of $C_{r+1,k+1}(V)$ and $C_{r+1,k+1}(W)$ in different positions. By Observation 1, for each t, $1 \leq t \leq m$, there is exactly one entry of the form z(t,*) (that is; a "z entry" with first coordinate entry t) in any vector in the image of the map $C_{m,n}$. Thus z(r+1,i) can appear only once in $C_{r+1,k+1}(V)$ and $C_{r+1,k+1}(W)$, so $C_{r+1,k+1}(V) \neq C_{r+1,k+1}(W)$, and thus $C_{r+1,k+1}$ is one to one.

We now prove b) for $C_{r+1,k+1}$ assuming it is true for $C_{r,k}$. Again let V and W be distinct (r+1)-dimensional vectors disagreeing in exactly one position, say the d'th coordinate, $1 \leq d \leq r+1$. As above, let V' and W' be their length r prefixes. For brevity, set $z_1 = C_{r+1,k+1}(V)$, $z_2 = C_{r+1,k+1}(W)$, $y_1 = C_{r,k}(V')$, and $y_2 = C_{r,k}(W')$.

Assume first that d = r+1, say with $V_{r+1} = a$, and $V_{r+1} = b$. Then V' = W', so $y_1 = y_2 = Y$ for some k-dimensional vector Y. Let p (resp. q) be the coordinate position of Y where s_a (resp. s_b) occurs. Then by Observation 2, it follows that z_1 and z_2 can disagree only in the three positions p,q, and r+1.

Assume then that d < r+1. We claim that z_1 and z_2 agree in one more coordinate than do y_1 and y_2. To see this, let $t = (z_1)_{r+1} = (z_2)_{r+1}$. If t = 0, then by definition $z_1 = y_1;z(r+1,0)$ and $z_2 = y_2;z(r+1,0)$, so z_1 and z_2 have the last coordinate in common (with entry $z_{r+1}(0)$) in addition to all coordinates that y_1 and y_2 had in common. Suppose then that t ≠ 0. Then by Observation 2 we see that again z_1 and z_2 have the last coordinate in common (this time with entry s_t) in addition to the coordinates that y_1 and y_2 had in common. The claim follows. But now z_1 and z_2 are vectors having one more entry than the vectors y_1 and y_2. The claim then implies that z_1 and z_2 can disagree in no greater number of coordinates than do y_1 and y_2. But by induction y_1 and y_2 disagree in at most three coordinates, so the same is true of z_1 and z_2, proving property (b). ∎

We now prove the critical property that we need from the map f, namely, that it sends adjacent points to points with few coordinate disagreements.

Lemma 2: Let $2 \le k \le 2^p$, and $1 \le r \le k-1$. Then the map

$$f: Q(kp+r) \rightarrow S(\, k, \lceil \frac{k+r}{k} 2^p \rceil + k-1 \,)$$

produced in the labeling procedure satisfies

(a) f is one to one.

(b) For any two adjacent points x and y in Q(kp+r), the image points f(x) and f(y) differ in at most six coordinates.

Proof: For part (a), it suffices to show that the map g constructed in Step 3 of the procedure is one to one, since this property is preserved after introduction of the redundancy breaking variables N_2, N_3, ..., N_k to form f.

Suppose then that g(x) = g(y) for distinct $x,y \in Q(kp+r)$. If Code(x) ≠ Code(y), then the injectivity of $C_{r,k}$ implies that g(x) ≠ g(y), a contradiction. Suppose then that Code(x) = Code(y). Then by Observation 3 it follows that x = y, proving part (a).

Now consider part (b), and let x and y be adjacent vertices of Q(kp+r), so that they disagree in exactly one coordinate. Assume first that Code(x) = Code(y). Then from the defintion of g(x), we see that the coordinate entries of g(x) are identical functions of the coordinates of x as the coordinate entries of g(y) are of y. It follows that g(x) and g(y) can disagree in only one coordinate, namely, the unique one which depends only on the coordinate at which x and y disagree.

We may then assume that Code(x) ≠ Code(y). Then by Lemma 1, $C_{r,k}$(Code(x)) and $C_{r,k}$(Code(y)) disagree in at most three coordinates. Let B(x,y) be the set of the remaining (at least) k-3 coordinates. Now let $t \in B(x,y)$. Then for some u, $1 \le u \le k$, the entries $g(x)_t$ and $g(y)_t$ are identical linear shifts of the coordinate entries x_u and y_u of x and y respectively.

Claim: For such a choice of t and u, we have $x_u = y_u$.

Proof of claim: Suppose not, and let $\delta = C_{r,k}(\text{Code}(x))_t = C_{r,k}(\text{Code}(y))_t$. Either $\delta = s_i$ for some i, $1 \le i \le k-r$, or $\delta = z(m,\alpha)$ for suitable m and α. In the former case , part a) in step 3b of the labeling procedure shows that u = i, so x and y disagree in the i'th coordinate. In the latter case we know that g(x) (resp. g(y)) is a linear shift of x_{k-r+m} (resp. y_{k-r+m}), so u = k-r+m. Thus x and y disagree either in coordinate k-r+m, $m \ge 1$, or in coordinate i, $1 \le i \le k-r$.

On the other hand we know that Code(x) and Code(y) disagree at some coordinate e, so $C_{r,k}$(Code(x)) and $C_{r,k}$(Code(y)) disagree at some corresponding coordinate position p, where they must have entries z(e,β) and z(e,γ) respectively with $\beta \neq \gamma$. It follows that $x_{k-r+e} \neq y_{k-r+e}$ since by the meaning of the "z" entries we have $x_{k-r+e} \in I_\beta$ while $y_{k-r+e} \in I_\gamma$. We also have $e \neq m$ (where m is from the above paragraph), since z(e,β) is an entry of $C_{r,k}$(Code(x)) appearing in a position $p \notin B(x,y)$, while z(m,α) appears in a position $t \in B(x,y)$.

Summarizing, we have found that x and y disagree in at least two coordinates; these being the pair {k-r+m, k-r+e} for e,m≥1, or the pair {i,k-r+e} for 1≤i≤k-r and e≥1. This contradicts x and y being adjacent in Q(kp+r), proving the claim.

The claim, together with Observation 3, show that g(x) and g(y) agree in all coordinates $t \in B(x,y)$. Hence g(x) and g(y) can disagree in at most three coordinates, namely, the ones outside the set B(x,y).

