SIMULATION OF LARGE NETWORKS ON SMALLER NETWORKS

H.L. Bodlaender and J. van Leeuwen

RUU-CS-84-4
April 1984

Rijksuniversiteit Utrecht

Vakgroep informatica

Budapestlaan 6 3584 CD Utrecht
Corr. adres: Postbus 80.012 3508 TA Utrecht
Telefoon 030-53 1454
The Netherlands

SIMULATION OF LARGE NETWORKS ON SMALLER NETWORKS

H.L. Bodlaender and J. van Leeuwen

Technical Report RUU-CS-84-4
April 1984

Department of Computer Science
University of Utrecht
P.O. Box 80.012, 3508 TA Utrecht
the Netherlands

SIMULATION OF LARGE NETWORKS ON SMALLER NETWORKS

H.L.Bodlaender* and J. van Leeuwen

Department of Computer Science, University of Utrecht
P.O. Box 80.012, 3508 TA Utrecht, the Netherlands.

Abstract. Parallel algorithms are normally designed for execution on networks of N processors, with N depending on the size of the problem to be solved. In practice there will be a varying problem size but a fixed network size. In [3] the notion of network emulation was proposed, to obtain a structure preserving simulation of large networks on smaller networks. We present a detailed analysis of the possible emulations for some important classes of networks, namely: the shuffle-exchange network, the cube network, the ring network, and the 2-dimensional grid. We also study the possibility of cross-emulations, and characterize the networks that can be emulated at all on a given network using some class of emulation functions.

1. Introduction. Parallel algorithms are normally designed for execution on a suitable network of N processors (viewed as SIMD- or MIMD-machine [12]), with N depending on the size of the problem to be solved. In practice N will be large and varying, whereas processor networks will be small and fixed. The resulting disparity between algorithm design and implementation must be resolved by simulating a network of some size N on a fixed and smaller size network of a similar or different kind, in a structure preserving and efficient manner. Notions of simulation are well-understood in e.g. automata theory (see [6]), and suitable analogs can be brought to bear on networks of processors. In this paper we study a notion of simulation, termed: emulation, that was recently proposed by Fishburn and Finkel [3].

---

* The work of this author was supported by the Foundation for Computer Science (SION) of the Netherlands Organization for the Advancement of Pure Research (ZWO).

Definition. Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be networks of processors (graphs). We say that G can be emulated on H if there exists a function $f: V_G \rightarrow V_H$ such that for every edge $(g,g') \in E_G$: $f(g) = f(g')$ or $(f(g),f(g')) \in E_H$. The function f is called an emulation function, or in short, an emulation of G on H.

Clearly, emulation between networks is transitive. We shall only be interested in emulations f that are "onto".

Let f be an emulation of G on H. Any processor $h \in V_H$ must actively emulate the processors $\in f^{-1}(h)$ in G. When $g \in f^{-1}(h)$ communicates information to a neighboring processor g', then h must communicate the corresponding information "internally" when it emulates g' itself or to a neighboring processor $h' = f(g')$ in H otherwise. If all processors act synchronously in G, then the emulation will be slowed by a factor proportional to $\max_{h \in V_H} |f^{-1}(h)|$. For a set A, we use $|A|$ to denote the cardinality of A.

Definition. Let G, H, and f be as above. The emulation f is said to be (computationally) uniform if for all h, $h' \in V_H$: $|f^{-1}(h)| = |f^{-1}(h')|$.

Every uniform emulation f has associated with it a fixed constant c, called: the computation factor, such that for all $h \in V_H$: $|f^{-1}(h)| = c$. It means that every processor of H emulates the same number of processors of G. Again, uniform emulation between networks is transitive. When G can be uniformly emulated on H and H can be uniformly emulated on G, then G and H are necessarily isomorphic. (Thus uniform emulation establishes a partial ordering of networks.) For graphs A, B let A[B] denote the composition of A and B (cf. [5]).

Lemma 1.1. G can be uniformly emulated on H if and only if there exists a graph G' such that G is a spanning subgraph of H[G'].

Proof.

$\Rightarrow$Let f be a uniform emulation of G on H with computation factor c. The sets $\{f^{-1}(h)\}$, h $\in$ H, partition G into blocks of size c. Let G' be any graph on c nodes such that the induced subgraph of every block (in G) is contained in G'. Next observe that for any two nodes g $\in f^{-1}(h)$ and g' $\in f^{-1}(h')$ of G : (g,g') $\in E_G \Rightarrow$ h = h' (and the edge is in G') or (h,h') $\in E_H$. It follows that G is a spanning subgraph of H[G'].

$\Leftarrow$ From the definition of composition (cf. [5]), by projection on H. □

G' can always be chosen to be equal to K_c, the complete graph on c nodes. When G is uniformly emulated on H, then H can be viewed as a "factor" of G (and, in particular: $|V_H| \mid |V_G|$). When $|V_G|=|V_H|$, then G can be uniformly emulated on H if and only if G is isomorphic to a subgraph of H. With this observation it is not hard to show that the general UNIFORM NETWORK EMULATION problem is NP-complete. (Cf. [4]. Reduce from HAMILTONIAN CIRCUIT. Let H be an instance of HAMILTONIAN CIRCUIT and let G be a ring with $|V_H|$ nodes.) Another useful property is the following.

Definition. For directed graphs G = (V,E) let G^R be the (directed) graph obtained from G by reversing the direction of the edges, i.e., $G^R = (V, E^R)$ with $E^R = \{(g',g) \mid (g,g') \in E\}$.

Lemma 1.2. f is a (uniform) emulation of G on H if and only if f is a (uniform) emulation of G^R on H^R.

Proof.

$\Rightarrow$Let f be an emulation of G on H. It means that for every edge $(g,g') \in E_G$: $f(g) = f(g')$ or $(f(g), f(g')) \in E_H$. Thus, by simple translation, we have for every edge $(g',g) \in E_{G^R}$: $f(g') = f(g)$ or $(f(g'),f(g)) \in E_{H^R}$. Hence f is an emulation of G^R on H^R.

$\Leftarrow$ By a similar argument, observing that $(G^R)^R = G$ for all graphs G. Finally we note that uniformity is preserved in the constructions. □

The relevant question is whether (large) networks of some class C can be uniformly emulated by networks of a smaller size within the same class C. Fishburn and Finkel [3] answered this question affirmatively for the following classes of processor networks : the shuffle-exchange network, the grid-connected network, the n-dimensional cube, the plus-minus network, the binary lens, and the cube-connected cycles. (For definitions of these networks, see [3].) In this paper we shall take a more fundamental approach and develop a detailed analysis of all possible emulations in selected classes of networks : the shuffle-exchange network, the n-dimensional cube, the ring, and the 2-dimensional grid. The results will be presented in Sections 2 to 4. In Section 5 we consider the question of emulating networks of some class C on (smaller) networks of some class C'. In Section 6 we show that there is a natural way to describe the networks that can be emulated on a given network H, using a set of permissible emulations. The results lead to interesting characterisations of all networks considered in terms of their emulated behaviour. Some proofs are deferred to appendix A and B.

2. Emulations of the shuffle-exchange network. Let S_n denote the shuffle-exchange network with 2^n nodes. Our main result will be that there are exactly 6 different uniform emulations of S_n on S_{n-1}. We also show that there are at least $2.2^{2^k}-2^{2^{k-1}}$ uniform emulations of S_n on S_{n-k} ($k \geq 1$). In Section 2.1. we give some preliminary definitions and results, in Section 2.2 we give the analysis leading to our main result. The proof of the main theorem is deferred to appendix A. In Section 2.3 we discuss the uniform emulation of S_n on S_{n-k} in general and argue that the results hold for the uniform emulation of the inverse shuffle-exchange network as well.

2.1. Preliminaries. The shuffle-exchange network was proposed initially by Stone [10], and has been successfully used as the interconnection network underlying a variety of parallel processing algorithms. The nodes are given n-bit addresses in the range $o..2^n-1$, and there is an edge from node b to node c if and only if b can be "shuffled" (move leading bit to tail position) and "exchanged" (flip the tail bit) into

c. Computations proceed by iterating through the network some n or more times, in a synchronised manner. We use the following notations.

$\frac{o}{1}$: a bit that can be o or 1
$\bar{\alpha}$: the complement of bit ($\bar{o} = 1$, $\bar{1} = o$)
$\alpha \equiv \beta$: the 'equivalence' test on bits ($0{\equiv}0 = 1$; $0{\equiv}1 = 0$; $1{\equiv}0 = 0$; $1{\equiv}1 = 1$)
b : the n-bit address $b_1..b_n$
$b|_i$: $b_1..b_i$ (truncation after the i^{th} bit)
$_i|b$: $b_i..b_n$ (truncation "before" the i^{th} bit)
$(b)_i$: b_i (the i^{th} bit).

For functions f defined on n-bit numbers b we use :

$f_i(b)$: $(f(b))_i$ (projection on the i^{th} bit)

We use b, c, .. to denote full addresses and x, y, .. to denote segments of bits. Individual bits are denoted α, β, .. .

Definition. The shuffle-exchange network is the graph $S_n = (V_n, E_n)$ with $V_n = \{ (b_1..b_n) \mid \forall_{1\le i\le n}\ b_i = \frac{o}{1}\}$ and $E_n = \{ (b,c) \mid b, c \in V_n \text{ and } \forall_{2\le i\le n}\ b_i = c_{i-1} \}$. The inverse shuffle-exchange network is the graph $\tilde{S}_n = (V_n, \tilde{E}_n)$ with $\tilde{E}_n = \{ (b, c) \mid b, c \in V_n \text{ and } \forall_{2\le i\le n}\ b_{i-1} = c_i \}$.

It follows that in S_n a node $b_1..b_n$ is connected to $b_2..b_n o$ and $b_2..b_n 1$, in $\tilde{S}_n$ to $ob_1..b_{n-1}$ and $1b_1..b_{n-1}$. The fact that S_n can be (uniformly) emulated on S_{n-1} and, hence, on every S_{n-k} ($k\ge 1$) derives from the following observation, using lemma 1.1. (Compare [3], theorem 2.)

Lemma 2.1. S_n is a spanning subgraph of $S_{n-1}[\bar{K}_2]$, for $n\ge 2$.
Proof.

Consider the mapping $h : S_n \to S_{n-1}[\bar{K}_2]$ defined by $h(b_1..b_n) = \langle b_1..b_{n-1}, b_n\rangle$, which clearly is 1-1 and onto on the set of nodes. Let $(b,c) \in E_n$ with $c = b_2..b_n \frac{o}{1}$ (necessarily). Then $(b_1..b_{n-1}, b_2..b_n) \in E_n$, hence $h(b)$ and $h(c)$ are adjacent in $S_{n-1}[\bar{K}_2]$. Thus S_n is isomorphic to a spanning subgraph of $S_{n-1}[\bar{K}_2]$. □

Corollary 2.2. S_n is a spanning subgraph of $S_{n-k}[\bar{K}_{2^k}]$, for every $1 \le k < n$.

By using a mapping h defined by $h(b_1..b_n) = \langle b_{n-1}..b_1, b_n \rangle$, a similar argument shows that S_n is a spanning subgraph of $\tilde{S}_{n-1}[K_2]$ and (hence) that S_n can be uniformly emulated on $\tilde{S}_{n-1}$ and any smaller inverse shuffle-exchange network. Clearly $\tilde{S}_n \cong S_n$.

Lemma 2.3. f is an emulation of S_n on S_{n-k} if and only if for all $x \in (\frac{0}{1})^{n-1}$, $y \in (\frac{0}{1})^{n-k-1}$ and $\alpha, \beta \in (\frac{0}{1})$: if $f(\alpha x) = \beta y$ then $(f(x0) = \beta y \vee f(x0) = y\frac{0}{1})$ and $(f(x1) = \beta y \vee f(x1) = y\frac{0}{1})$.

(The proof follows straight from the definitions involved.) For a mapping f, define its "companion" $\bar{f}$ by $\bar{f}_i(b) = \overline{f_i(b)}$ for all $1 \le i \le n$.

Lemma 2.4. If f is an emulation of S_n on S_{n-k}, then so is $\bar{f}$.

Proof.

Immediate from lemma 2.3. □

2.2. Uniform emulations of S_n on S_{n-1}. The uniform emulations of S_n on S_{n-1} will be shown to be "step-simulating" in a very precise sense. Our aim will be to characterize all step-simulations of S_n on S_{n-1}, and to derive from it all uniform emulations.

Definition. A mapping $g : S_n \to S_{n-1}$ is called step-simulating (or : a "step-simulation" of S_n on S_{n-1}) if and only if for all $x \in (\frac{0}{1})^{n-1}$, $y \in (\frac{0}{1})^{n-2}$ and $\alpha, \beta \in (\frac{0}{1})$: if $g(\alpha x) = \beta y$, then $g(x0) = y\frac{0}{1}$ and $g(x1) = y\frac{0}{1}$.

Lemma 2.5. Every step-simulation g of S_n on S_{n-1} is an emulation.

Proof.

Immediate. (Compare lemma 2.3.) □

We shall call a step-simulation "uniform" when it is uniform as an emulation. When g is a step-simulation, then so is $\bar{g}$.

<u>Lemma 2.6</u>. A mapping $g : S_n \to S_{n-1}$ is step-simulating if and only if for all $x \in (\frac{o}{1})^{n-1}$, $y \in (\frac{o}{1})^{n-2}$ and $\alpha,\beta \in (\frac{o}{1})$: if $g(x\alpha) = y\beta$ then $g(ox) = \frac{o}{1}y$ and $g(1x) = \frac{o}{1}y$.

Proof.

By verifying equivalence with the definition of step-simulation. (Use the string character of x and y.) □

Lemma 2.6. can be interpreted as stating that the (uniform) step-simulations of S_n on S_{n-1} act at the same time as (uniform) step-simulations of $\tilde{S}_n$ on $\tilde{S}_{n-1}$. Note the following useful properties of step-simulations g:

$$g(b_1..b_{n-1}o)|_{n-2} = {}_2|g(ob_1..b_{n-1})$$
$$g(b_1..b_{n-1}1)|_{n-2} = {}_2|g(1b_1..b_{n-1})$$

We shall now aim for a characterisation of the possible step-simulations and uniform step-simulations of S_n on S_{n-1}.

<u>Definition</u>. For n≥3, define the operators $\Pi^n : [V_n \to V_{n-1}] \to [V_{n-1} \to V_{n-2}]$ and $T^n : [V_{n-1} \to V_{n-2}] \to [V_n \to V_{n-1}]$ as follows:

$$\Pi^n(g)\ (b_1..b_{n-1}) = g(b_1..b_{n-1}o)|_{n-2}$$
$$T^n(h)\ (b_1..b_n) = h(b_1..b_{n-1}).h_{n-2}(b_2..b_n)$$

<u>Theorem 2.7</u>. For n≥3,

(i) if g is a step-simulation of S_n on S_{n-1}, then $\Pi^n(g)$ is a step-simulation of S_{n-1} on S_{n-2}.

(ii) if h is a step-simulation of S_{n-1} on S_{n-2}, then $T^n(h)$ is a step-simulation of S_n on S_{n-1}.

(iii) restricted to step-simulations, Π^n and T^n are inverses.

(iv) restricted to step-simulations, Π^n preserves uniformity.

Proof.

(i) Verify the condition of lemma 2.6. : $\Pi^n(g)(x\alpha) = y\beta \Rightarrow g(x\alpha o) = y\beta\frac{o}{1}$ (definition of Π^n) $\Rightarrow g(ox\alpha) = \frac{o}{1}y\beta$ and $g(1x\alpha) = \frac{o}{1}y\beta \Rightarrow g(oxo) = \frac{o}{1}y\frac{o}{1}$ and $g(1xo) = \frac{o}{1}y\frac{o}{1}$ (by shifting right and then left) $\Rightarrow \Pi^n(g)\ (ox) = \frac{o}{1}y$ and $\Pi^n(g)\ (1x) = \frac{o}{1}y$.

(ii) Similarly $T^n(h)\ (x\gamma a) = y\delta\beta \Rightarrow h(x\gamma) = y\delta \Rightarrow T^n(h)(ox\gamma) =$

$h(ox).h_{n-2}\ (x\gamma) = \frac{o}{1}y\delta$ and $T^n(h)\ (1x\gamma) = h(1x).h_{n-2}\ (x\gamma) = \frac{o}{1}y\delta$.

(iii) Let g be a step-simulation of S_n on S_{n-1}. Then $T^n \circ \Pi^n(g)\ (\gamma x\delta) = \Pi^n(g)(\gamma x).\Pi^n(g)_{n-2}\ (x\delta) = g(\gamma xo)|_{n-2}.g_{n-2}(x\delta o) = g(\gamma x\delta)|_{n-2}.g_{n-1}(\gamma x\delta) = g(\gamma x\delta)$ for all γ,x,δ. Hence $T^n \circ \Pi^n$ = id. Conversely, let h be a step-simulation of S_{n-1} on S_{n-2}. Then $\Pi^n \circ T^n(h)\ (\gamma x) = T^n(h)\ (\gamma xo)|_{n-2} = (h(\gamma x).h_{n-2}(xo))|_{n-2} = h(\gamma x)$ for all γ,x. Hence also $\Pi^n \circ T^n$ = id. It follows that Π^n and T^n are inverses to one another when considered as operators on step-simulations.

(iv) Let g be a uniform step-simulation of S_n on S_{n-1}. Suppose $\Pi^n(g)$ is not uniform. Then there must be a $y \in V_{n-2}$ such that $|\Pi^n(g)^{-1}(y)| > 2$. Let $x^{(1)}$, $x^{(2)}$, $x^{(3)}$ be distinct elements of $\Pi^n(g)^{-1}(y)$. It follows that $g(x^{(1)}o), g(x^{(2)}o), g(x^{(3)}o) \in \{yo,y1\}$. Because g is step-simulating we have, in fact : $g(x^{(1)}o)$, $g(x^{(1)}1)$, $g(x^{(2)}o)$, $g(x^{(2)}1)$, $g(x^{(3)}o)$, $g(x^{(3)}1) \in \{yo,y1\}$ and hence $|g^{-1}(yo)|\geq 3$ or $|g^{-1}(y1)|\geq 3$. This contradicts the uniformity of g. □

Theorem 2.7. (i)-(iii) shows that there is a 1-1 onto correspondence between the step-simulations of S_n on S_{n-1} and the step-simulations of S_{n-1} on S_{n-2}, for $n\geq 3$. Theorem 2.7.(iv) does not quite show that this correspondence holds for the subclass of uniform step-simulations, but in the next theorem we will show that it is the case.

<u>Theorem 2.8</u>. For $n\geq 2$,

(i) there are exactly 16 possible step-simulations of S_n on S_{n-1}.

(ii) There are exactly 6 possible uniform step-simulations of S_n on S_{n-1} (see table A).

<u>Proof</u>.

(i) By theorem 2.7.(i)-(iii) the number of step-simulations of S_n on S_{n-1} is equal to the number of step-simulations of S_{n-1} on S_{n-2}, for $n\geq 3$ (because Π^n is bijective). By induction this number is equal to the number of step-simulations of S_2 on S_1. Clearly every mapping $\in [V_2 \to V_1]$ is step-simulating. There are exactly $2^4 = 16$ mappings in this set.

(ii) There are exactly $\binom{4}{2} = 6$ mappings $\in [V_2 \to V_1]$ that are uniform and step-simulating. By theorem 2.7.(i)-(iv) the number of uniform step-simulations of S_n on S_{n-1} ($n\geq 3$) is not larger than the number of uniform step-simulations of S_{n-1} on S_{n-2} and thus, by induction, not larger than 6. On the other hand at least 6 uniform step-simulations of S_n on S_{n-1} can be explicitly given, see table A. (The verification of the mappings is immediate from the definition.) □

The remaining problem is to determine whether any other uniform emulation of S_n on S_{n-1} exists. Our main result is the following.

<u>Theorem 2.9</u>. (Characterisation Theorem) Every uniform emulation of S_n on S_{n-1} is step-simulating, and thus equal to one of the mappings listed in table A.

The proof of theorem 2.9. is long and tedious, and is given in appendix A.

$$f_1 : f_1(b_1..b_n) = b_1..b_{n-1}$$
$$\bar{f}_1 : \bar{f}_1(b_1..b_n) = \bar{b}_1..\bar{b}_{n-1}$$

$$f_2 : f_2(b_1..b_n) = b_2..b_n$$
$$\bar{f}_2 : \bar{f}_2(b_1..b_n) = \bar{b}_2..\bar{b}_n$$

$$f_3 : f_3(b_1..b_n) = c_1..c_{n-1} \text{ with } c_i = (b_i \equiv b_{i+1}),\ 1 \leq i \leq n-1$$
$$\bar{f}_3 : \bar{f}_3(b_1..b_n) = \bar{c}_1..\bar{c}_{n-1} \text{ with } c_i = (b_i \equiv b_{i+1}),\ 1 \leq i \leq n-1$$

Table A. Listing of the 6 possible uniform step-simulations of the shuffle-exchange network with 2^n nodes on the shuffle-exchange network with 2^{n-1} nodes.

2.3. Uniform emulations of S_n on S_{n-k}. We will extend the notion of 'step-simulation' of S_n on S_{n-k}, in order to attempt a characterisation of the uniform emulations in general. We show that the step-simulations of S_n on S_{n-k} (which are not all uniform) can again be characterized in terms of the step-simulations of S_{k+1} on S_1 (cf. theorem 2.8.). It remains an open question whether all uniform emulations of S_n on S_{n-k} are step-simulating, and thus whether a suitable analogue of theorem 2.9. holds for $k \geq 1$. We show that there are at least $2.2^{2^k}-2^{2^{k-1}}$ uniform step-simulations of S_n on S_{n-k}. We also discuss the uniform emulations of $\tilde{S}_n$ on $\tilde{S}_{n-k}$.

Definition. A mapping $g : S_n \rightarrow S_{n-k}$ is called step-simulating (or : a "step-simulation" of S_n on S_{n-k}) if and only if for all $x \in (\frac{o}{1})^{n-1}$, $y \in (\frac{o}{1})^{n-k-1}$ and $\alpha, \beta \in \frac{o}{1}$: if $g(\alpha x) = \beta y$ then $g(x\frac{o}{1}) = y\frac{o}{1}$.

Every step-simulation clearly is an emulation (verify lemma 2.3.) and also the following analogue of lemma 2.6. holds.

Lemma 2.10. A mapping $g : S_n \rightarrow S_{n-k}$ is step-simulating if and only if for all $x \in (\frac{o}{1})^{n-1}$, $y \in (\frac{o}{1})^{n-k-1}$ and $\alpha, \beta \in (\frac{o}{1})$: if $g(x\alpha) = y\beta$ then $g(ox) = \frac{o}{1}y$ and $g(1x) = \frac{o}{1}y$.

We now aim for a characterization of all step-simulations of S_n on S_{n-k}.

Definition. For $n \geq k+2$, define the operators $\Pi^{n,k} : [V_n \rightarrow V_{n-k}] \rightarrow [V_{n-1} \rightarrow V_{n-k-1}]$ and $T^{n,k} : [V_{n-1} \rightarrow V_{n-k-1}] \rightarrow [V_n \rightarrow V_{n-k}]$ as follows:

$$\Pi^{n,k}(g)\ (b_1..b_{n-1}) = g(b_1..b_{n-1}o)|_{n-k-1}$$
$$T^{n,k}(h)\ (b_1..b_n) = h(b_1..b_{n-1}).h_{n-k-1}(b_2..b_n)$$

Theorem 2.11. For $n \geq k+2$,

(i) if g is a step-simulation of S_n on S_{n-k}, then $\Pi^{n,k}(g)$ is a step-simulation of S_{n-1} on S_{n-k-1}.

(ii) if h is a step-simulation of S_{n-1} on S_{n-k-1}, then $T^{n,k}(h)$ is a step-simulation of S_n on S_{n-k}.

(iii) restricted to step-simulations, $\Pi^{n,k}$ and $T^{n,k}$ are inverses.

(iv) restricted to step-simulations, $\Pi^{n,k}$ preserves uniformity.

Proof.

(The proof is virtually the same as for theorem 2.7. and therefore left to the reader.) □

We conclude the following results (cf. theorem 2.8.) :

Theorem 2.12. For $n \geq k+2$,

(i) there is a bijection from the set of step-simulations of S_n on S_{n-k} to the set of step-simulations of S_{n-1} on S_{n-k-1} and (hence) to the set of step-simulations of S_{k+1} on S_1.

(ii) there is an injection from the set of uniform step-simulations of S_n on S_{n-k} to the set of uniform step-simulations of S_{n-1} on S_{n-k-1} and (hence) to the set of uniform step-simulations of S_{k+1} on S_1.

Theorem 2.12. is important, as it characterizes the step-simulations of S_n on S_{n-k}. Clearly every mapping $\in [V_{k+1} \to V_1]$ is step-simulating, and thus there are precisely $2^{2^{k+1}}$ step-simulations of S_n on S_{n-k}.

Corollary 2.13. For $n \geq 1$, S_n admits precisely 2 graph-isomorphisms.

Proof.

Every isomorphism of S_n must be step-simulating. By theorem 2.12.(i) the step-simulations of S_n on S_n are in 1-1 correspondence to the step-simulations of S_1 on S_1. There are four mappings of this kind and thus precisely four step-simulations of S_n on S_n : $g_1(b_1..b_n) = b_1..b_n$, $g_2(b_1..b_n) = \bar{b}_1..\bar{b}_n$, $g_3(b_1..b_n) = o..o$, $g_4(b_1..b_n) = 1..1$. Clearly, only g_1 and g_2 are isomorphisms. □

The 1-1 correspondence referred to in theorem 2.12.(i) can be made explicit as follows. Given a step-simulation g of S_n on S_{n-k}, the uniquely corresponding step-simulation $\tilde{g}$ of S_{k+1} on S_1 is defined by the formula $\tilde{g}(b_1..b_{k+1}) = g(b_1..b_{k+1}o..o)|_1$. Conversely, given a step-simulation h of S_{k+1} on S_1, the uniquely corresponding step-simulation $\tilde{h}$ of S_n on S_{n-k} is defined by $\tilde{h}(b_1..b_n) = h(b_1..b_{k+1}).h(b_2..b_{k+2})..h(b_{n-k}..b_n)$. While the correspondence $g \to \tilde{g}$ preserves uniformity (cf. theorem 2.11. (iv)), it does not induce a bijection from the uniform step-simulations of S_n on S_{n-k} to the uniform step-simulations of S_{k+1} to S_1 for k>1. The existence of such a bijection for k = 1 (cf. theorem 2.8.(ii)) was the key to the complete characterisation of the uniform step-simulations of S_n on S_{n-1} and of the uniform emulations of S_n on S_{n-1} (cf. theorem 2.9.). A similar characterisation of the uniform step-simulations and of the uniform emulations of S_n on S_{n-k} for k > 1 remains a challenging open problem. We can characterize a large class of uniform step-simulations.

<u>Theorem 2.14</u>. Let $n \geq k+1$, and let g be a step-simulation of S_n on S_{n-k}.

(i) if $\tilde{g}(b_1..b_{k+1}) = \overline{\tilde{g}(\bar{b}_1 b_2..b_{k+1})}$ for all $b_1..b_{k+1} \in (\frac{o}{1})^{k+1}$, then g is uniform.

(ii) if $\tilde{g}(b_1..b_{k+1}) = \overline{\tilde{g}(b_1..b_k \bar{b}_{k+1})}$ for all $b_1..b_{k+1} \in (\frac{o}{1})^{k+1}$, then g is uniform.

<u>Proof</u>.

We only prove (i) as the proof of (ii) is similar. Induct on n. For n = k+1, observe from the assumption that of every pair $b_1..b_{k+1}$, $\bar{b}_1 b_2..b_{k+1}$ $\tilde{g}$ will map one to $o \in V_1$ and one to $1 \in V_1$. Thus $g = \tilde{g}$ is uniform. Assume it holds up to $n-1 \geq k+1$. Let g be a step-simulation of S_n on S_{n-k} for which the constraint on $\tilde{g}$ is satisfied. Let g' be the uniquely corresponding step-simulation of S_{n-1} on S_{n-k-1} (cf. theorem 2.12.(i)) defined by the formula $g'(b_1..b_{n-1}) = g(b_1..b_{n-1}o)|_{n-1}$. Observe that for all $b_o..b_{n-1} \in (\frac{o}{1})^n$: $g(b_o b_1..b_{n-1}) = \tilde{g}(b_o b_1..b_k).\tilde{g}(b_1..b_{k+1})..\tilde{g}(b_{n-k-2}..b_{n-1})$ and likewise for $g'(b_1..b_{n-1})$, hence $g(b_o b_1..b_{n-1}) = \tilde{g}(b_o b_1..b_k).g'(b_1..b_{n-1})$. Since $\tilde{g}' = \tilde{g}$, it follows by induction that g' is uniform. Thus for every $c_1..c_{n-k-1} \in (\frac{o}{1})^{n-k-1}$: $|(g')^{-1}(c_1..c_{n-k-1})| = 2^k$. Let $b_1..b_{n-1} \in (g')^{-1}(c_1..c_{n-k-1})$. By assumption it follows that of the pair $ob_1..b_k$, $1b_1..b_k$ $\tilde{g}$ will map one to $o \in V_1$ and one to $1 \in V_1$, and thus g will map one of the strings $ob_1..b_{n-1}$, $1b_1..b_{n-1}$ to $oc_1..c_{n-k-1}$ and the other to $1c_1..c_{n-k-1}$. It follows that for all $c_o c_1 .. c_{n-k-1} \in (\frac{o}{1})^{n-k}$: $|g^{-1}(c_o c_1 .. c_{n-k-1})| = |(g')^{-1}(c_1 .. c_{n-k-1})| = 2^k$, which implies that g is uniform. This completes the inductive argument. □

<u>Theorem 2.15</u>. For $n \geq k+1$, there are at least $2.2^{2^k} - 2^{2^{k-1}}$ uniform step-simulations of S_n on S_{n-k}.

<u>Proof</u>.

For k=1 the result follows from theorem 2.8.(ii). For k>1 we use the characterisation from theorem 2.14. By induction on k one easily derives that there exist 2^{2^k} functions $\tilde{g} : V_{k+1} \to V_1$ that satisfy the constraint $\tilde{g}(b_1 .. b_{k+1}) = \tilde{g}(\bar{b}_1 b_2 .. b_{k+1})$, 2^{2^k} functions $\tilde{g} : V_{k+1} \to V_1$ that satisfy the constraint $\tilde{g}(b_1 .. b_{k+1}) = \tilde{g}(b_1 .. b_k \bar{b}_{k+1})$, and $2^{2^{k-1}}$ functions $\tilde{g}$ that satisfy both constraints simultaneously. Using the unique correspondence of $\tilde{g}$ and g, the given bound follows. □

By lemma 1.2. every uniform emulation $f : S_n \to S_{n-k}$ $(n,k \geq 1)$ also is a uniform emulation of $\tilde{S}_n$ on $\tilde{S}_{n-k}$, and conversely. (Note that $\tilde{S}_n = (S_n)^R$.) It follows that all results concerning the uniform emulations of S_n on S_{n-k} hold ipso facto for the uniform emulations of $\tilde{S}_n$ on $\tilde{S}_{n-k}$.

3. <u>Emulations of the cube network</u>. Let C_n denote the cube network with 2^n nodes. Our main result will be a complete characterisation of the uniform emulations of C_n on C_{n-1}, in terms of the uniform emulations of C_3 on C_2. This Section will be devoted to various auxiliary results and the proof of the main theorem. The argument depends on a crucial lemma (theorem 3.5.) whose lengthy proof is deferred to appendix B.

The cube network with 2^n nodes (also called an n-cube) has perhaps been the first proposal ever for processor interconnection. The nodes in the network again are given n-bit addresses in the range $o..2^n-1$, and there is an edge from node b to node c if and only if c is obtained by flipping precisely one bit in b. Information can be routed from a source b to a destination c in at most n steps, by flipping the bits b_i to the corresponding bits c_i in some order. Since nodes thus have degree n, the cube network is considered practical only for small values of n. We use b,c,.. to denote full addresses and x,y.. to denote segments of bits. The i^{th} bit of an address b is denoted by b_i ($1 \le i \le n$). For $|x| = |y|$, let d(x,y) be the Hamming distance between the bitstrings x and y, i.e., the number of bit-positions in which x and y differ. (See, for example, Deo [2] sect. 12-5)

Definition. The cube network (or n-cube) is the graph $C_n = (V_n, E_n)$ with $V_n = \{ (b_1..b_n) | \forall_{1 \le i \le n}\ b_i = {o \atop 1} \}$ and $E_n = \{(b,c) \mid b,c \in V_n$ and $d(b,c) = 1 \}$.

The fact that C_n can be (uniformly) emulated on every C_{n-k} for $k \ge 1$ easily derives from the following observation, using lemma 1.1.

Propostion 3.1. For $k \ge 1$, C_n is isomorphic to $C_{n-k}[C_k]$.

Proof.