Finally turning our attention to the map f, we claim that for any points x and y of Q(kp+r), if g(x) and g(y) disagree in C coordinates then f(x) and f(y) can disagree in at most 2C coordinates. For let c be be a coordinate such that $g(x)_c = g(y)_c$ but $f(x)_c \neq f(y)_c$. Then we must have $g(x)_c = g(y)_c = d$ (for some d) which is repeated in (wlog) g(y), and that $f(x)_c = N_c$ while $f(y)_c = d$. This happens only if coordinate c is the second occurrence (in order of coordinate index) of the letter d in g(y), while the coordinate b of the first occurrence of d in g(y) satisfies $d = g(x)_b \neq g(y)_b$. Thus we see that each disagreement between f(x) and f(y) not present between g(x) and g(y) arises from some "original" disagreement between g(x) and g(y) (i.e. at coordinate b), and that each original disagreement between g(x) and g(y) gives rise to at most one new disagreement between f(x) and f(y). The claim follows.

Since there are at most 3 disagreements in the coordinates of images of adjacent points under the map g, there can be at most 6 disagreements in such images under f. The lemma is thus proved. ∎

3. The Main Result

With the preceding lemma we have found a way of labelling 2^{kp+r} many k-letter strings (which will play the role of suffixes) by vertices of Q(kp+r) so that adjacent vertices label strings with few disagreements. Recall that in our inductive approach we assume a map Q(d)→S(n), and we partition S(n+k) into (n+1)(n+1)···(n+k) atoms, each atom consisting of all points in S(n+k) having a given length k string (over an alphabet of size n+k) as suffix. Our goal is to obtain a map Q(d+kp+r)→S(n+k) by overlaying the Q(kp+r) structure on suffixes with the assumed Q(d) in each of 2^{kp+r} atoms to get the desired Q(d+kp+r) structure in S(n+k), and to know also that adjacent vertices in Q(d+kp+r) are not far apart in their host S(n+k). To accomplish this we need to find maps $m_{ij}:A_i \to A_j$ between pairs of atoms A_i and A_j such that if the suffixes of A_i and A_j have few disagreements, then $dist_{S(n+k)}(z,m_{ij}(z))$ is small for any $z \in A_i$. These maps in effect tie together the 2^{kp+r} different copies of Q(d) in our 2^{kp+r} atoms to yield the desired Q(d+kp+r). The next lemma builds these maps, after we first develop some useful notation.

Let the star network S = S(d+k) be 1-represented on the symbol set {1,2,...,d+k}, with d,k > 0. With each k-tuple $I = (i_1, i_2, .., i_k)$, $1 \le i_t \le d+k$, we may associate the vertex subset <u>S(I)</u> of S(d+k) consisting of all points whose suffix of length k is I, i.e. $S(I) = \{z \in S(d+k): z_{d+t} = i_t,\ 1 \le t \le k\}$. Now fix a particular k-tuple, say $I_0 = (1,2,..,k)$. For any other k-tuple $I = (i_1, i_2, .., i_k)$ we define a digraph <u>D(I)</u> and an associated one-to-one map $\beta_I: S(I_0) \to S(I)$. The vertices of D(I) are the symbols 1,2,...,d+k. The arc set E(D(I)) of D(I) will be a disjoint union $E(D(I)) = E_1 \cup E_2$ of two sets defined as follows. First, we let

$$E_1 = \{ r \to i_r : 1 \le r \le k \} \cup \{ r \to r : r \notin \{i_1, i_2, .., i_k\} \cup \{1,2,..,k\} \}.$$

Observe that the (weakly) connected components of the digraph $\langle E_1 \rangle$ induced by the arcs in E_1 are directed cycles (possibly loops) and directed paths. Each directed path P has an initial vertex, call it <u>initial(P)</u>, of indegree 0, and a terminal vertex, call it <u>terminal (P)</u>, of outdegree 0. We now let

$$E_2 = \{ \text{terminal}(P) \to \text{initial}(P) : P \text{ is a directed path component in } \langle E_1 \rangle \}.$$

With this, the digraph D(I) has been defined, and we observe that each connected component of D(I) is a directed cycle. For future reference note also that each component has at most one vertex u satisfying u > k. Figure 4 shows the digraphs D(I) for three different 3-tuples I in the case d=k=3.

(Put Figure 4 here.)

We can now proceed to the map β_I. First define α_I to be the one-to-one map from the set {1,2,..,d+k} to itself associated with D(I); that is, for any letters x,y in this set we let $\alpha_I(p) = q$ if and only if $p \to q$ is an arc in D(I). Now we define the map $\beta_I: S(I_0) \to S(I)$ by applying α_I coordinatewise; that is, for any point $x = (x_1, x_2, .., x_{d+k})$ of $S(I_0)$ we let $\beta_I(x) = (\alpha_I(x_1), \alpha_I(x_2), .., \alpha_I(x_{d+k}))$. We see that indeed $\beta_I(S(I_0)) = S(I)$, using the definition of E_1. Now given any two of the subsets S(J) and S(I) in our partition of S(d+k), the map between them which interests us is $\beta_I\beta_J^{-1}: S(J) \to S(I)$.

<u>Lemma 3</u>: Let I and J be two k-tuples in S(k,d+k) disagreeing in s positions. Then for any $z \in S(J)$ we have $dist_{S(d+k)}(z,\beta_I(\beta_J^{-1}(z))) \le \lfloor \frac{9s}{2} \rfloor$.

<u>Proof</u>: Let $H = \{p: \alpha_I(\alpha_J^{-1}(p)) \ne p\}$, and write $h = |H|$. The restriction α of $\alpha_I\alpha_J^{-1}$ to the set $H \cup \{1\}$ is a permutation on $H \cup \{1\}$. Thus

$$dist_{S(d+k)}(z,\beta_I(\beta_J^{-1}(z))) \le diam(S(h+1)) = \lfloor \tfrac{3}{2} h \rfloor,$$

where the inequality holds because α acts as a permutation in S(h+1) when 1-represented on the symbol set $H\cup\{1\}$.

We will relate h to s. Note that $h = |\{p: \alpha_I^{-1}(p) \neq \alpha_J^{-1}(p)\}|$, because $\alpha_I(\alpha_J^{-1}(p)) \neq p$ if and only if $\alpha_J^{-1}(p) \neq \alpha_I^{-1}(p)$. Also, $|\{p: \alpha_I^{-1}(p) \neq \alpha_J^{-1}(p)\}| = |\{p: \alpha_I(p) \neq \alpha_J(p)\}|$, so to bound h it suffices to bound the number of elements in $A = \{p: \alpha_I(p) \neq \alpha_J(p)\}$.