One verifies that the mapping $h : C_n \to C_{n-k}[C_k]$ defined by the formula $h(b_1..b_n) = \langle b_1..b_{n-k}, b_{n-k+1}..b_n \rangle$ is an isomorphism. □

Lemma 3.2. f is an emulation of C_n on C_{n-k} if and only if for all b, c $\in V_n$: if d(b,c) = 1 then $d(f(b),f(c)) \le 1$.

(The proof follows directly from the definition of emulation. Note that the condition can be equivalently written as: $d(f(b),f(c)) \le d(b,c)$.) We shall be interested in characterizing the uniform emulations of C_n on C_{n-1}.

The distinguishing feature of C_n is that it admits many more isomorphisms than e.g. S_n (cf. corollary 2.13). This immediately has consequences for the characterization of uniform emulations, because of the following fact.

Lemma 3.3. Let I, I' be isomorphisms of C_n, C_{n-1} respectively. For every f, if f is a uniform emulation of C_n on C_{n-1} then so is $I' \circ f \circ I$ (and conversely).

(The easy proof of lemma 3.3. is left as an exercise.) The isomorphisms of C_n can be characterized. For permutations Π let I_Π be the isomorphism defined by $I_\pi\ (b_1..b_n) = b_{\pi(1)}..b_{\pi(n)}$, and for index sets $J \subseteq \{1,...,n\}$ let I_J be the isomorphism defined by

$$(I_J)_i\ (b_1..b_n) = \begin{cases} \bar{b}_i & \text{if } i \in J \\ b_i & \text{otherwise} \end{cases}$$

for $1 \leq i \leq n$. Thus, I_J flips the bits in the positions with index in J.

<u>Theorem 3.4</u>. I is an isomorphism of C_n if and only if there are J, Π such that $I = I_J \circ I_\Pi$.

<u>Proof</u>.

The "if"-part is obvious. To prove the "only-if"-part, proceed as follows. Consider I(o..o) and choose J such that $i \in J$ if and only if $I_i(o..o) = 1$. Furthermore choose Π such that if the i^{th} bit of o..o is flipped, then so is the $\Pi(i)^{th}$ bit of I(o..o). Observe that such a permutation Π must exist. Define the weight w(b) of a bitstring b as the number of nonzero bits in b. We prove by induction on w(b) that for all $b \in V_n : I_J^{-1} \circ I = I_\Pi$. For $w(b) \leq 1$ it holds : observe that $I_J^{-1} \circ I(o..o) = (o..o)$ and that if the i^{th} bit of o..o is flipped, then so is the $\Pi(i)^{th}$ bit of $I_J^{-1} \circ I(o..o)$. Suppose it holds for all b with $w(b) \leq m$ for some $m \geq 1$. Consider $b \in V_n$ with $w(b) = m+1$ and choose $c,c' \in V_n$ of weight m, with $c \neq c'$ and $d(b,c) = d(b,c') = 1$. Suppose b is obtained from c,c' by flipping the i^{th}, j^{th} bit from o to 1 respectively, for some $i \neq j$. By induction $I_J^{-1} \circ I(c) = I_\Pi(c)$ and $I_J^{-1} \circ I(c') = I_\Pi(c')$ and clearly $I_\Pi(c)$ and $I_\Pi(c')$ differ in the $\Pi(i)^{th}$ and $\Pi(j)^{th}$ position. If $I_J^{-1} \circ I(b)$ is obtained from $I_\Pi(c)$ by flipping a bit in a position $\notin \{\Pi(i),\Pi(j)\}$ then it will have a distance ≥ 2 from $I_\Pi(c')$. Contradiction. Suppose $I_J^{-1} \circ I(b)$ is obtained from $I_\Pi(c)$ by flipping the $\Pi(j)^{th}$ bit. Clearly $c_j = 1$. Let c" be the string obtained from c by setting the j^{th} bit to o. $I_\Pi(c")$ is obtained from $I_\Pi(c)$ by flipping the $\Pi(j)^{th}$ bit, so $I_J^{-1} \circ I(b) = I_\Pi(c")$. It follows that $w(c") = m-1$ and (hence) $b \neq c"$ and (by induction) $I_J^{-1} \circ I(b) = I_\Pi(c") = I_J^{-1} \circ I(c")$, contradicting that $I_J^{-1} \circ I$ is 1-1. Thus $I_J^{-1} \circ I(b)$ is obtained from $I_{\Pi(c)}$ by flipping the $\Pi(i)^{th}$ bit and thus $I_J^{-1} \circ I(b) = I_\Pi(b)$. This completes the induction. We conclude that $I_J^{-1} \circ I = I_\Pi$, or $I = I_J \circ I_\Pi$. □

Viewing C_n as the n-dimensional unit cube brings the analysis of emulations into the realm of combinatorial topology.

Definition. For $o \leq m \leq n$, an m-face of C_n is any subgraph (subcube) of 2^m nodes of C_n that have identical bits in n-m corresponding positions.

Crucial for the characterization of uniform emulations is the following result, the proof of which is deferred to appendix B.

Theorem 3.5. (Topological Reduction Theorem). Let $n \geq 4$, and let f be a uniform emulation of C_n on C_{n-1}. Then there exists an (n-1)-face A of C_n such that f(A) is an (n-2)-face of C_{n-1}.

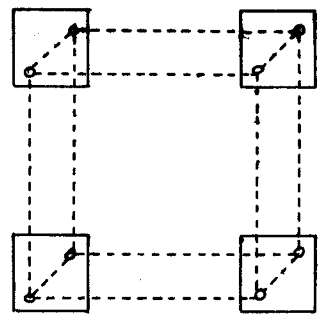
Definition. For mappings $g : V_3 \to V_2$, let $\hat{g} : V_n \to V_{n-1}$ be the mapping defined by $\hat{g}(b_1..b_n) = g(b_1b_2b_3)b_4..b_n$ $(n \geq 4)$.

Theorem 3.6. (Characterization Theorem). For $n \geq 3$, f is a uniform emulation of C_n on C_{n-1} if and only if there are isomorphisms I and I' of C_n and C_{n-1} respectively and a uniform emulation g of C_3 on C_2 such that $f = I' \circ \hat{g} \circ I$.

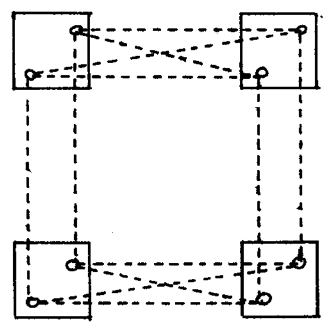
Proof.

The "if"-part is obvious. For the "only if"-part we induct on n. The characterization is obvious for n=3. Assume it holds up to $n-1 \geq 3$, and consider a uniform emulation f of C_n on C_{n-1}. By theorem 3.5. there is an (n-1)-face A of C_n such that f(A) is an (n-2)-face of C_{n-1}. Up to isomorphisms of C_n and C_{n-1} we may assume that A is determined by elements b that have identical b_n and that f(A) is determined by elements c that have identical c_{n-1}. Because of uniformity no elements of the complementary face A^c (i.e., the elements with bit b_n flipped) can be mapped into f(A). It follows that A^c is mapped to $f(A)^c$ (i.e., the elements of f(A) with bit c_{n-1} flipped) and, because f emulates, that $f(b_1..b_{n-1}b_n)$ and $f(b_1..b_{n-1}\overline{b}_n)$ are equal in the first n-2 bits for all

Type 1 : 1-face → o-face



Type 2 : 2-face → 1-face



Type 3 : 3-face → 2-face

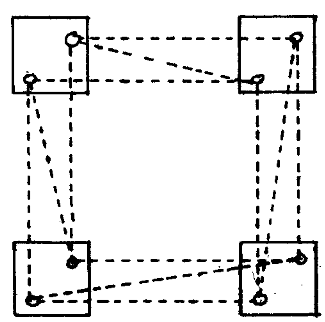


Table B. Classification of the uniform emulations of C_3 on C_2 according to the smallest m for which an m-face is mapped to an (m-1)-face.

$b_1..b_n \in V_n$. It follows that, restricted to $A \cong V_{n-1}$, f reduces to a mapping f' depending on $b_1..b_{n-1}$ only and $f(b_1..b_n) = f'(b_1..b_{n-1})b_n$ for all $b_1..b_n \in V_n$ or $f(b_1..b_n) = f'(b_1..b_{n-1})\bar{b}_n$ for all $b_1..b_n \in V_n$. Up to another isomorphism of C_{n-1} we can assume the former. As a mapping from $A \cong V_{n-1}$ to $f(A) \cong V_{n-2}$, f' is seen to act as a uniform emulation of C_{n-1} on C_{n-2}. The induction hypothesis now applies to obtain the desired form for f. □

The characterisation of theorem 3.6. is complete once the uniform emulations of C_3 on C_2 are explicity given. Clearly there are many that are similar, by lemma 3.3. Characterized by the smallest m such that an m-face is mapped to an (m-1)-face, only three essentially different uniform emulations of C_3 on C_2 can arise. The different types are given in table B.

It is open whether a similar, complete characterisation can be given of the uniform emulations of C_n on C_{n-k} for $k > 1$.

4. Emulations of the ring and the two-dimensional grid network.

Throughout this Section let n be even, unless stated otherwise. Let R_n be the ring network with n nodes, and let GR_n be the n x n grid network (with n^2 nodes) with wrap-around connections. In Section 4.1. we give a complete characterization of the uniform emulations of R_n on $R_{n/2}$. In Section 4.2. we show that the number of uniform emulations of GR_n on $GR_{n/2}$ is at least exponential in n.

4.1. Uniform emulations of R_n on $R_{n/2}$. The ring network is important in practice (cf. Tanenbaum [11]), but hardly occurs as an interconnection network for multiprocessor algorithms. Indeed the analysis in this Section only prepares for the later study of GR_n, because $GR_n \cong R_n \times R_n$. The nodes of R_n are named o,1,..., n-1 in consecutive order.

Definition. The ring network (or n-ring) is the graph $R_n = (V_n, E_n)$ with $V_n = \{ i | i \in N \text{ and } o \le i \le n-1 \}$ and $E_n = \{ (i, i\underline{+}1) | i \in V_n \}$, where "+" is the addition modulo n.

By "wrapping" it around $R_{n/2}$ twice, it follows that R_n can be uniformly emulated on $R_{n/2}$. Our aim will be to characterize all possible uniform emulations of R_n on $R_{n/2}$.

It will be helpful to view R_n (hence $R_{n/2}$) as a subdivision of the unit circle S^1 in the plane. Clearly, every emulation of R_n on $R_{n/2}$ induces a continuous mapping from S^1 to itself. It is well-known that such mappings can be characterized by their topological degree or "winding number". The winding number indicates the number of times the image of S^1 is wrapped around the unit circle when S^1 is traversed once. By analogy we can speak of the winding number of an emulation.

Proposition 4.1. The winding number of an emulation of R_n on $R_{n/2}$ is -2,-1,o,+1, or +2.

Proof.

Let f be an emulation of R_n on $R_{n/2}$. If the image of R_n wraps around $R_{n/2}$ 3 times or more, then the n nodes of R_n are mapped to a trajectory of at least $\frac{3}{2}$ n consecutive points on $R_{n/2}$. This is impossible. □

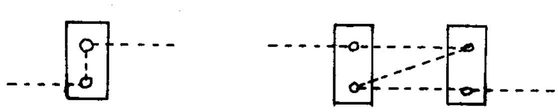
It is relatively straightforward to classify the possible uniform emulations of R_n on $R_{n/2}$ by their (positive) winding number.

Case I. Winding number = o.

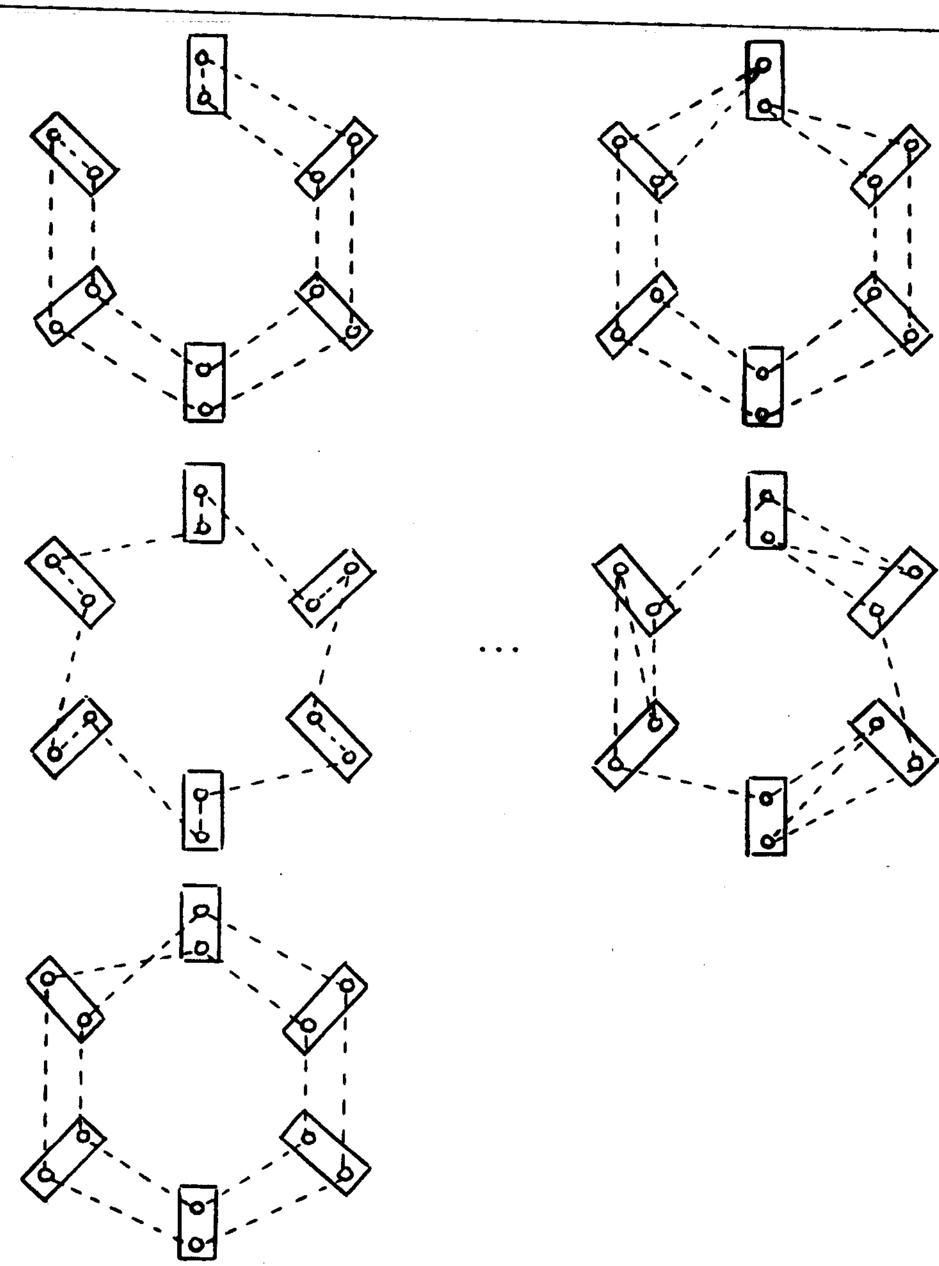
If $f(R_n)$ cannot make a full turn around $R_{n/2}$ then the condition of uniformity forces f to be one of the two forms suggested in table C (a). We shall refer to the emulations as being of "type 1".

Case II. Winding number = 1.

One verifies that f(R) must be composed of a number of "hooks" and "zigzags":.



"hook" "zigzag"



(a) Type 1.

(b) Type 2.

(c) Type 3.

Table C. Classification of the uniform emulations of R_n on $R_{n/2}$ by winding number.

Conversely, any combination of hooks and zigzags defines a uniform emulation of R_n on $R_{n/2}$ with winding number 1. We shall refer to the emulations of this kind as being of "type 2". Table C(b) shows two extreme examples of emulations of type 2.

Case III. Winding number = 2.

$f(R_n)$ is necessarily of the kind suggested in table C(c). We shall refer to the emulations of this kind as being of "type 3".

We conclude the following.

Theorem 4.2. (Characterization Theorem). For n even, f is a uniform emulation of R_n on $R_{n/2}$ if and only if it is of type 1, type 2, or type 3.

Corollary 4.3. The number of different uniform emulations of R_n on $R_{n/2}$ is exponential in n.

Proof.

(Two emulations f and g are said to be "different" if g cannot be obtained by a rotational shift of f.) Clearly the number of uniform emulations of R_n on $R_{n/2}$ of type 2 is exponential in n. □

4.2. Uniform emulations of GR_n on $GR_{n/2}$. The two-dimensional grid (or mesh) has been used as a processor interconnection network, and extensive studies have been made of algorithms to be executed on a grid (e.g. Nassimi & Sahni [7]). We use a version of the grid with "wrap-around" connections along the boundaries, which gives the nodes a uniform neighbourhood structure. The nodes of GR_n are named by their plane coordinates (i,j) with $0 \leq i,j \leq n-1$.

Definition. The two-dimensional grid network is the graph $GR_n = (V_n, E_n)$ with $V_n = \{ (i,j) \mid i,j \in N \text{ and } 0 \leq i,j \leq n-1 \}$ and $E_n = \{ ((i,j),(i',j')) \mid (i,j),(i',j') \in V_n \text{ and } (i=i' \wedge j=j' \underline{+} 1) \text{ or } (i=i' \underline{+} 1 \wedge j=j') \}$, where "+" is the addition modulo n.

By "folding" GR_n, it follows that GR_n can be uniformly emulated on

$GR_{n/2}$. Every uniform emulation of GR_n on $GR_{n/2}$ has computation factor 4. The classification of the uniform emulations is presently open, but some useful observations can be made.

As $GR_n \cong R_n \times R_n$, it can effectively be viewed as a torus. Let $n \geq 10$ and let f be a uniform emulation of GR_n on GR_n. Every cycle with 4 nodes, i.e., a "square" in GR_n must be mapped on $GR_{n/2}$ by f in one of the ways shown in table D.

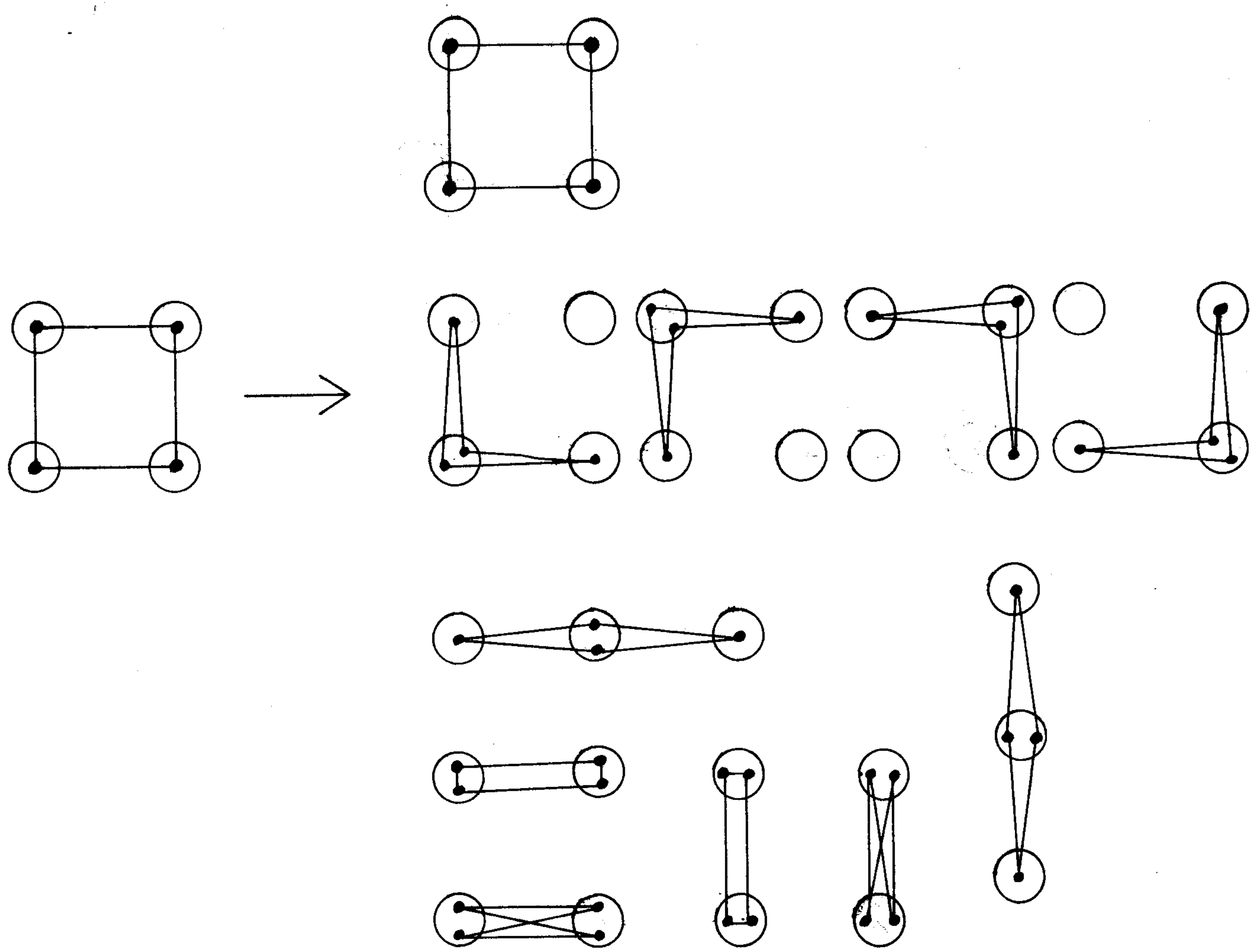
From this one easily derives that f induces a continuous mapping of the torus to itself. Again the notion of topological degree (winding number) can be introduced, as expounded in homology theory. Let GR_n be "spanned" by the oriented cycles $\vec{a} \cong \{ (o,j) \mid o \leq j \leq n-1 \}$ and $\vec{b} \cong \{ (i,o) \mid o \leq i \leq n-1 \}$. A closed curve C can be classified by the pair (k,l), where k is the number of times C is wrapped in the $\vec{a}$-direction and l is the number of times C is wrapped in the $\vec{b}$-direction. One can now classify (uniform) emulations by the topological degrees of $f(\vec{a})$ and $f(\vec{b})$, considered as closed curves on the torus $GR_{n/2}$. The underlying reason is the following fact from homology theory : if closed curves C,C' on the torus have equal topological degree and f is continuous, then f(C) and f(C') have equal topological degree on the torus also.

For $n \leq 8$, the same analysis does not necessarily hold. In fig. 1 we give an example of a uniform emulation of GR_6 on GR_3 for which $f(\vec{a}) = f(\{(o,j) \mid o \leq j \leq 5\})$ has topological degree (1,1) and $f(\{(1,j) \mid o \leq j \leq 5\})$ has topological degree (-1,1). (Hence f cannot induce a continuous mapping of the torus to itself.)

Proposition 4.4. Let f be an emulation of GR_n on $GR_{n/2}$. The topological degree (k,l) of $f(\vec{a})$ and $f(\vec{b})$ satisfies $|k|+|l| \leq 2$.

(The proof follows by observing that the n points of $\vec{a}$ or $\vec{b}$ can be mapped to a trajectory of at most n points on $GR_{n/2}$.)

Theorem 4.5. The number of uniform emulations of GR_n on $GR_{n/2}$ is at least exponential in n.



Tabel D. Possible mappings of a cycle with 4 nodes by an emulation f of GR_n on $GR_{n/2}$, for $n \geq 10$.

Proof.

Let g,h be uniform emulations of R_n on $R_{n/2}$. Clearly the mapping f defined by f(i,j) = (g(i), h(j)) is a uniform emulation of GR_n on $GR_{n/2}$. By corollary 4.3. at least exponentionally many uniform emulations are obtained. □

For the uniform emulations f defined in the proof of theorem 4.5., the topological degrees of $f(\vec{a})$ and $f(\vec{b})$ are of the form (k,o) and (o,l) respectively. Figure 2 shows an example of a uniform emulation f of GR_8 on GR_4 for which $f(\vec{a})$ has topological degree (1,1) and $f(\vec{b})$ has topological degree (1,-1). (The example can easily be generalized to obtain uniform emulations f of GR_n on $GR_{n/2}$ for which $f(\vec{a})$ has topological degree (1,1) and $f(\vec{b})$ has topological degree (1,-1), for all even $n \geq 6$.)

23	32	03	13	35	44
34	43	30	31	53	55
01	10	02	14	15	25
24	42	20	41	51	52
00	11	04	12	05	45
22	33	21	40	50	54

Figure 1. A uniform emulation of GR_6 on GR_3 that does not induce a continuous mapping of the torus to itself.

07	32	24	33	14	25	06	15
43	76	60	77	50	61	42	51
22	31	12	23	04	13	05	30
66	75	56	67	40	57	41	74
10	21	02	11	03	36	20	37
54	65	46	55	47	72	64	73
00	17	01	34	26	35	16	27
44	53	45	70	62	71	52	63

Figure 2. A uniform emulation of GR_8 on GR_4 that is not the direct product of two uniform emulations of R_8 on R_4.

Similar results can be obtained for the d-dimensional analogue of GR_n. Let GR_n^d be the d-dimensional grid network (with wrap-around) with size n in each dimension, i.e., a "hypertorus" with n^d nodes.

Theorem 4.6. The number of uniform emulations of GR_n^d on $GR_{n/2}^d$ is at least exponential in dn.

The proof is a straightforward extension of the argument used for theorem 4.5.

5. Cross emulations. By cross-emulation we refer to the emulation of a

network G belonging to some class C_1 on a network H belonging to a different class C_2. The question of cross-emulating G on H can be important if algorithms must be transported from one type of interconnection network to another. We only consider situations with $|G| = |H|$, which means that the resulting uniform (cross-) emulations will necessarily have computation factor 1. Several results of Parker [8] concerning the "topological" equivalence of some common types of multi-stage networks are easily put into this framework. We only consider cross-emulations between S_n, C_n, R_n, and GR_n (as defined in Section 2-4).

In a number of cases the existence of cross-emulations is impossible by degree arguments. For example S_n, C_n, and GR_n cannot be emulated on a ring network of the same number of nodes. C_n and GR_n cannot be emulated on a shuffle-exchange network with a corresponding number of nodes, and neither can S_n be cross-emulated on the grid network of an equal number of nodes.

Proposition 5.1. For $n \geq 2$, S_n cannot be uniformly emulated on C_n.

Proof.

Suppose there was a uniform emulation f of S_n on C_n. Clearly $f(oo^{n-1})$, $f(1o^{n-1})$, and $f(o^{n-1}1)$ must be adjacent to one another in C_n, as the arguments are in S_n. Thus C_n contains a triangle. Contradiction. □

On the positive side, consider GR_{2^n} (with 2^{2n} nodes).

Theorem 5.2. For $n \geq 1$, GR_{2^n} can be uniformly emulated on C_{2n}.

Proof.

We prove the following, slightly stronger claim : for every $m,n \geq 1$ there is a uniform emulation of the $2^m \times 2^n$ grid network (with wrap-around connections) on C_{m+n}. Putting m=n proves the theorem. To prove the claim, induct on m and n. For m=n=1 the result is immediate. Assume the claim holds for some $m,n \geq 1$. Let f be a uniform emulation of the $2^m \times 2^n$ grid network on C_{m+n}. Consider the $2^{m+1} \times 2^n$ grid network, and map it to C_{m+n+1} using the mapping f' defined by

$$f'(i,j) = \begin{cases} o.f(i,j) & \text{if } o \le i < 2^m, \\ 1.f(2^{m+1}-i-1,j) & \text{otherwise } (2^m \le i < 2^{m+1}) \end{cases}$$

One easily verifies that f' is a uniform emulation. Likewise the 2^m x 2^{n+1} grid network can be uniformly emulated on C_{m+n+1}. This completes the inductive argument. □

By a degree argument it easily follows that, conversely, C_{2n} can be uniformly emulated on $GR_{2}n$ only for n=2.

Theorem 5.3. For values of n as indicated, R_n can be uniformly emulated on the following networks :

(i) for $n=k^2$, on GR_k.

(ii) for $n=2^k$, on S_k and on C_k.

Proof.

(The results are equivalent to claiming that GR_k, S_k and C_k are hamiltonian.)

(i) Left as an exercise.

(ii) By the existence of binary de Bruijn sequences ([1]) it follows that every S_k has a hamiltonian circuit. To obtain the result for C_k, write $k=k_1+k_2$. As the result is obvious for k=1, we may assume that $k_{1,2} \ge 1$. It is easy to show that R_n can be uniformly emulated on the 2^{k_1} x 2^{k_2} grid network (with wrap-around connections). In the proof of theorem 5.2. it was shown that the 2^{k_1} x 2^{k_2} grid network can be uniformly emulated on $C_{k_1+k_2}=C_k$. By transitivity the result follows. □

Observe that every uniform emulation of the ring of 2^k elements on C_k corresponds to a Gray code of length k (cf. [9]).

6. Defining networks by emulation. Every network $H = (V_H,E_H)$ can act as a "host" under emulation for many different, larger networks. If we restrict the class of admissible (uniform) mappings that should act as emulations, then the set of graphs G that can be emulated on H will likely

be restricted also. Our main result will be that S_n, C_n, R_n and GR_n are uniquely defined by their emulations on S_{n-1}, C_{n-1}, $R_{n/2}$ and $GR_{n/2}$ respectively. In Section 6.1. we derive some general results on defining networks by admissible sets of emulations. In Section 6.2. we prove the main results.

6.1. General characterizations. Let $H = (V_H, E_H)$ be a given host network, V a set of nodes with $|V| \geq |V_H|$, and F a collection of functions from V onto V_H.

Definition. A network G = (V,E) is said to be F-emulated on H if every f $\in$ F is an emulation of G on H.

Our aim will be to characterize all networks G that are F-emulated on H, given F and H. We assume H and V to be fixed, and F to be variable.

Definition. $E_F = \{ (v,v') \mid v,v' \in V,\ v \neq v' \text{ and } \forall_{f \in F} : f(v) = f(v') \text{ or } (f(v),f(v')) \in E_H \}$.

Theorem 6.1. (Characterization Theorem) G is F-emulated on H if and only if G is a spanning subgraph of (V,E_F).

Proof.

Let G = (V,E) be F-emulated on H, and let $(v,v') \in E$. By definition (of emulation) we have that for every $f \in F$: $f(v) = f(v')$ or $(f(v),f(v')) \in E_H$. Thus $(v,v') \in E_F$. It follows that $E \subseteq E_F$, and G is a spanning subgraph of (V,E_F). Conversely, it is clear that (V,E_F) is F-emulated on H by definition. Clearly, every spanning subgraph is F-emulated on H also. □

It follows that (V,E_F) is the maximal graph that can be F-emulated on H. Define $f : V \to V_H$ to be uniform if for all $h \in V_H$: $|f^{-1}(h)| = c$, for some constant $c = |V|/|V_H|$ (the computation factor).

Theorem 6.2. Let $f,f' : V \to V_H$ be uniform functions. Then $(V,E_{\{f\}})$ and $(V,E_{\{f'\}})$ are isomorphic graphs.

Proof.

Since $f, f' : V \to V_H$ are uniform (and thus map equal sized piles of elements onto every node of H) there exists a permutation $\Pi : V \to V$ such that for all $v \in V : f(v) = f'(\Pi(v))$. One easily verifies that Π is, in fact, an isomorphism of $(V, E_{\{f\}})$ and $(V, E_{\{f'\}})$. □

We derive a further result to characterize $(V, E_{\{f\}})$ when f is uniform. Let c be as defined above.

Lemma 6.3. Let $f : V \to V_H$ be uniform. Let d_{out} and d_{in} be the maximum out-degree and the maximum in-degree of the nodes in H, respectively. (If H is undirected, let $d_{out} = d_{in}$ be the maximum degree in H.)

(i) the maximum out-degree in $(V, E_{\{f\}})$ equals $(d_{out}+1).c-1$.

(ii) the maximum in-degree in $(V, E_{\{f\}})$ equals $(d_{in}+1).c-1$.

(iii) If G and H are undirected graphs, then $|E_{\{f\}}| = \frac{1}{2}c(c-1)|V_H| + c^2.|E_H|$.

(iv) If G and H are directed graphs, then $|E_{\{f\}}| = c(c-1)|V_H| + c^2.|E_H|$.

Proof.

(i) Consider any node $v \in V$. By uniformity there are precisely $c-1$ nodes $v' \neq v$ with $f(v) = f(v')$, which thus accounts for $c-1$ outgoing edges with this property. Next there are at most $d_{out}.c$ nodes v' with $(f(v), f(v')) \in E_H$. This accounts for a maximal out-degree of $c-1+d_{out}.c = (d_{out}+1)c-1$. By choosing v such that f(v) has maximum out-degree, it is clear that the bound is attained.

(ii) Similar to (i).