Let $Y = A\cap\{1,2,...,k\}$. By the definition of s, we have that $|Y| = s$. We prove that $|A\backslash Y| \leq 2s$. Consider any $p\in A\backslash Y$. Let $C_I(p)$ and $C_J(p)$ denote the cycles containing p in D(I) and D(J) respectively, recalling that all points other than p in these cycles are numbers less than or equal to k. We show that for each $p\in A\backslash Y$ there exists a corresponding $y(p)\in Y$ such that $y(p)\in C_I(p)\cup C_J(p)$. Suppose for contradiction that neither $C_I(p)$ nor $C_J(p)$ contains a point in Y. Backtracking around these cycles, we have $\alpha_I^{-1}(p) = \alpha_J^{-1}(p)$, and $\alpha_I^{-1}(\alpha_I^{-1}(p)) = \alpha_J^{-1}(\alpha_J^{-1}(p))$, and $\alpha_I^{-1}(\alpha_I^{-1}(\alpha_I^{-1}(p))) = \alpha_J^{-1}(\alpha_J^{-1}(\alpha_J^{-1}(p)))$ and so on until we reach an equation whose sides are both $\alpha_I(p)$ or both $\alpha_J(p)$. These two possibilities cannot occur simultaneously, else $C_I(p) = C_J(p)$, contradicting that $p\in A$. If both sides equal $\alpha_I(p)$, then $\alpha_J^{-1}(\alpha_I(p))$ is in Y, a contradiction. A similar contradiction follows if both sides equal $\alpha_J(p)$. We have thus shown the existence of the desired point y(p) in $[C_I(p)\cup C_J(p)]\cap Y$. But observe that no point of Y is associated with more than two p's in $A\backslash Y$, because each point of Y is in only one cycle of D(I) and one cycle of D(J). Therefore $|A\backslash Y| \leq 2|Y| = 2s$.

Putting these facts together we obtain $h = |Y| + |A\backslash Y| \leq s + 2s = 3s$, and hence for any $z\in S(J)$ we have $\text{dist}_{S(d+k)}(z,\beta_I(\beta_J^{-1}(z))) \leq \lfloor \frac{3}{2} h \rfloor \leq \lfloor \frac{9s}{2} \rfloor$. ∎

In the next lemma we tie together the 2^{kp+r} many Q(d)'s embedded inside the atoms by using the maps $\beta_I\beta_J^{-1}$, and we obtain an upper bound for the dilation of the resulting map $Q(d+kp+r)\rightarrow S(n+k)$.

<u>Lemma 4</u>: Let n,k be integers satisfying $2^p \leq n < n+k < 2^{p+1}$. Suppose that $n \geq (1 + \frac{r}{k})2^p - 1$, where $r \leq k-1$. Then for any injective map $g:Q(d)\rightarrow S(n)$, there is an injective map $h:Q(d+kp+r)\rightarrow S(n+k)$ of dilation at most max{dilation(g), 27}.

<u>Proof</u>: Before defining the map h formally, we explain its basic idea. Having partitioned S(n+k) into (n+1)(n+2)···(n+k) atoms, a typical atom being a set S(I), we find an "image" of Q(d) in each S(I) in the following way. Using the map g of the hypothesis, construct a map $s:Q(d)\rightarrow S(I_0)$ (where I_0 is the k-tuple (1,2,..,k)) by the following steps. Let $z\in Q(d)$.

1) Find $g(z) = (g_1, g_2, ..., g_n)$, and let $g'(z) = (g_1+k, g_2+k, ..., g_n+k)$.

2) Let $s(z) = g'(z);I_0$.

We have thus found an image of Q(d) in $S(I_0)$, and now for any k-tuple I we let β_I(s(Q(d)) be an image of Q(d) in S(I). Finally, we tie together these images to get an image of Q(d+kp+r) [in S(n+k)] by regarding β_I(s(Q(d))) as the image of the subcube of Q(d+kp+r) of dimension d consisting of the points having the binary string $f^{-1}(I)$ as length kp+r suffix, where f is the map of lemma 2.

We now define the map h formally. Consider the image C of the map in lemma 2. Observe that any point of Q(d+kp+r) may be written in the concatenated form $z;f^{-1}(I)$ for some $z \in Q(d)$ where by lemma 2 each coordinate c of I satisfies $1 \le c \le \lceil \frac{k+r}{k} 2^p \rceil + k-1 \le n+k$, so that I is a k-letter suffix of points in S(n+k). We then define our map by $h(z;f^{-1}(I)) = \beta_I(s(z))$.

To verify the statement on dilation, let $x_1 = z_1;f^{-1}(I_1)$ and $x_2 = z_2;f^{-1}(I_2)$ be adjacent points in Q(d+kp+r).

Suppose first that $I_1 = I_2 = I$ for some k-tuple I, so that z_1 and z_2 are adjacent points of Q(d). Thus $h(x_1)$ and $h(x_2)$ belong to the same image $\beta_I(s(Q(d)))$ of Q(d). Observe that for any $x,y \in S(I_0)$ we have

$$\text{dist}_{S(I_0)}(x,y) = \text{dist}_{S(I)}(\beta_I(x),\beta_I(y)).$$

To see this, let $\pi = \pi_1\pi_2\cdots\pi_t$ be a product of transpositions through the first position keeping the suffix I_0 fixed and taking x to y. Then $\pi' = \pi_1'\pi_2'\cdots\pi_t'$ is a product of transpositions through the first position keeping the suffix I fixed and taking $\beta_I(x)$ to $\beta_I(y)$, where π_i' is obtained from π_i by replacing any symbol $u > k$ by $\alpha_I(u)$ (since β_I is just the permutation on length n+k strings obtained by applying α_I coordinatewise). It now follows that

$\text{dist}_{S(I)}(h(x_1),h(x_2)) = \text{dist}_{S(I_0)}(s(z_1),s(z_2)) = \text{dist}_{S(n)}(g(z_1),g(z_2)) \le \text{dilation}(g)$.

Next assume that $z_1 = z_2$, so that $f^{-1}(I_1)$ and $f^{-1}(I_2)$ are adjacent points of Q(kp+r), while $h(x_1)$ and $h(x_2)$ are corresponding points in different images $\beta_{I_1}(s(Q(d)))$ and $\beta_{I_2}(s(Q(d)))$ of Q(d). Thus $h(x_2) = \beta_{I_2}(\beta_{I_1}^{-1}(h(x_1)))$, so by lemma 3 we have $\text{dist}_{S(n+k)}(h(x_1),h(x_2)) \le \lfloor \frac{9s}{2} \rfloor$, where s is the number of disagreements between I_1 and I_2. But since $f^{-1}(I_1)$ and $f^{-1}(I_2)$ are adjacent points of Q(kp+r), we know by lemma 2 that $s \le 6$. Thus $\text{dist}_{S(n+k)}(h(x_1),h(x_2)) \le 27$, and the lemma is proved. ∎

Finally it remains to apply iteratively the inductive approach embodied in lemma 4. The only thing left to do in order to complete the proof of Theorem 1 is to estimate the asymptotic behavior of d from the map $Q(d)\to S(n)$ that results from these iterations.

The following notation will be used. Let $n_1< n_2< ...< n_t$ and $m_1< m_2< ...< m_t$ be two increasing sequences of integers. A sequence of one-to-one maps h_1: $Q(n_1)\to S(m_1)$, h_2:$Q(n_2)\to S(m_2)$, ..., h_t: $Q(n_t)\to S(m_t)$ will be called a mapping sequence, and will be denoted $S(m_1)\to S(m_2)\to \cdots\to S(m_t)$. (Mention of the maps h_i and the $Q(n_i)$ is omitted in this notation since they will be fixed by context.) We define the bit accumulation of this mapping sequence, denoted $\text{bit}(S(m_1)\to S(m_2)\to \cdots\to S(m_t))$, to be the integer $n_t - n_1$.

Proof of theorem 1: Let $n = 2^{p+1}$. The plan is to build a mapping sequence $S(4)\to\cdots\to S(2^{p+1})$ in which each step is obtained by applying lemma 4 or Theorem 2. It would then follow that there is a map $Q(d)\to S(n)$ of bounded dilation, where $d = \mathrm{bit}(S(4)\to\cdots\to S(2^{p+1}))$. Hence it suffices to show that the bit accumulation of the mapping sequence we build is at least $n\log(n) - (\frac{3}{2} + o(1))n$.

We describe the construction of the mapping subsequence $S(2^p)\to\cdots\to S(2^{p+1})$, the full sequence being obtained by composing these subsequences. Let $t = \lceil 2^{p/3} \rceil$, and consider the sequence $2^p=b_0 < b_1 < b_2 < ...< b_t = 2^{p+1}$ defined by $b_i = \lceil (1 + \frac{i}{t}) 2^p \rceil$. Suppose that for some i we have already built the mapping sequence $S(2^p)\to\cdots\to S(b_i)$. We then build the sequence $\mu_i=[S(b_i)\to\cdots\to S(b_{i+1})]$, which when composed with the first sequence gives us the extended sequence $S(2^p)\to\cdots\to S(b_{i+1})$. Repeating this process t times gives us the desired sequence $S(2^p)\to\cdots\to S(2^{p+1})$.

To build μ_i, suppose that the subsequence $S(b_i)\to\cdots\to S(u)$ has been constructed, $b_i \le u < b_{i+1}$. We then extend this subsequence by applying either lemma 4 or Theorem 2, depending on where u is in the interval (b_i,b_{i+1}).

(a) If $b_{i+1}-u \ge t$, then apply lemma 4 with $d = u$, $r = i$, and $k = t$. The result is the extended sequence $S(b_i)\to\cdots\to S(u)\to S(u+t)$, and the addition to the bit accumulation is $tp+i$.

(b) If $b_{i+1}-u < t$, then apply Theorem 2 iteratively $b_{i+1}-u$ times, obtaining the extended sequence $S(b_i)\to\cdots\to S(b_{i+1})$. The addition to the bit accumulation is $\mathrm{bit}(S(u)\to\cdots\to S(b_{i+1})) = (b_{i+1}-u)p$.

By applying either (a) or (b) when indicated, we eventually obtain the sequence μ_i, with

$$\mathrm{bit}(\mu_i) \ge (b_{i+1}-b_i)p + \frac{i}{t}(b_{i+1}-b_i) - i = (b_{i+1}-b_i)p + \frac{i}{t^2} 2^p - i.$$

Composing these sequences we obtain

$$\begin{aligned}
\mathrm{bit}(S(2^p)\to\cdots\to S(2^{p+1})) &= \sum_{i=0}^{t-1} \mathrm{bit}(\mu_i) \\
&\ge p2^p + \sum_{i=0}^{t-1} \left(\frac{i}{t^2} 2^p - i \right) \\
&= p2^p + 2^{p-1} + O(t^2) \\
&= p2^p + 2^{p-1} + O(2^{2p/3}).
\end{aligned}$$

Finally using the identity $\sum_{s=1}^{d} s2^s = 2^{d+1}(d-1) + 2$ we obtain

$$\begin{aligned}
\mathrm{bit}(S(4)\to\cdots\to S(2^{p+1})) &= \sum_{s=1}^{p} (s2^s + 2^{s-1} + O(2^{2s/3})) \\
&= n\log(n) - (\frac{3}{2} + o(1))n. \qquad \blacksquare
\end{aligned}$$

We now interpolate the result of Theorem 1. That is, for a given n we ask how large we can make d and still be able to construct a one to one map $Q(d)\rightarrow S(n)$ of dilation $O(1)$ when n is arbitrary and not necessarily a power of 2. An answer is already provided in the proof of Theorem 1 just by computing the bit accumulation of the mapping sequence $S(2^p)\rightarrow\cdots\rightarrow S(n)$, $p = \lfloor \log(n) \rfloor$, specified in the proof of the theorem, and then adding this to $bit(S(4)\rightarrow\cdots\rightarrow S(2^p))$ as given in the theorem itself applied to powers of 2. The result is the following.