(iii) E_H contains $|V_H|.\frac{1}{2}c(c-1)$ edges (v,v') with $f(v) = f(v')$, because c nodes of V are mapped to every $h \in V_H$. Every edge $(h,h') \in E_H$ accounts for c^2 edges (v,v') with $f(v) = h$ and $f(v') = h'$. By definition E_F contains no other edges than the ones that were distinguished.

(iv) Similar to (iii). □

Lemma 6.3. will be useful later because, whenever $f \in F$ and G is F-emulated on H, then G is a spanning subgraph of $(V, E_{\{f\}})$.

6.2. Characterization of the shuffle-exchange, the cube, the ring and the grid networks by emulation. We use the definitions and results concerning S_n, C_n, R_n and GR_n as presented in Sections 2-4. First we consider S_n, the shuffle-exchange graph on 2^n nodes. From table A we recall the following uniform emulations of S_n on S_{n-1} :

$$f_1 : f_1(b_1..b_n) = b_1..b_{n-1},$$
$$f_2 : f_2(b_1..b_n) = b_2..b_n,$$
$$f_3 : f_3(b_1..b_n) = c_1..c_{n-1} \text{ with } c_i = (b_i \equiv b_{i+1}) \text{ for } 1 \le i \le n-1$$

We show that S_n is uniquely characterized by these three emulations on S_{n-1}. Let $V \tilde{=} V_n$ $(=(\frac{0}{1})^n)$, $n \ge 2$.

Theorem 6.4. $(V,E_{\{f_1,f_2,f_3\}}) = S_n$.
Proof.

Clearly S_n is a spanning subgraph of $(V,E_{\{f_1,f_2,f_3\}})$, by definition (or theorem 6.1.). We show that every edge of $(V,E_{\{f_1,f_2,f_3\}}$ must be an edge of $S_n = (V_n,E_n)$. Consider any edge $(b_1..b_n,c_1..c_n) \in E_{\{f_1,f_2,f_3\}}$. We distinguish the following cases:

(a) $f_i(b_1..b_n) = f_i(c_1..c_n)$ for $1 \le i \le 2$. It follows that $b_1..b_{n-1} = c_1..c_{n-1}$ and $b_2..b_n = c_2..c_n$ and (hence) $b_1..b_n = c_1..c_n$, contradicting that we had an edge between distinct points.

(b) $f_i(b_1..b_n) = f_i(c_1..c_n)$ for $i = 1,3$ (and $(f_2(b_1..b_n),f_2(c_1..c_n)) \in E_{n-1}$). It follows that $b_1..b_{n-1} = c_1..c_{n-1}$ and $(b_{n-1} \equiv b_n) = (c_{n-1} \equiv c_n)$, so $b_1..b_n = c_1..c_n$, a contradiction.

(c) $(f_1(b_1..b_n), f_1(c_1..c_n)) \in E_{n-1}$ and $f_2(b_1..b_n) = f_2(c_1..c_n)$. It follows that $b_2..b_{n-1}\alpha = c_1..c_{n-1}$ and $b_2..b_n = c_2..c_n$ for some α, hence $b_1..b_n = b_1\alpha^{n-1}$ and $c_1..c_n = \alpha^n$. Clearly $(b_1\alpha^{n-1},\alpha^n) \in E_n$.

(d) $f_1(b_1..b_n) = f_1(c_1..c_n)$ and $(f_i(b_1..b_n), f_i(c_1..c_n)) \in E_{n-1}$ for $i = 2,3$. It follows that $b_1..b_{n-1} = c_1..c_{n-1}$ and $b_3..b_n\alpha = c_2..c_n$ for some α, hence $b_1..b_n = b_1b_n^{n-1}$ and $c_1..c_n = b_1b_n^{n-2}\alpha$. Now $(f_3(b_1b_n^{n-1}),f_3(b_1b_n^{n-2}\text{alpha})) \in E_{n-1}$ implies $b_1 = b_n$, and clearly $(b_1b_n^{n-1},b_1b_n^{n-2}\alpha) \in E_n$.

(e) $(f_i(b_1..b_n),f_i(c_1..c_n)) \in E_{n-1}$ for $i = 1,2$. It follows that

$b_2..b_{n-1}\alpha = c_1..c_{n-1}$ and $b_3..b_n\beta = c_2..c_n$ for suitable α and β, hence $b_1..b_n = b_1c_1..c_{n-1}$. Clearly $(b_1c_1..c_{n-1}, c_1..c_n) \in E_n$. □

(It can be verified that no subset of $\{f_1,f_2,f_3\}$ is sufficient to characterize S_n.) Next consider C_n, the cube network on 2^n nodes. We select the following uniform emulations of C_n on C_{n-1}:

$$f_1 : f_1(b_1..b_n) = b_1..b_{n-1}$$

$$f_4 : f_4(b_1..b_n) = (b_1 \equiv b_2)b_3..b_n$$

Theorem 6.5. For $n \geq 3$, $(V,E_{\{f_1,f_4\}})=C_n$.

Proof.

Clearly C_n is a spanning subgraph of $(V,E_{\{f_1,f_4\}})$. Consider any edge $(b_1..b_n,c_1..c_n) \in E_{\{f_1,f_4\}}$. We distinguish the following cases :

(a) $f_1(b_1..b_n) = f_1(c_1..c_n)$. It follows that $b_1..b_{n-1} = c_1..c_{n-1}$, and either $b_1..b_n = c_1..c_n$ (a contradiction) or $b_1..b_n = c_1..c_{n-1}\bar{c}_n$. It follows that $(b_1..b_n,c_1..c_n) \in E_n$.

(b) $(f_1(b_1..b_n),f_1(c_1..c_n)) \in E_{n-1}$ and $f_4(b_1..b_n) = f_4(c_1..c_n)$. It follows that $d(b_1..b_{n-1},c_1..c_{n-1}) = 1$ and $b_1..b_n = b_1b_2c_3..c_n$ with $(b_1 \equiv b_2) = (c_1 \equiv c_2)$. It follows that $b_n=c_n$, and thus $(b_1..b_n,c_1..c_n) \in E_n$.

(c) $(f_1(b_1..b_n),f_1(c_1..c_n)) \in E_{n-1}$ and $(f_4(b_1..b_n),f_4(c_1..c_n)) \in E_{n-1}$. It follows that $d(b_1..b_{n-1},c_1..c_{n-1}) = 1$ and $d(\alpha b_3..b_n,\beta c_3..c_n) = 1$, with $\alpha = (b_1 \equiv b_2)$ and $\beta = (c_1 \equiv c_2)$. If $\alpha=\beta$ then necessarily $b_1b_2 = c_1c_2$ and $d(b_1..b_n,c_1..c_n) = 1$ thus $(b_1..b_n,c_1..c_n) \in E_n$. If $\alpha \neq \beta$ then $b_3..b_n = c_3..c_n$ and (hence) $d(b_1b_2,c_1c_2) = 1$. Clearly it follows that $(b_1..b_n,c_1..c_n) \in E_n$.

We conclude that every edge of $(V,E_{\{f_1,f_4\}})$ also is an edge of C_n. □

Theorem 6.5. is "minimal" in the sense that C_n cannot be characterized from C_{n-1} by means of just one uniform emulation.

Proposition 6.6. There does not exist a uniform emulation f of C_n on C_{n-1} such that $(V,E_{\{f\}}) = C_n$.

Proof.

Observe that (the undirected graph) C_{n-1} has 2^{n-1} nodes and $\frac{1}{2}.2^{n-1}(n-1)$ edges. Suppose a uniform mapping $f : V \to V_{n-1}$ exists with $(V,E_{\{f\}}) = C_n$. By lemma 6.3. (iii) $(V,E_{\{f\}})$ must have $n.2^n-2^{n-1}$ edges (c=2), which is more than C_n can have. □

Consider R_n, the ring on n nodes. Define the following uniform emulations of R_n on $R_{n/2}$ (n even) :

$$g_1 : g_1(i) = \lfloor \tfrac{i}{2} \rfloor$$
$$g_2 : g_2(i) = (i \bmod n/2)$$

Theorem 6.7. For n>8, $(V,E_{\{g_1,g_2\}}) = R_n$.

Proof.

Clearly R_n is a spanning subgraph of $(V,E_{\{g_1,g_2\}})$. Consider any edge $(i,j) \in E_{\{g_1,g_2\}}$. If $g_1(i) = g_1(j)$ then $|i-j|=1$ and $(i,j) \in E_n$. If $(g_1(i),g_1(j)) \in E_{n/2}$, then we may assume without loss of generality that $\lfloor i/2 \rfloor = \lfloor j/2 \rfloor+1 \pmod{n/2}$. It follows that $i \equiv j+2+\delta_i-\delta_j \pmod n$, with δ_i and δ_j Kronecker δ's. Now, in addition, $g_2(i) = g_2(j)$ or $(g_2(i), g_2(j)) \in E_{n/2}$. If $g_2(i) = g_2(j)$ then $|i-j| \equiv o \pmod{n/2}$, hence $2+\delta_i-\delta_j \equiv o$ (mod n/2) and, by the assumption on n, necessarily $2+\delta_i-\delta_j = o$ and $i=j$. Contradiction. If $(g_2(i),g_2(j)) \in E_{n/2}$ then $i \equiv j \pm 1 \pmod{n/2}$, hence $2+\delta_i-\delta_j \equiv \pm 1 \pmod{n/2}$. Since n>8, we have $2+\delta_i-\delta_j = 1$ and $i \equiv j+1 \pmod n$. Thus $(i,j) \in E_n$. We conclude that every edge of $(V,E_{\{g_1,g_2\}})$ is an edge of R_n. □

Finally consider GR_n, the grid network on n^2 nodes. Define the following uniform emulations of GR_n on $GR_{n/2}$ (n even) :

$$h_1 : h_1(i,j) = (\lfloor \tfrac{i}{2} \rfloor, \lfloor \tfrac{j}{2} \rfloor)$$
$$h_2 : h_2(i,j) = (i \bmod n/2,\ j \bmod n/2)$$

Theorem 6.8. For n>8, $V,E_{\{h_1,h_2\}}) = GR_n$.

Proof.

Similar to the proof of theorem 6.7. □

Acknowledgement. P.W.H. Lemmens pointed out several errors in the original version of Section 4.2. and provided the example shown in figure 1. We thank the referee for his/her very careful reading of the manuscript.

7. References.

[1] de Bruijn, N.G., A combinational problem, Indag. Math. VIII (1946) 461-467.

[2] Deo, N., Graph theory with applications to Engineering and Computer Science, Prentice Hall, Inc., Englewood Cliffs, NJ., 1974.

[3] Fishburn, J.P., and R.A. Finkel, Quotiënt networks, IEEE Trans. Comput. C-31 (1982) 288-295.

[4] Garey, M.R., and D.S. Johnson, Computers and intractability: a guide to the theory of NP-completeness, W.H. Freeman, San Francisco, Calif., 1979.

[5] Harary, F., Graph theory, Addison Wesley Publ. Comp., Reading Mass., 1969.

[6] Herman, G.T., When is a sequential machine the realization of another, Math. Syst. Th. 5(1971) 115-127.

[7] Nassimi, D., and S. Sahni, An optimal routing algorithm for mesh-connected parallel computers, J.ACM 27(1980) 6-29.

[8] Parker, D.S., Notes on shuffle/exchange-type switching networks, IEEE Trans. Comput. C-29 (1980) 213-222.

[9] Reingold, E.M., J. Nievergelt, and N. Deo, Combinatorial algorithms: theory and practice, Prentice Hall, Inc., Englewood Cliffs, NJ, 1977.

[10] Stone, H.S., Parallel processing with the perfect shuffle, IEEE Trans. Comput. C-20 (1971) 153-161.

[11] Tanenbaum, A.S., Computer networks, Prentice Hall, Inc., Englewood Cliffs, NJ., 1982.

[12] van Leeuwen, J., Parallel computers and algorithms, in : J. van Leeuwen and J.K. Lenstra (eds.), Parallel computers and computations, CWI Syll. 9, Centre for Mathematics and Computer Science, Amsterdam, 1985, pp. 1-32.

Appendix A. The proof of the Characterisation Theorem for the uniform emulations of S_n on S_{n-1} (theorem 2.9.).

We use the notations and terminology of Section 2. Our aim is to prove the following result.

Theorem 2.9. (Characterisation Theorem). Every uniform emulation of S_n on S_{n-1} is step-simulating, and thus equal to one of the mappings listed in table A.

The proof is based on the lemma below and a subsequent analysis of cases. Some further notational conventions will be helpful to deal with the elements of $(\frac{o}{1})^n$ and similar sets as strings:

[o] : zero or one occurence of bit o (i.e., "empty" or "o")
[1] : zero or one occurence of bit 1 (i.e., "empty" or "1")
(o1)* : zero or more repetitions of the string o1 (as required)
(1o)* : zero or more repetitions of the string 1o (as required)

The length (n) of a bitstring will always be clear from the context, and is usually not given by separate indices. For example, the notation (o1)*[o] for n odd will denote the string $(o1)^{\lfloor n/2 \rfloor}o$. For n even it will denote the string $(o1)^{n/2}$. Assume $n>2$.

For the proof of theorem 2.9., assume that there exists a uniform emulation f of S_n on S_{n-1} that is not step-simulating. It follows that there must be an $x \in (\frac{o}{1})^{n-1}$, $y \in (\frac{o}{1})^{n-3}$ and $\alpha,\beta,\gamma,\delta \in (\frac{o}{1})$ such that $f(\alpha x) = \beta y \delta$ and $f(x\gamma) = \beta y \delta$, with $\beta y \neq y \delta$. (Cf. lemma 2.3. and lemma 2.6.) We will fix the notation throughout the remainder of this section.

Claim 2.9.1. Under the assumption stated, one of the following situations must hold

(i) $x = o^{n-1}$ and $(\alpha=o \vee \gamma=o)$
(ii) $x = 1^{n-1}$ and $(\alpha=1 \vee \gamma=1)$
(iii) $\beta y \delta = (o1)^*[o]$
(iv) $\beta y \delta = (1o)^*[1]$

Proof.

In addition to $f(\alpha x) = f(x\gamma) = \beta y\delta$ we must have : $(f(\bar{\alpha}x) = \beta y\delta \vee f(\bar{\alpha}x) = {}^{o}_{1}\beta y)$ and $(f(x\bar{\gamma}) = \beta y\delta \vee f(x\bar{\gamma}) = y\delta{}^{o}_{1})$, from the emulation property. Because f is uniform, only two nodes can be mapped to $\beta\gamma\delta$. The following situation can be distinguished :

(a) $f(\bar{\alpha}x) = f(\alpha x) = \beta y\delta$. Because $f(x\gamma) = \beta y\delta$ also, we have $x\gamma = \bar{\alpha}x$ ($\Rightarrow x=o^{n-1}$ and $\alpha=1$ and $\gamma=o$, or $x=1^{n-1}$ and $\alpha=o$ and $\gamma=1$) or $x\gamma = \alpha x$ ($\Rightarrow x=o^{n-1}$ and $\alpha=o$ and $\gamma=o$, or $x=1^{n-1}$ and $\alpha=1$ and $\gamma=1$).

(b) $f(x\bar{\gamma}) = f(x\gamma) = \beta y\delta$. Now also $f(\alpha x) = \beta y\delta$, and the same cases as under (a) result.

(c) $f(\bar{\alpha}x) = {}^{o}_{1}\beta y$ and $f(x\bar{\gamma}) = y\delta{}^{o}_{1}$. Clearly $f(x\bar{\gamma}) = {}^{o}_{1}\beta y$ or $f(x\bar{\gamma}) = \beta y\bar{\delta}$, hence ${}^{o}_{1}\beta y = y\delta{}^{o}_{1}$ or $\beta y\bar{\delta} = y\delta{}^{o}_{1}$. Because $\beta y \neq y\delta$ only the former case can arise : ${}^{o}_{1}\beta y = y\delta{}^{o}_{1}$. It follows that $\beta y\delta = (o1)^*[o]$ or $\beta y\delta = (1o)^*[1]$. (The "solutions" $\beta y\delta = o^{n-1}$ and $\beta y\delta = 1^{n-1}$ are not valid, because it would yield $\beta y = y\delta$.) □

We now obtain the basic step for the further case analysis.

Lemma 2.9.2. Under the assumption stated, one of the following six cases must hold :

(I) $f((o1)^*[o]) = o^{n-1}$, $f((1o)^*[1]) = o^{n-1}$

(II) $f((o1)^*[o]) = 1^{n-1}$, $f((1o)^*[1]) = 1^{n-1}$

(III) $f((o1)^*[o]) = (o1)^*[o]$, $f((1o)^*[1]) = (o1)^*[o]$

(IV) $f((o1)^*[o]) = (o1)^*[o]$, $f((1o)^*[1]) = (1o)^*[1]$

(V) $f((o1)^*[o]) = (1o)^*[1]$, $f((1o)^*[1]) = (o1)^*[o]$

(VI) $f((o1)^*[o]) = (1o)^*[1]$, $f((1o)^*[1]) = (1o)^*[1]$.