<u>Corollary 1.1</u>: Let n be a positive integer and $M = 2^p$, where $p = \lfloor \log(n) \rfloor$. Then there is map $Q(d)\rightarrow S(n)$ of dilation $O(1)$, provided

$$d \leq M\log(M) - (\frac{3}{2} + o(1))M + p(n - M) + \frac{(n - M)^2}{2M} .$$

<u>Sketch of proof</u>: As indicated in the above discussion we can build a mapping sequence $S(4)\rightarrow\cdots\rightarrow S(n)$ in the two stages $S(4)\rightarrow\cdots\rightarrow S(2^p)$ followed by $S(2^p)\rightarrow\cdots\rightarrow S(n)$ as in the proof of Theorem 1. The bit accumulation of the first stage is given by the first two terms on the right according to Theorem 1. The bit accumulation of the second stage is $p(n - M) + \sum_{i=0}^{m} \{\frac{i}{t^2} 2^p - i\}$, where m is the integer for which $(1 + \frac{m}{t})2^p \leq n < (1 + \frac{m+1}{t})2^p$. The sum becomes $2^p(\frac{(n - M)^2}{2M^2})$, where we have neglected error terms which can be subsumed in the $o(1)M$ term already present. The corollary follows. ∎

Finally we note that Theorem 1 remains true when S(n) is replaced by the pancake graph P(n) or the transposition graph TranS(n) discussed in the introduction, only with different constants in the O(1). This is clear for TranS(n) since S(n) is a subgraph of TranS(n). To see this for P(n), it suffices to show that any transposition (1,i) applied to a string in S(n) is a product of at most D of the prefix reversals ρ_k for a constant D. For then any two strings x and y of S(n) satisfying $dist_{S(n)}(x,y) \leq L$ must also satisfy $dist_{P(n)}(x,y) \leq DL$, from which the corresponding O(1) bound on dilation of maps into P(n) follows.

We show that D = 4. Let AxBy be a string in S(n), positionally represented, where A and B are letters while x and y are strings. Also denote by x^R the reversal of the string x. Then a sequence of four prefix reversals leading from AxBy to BxAy is given by $AxBy \rightarrow x^RABy \rightarrow xABy \rightarrow Ax^RBy \rightarrow BxAy$.

4. Appendix: Embeddings into small star networks

Our main result, Theorem 1, gives asymptotics (as $n\rightarrow\infty$) for the maximum d such that an iterated application of lemma 4 yields a O(1) dilation map $Q(d)\rightarrow S(n)$. On the other hand, there may be practical reasons for limiting ourselves to finding such maps only when n is bounded, perhaps even small. In that case we might hope to obtain a bound on the dilation D which is even better than the bound $D \le 27$ given by the theorem. The latter bound comes from applying lemma 4 iteratively with k arbitrary large, and then using lemmas 2 and 3 (with s = 6) to deduce that $D \le 27$. If we apply lemma 4 with k tightly upper bounded, then we can avoid using the uniform bound $s \le 6$ (and the resulting $D \le 27$) that applied to all k and instead calculate D directly, taking advantage of k being small.

We illustrate this approach, summarizing our results in Table 1. As our first example, we limit ourselves to using lemma 4 with k=2 and r=1, together with Theorem 2. Under this limitation, we show that the conclusion of lemma 4 holds with 27 replaced by 6.

Following along in the proof of lemma 4, the assumption k=2 and r=1 takes effect in the last paragraph. In the notation used there, we must show that $dist_{S(d+k)}(h(x_1),h(x_2)) \le 6$, where $h(x_1)$ and $h(x_2)$ are related by $h(x_2) = \beta_{I_2}(\beta_{I_1}^{-1}(h(x_1)))$. The 2-tuples I_1 and I_2 are in the image of the map f of lemma 2 in the case k=2 and r=1. The pair $h(x_1)$ and $h(x_2)$ may be classified into to one of several categories, defined by whether their respective 2-tuple suffixes I_1 and I_2 agree in some coordinate and whether either of their two coordinate entries have value at most 2. These different categories lead to different upper bounds on $dist(h(x_1),h(x_2))$.

We describe the case leading to the largest such bound; namely, the one in which I_1 and I_2 disagree in both coordinates and these coordinates have values bigger than 2 in both I_1 and I_2. In that case we can write $h(x_1) = P_1;AB$ and $h(x_2) = P_1;CD$, where I_1=AB, I_2=CD, while A,B,C,D are distinct letters all bigger than 2 and the P_i are the length d+k-2 prefixes of the $h(x_i)$. As an aid in deriving the distance bound, we draw in Figure 5a the the "relevant" connected components of the digraphs $D(I_1)$ and $D(I_2)$. These are the components which $D(I_1)$ and $D(I_2)$ do not have in common. We note that the permutation $\alpha_{I_2}\alpha_{I_1}^{-1}$ leaves fixed every letter not in one of the relevant components. As for the eight letters in the relevant components, the digraph shows that the cycle structure of $\alpha_{I_2}\alpha_{I_1}^{-1}$ on these is (1AC)(2BD) = (1C)(1A)(1D)(1B)(12)(1D). Thus $dist(h(x_1),h(x_2)) = 6$. It can be shown that in all other possible categories defining the pair $h(x_1),h(x_2)$ we have $dist(h(x_1),h(x_2)) \le 6$. It follows that lemma 4, with k and r limited by k=2 and r=1, holds with the 27 replaced by 6, as claimed.

(Put Figure 5 here.)

The results indicated in the dilation 6 column of Table 1 can now be obtained using Theorem 2 and the "new" lemma 4 for k=2 and r=1. The initial entries corresponding to $1 \le n \le 5$ are found by noting that in these cases we have $diam(S(n)) = \left\lfloor \frac{3(n-1)}{2} \right\rfloor \le 6$. Hence <u>any</u> one to one embedding $Q(d) \rightarrow S(n)$, $2^d \le n!$, has dilation≤6, and in particular this is so when $d = \lfloor \log(n!) \rfloor$. So we can fill in the first five entries a_n, $1 \le n \le 5$, with $a_n = \lfloor \log(n!) \rfloor$. The typical step in filling out the remaining entries is as follows. Given that the entries a_i, $1 \le i \le n$, of the first n rows have been computed, the entry a_{n+1} can be computed in one of two ways.

(a) Use Theorem 2 with n playing the role of d. The result is $a_{n+1} = a_n + \lfloor \log(n+1) \rfloor$

(b) If the hypothesis of lemma 4 is applicable for k=2,r=1, then apply the lemma with n-1 playing the role of d; that is, if $n-1 \ge \frac{3}{2} 2^p - 1$, where $p = \lfloor \log(n) \rfloor$, then apply lemma 4. The result is $a_{n+1} = a_{n-1} + 2p + 1$.

One chooses the computation that gives the greatest value for a_{n+1}, choosing either one if there is a tie.

As a second example, suppose we limit our application of lemma 4 to cases in which $k \le 3$. This allows the case k=2, r=1 considered above, as well as the new cases k=3,r=1 and k=3,r=2. It can be shown that lemma 4 restricted to these cases yields a conclusion in which the 27 is replaced by 8. One can check that in all these cases we have $dist(h(x_1),h(x_2)) \le 8$, from which this conclusion follows. We sketch the analysis behind this distance bound in one possible category of pairs $h(x_1)$ and $h(x_2)$, one which in fact achieves the maximum $dist(h(x_1),h(x_2)) = 8$. This category, arising from the case k=3, r=2, is defined by $h(x_1) = P_1;I_1$, $h(x_2) = P_2;I_2$ where $I_1 = AN_2B$ and $I_2 = CAD$, A,B,C,D are distinct and all greater than 2, and N_2 is one of the letters used (in the proof of lemma 2) for breaking redundancies. (Recall that for k=3 and r=2 the points g(z) and g(z') in the proof of lemma 2 can disagree in at most two coordinates. Hence the 3-tuples $g(z)=J_1$ and $g(z')=J_2$ have at least one coordinate in common, so that we could have J_1=AAB and J_2=CAD. Now $f(z)=I_1$ is obtained from J_1 by introducing N_2 as shown.) The digraphs D_{I_1} and D_{I_2} are shown in Figure 5b, with the result that the cycle structure of $\alpha_{I_2}\alpha_{I_1}^{-1}$ on non-fixed points is $(12N_2AC)(BD3)$, thereby yielding $dist(h(x_1),h(x_2)) = 8$ as claimed. We omit an analysis of the remaining categories for brevity.

Now the column of Table 1 headed by dilation 8 can be computed. Again the initial values a_n for $n \le 6$ satisfy $a_n = \lfloor \log(n!) \rfloor$ since for these n we have $diam(S(n)) \le 8$. For $n > 6$, the typical step in calculating a_{n+1} from previously calculated values a_i, $i \le n$, can take one of the following routes;

(a) and (b) described above in the case k=2 and r=1.

(c) If the hypothesis of lemma 4 is applicable for k=3,r=1, then apply the lemma with n-2 playing the role of d; that is, if $n-2 \ge \lceil \frac{4}{3} 2^p \rceil - 1$, where $p = \lfloor \log(n) \rfloor$ then apply lemma 4. The result is $a_{n+1} = a_{n-2} + 3p + 1$.

(d) If the hypothesis of lemma 4 is applicable for k=3,r=2, then apply the lemma with n-2 playing the role of d; that is, if $n-2 \ge \lceil \frac{4}{3} 2^p \rceil - 1$, where $p = \lfloor \log(n) \rfloor$ then apply lemma 4. The result is $a_{n+1} = a_{n-2} + 3p + 2$.

Again one chooses the way which yields the maximum value for a_{n+1}.

References

[AHK] S. Akers, D. Harel, and B. Krishnamurthy, "The star graph: an attractive alternative to the n-cube", Proc. of 1987 International Conference on Parallel Processing, pp. 393-400.

[AK1] S. Akers and B. Krishnamurthy, "Group graphs as interconnection networks", Proc. of 14'th International Conference on Fault Tolerant Computing, (1984) pp.422-427.

[AK2] S. Akers and B. Krishnamurthy, "A group-theoretic model for symmetric interconnection networks", IEEE Transactions on Computers, Vol. 38, No. 4 (1989), pp.555-566.

[BCHLR] S. Bhatt, F. Chung, J-W. Hong, F.T. Leighton, and A. Rosenberg, "Optimal simulations by butterfly networks", Proc. of 23rd Annual ACM Theory of Comp. Sci. Conf. (1988), pp.192-204.

[Lei] F. T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufmann Publishers, San Mateo, Ca. (1992), p.776, problem 3.355.

[JLD] J.S. Jwo, S. Lakshmivarahan, and S.K. Dhall, "Embedding of cycles and grids in star graphs", Journal of Circuits, Systems, and Computers Vol. 1, No. 1 (1991) 43-74.

[MPS] Z. Miller, D. Pritikin, and I.H. Sudborough, "Near embeddings of hypercubes into Cayley graphs on the symmetric group", IEEE Transactions on Computers, Vol. 43, No. 1, (1994) 13-22.

[MS] B. Monien and I.H. Sudborough, "Simulating binary trees on hypercubes", Proc. of 1988 AWOC Conference on VLSI Algorithms and Architectures (July 1988), in Lecture Notes in Computer Science, Vol. 319, (Springer Verlag) pp. 170-180.

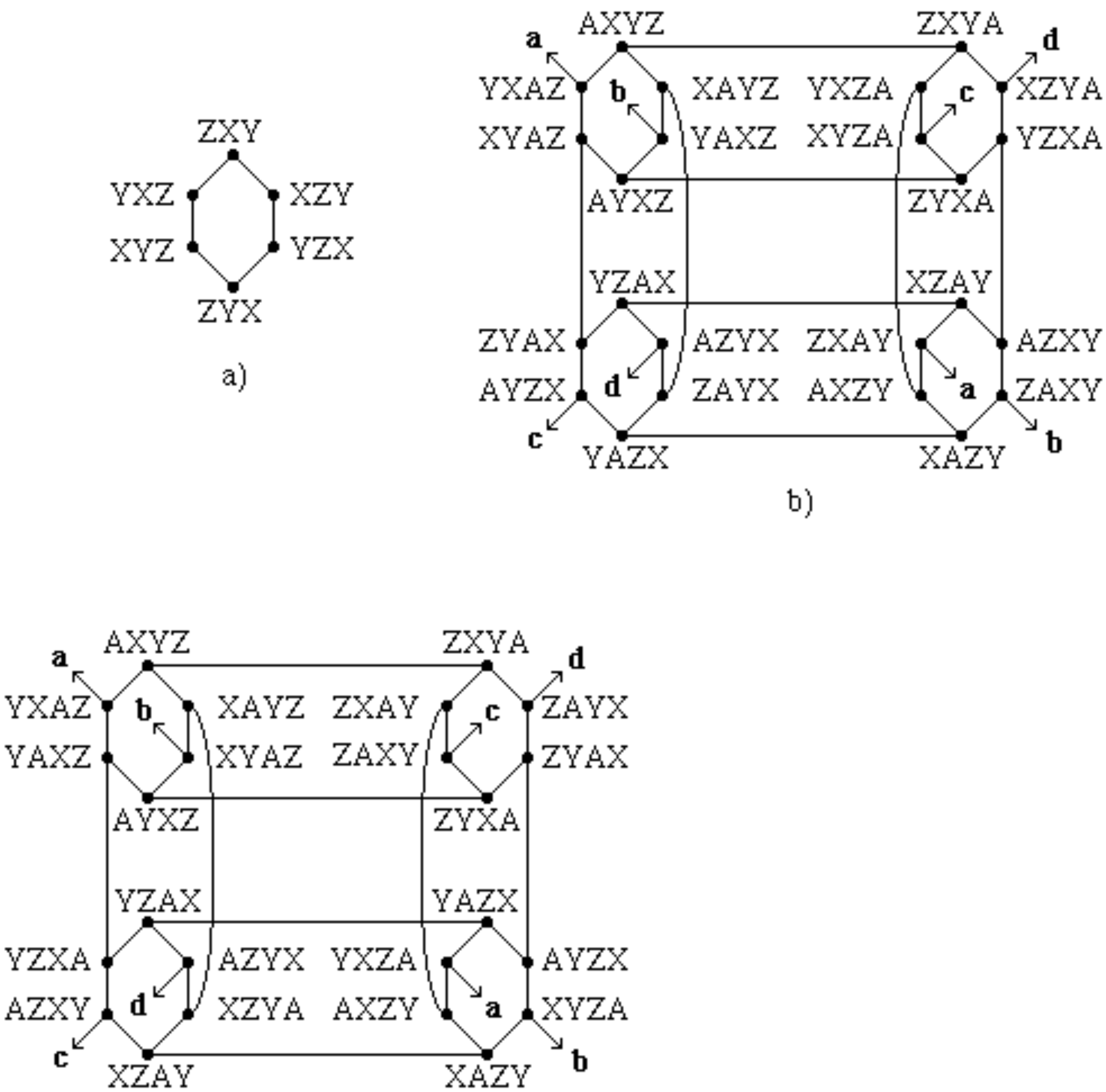
[NSK] M. Nigam, S. Sahni, and B. Krishnamurthy, "Embedding hamiltonians and hypercubes in star interconnection graphs", Proceedings of International Conference on Parallel Processing (1990).