Proof.

Let $f((o1)^*[o]) = u_1..u_{n-1}$ and $f((1o)^*[1]) = v_1..v_{n-1}$. Because $(o1)^*[o]$ and $(1o)^*[1]$ are adjacent in S_n and f is an emulation, the following situations can arise :

(a) $u_1..u_{n-1} = v_1..v_{n-1}$. Write $u_1..u_{n-1} = \beta y\delta$. (Note that we cannot assume that $\beta y \neq y\delta$.) By the analysis under claim 2.9.1. it follows that ${}^{o}_{1}\beta y = y\delta{}^{o}_{1}$ (hence $\beta y\delta = o^{n-1}$, 1^{n-1}, $(o1)^*[o]$, or $(1o)^*[1]$) or $\beta y\bar{\delta} = y\delta{}^{o}_{1}$ (hence $\beta y\delta = o^{n-1}$ or 1^{n-1}). This proves cases I, II, III, and VI.

(b) $u_1..u_{n-1} \neq v_1..v_{n-1}$, but $u_1..u_{n-1} = v_2..v_{n-1}{}^{o}_{1} = {}^{o}_{1}v_1..v_{n-2}$. It follows that $v_1..v_{n-1} = o^{n-1}$, 1^{n-1}, (o1)*[o], (1o)*[1] but only for the latter two cases can $u_1..u_{n-1}$ be chosen to satisfy the constraint (namely $u_1..u_{n-1}$ =(1o)*[1], (o1)*[o] respectively). This proves cases IV and V. □

We proceed by analysing the cases of lemma 2.9.2. and showing that in each case a contradiction must arise. (Recall the assumption that f is uniform and not step-simulating.)

Case I. $f((o1)*[o]) = f((1o)*[1]) = o^{n-1}$.
We show that this forces f to be equal to f_3, one of the six step-simulations listed in table A.

Claim 2.9.3. For $1 \le i \le n-1$ and $b \in \binom{o}{1}^n$, $f_i(b_1..b_n) = (b_i \equiv b_{i+1})$.

Proof.

Define $B_i^n = \{b_1..b_n | \forall_j : 1 \le j \le i-1 \Rightarrow b_j \ne b_{j+1}\} \subseteq \binom{o}{1}^n$ and $C_{i-1}^{n-1} = \{c_1..c_{n-1} | \forall_j : 1 \le j \le i-1 \Rightarrow c_j = o\} \subseteq \binom{o}{1}^{n-1}$. Note that B_n^n = {(o1)*[o], (10)*[1]} and $C_{n-1}^{n-1} = \{o^{n-1}\}$, and hence that $f(B_n^n)=C_{n-1}^{n-1}$ and $f^{-1}(C_{n-1}^{n-1}) = B_n^n$ (by uniformity). We claim that for all $1 \le i \le n$, $f(B_i^n)=C_{i-1}^{n-1}$ and $f^{-1}(C_{i-1}^{n-1})=B_i^n$. For a proof, use downward induction starting with i=n, for which the claim clearly holds. Suppose it holds for some $i \ge 1$. Consider any $b_1..b_n \in B_{i-1}^n$. It follows that $\bar{b}_1 b_1..b_{n-1} \in B_i^n$ and thus that $f(\bar{b}_1 b_1..b_{n-1}) \in C_{i-1}^{n-1}$. Since f is an emulation, we must have $f(b_1..b_n) \in C_{i-1}^{n-1}$ or $f(b_1..b_n) \in C_{i-2}^{n-1}$. In either case $f(b_1..b_n) \in C_{i-2}^{n-1}$, and we have $f(B_{i-1}^n) \subseteq C_{i-2}^{n-1}$. Because $|B_{i-1}^n| = 2|C_{i-2}^{n-1}|$ and f is uniform, we have in fact $f(B_{i-1}^{n-1})=C_{i-2}^{n-1}$ and ipso facto $f^{-1}(C_{i-2}^{n-1})=B_{i-1}^n$. This completes the inductive argument.

We immediately conclude (take i=2) that for all $x \in \binom{o}{1}^{n-2}$, $f_1(o1x) = o$ and $f_1(1ox) = o$. Because of uniformity this forces $f_1(oox) = f_1(11x) = 1$ for all $x \in \binom{o}{1}^{n-2}$. Define $\tilde{B}_i^n = \{b_1..b_n | b_n..b_1 \in B_i^n\}$ and $\tilde{C}_{i-1}^{n-1} = \{c_1..c_{n-1} | c_{n-1}..c_1 \in C_{i-1}^{n-1}\}$. As before one shows that for all $1 \le i \le n$, $f(\tilde{B}_i^n)=\tilde{C}_{i-1}^{n-1}$ and $f^{-1}(\tilde{C}_{i-1}^{n-1})=\tilde{B}_i^n$. We now argue by downward induction on i that for all $x \in \tilde{B}_i^n$, $f(x)=f_3(x)$ (with f_3 as in table A). For i=n we have $\tilde{B}_n^n$ = {(o1)*[o], (1o)*[1]} and f((o1)*[o]) = f((1o*)[1]) = o^{n-1}, which indeed coincides with f_3. Suppose it holds for some $i \ge 1$. Consider any

$x_1..x_{n-i+1}(o1)^*[o] \in \tilde{B}^n_{i-1}$. If $x_{n-i+1} = 1$ then f and f_3 coincide on the argument by induction. Let $x_{n-i+1} = o$. Observe that $f(x_2..x_{n-i+1}(o1)^*[o]) \in f(\tilde{B}^n_i)=\tilde{C}^{n-1}_{i+1}$ and that $f(x_1..x_{n-i+1}(o1)^*[o]) \in f(\tilde{B}^n_{i-1}\backslash\tilde{B}^n_i)=\tilde{C}^{n-1}_{i-2}\backslash\tilde{C}^{n-1}_{i-1}$, where the latter holds because $x_{n-i+1} = o$ and uniformity of f. It follows that $f(x_1..x_{n-i+1}(o1)^*[o]) \neq f(x_2..x_{n-i+1}(o1)^*[o])$ and thus, because f is an emulation, necessarily $f(x_1..x_{n-i+1}(o1)^*[o]) = \frac{o}{1}.f(x_2..x_{n-i+1}(o1)^*[o])|_{n-2}$. Using the inductive assertion it follows that $f_i(x_1..x_{n-i+1}(o1)^*[o]) = (f_3)_i(x_1..x_{n-i+1}(o1)^*[o])$ for all $2 \leq i \leq n-1$. At the beginning of this paragraph we showed that this must hold also for i=1. Thus f and f_3 coincide on $\tilde{B}^n_{i-1}$, which completes the inductive argument. Because $\tilde{B}^n_1 = V_n$ this shows that f and f_3 coincide for all arguments, which proves the claim. □

Because f was assumed nòt to be step-simulating, claim 2.9.3. clearly proves that case I is contradictory.

Case II. $f((o1)^*[o]) = f((1o)^*[1]) = 1^{n-1}$.
The proof of claim 2.9.3. can be completely dualized to show that in this case f must be equal to $\overline{f}_3$, another one of the six step-simulations listed in table A. Because f was assumed nòt to be step-simulating, this case is also contradictory.

Case III. $f((o1)^*[o]) = f((1o)^*[1]) = (o1)^*[o]$.
We show that for n>2 no emulation f of S_n on S_{n-1} with this property exists. Suppose on the contrary that an f does exist. We derive a contradiction as follows.

First let n be odd, which implies that the assumption turns into $f((o1)^*o) = f((1o)^*1) = (o1)^*$. Since f is uniform no other nodes can be mapped to $(o1)^*$, and we necessarily obtain: $f(oo(1o)^*1) = \frac{o}{1}(o1)^*o$, $f(1(o1)^*oo) = 1(o1)^*\frac{o}{1}$, $f(11(o1)^*o) \in \{1(o1)^*\frac{o}{1}, 11(o1)^*\}$, $f(o(1o)^*11) \in \{\frac{o}{1}(o1)^*o, (o1)^*oo\}$. Observing that necessarily $(f(11(o1)^*o), f((1o)^*1)) \in E_{n-1}$ and $(f((1o)^*1), f(o(1o)^*11)) \in E_{n-1}$ it follows that $f(11(o1)^*o) = 1(o1)^*\frac{o}{1}$ and $f(o(1o)^*11) = \frac{o}{1}(o1)^*o$, and the emulation property now forces that $f(1(o1)^*oo) = f(11(o1)^*o) = 1(o1)^*\frac{o}{1}$ and $f(oo(1o)^*1) = f(o(1o)^*11) =$

$\frac{o}{1}$(o1)*o. From the assumption one easily derives f(11(o1)*o) = $\frac{o}{1}$(o1)*o and f(o(1o)*11) = 1(o1)*$\frac{o}{1}$, thus forcing all four nodes to be mapped to 1(o1)*o. This contradicts uniformity.

For n even we have f((o1)*) = f((1o)*) = (o1)*o. By uniformity again no other nodes are mapped to (o1)*o, and we necessarily obtain: f(oo(1o)*) = $\frac{o}{1}$(o1)*, f((o1)*oo) = (1o)*$\frac{o}{1}$, f((1o)*11) = (1o)*$\frac{o}{1}$, f(11(o1)*) = $\frac{o}{1}$(o1)*. The emulation property forces f(oo(1o)*) and f((o1)*oo) to be adjacent in S_{n-1} (impossible) or equal, hence f(oo(1o)*) = f((o1)*oo) = 1(o1)*. By the same argument f((1o)*11) = f(11(o1)*) = 1(o1)*. Thus four nodes are mapped to 1(o1)*, contradicting uniformity.

<u>Case IV</u>. f((o1)*[o]) = (o1)*[o], f((1o)*[1]) = (1o)*[1].
A more tedious argument is required to show that in this case again every uniform emulation f that satisfies the contraint must be step-simulating, contrary to our basic assumption.

First let n=3, which turns the contraint into f(o1o) = o1 and f(1o1) = 1o. We show that f must be equal to the step-simulations f_1 or $\overline{f}_2$ from table A. By emulation f(oo1) ∈ {o1,oo,1o}, f(1oo) ∈ {o1,1o,11}, f(o11) ∈ {1o,oo,o1}, f(11o) ∈ {1o,o1,11}. Uniformity is heavily used in the following further analysis:

(a) Suppose f(oo1) = f(o1o) = o1. Then f(1oo) = $\frac{o}{1}$o = 1$\frac{o}{1}$, hence f(1oo) = f(1o1) = 1o. It follows that f(o11) = 1$\frac{o}{1}$ = o$\frac{o}{1}$, contradiction.

(b) Suppose f(oo1) = oo. It follows that f(1oo) = 1o (=f(1o1)) and f(ooo) ∈ {oo,1o}, hence f(ooo) = f(oo1) = oo and thus f(o11) = o1 (=f(o1o)) and f(11o) = 11. Necessarily f(111) = 11, and f is proved to coincide with f_1 from table A.

(c) Suppose f(oo1) = f(1o1) = 1o. It follows that f(1oo) ∈ {o1,11}. If f(1oo) = f(o1o) = o1, then f(11o) = oo and this is impossible. Thus f(1oo) = 11 and necessarily f(11o) = o1 (=f(o1o)) and f(o11) = oo. It follows by emulation that f(ooo) = f(1oo) = 11, and f(111) = oo. This proves f equal to $\overline{f}_2$ from table A.

Now let n≥4. We shall first derive a number of auxiliary facts that are needed later.

Claim 2.9.4. For $n \geq 4$, $f(o^n) \in \{o^{n-1}, 1^{n-1}\}$ and $f(1^n) \in \{o^{n-1}, 1^{n-1}\}$.

Proof.

We only consider $f(o^n)$, as the argument for $f(1^n)$ is similar. Let $f(o^n) = u_1..u_{n-1}$. Then $f(1o^{n-1}) \in \{u_1..u_{n-1}, \frac{o}{1}u_1..u_{n-2}\}$ and $f(o^{n-1}1) \in \{u_1..u_{n-1}, u_2..u_{n-1}\frac{o}{1}\}$. The following cases can arise :

(a) $f(1o^{n-1}) = f(o^n) = u_1..u_{n-1}$. Because of uniformity we must have that $f(o^{n-1}1) = u_2..u_{n-1}\frac{o}{1} \neq u_1..u_{n-1}$, and also $f(o1o^{n-2}) = \frac{o}{1}u_1..u_{n-2}$ and $f(1o^{n-2}1) \in \{\frac{o}{1}u_1..u_{n-2}, u_1..u_{n-2}\frac{o}{1}\}$. If $f(1o^{n-2}1) = f(o1o^{n-2}) = \frac{o}{1}u_1..u_{n-2}$ then by uniformity $f(o^{n-2}1o) = u_1..u_{n-2}\frac{o}{1}$ and $f(o^{n-2}11) = u_1..u_{n-2}\frac{o}{1}$, hence $f(o^{n-2}1o) = f(o^{n-2}11) = u_1..u_{n-2}\bar{u}_{n-1}$. Thus $f(o^{n-1}1) = u_2..u_{n-1}\frac{o}{1} = \frac{o}{1}u_1..u_{n-2}$ and necessarily $u = o^{n-1}$ or $u = 1^{n-1}$. In either case uniformity is contradicted. Thus $f(1o^{n-2}1) = u_1..u_{n-2}\frac{o}{1}$, which implies in fact that $f(1o^{n-2}1) = u_1..u_{n-2}\bar{u}_{n-1}$ and hence $f(o^{n-2}1o) \in \{u_1..u_{n-2}\bar{u}_{n-1}, u_2..u_{n-2}\bar{u}_{n-1}\frac{o}{1}\}$. If $f(o^{n-2}1o) = f(1o^{n-2}1) = u_1..u_{n-2}\bar{u}_{n-1}$ then $f(o^{n-1}1) = u_2..u_{n-1}\frac{o}{1} = \frac{o}{1}u_1..u_{n-2}$ and necessarily $u = o^{n-1}$ or $u = 1^{n-1}$. In either case uniformity is contradicted again. Thus $f(1o^{n-2}1) = u_1..u_{n-2}\bar{u}_{n-1}$ and $f(o^{n-2}1o) = u_2..u_{n-2}\bar{u}_{n-1}\frac{o}{1}$, and thus $f(o^{n-1}1) = u_2..u_{n-1}\frac{o}{1} = \frac{o}{1}u_2..u_{n-2}\bar{u}_{n-1}$. It follows that $u_2..u_{n-1} = \alpha^{n-2}$ (with $\alpha = o$ or $\alpha = 1$) and $f(o^{n-1}1) = \alpha^{n-2}\bar{\alpha}$. If $u_1 = \alpha$ then we are finished. Thus assume that $u_1 = \bar{\alpha}$, hence $f(o^n) = f(1o^{n-1}) = \bar{\alpha}\alpha^{n-2}$. Consider $b_1..b_n \in f^{-1}(\alpha^{n-1})$, thus $f(b_1..b_n) = \alpha^{n-1}$. Because of uniformity it follows that $f(ob_1..b_{n-1}) = f(1b_1..b_{n-1}) = \alpha^{n-1}$ (using that $b_1..b_{n-1} = o^{n-1}$), and likewise $f(o1b_1..b_{n-2}) = f(11b_1..b_{n-2}) = \alpha^{n-1}$ and, provided $b_1..b_{n-2} \neq o^{n-2}$, also $f(oob_1..b_{n-2}) = \alpha^{n-1}$. This shows that at least 3 nodes are mapped to α^{n-1}, contradicting uniformity.

(b) $f(o^{n-1}1) = f(o^n) = u_1..u_{n-1}$. The argument is analogous to case (a) by 'reversing' the orientation of the strings.