Dimension n of star	$\lfloor \log_2(n!) \rfloor$	dilation 1*	dilation 2*	dilation 3	dilation 4	dilation 6	dilation 8
1	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1
3	2	1	2	2	2	2	2
4	4	3	4	4	4	4	4
5	6	3	4	6	6	6	6
6	9	3	5	7	8	8	9
7	12	3	5	9	10	11	12
8	15	4	7	12	13	14	15
9	18	4	7	15	16	17	18
10	21	4	8	16	19	20	21
11	25	4	9	18	22	23	24
12	28	4	10	20	25	26	27
13	32	5	11	23	28	30	31
14	36	5	12	25	31	33	34
15	40	5	13	28	34	37	38
16	44	6	13	32	38	41	42
17	48	6	13	36	42	45	46
18	52	6	14	37	46	49	50
19	56	7	15	39	50	53	54
20	61	7	16	41	54	57	58
21	65	7	17	44	58	61	62
22	69	8	18	46	62	65	66
23	74	8	19	49	66	69	70
24	79	8	20	52	70	73	75
25	83	9	21	56	74	78	79
26	88	9	22	58	78	82	84
27	93	9	23	61	82	87	88
28	97	10	24	64	86	91	93
29	102	10	24	68	90	96	98
30	107	10	24	71	94	100	102
31	112	11	25	75	98	105	107
32	117	11	26	80	103	109	112

Table 1

Largest k for which it is known that Q(k) has a dilation d embedding into S(n).

* Indicates that one-to-many embeddings have been allowed.



c)

<u>Figure 1</u>

a) S(3) , positionally represented on XYZ.

b) S(4) , positionally represented on AXYZ.

c) S(4) , A-represented on AXYZ.

$0 \rightarrow (s_1, s_2, z(1,0))$ $1 \rightarrow (z(1,1), s_2, s_1)$ $2 \rightarrow (s_1, z(1,2), s_2)$

Figure 2A: The vector V and corresponding vector [V;r,k] in the case r = 1 and k = 3

$(0,0) \rightarrow (s_1, s_2, z(1,0), z(2,0))$ $(1,0) \rightarrow (z(1,1), s_2, s_1, z(2,0))$ $(2,0) \rightarrow (s_1, z(1,2), s_2, z(2,0))$

$(0,1) \rightarrow (z(2,1), s_2, z(1,0), s_1)$ $(1,1) \rightarrow (z(1,1), s_2, z(2,1), s_1)$ $(2,1) \rightarrow (z(2,1), z(1,2), s_2, s_1)$

$(0,2) \rightarrow (s_1, z(2,2), z(1,0), s_2)$ $(1,2) \rightarrow (z(1,1), z(2,2), s_1, s_2)$ $(2,2) \rightarrow (s_1, z(1,2), z(2,2), s_2)$

Figure 2B: The vector V and corresponding vector [V;r,k] in the case r = 2 and k = 4

Figure 2 : The vector [V;r,k].

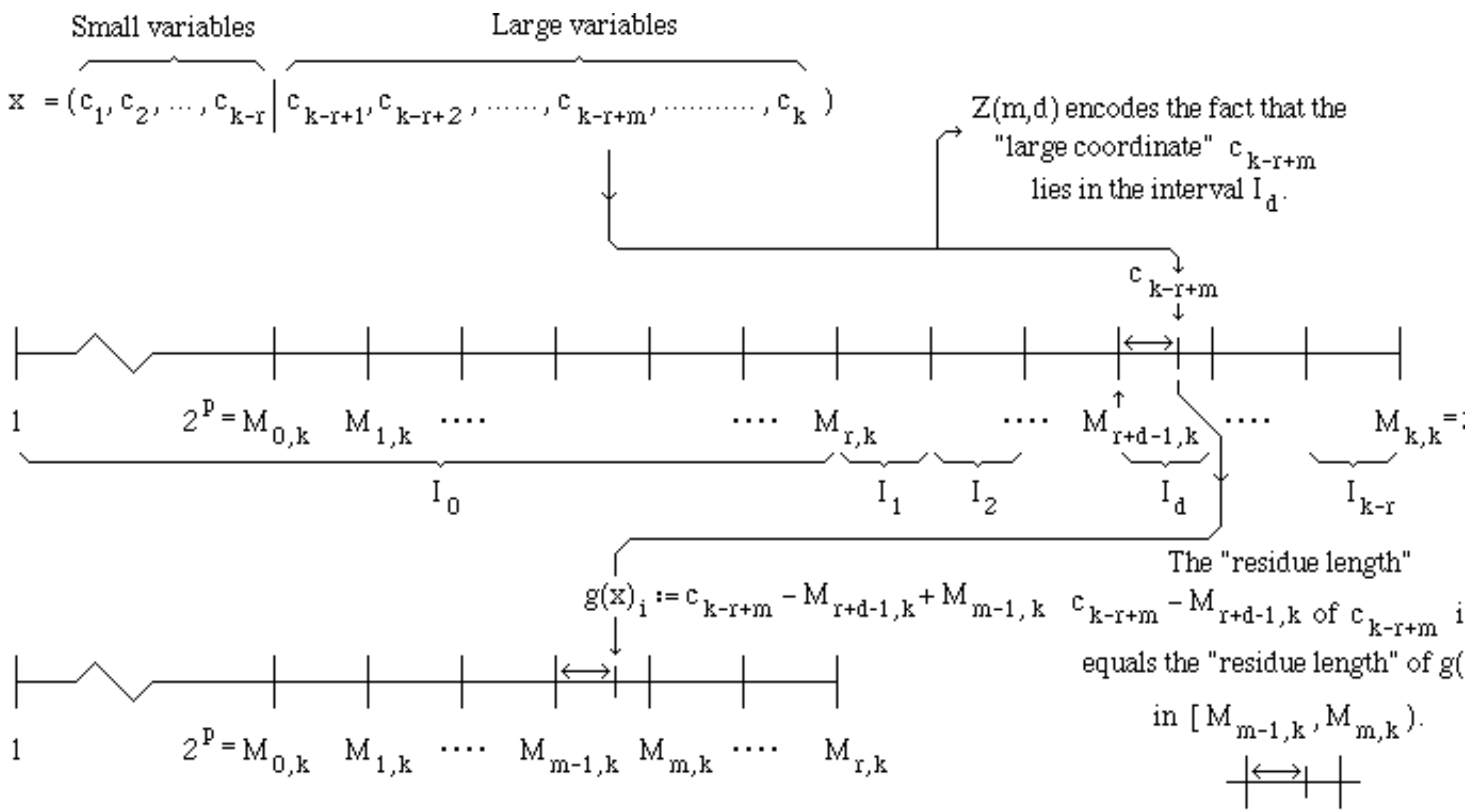


Figure 3: Interpreting $[V; r, k]_i = Z(m,d)$ as an instruction for computing $g(x)_i$ by shifting.

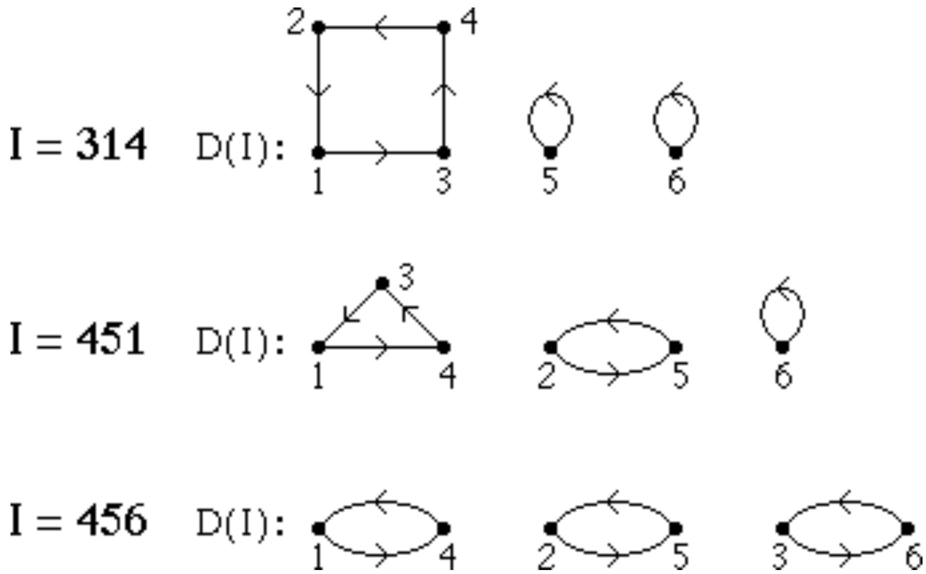


Figure 4 : Digraphs D(I) when d = k = 3.

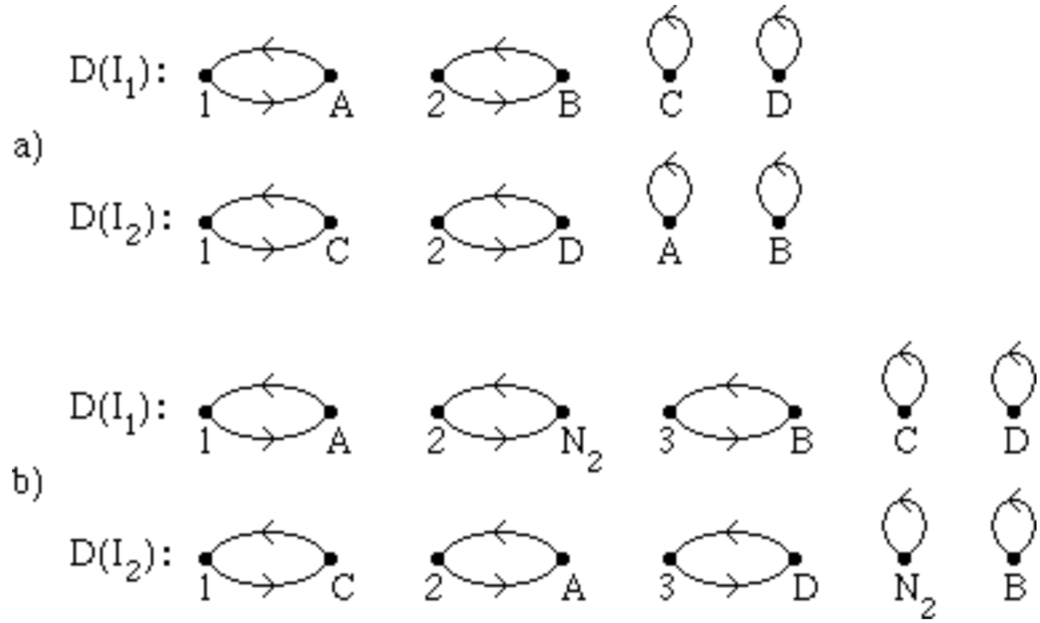


Figure 5 : Calculating $\alpha_{I_2}\alpha_{I_1}^{-1}$.