(c) $f(1o^{n-1}) = \frac{o}{1}u_1..u_{n-2}$ and $f(o^{n-1}1) = u_2..u_{n-1}\frac{o}{1}$. If $f(1o^{n-1}) = f(o^{n-1}1)$ then necessarily $u_1..u_{n-1} = (\alpha\beta)^*[\alpha]$. Because of uniformity $\alpha=\beta$ (otherwise one of $(o1)^*[o]$ and $(o1)^*[1]$ would be mapped to $(\alpha\beta)^*[\alpha]$ too), and thus $u_1..u_{n-1} = o^{n-1}$ or $u_1..u_{n-1} = 1^{n-1}$. It follows that $f(1o^{n-1}) = f(o^{n-1}1) = f(o^n)$, contradicting uniformity. Thus $f(1o^{n-1}) \neq f(o^{n-1}1)$ and by emulation necessarily $f(o^{n-1}1) = u_2..u_{n-1}\frac{o}{1} = u_1..u_{n-2}\frac{o}{1}$, hence $u_1..u_{n-1} = o^{n-1}$ or $u_1..u_{n-1} = 1^{n-1}$. □

(The condition $n \geq 4$ was used in case (a), to make sure that $o^{n-2}1o \neq o1o^{n-2}$ and (hence) $1o^{n-2}1 \neq (1o)^*[1]$ and $o^{n-2}1o \neq (o1)^*[o]$.) Next observe from $f((o1)^*[o]) = (o1)^*[o]$ that $f(oo(1o)^*[1]) \in \{(o1)^*[o], (1o)^*[1], oo(1o)^*[1]\}$ and from $f((1o)^*[1]) = (1o)^*[1]$ that $f(11(o1)^*[o]) \in \{o1)^*[o], (1o)^*[1], 11(o1)^*[o]\}$.

We tackle a particular combination first, because it will be central in the remainder of the proof.

<u>Claim 2.9.5</u>. For $n \geq 4$,

(i) if $f(oo(1o)^*[1]) = oo(1o)^*[1]$, then for all $b_1..b_{n-3} \in (\frac{o}{1})^{n-3}$ there exist $c_1..c_{n-3}$, $c'_1..c'_{n-3}$, $\in (\frac{o}{1})^{n-3}$ such that $f(b_1..b_{n-3}ooo) = c_1..c_{n-3}oo$ and $f(b_1..b_{n-3}oo1) = c'_1..c'_{n-3}$.

(ii) if $f(11(o1)^*[o]) = 11(o1)^*[o]$, then for all $b_1..b_{n-3} \in (\frac{o}{1})^{n-3}$ there exist $c_1..c_{n-3}$, $c'_1..c'_{n-3}$, $\in (\frac{o}{1})^{n-3}$ such that $f(b_1..b_{n-3}11o) = c_1..c_{n-3}11$ and $f(b_1..b_{n-3}111) = c'_1..c'_{n-3}11$.

Proof.

We only prove (i), as (ii) is similar. First we induct on i to show that for all $b_1..b_i \in (\frac{o}{1})^i$ there exist a $c_1..c_i \in (\frac{o}{1})^i$ with $f(b_1..b_i oo1(o1)^*[o]) = c_1..c_i oo(1o)^*[1]$. Since $f(oo1(o1)^*[o]) = oo(1o)^*[1]$ by assumption, we have for $i=1$: $f(b_1 oo1(o1)^*[o]) \in \{oo1(o1)^*[o], \frac{o}{1}oo(1o)^*[1]\}$. If $f(b_1 oo1(o1)^*[o]) = f(oo1(o1)^*[o]) = oo(1o)^*[1]$ then one easily verifies that claim 2.9.1. is contradicted. (Use $\alpha=b_1$, $x=oo1(o1)^*[o]$, $\gamma=o$ or 1.) Thus $f(b_1 oo1(o1)^*[o]) = c_1 oo(1o)^*[1]$, for some $c_1 \in \frac{o}{1}$. Suppose it holds for some i, $1 \leq i < n-3$. Consider $f(b_1..b_{i+1}oo1(o1)^*[o])$. By induction there exists a $c_2..c_{i+1} \in (\frac{o}{1})^i$ such that $f(b_2..b_{i+1}oo1(o1)^*[o]) = c_2..c_{i+1}oo(1o)^*[1]$, and thus $f(b_1b_2..b_{i+1}oo1(o1)^*[o]) \in \{c_2..c_{i+1}oo(1o)^*[1], \frac{o}{1}c_2..c_{i+1}oo(1o)^*[1]\}$. If $f(b_1b_2..b_{i+1}oo1(o1)^*[o]) = f(b_2..b_{i+1}oo1(o1)^*[o]) = c_2..c_{i+1}oo(1o)^*[1]$, then one easily verifies again that claim 2.9.1. is contradicted. Thus $f(b_1..b_{i+1}oo1(o1)^*[o]) = c_1c_2..c_{i+1}oo(1o)^*[1]$, for some $c_1 \in \frac{o}{1}$. This completes the inductive argument. We conclude in particular (take $i=n-3$) that for every $b_1..b_{n-3} \in (\frac{o}{1})^{n-3}$ there exists a $c_1..c_{n-3} \in (\frac{o}{1})^{n-3}$ such that $f(b_1..b_{n-3}oo1) = c_1..c_{n-3}o$.

Next consider $f(b_1..b_{n-3}ooo)$. Since $f(b_2..b_{n-3}ooo1) = c_1..c_{n-3}oo$ for suitable $c_1..c_{n-3} \in (\frac{o}{1})^{n-3}$, it follows that $f(b_1b_2..b_{n-3}ooo) \in$

$\{c_1..c_{n-3}oo,\ \frac{o}{1}c_1..c_{n-3}o\}$. If $b_1..b_{n-3} = o^{n-3}$, then necessarily $f(b_1..b_{n-3}ooo) = o^{n-1}$ by claim 2.9.4. and the form claimed under (i) holds. Thus let $b_1..b_{n-3} \neq o^{n-3}$. If $f(b_1..b_{n-3}ooo) = c_1..c_{n-3}oo$, then the form claimed under (i) holds too. Hence let $f(b_1..b_{n-3}ooo) = \beta c_1..c_{n-3}o$ and, consequently, $f(b_ob_1..b_{n-3}oo) \in \{\beta c_1..c_{n-3}o,\ \frac{o}{1}\beta c_1..c_{n-3}\}$ for some $\beta \in \frac{o}{1}$ and $b_o \in \frac{o}{1}$. (Note that necessarily $b_ob_1..b_{n-3}oo \neq b_1..b_{n-3}ooo$.) If $f(b_o..b_{n-3}oo) = f(b_1..b_{n-3}ooo) = \beta c_1..c_{n-3}o$, then it follows from claim 2.9.1. that $\beta c_1..c_{n-3}o \in \{o^{n-1}, (o1)^*[o], (1o)^*[1]\}$. In each of the three cases uniformity is contradicted. (Note that $o^n \in f^{-1}(o^{n-1})$ by claim 2.9.4. in this case, and $(o1)^*[o] \in f^{-1}((o1)^*[o])$ and $(1o)^*[1] \in f^{-1}((1o)^*[1])$.) Thus $f(b_o..b_{n-3}oo) = \frac{o}{1}\beta c_1..c_{n-3}$. Since $f(b_1..b_{n-3}oo1) = c'_1..c'_{n-3}oo$ for suitable $c'_1..c'_{n-3} \in (\frac{o}{1})^{n-3}$, it follows that $f(b_ob_1..b_0oo) \in \{c'_1..c'_{n-3}oo, \frac{o}{1}c'_1..c'_{n-3}o\}$ and thus ends with a "o". Hence $c_{n-3} = o$, and $f(b_1..b_{n-3}ooo) = c_1..c_{n-4}oo$ as claimed. □

We now begin our case analysis.

Claim 2.9.6. For n≥4, the case f(oo(1o)*[1]) = oo(1o)*[1] and f(11(o1)*[o]) = 11(o1)*[o] is contradictory.

Proof.

By claim 2.9.5. the 2^{n-2} strings of $(\frac{o}{1})^{n-3}ooo \cup (\frac{o}{1})^{n-3}oo1$ are mapped to the 2^{n-3} strings of $(\frac{o}{1})^{n-3}oo$. By uniformity it follows that no other strings can be mapped to $(\frac{o}{1})^{n-3}oo$. Likewise no other strings than the elements of $(\frac{o}{1})^{n-3}11o \cup (\frac{o}{1})^{n-3}111$ are mapped to $(\frac{o}{1})^{n-3}11$. Let $b_1..b_{n-3} \in (\frac{o}{1})^{n-3}$. By claim 2.9.5. we have $f(b_2..b_{n-3}o11o) = c_1..c_{n-3}11$ and $f(b_2..b_{n-3}1ooo) = c'_1..c'_{n-3}oo$ and, consequently, $f(b_1..b_{n-3}o11) \in \{c_1..c_{n-3}11, \frac{o}{1}c_1..c_{n-3}1\}$ and $f(b_1..b_{n-3}1oo) \in \{c'_1..c'_{n-3}oo,\ \frac{o}{1}c'_1..c'_{n-3}o\}$. The cases that $f(b_1..b_{n-3}o11) = f(b_2..b_{n-3}o11o) = c_1..c_{n-3}11$ or $f(b_1..b_{n-3}1oo) = f(b_2..b_{n-3}1ooo) = c'_1..c'_{n-3}oo$ clash with claim 2.9.1. Thus $f(b_1..b_{n-3}o11) = \frac{o}{1}c_1..c_{n-3}1$ and $f(b_1..b_{n-3}1oo) = \frac{o}{1}c'_1..c'_{n-3}o$ and, since neither one can "end" with oo or 11, we have in fact that $f(b_1..b_{n-3}o11) = d_1..d_{n-3}o1$ and $f(b_1..b_{n-3}1oo) = d'_1..d'_{n-3}1o$ (for suitable $d_1..d_{n-3}$ and $d'_1..d'_{n-3}$). By a very similar argument one now shows that $f(b_1..b_{n-3}o1o) = e_1..e_{n-3}o1$ and $f(b_1..b_{n-3}1o1) = e'_1..e'_{n-3}1o$, for suitable $e_1..e_{n-3}$ and $e'_1..e'_{n-3}$. (It follows that f 'resembles' f_1 of table A.)

As f was assumed not to be step-simulating, there must be $x \in (\frac{o}{1})^{n-1}$ and $y \in (\frac{o}{1})^{n-3}$ and $\alpha,\beta,\gamma,\delta \in (\frac{o}{1})$ such that $f(\alpha x) = f(x\gamma) = \beta y\delta$ and $\beta y \neq y\delta$. By claim 2.9.1. one of the strings αx, $x\gamma$ is o^n or 1^n and hence (by claim 2.9.4.) $\beta y\delta \in \{o^n, 1^n\}$, or $\beta y\delta \in \{(o1)^*[o], (1o)^*[1]\}$. In the former case the condition $\beta y \neq y\delta$ is violated. In the latter case we necessarily have $\beta y\delta = \beta y'\overline{\delta}\delta$ (for a suitable y') and hence, by our earlier analysis, necessarily $\alpha x = \alpha x'\overline{\delta}\delta\frac{o}{1}$ for suitable x'. It follows that $x\gamma = x'\overline{\delta}\delta\frac{o}{1}\gamma$ and thus $f(x\gamma)$ ends in $\delta\frac{o}{1}$, contradicting that it equals $\beta y\delta$ and thus ends in $\overline{\delta}\delta$. □

Next let $f(oo(1o)^*[1]) = (o1)^*[o]$ and $f(11(o1)^*[o]) = 11(o1)^*[o]$. Observe that $f(oo(1o)^*[1]) = f((o1)^*[o]) = (o1)^*[o]$ and thus by uniformity, $f(1oo(1o)^*[1]) = \frac{o}{1}(o1)^*[o]$.

Claim 2.9.7. For $n \geq 4$, the case $f(oo(1o)^*[1]) = (o1)^*[o]$ and $f(11(o1)^*[o]) = 11(o1)^*[o]$ is contradictory.

Proof.

We distinguish two further cases.

(a) $f(1oo(1o)^*[1]) = o(o1)^*[o]$. As in the proof of claim 2.9.5. one shows by induction that for all $1 \leq i \leq n-3$ and $b_1..b_i \in (\frac{o}{1})^i$ there exists $c_1..c_i \in (\frac{o}{1})^i$ such that $f(b_1..b_i 1oo(1o)^*[1]) = c_1..c_i o(o1)^*[o]$. Thus for $b_1..b_{n-3} \in (\frac{o}{1})^{n-3}$ we have $f(b_1..b_{n-3}1oo) = c_1..c_{n-3}oo$. It also follows that for every $b_o \in (\frac{o}{1})$ $f(b_o b_1..b_{n-3}1o) = \frac{o}{1}c_1..c_{n-3}o$. For $b_{n-3}=1$ this contradicts claim 2.9.5. (ii).

(b) $f(1oo(1o)^*[1]) = 1(o1)^*[o] = f((1o)^*[1])$. By uniformity one must have $f(o1oo(1o)^*[1]) = 11(o1)^*[o]$. By induction one shows that for all $1 \leq i \leq n-4$ and $b_1..b_i \in (\frac{o}{1})^i$ there exists $c_1..c_i \in (\frac{o}{1})^i$ with $f(b_1..b_i o1oo(1o)^*[1]) = c_1..c_i 11(o1)^*[o]$. Thus for $i=n-4$ we have $f(b_1..b_{n-4}o1oo) = c_1..c_{n-4}11o$, and it follows that also for every $b_o \in \frac{o}{1}$ that $f(b_o b_1..b_{n-4}o1o) = \frac{o}{1}c_1..c_{n-4}11$. (For if $f(b_o b_1..b_{n-4}o1o) = f(b_1..b_{n-4}o1oo) = c_1..c_{n-4}11o$, one easily derives a contradiction with claim 2.9.1.) By claim 2.9.5. and a uniformity argument (cf. the proof of claim 2.9.6.), no other strings than the elements of $(\frac{o}{1})^{n-3}11o \cup (\frac{o}{1})^{n-3}111$ can be mapped to $(\frac{o}{1})^{n-3}11$. This contradicts the assertion for

$f(b_o b_1 .. b_{n-4} o1o)$. □

By a similar argument the following cases are proved contradictory as well: f(oo(1o)*[1]) = (1o)*[1] and f(11(o1)*[o]) = 11(o1)*[o], f(oo(1o)*[1]) = oo(1o)*[1] and f(11(o1)*[o]) = (o1)*[o], and f(oo(1o)*[1]) = oo(1o)*[1] and f(11(o1)*[o]) = (1o)*[1].

Claim 2.9.8. For n≥4, the case f(oo(1o)*[1]) = (o1)*[o] and f(11(o1)*[o]) = (1o)*[1] is contradictory.

Proof.

By uniformity (recall that f((o1)*[o]) = (o1)*[o] and f((1o)*[1]) = (1o)*[1]) we necessarily have f([1](o1)*oo) = [1](o1)*oo, and also f([o](1o)*11) = [o](1o)*11. Thus we have a situation similar to the one considered in claim 2.9.5. and 2.9.6., with the orientation of the strings involved "reversed". Clearly a contradiction is again derived. □

The case f(oo(1o)*[1]) = (1o)*[1] and f(11(o1)*[o]) = (o1)*[o] is proved contradictory in the same way. By noting that the cases f(oo(1o)*[1]) = f(11(o1)*[o]) = (o1)*[o] and f(oo(1o)*[1]) = f(11(o1)*[o]) = (1o)*[1] cannot occur because of uniformity, the case analysis is complete.

Case V. f((01)*[o]) = (1o)*[1], f((1o)*[1]) = (o1)*[o].

This case is "dual" to case IV, which was shown to be contradictory.

Case VI. f((o1)*[o]) = f((1o)*[1]) = (1o)*[1].

This case is "dual" to case III, which was shown to be contradictory.

This completes the proof of theorem 2.9. □

Appendix B. The proof of the topological reduction theorem for emulations of C_n on C_{n-1} (theorem 3.5.)

We use the notations and terminology of Section 3. Our aim is to prove the following result.

Theorem 3.5. (Topological Reduction Theorem). Let $n \geq 4$, and let f be a uniform emulation of C_n on C_{n-1}. Then there exists an $(n-1)$-face A of C_n such that $f(A)$ is an $(n-2)$-face of C_{n-1}.

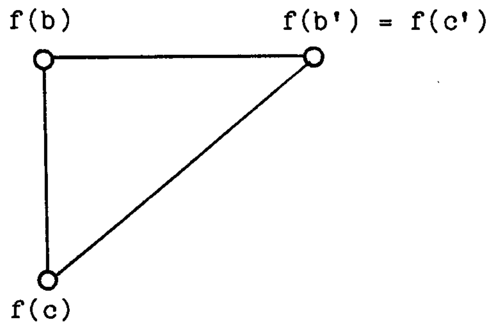
The proof proceeds by way of contradiction. Let $n \geq 4$, and let f be a uniform emulation of C_n on C_{n-1}. Suppose that there does not exist an $(n-1)$-face A of C_n such that $f(A)$ is an $(n-2)$-face of C_{n-1}.

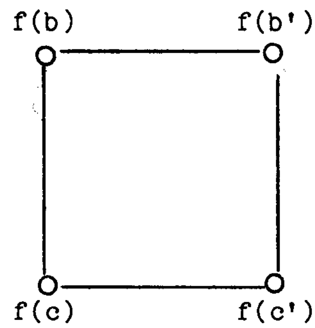
Claim 3.5.1. For every k with $1 \leq k \leq n-1$, there does not exist a k-face A of C_n such that $f(A)$ is a $(k-1)$-face of C_{n-1}.

Proof.

Without loss of generality let $k<n-1$. Suppose the claim is false. Let k be the largest integer $\in 1..n-2$ for which there exists a k-face A of C_n such that $f(A)$ is a $(k-1)$-face of C_{n-1}. Without loss of generality we may assume that the elements of A have identical bits in the last n-k positions, hence $A = \{x\alpha u | x \in (\frac{0}{1})^k\}$ for certain $\alpha \in (\frac{0}{1})$ and $u \in (\frac{0}{1})^{n-k-1}$. Consider the $(k+1)$-face $A' = \{x\frac{0}{1}u | x \in (\frac{0}{1})^k\} = \{x\alpha u | x \in (\frac{0}{1})^k\}$ $\{x\bar{\alpha}u | x \in (\frac{0}{1})^k\}$. For every $b = x\alpha u \in A$, let $b' = x\bar{\alpha}u$. Because of uniformity no elements b' can be mapped into $f(A)$. It follows that $f(b')$ is obtained from $f(b)$ by flipping one bit. We claim that for all $b,c \in A$ one has $f(b)-f(b') = f(c)-f(c')$, where "-" denotes the component-wise subtraction, i.e., $(b_1..b_k)-(c_1..c_k) = (b_1-c_1..b_k-c_k) \in \{-1,0,1\}^k$. It is sufficient to prove this for pairs $b,c \in A$ with $d(b,c) = 1$. Note that $f(b'), f(c') \notin f(A)$. Suppose $f(b') = f(c')$. If $f(b) = f(c)$, then $f(b)-f(b') = f(c)-f(c')$ and we are finished. If $f(b) \neq f(c)$, then necessarily $d(f(b),f(c)) = 1$ and $(f(b),f(c))$ is an edge of C_{n-1} (in fact, of $f(A)$). However, both $f(b)$ and $f(c)$ are connected to $f(b') = f(c') \notin f(A)$ too. It follows that C_{n-1} contains a triangle, which is impossible. Next suppose $f(b') \neq f(c')$. If $f(b) = f(c)$, then one easily argues again that C_{n-1} contains a triangle, and a contradiction arises. If $f(b) \neq f(c)$, then the nodes must form a 4-cycle and hence (necessarily) a 2-face of

C_{n-1}. It follows that f(b)-f(b') = f(c)-f(c').





Using the claim we now argue that f(A') is a k-face of C_{n-1}. Note that A' = A ∪ {b'|b ∈ A}. Fix a b ∈ A and assume that f(b') is obtained from f(b) by flipping the i^{th} bit, where i belongs to the bit-positions with fixed values for face f(A). For arbitrary c ∈ A, the identity f(b)-f(b') = f(c)-f(c') forces that f(c') is obtained from f(c) by flipping exactly the same i^{th} bit. Thus f(A') ⊃ f(A) is a k-face of C_{n-1}. This contradicts that k was the largest integer for which a face of C_n with this property exists. □

We shall now prove a number of results that will eventually contradict claim 3.5.1., which thus proves that our initial assumption was false.

<u>Definition</u>. For $1 \leq k \leq n-1$, a k-face A of C_n is called stable if f(A) is a k-face of C_{n-1}.

<u>Claim 3.5.2</u>. There exists a 2-face A of C_n that is stable.
<u>Proof</u>.

Consider the 2-face A = {xoo..o|x ∈ $(\frac{o}{1})^2$}. Suppose A is not stable, i.e., f(A) is not a 2-face of C_{n-1}. By uniformity f(A) contains at least 2 elements, but by claim 3.5.1. it can not be a 1-face. It follows that f(A) contains precisely 3 elements. Observing adjacencies, the following two cases can arise:

(a) f(oooo..o) = f(11oo..o) and f(o1oo..o) ≠ f(1ooo..o). Consider f(oo1o..o). By uniformity it cannot be equal to f(oooo..o) and f(11oo..o). If f(oo1o..o) = f(o1oo..o), then either f(1o1o..o) =

f(1000..0) or f(1010..0) is different from f(0000..0), f(0010..0), and f(1000..0). In the former case f(0000..0), f(0100..0) and f(1000..0) will form a triangle, which is impossible in C_{n-1}. In the latter case one verifies that B = $\{\alpha o \beta o..o | \alpha,\beta \in \frac{o}{1}\}$ is a stable 2-face of C_n. If f(0010..00) = f(1000..0), then a similar argument shows that B'=$\{o\alpha\beta o..o | \alpha,\beta \in \frac{o}{1}\}$ must be a stable 2-face again. If f(0010..0) ≠ f(0100..0) and ≠ f(1000..0), then observe the following. If f(1010..0) would coincide with either f(0100..0), f(1000..0), or f(0010..0), then triangles are formed in C_{n-1}. Contradiction. Thus f(1010..0) is different from all these, and one verifies again that B is a stable 2-face.

(b) f(0000..0) ≠ f(1100..0) and f(0100..0) = f(1000..0). Consider f(1010..0) and distinguish cases as under (a). Once again triangles in C_{n-1} are formed (contradiction), or B = $\{\alpha o \beta o..o | \alpha,\beta \in (\frac{o}{1})\}$ or B' = $\{o\alpha\beta o..o | \alpha,\beta \in (\frac{o}{1})\}$ is proved a stable 2-face of C_n.

The cases "f(0000..0) = f(0100..0) and f(1100..0) ≠ f(0100..0)" and alike cannot arise, because it would lead to 1-faces being mapped to o-faces (points), contradicting claim 3.5.1. □

The proof of claim 3.5.2. shows, in fact, that either $(\frac{o}{1})^2 o^{n-2}$, $(\frac{o}{1})o(\frac{o}{1})o^{n-3}$ or $o(\frac{o}{1})^2 o^{n-3}$ must be a stable 2-face of C_n.

<u>Claim 3.5.3</u>. For every k with 2≤k≤n-2, there exists a k-face A of C_n that is stable.

Proof.

We induct on k. The case k=2 follows by claim 3.5.2. Assume it holds up to some k with 2≤k<n-2. Let A be a stable k-face of C_n. Without loss of generality we can let A = $\{x\alpha u | x \in (\frac{o}{1})^k\}$ for some $\alpha \in (\frac{o}{1})$ and $u \in (\frac{o}{1})^{n-k-1}$. Let A' = $\{x\bar{\alpha}u | x \in (\frac{o}{1})^k\}$ (a k-face), and for every b = $x\alpha u \in$ A' let b' = $x\bar{\alpha}u \in$ A'. We show that there must exist a stable (k+1)-face.

Suppose first that f(A) ∩ f(A') = ∅. As in the proof of claim 3.5.1. one shows that for all b,c ∈ A : f(b)-f(b') = f(c)-f(c'). Now note that f(A) is a k-face of C_{n-1}. As in the proof of claim 3.5.1. one shows that for all b ∈ A f(b') is obtained from f(b) by flipping the same bit (in a position with fixed value for the elements of f(A)). Thus f(A') is a k-face of C_{n-1} too, and one easily verifies that A ∪ A' =

$\{yu | y \in (\frac{0}{1})^{k+1}\}$ is a stable (k+1)-face of C_n.

Suppose next that $f(A) \cap f(A') \neq \emptyset$. If $f(A) = f(A')$, then $A \cup A'$ is a (k+1)-face of C_n whose image is a k-face (namely, f(A)) of C_{n-1} and a contradiction with claim 3.5.1. arises. Thus $f(A) \neq f(A')$, and it easily follows that $b', c' \in A'$ must exist with $d(b',c') = 1$ and $f(b') \notin f(A)$ and $f(c') \in f(A)$. (We assume that b,c are the corresponding nodes in A.) Let $\underline{b} \neq c$ be any other node $\in A$ adjacent to b, and let $\underline{c} \in A$ be obtained from c by flipping the same bit (as the one flipped to obtain $\underline{b}$ from b). We now claim : (i) $f(b) = f(c')$, (ii) $f(\underline{b}') \notin f(A)$, and (iii) $f(\underline{c}') \in f(A)$. For the proof, observe the following.

(i) Suppose $f(b) \neq f(c')$, and consider f(c). If $f(c) = f(c')$ then f(b), f(b'), and f(c) from a triangle in C_{n-1} (by observing adjacencies). Contradiction. If $f(c) \neq f(c')$, then note that also $f(b) \neq f(c)$ (because f is necessarily 1-1 as a mapping from k-face A onto k-face f(A)). Thus f(b), f(b'), f(c'), and f(c) form a 4-cycle, hence a 2-face of C_{n-1}. But with f(b),f(c'), and f(c) belonging to f(A) the entire 2-face must belong to C_{n-1}, hence $f(b') \in f(A)$. Contradiction. We conclude $f(b) = f(c')$.

(ii) Suppose $f(\underline{b}') \in f(A)$. By uniformity $f(\underline{b}') \neq f(b) = f(c')$. If $f(\underline{b}') = f(\underline{b})$ then f(b), f(b'), and $f(\underline{b}')$ form a triangle in C_{n-1}. Contradiction. If $f(\underline{b}') \neq f(\underline{b})$ then f(b),f(b'),$f(\underline{b})$, and $f(\underline{b}')$ form a 4-cycle, hence a 2-face of C_{n-1} with three nodes in the k-face f(A). It follows that also $f(b') \in f(A)$. Contradiction. We conclude $f(\underline{b}') \notin f(A)$.

(iii) Note that $f(\underline{c}) \neq f(\underline{b})$, (else a contradiction with claim 3.5.1. arises), so $f(\underline{c})$ is adjacent to $f(\underline{b})$, and $f(\underline{b})$ is adjacent to $f(b)=f(c')$. Hence the distance between f(c') and $f(\underline{c})$ is 2. $f(\underline{c}')$ must be adjacent to $f(c') \in f(A)$ and $f(\underline{c}) \in f(A)$, hence $f(c') \in f(A)$.

From the claim we derive that $\underline{b}',\underline{c}'$ is a pair exactly like b',c' and the argument can be repeated. In this way we can let b' range over all of A', and obtain that f(A') must be a k-face of C_{n-1} and $f(A) \cap f(A')$ is a (k-1)-face (because nodes are paired in adjacent couples with one mapped to $f(A) \cap f(A')$ and the other to $f(A') - f(A)$). Now consider two more k-faces A",A''' adjacent (parallel) to A obtained, say, by flipping the first and second bit of u respectively. (Note that $|u| \geq 2$, because $k<n-2$.) Either $f(A) \cap f(A") = \emptyset$ or $f(A) \cap f(A''') = \emptyset$ and we

would be finished by the first part of the proof, or both f(A) $\cap$ f(A") $\neq \emptyset$ and f(A) $\cap$ f(A''') $\neq \emptyset$. In the latter case one derives the same conclusion for f(A") and f(A''') as for f(A'). It follows that $f^{-1}(f(A))$ contains at least 2^{k-1} elements of each A', A", and A''', thus at least $2\frac{1}{2}.2^{k-1}$ elements in all. This contradicts the uniformity of f.

This completes the induction argument. □

We now derive a contradiction as follows. By claim 3.5.3. there exists a (n-2)-face A of C_n that is stable. Without loss of generality we can let A = $\{xoo \mid x \in (\frac{o}{1})^{n-2}\}$. Let A' = $\{x1o \mid x \in (\frac{o}{1})^{n-2}\}$, A" = $\{xo1 \mid x \in (\frac{o}{1})^{n-2}\}$, and A''' = $\{x11 \mid x \in (\frac{o}{1})^{n-2}\}$. From the proof of claim 3.5.3. one derives that A', A", and A''' must be stable (n-2)-faces of C_n as well, and that the f-images of adjacent (parallel) faces are either disjoint or intersect (pairwise) in a (n-3)-face. We distinguish the following cases for the pairwise intersections :

(a) f(A) $\cap$ f(A') is an (n-3)-face, f(A) $\cap$ f(A") is an (n-3)-face. If f(A') $\cap$ f(A''') = $\emptyset$ or f(A") $\cap$ f(A''') = $\emptyset$, then A' $\cup$ A''' = $\{x1\frac{o}{1} \mid x \in (\frac{o}{1})^{n-2}\}$ or A" $\cup$ A''' = $\{x\frac{o}{1}1 \mid x \in (\frac{o}{1})^{n-2}\}$ is stable (n-1)-face (as is its one parallel face A $\cup$ A" or A $\cup$ A', resp.) and either f(A) and f(A") or f(A) and f(A') must be disjoint respectively. Contradiction. We conclude that f(A') $\cap$ f(A''') and f(A") $\cap$ f(A''') both are (n-3)-faces too, in this case. Let b = xoo $\in$ A and c' = y1o $\in$ A' be such that f(b) = f(c') $\in$ f(A) $\cap$ f(A'). Without loss of generality let f(A) = $(\frac{o}{1})^{n-3}(\frac{o}{1})\beta$ and f(A') = $(\frac{o}{1})^{n-3}\alpha(\frac{o}{1})$. Because $f \restriction A$ and $f \restriction A'$ act like isomorphisms of C_{n-2} theorem 3.4. applies, and there must be literals l_i and l_i' corresponding to b_i ($1 \leq i \leq n-2$) and permutations Π and Π' such that $f(b_1..b_{n-2}oo) = l_{\Pi(1)}..l_{\Pi(n-3)}l_{\Pi(n-2)}\beta$ and $f(b_1..b_{n-2}1o) = l'_{\Pi'(1)}..l'_{\Pi'(n-3)}\alpha l'_{\Pi'(n-2)}$. By letting the argument $b_1..b_{n-2}$ range over $(\frac{o}{1})^{n-2}$ and observing that $f(b_1..b_{n-2}oo)$ and $f(b_1..b_{n-2}oo)$ and $f(b_1..b_{n-2}1o)$ must have distance $\leq$ 1, one easily concludes that $\Pi = \Pi'$ and $l_{\Pi(i)} = l'_{\Pi(i)}$ for $1 \leq i \leq n-3$. If f(xoo) = f(y1o) then necessarily x=y or d(x,y) = 1. Now let b' = x1o $\in$ A', b" = xo1 $\in$ A", b''' = x11 $\in$ A''', and let c = yoo $\in$ A, c" = yo1 $\in$ A", c''' = y11 $\in$ A'''. If x=y, then one obtains that the 1-face of C_n spanned by b and c' is mapped to a o-face (a point), contradicting claim 3.5.1. for k=1. If d(x,y) = 1, then b and c are adjacent and likewise are their primed companions. By a similar analysis of f(A') ∩ f(A''') and alike, one shows that necessarily : f(b') = f(c'''), f(b") = f(c), and f(b''') = f(c"). It follows that the 3-face of C_n spanned by b,b',b",b''',c,c',c",c''' is mapped to a 2-face of C_{n-1}. (The case that more f-value coincide is excluded by uniformity.) This contradicts claim 3.5.1. for k=3.

(b) f(A) ∩ f(A') is an (n-3)-face, f(A) ∩ f(A") = ∅. If f(A') ∩ f(A''') is an (n-3)-face, then one can use the argument under case (a) and derive a contradiction. Thus let f(A') ∩ f(A''') = ∅. It follows that both A ∪ A" = $\{xo(\frac{o}{1}) | x \in (\frac{o}{1})^{n-2}\}$ and A' ∪ A'''' = $\{x1(\frac{o}{1}) | x \in (\frac{o}{1})^{n-2}\}$ are stable (n-1)-faces, thus their images each span C_{n-1}. It follows that f(A) ∩ f(A''') and f(A") ∩ f(A''') cannot be empty, and thus must be (n-3)-faces. Now a similar argument as given under case (a) applies to derive a contradiction.

(c) f(A) ∩ f(A') = ∅, f(A) ∩ f(A") = ∅. We may assume that f(A') ∩ f(A''') = ∅ and f(A") ∩ f(A''') = ∅, otherwise analyses similar to case (a) and case (b) apply. It follows that f(A) = f(A''') and f(A') = f(A"), and the sets are complementary (n-2)-faces of C_{n-1}. Consider b=xoo ∈ A, b'=x1o ∈ A', b" = xo1 ∈ A", and b''' = x11 ∈ A'''. Note that there is exactly one node in the (n-2)-face f(A) that is adjacent to f(b') ∉ f(A). Hence f(b) = f(b'''). With a similar argument one shows f(b') = f(b"). It follows that the 2-face of C_n spanned by b,b',b",b''' is mapped to a 1-face. Contradiction with claim 3.5.1.

This ends the proof of theorem 3.5. □