A Combinatorial Treatment of Balancing Networks

COSTAS BUSCH
Brown University, Providence, Rhode Island

MARIOS MAVRONICOLAS
University of Cyprus, Nicosia, Cyprus

(APRIL 30, 1996)

A preliminary version of this work appears in the *Proceedings of the 13th Annual ACM Symposium on Principles of Distributed Computing* (Los Angeles, California, August 1994), ACM, New York, pp. 206–215.

This work was supported in part by ESPRIT III Basic Research Projects # 8144 – LYDIA ("Load Balancing on High Performance Parallel and Distributed Systems") and # 6253 – SHIPS ("Super-Computing Highly Parallel Systems"), by funds for the promotion of research at University of Cyprus (research project "Load Balancing Problems on Shared-Memory Multiprocessor Architectures"), and by NCR Research Funds.

The work of C. Busch was done while at the Department of Computer Science, University of Crete, and Institute of Computer Science, Foundation for Research and Technology – Hellas. The work of M. Mavronicolas was done in part while visiting Institute of Computer Science, Foundation for Research and Technology – Hellas.

Authors' addresses: C. Busch, Department of Computer Science, Brown University, P.O. Box 1910, Providence, RI 02912; M. Mavronicolas, Department of Computer Science, University of Cyprus, Nicosia, CY-1678, Cyprus.

Abstract. *Balancing networks,* originally introduced by Aspnes *et al.* (*Proc. of the 23rd Annual ACM Symposium on Theory of Computing,* pp. 348–358, May 1991), represent a new class of distributed, low-contention data structures suitable for solving many fundamental multi-processor coordination problems that can be expressed as *balancing problems*. In this work, we present a mathematical study of the combinatorial structure of balancing networks, and a variety of its applications.

Our study identifies important combinatorial *transfer parameters* of balancing networks. In turn, necessary and sufficient combinatorial conditions are established, expressed in terms of transfer parameters, which precisely characterize many important and well studied classes of balancing networks such as *counting networks* and *smoothing networks*. We propose these combinatorial conditions to be "balancing analogs" of the well known *Zero-One principle* holding for *sorting networks*.

Within the combinatorial framework we develop, our first application is in deriving combinatorial conditions, involving the transfer parameters, which precisely delimit the boundary between counting networks and sorting networks.

We next turn to use the necessity of the shown combinatorial conditions in deriving width inconstructibility results and lower bounds on depth-like measures for several classes of balancing networks; these results significantly improve upon previous ones shown by Aharonson and Attiya (*Proc. of the 3rd Annual ACM-SIAM Symposium on Discrete Algorithms,* pp. 104–113, January 1992), and Moran and Taubenfeld (*Proc. of the 12th Annual ACM Symposium on Principles of Distributed Computing,* pp. 251–259, August 1993) in terms of strength, generality and proof simplicity.

The sufficiency of the shown combinatorial conditions is employed in designing the first formal algorithms for mathematically verifying that a given network belongs to each of a collection of classes. These algorithms are simple, modular and easy to implement, consisting merely of multiplying matrices and evaluating matricial functions.

Categories and Subject Descriptors: C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design– *distributed networks, network topology*; C.2.4 [**Computer-Communication Networks**]: Distributed Systems– *distributed applications*; F.1.2 [**Computation by Abstract Devices**]: Modes of Computation– *parallelism and concurrency*; F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems– *sorting and searching*; G.2.1 [**Discrete Mathematics**]: Combinatorics– *combinatorial algorithms, counting problems*; G.2.2 [**Discrete Mathematics**]: Graph Theory– *graph algorithms, network problems*

General Terms: Theory, Verification, Algorithms

Additional Key Words and Phrases: Balancing networks, counting networks, smoothing networks, block-output networks, block-input networks, transfer parameters, incidence matrices, combinatorial characterizations, impossibility results

Contents

1 Introduction

Most interesting coordination problems in multi-processor computing require processors to balance their actions in some way. Typical examples of such *balancing problems* include assigning successive memory addresses to processors [18], balancing the computational load on a computer system while minimizing the maximum load on a server [6, 35, 37, 39], and implementing barrier data structures in order to synchronize processes operating at different speeds [1, 24, 30, 32].

In a seminal paper, Aspnes *et al.* [5] proposed *balancing networks* as a new approach to solving balancing problems. Balancing networks, resembling comparator networks (see, e.g., [15, Chapter 28] or [29, Section 5.3.4]) are constructed from simple multi-input, multi-output computing elements called *balancers*, connected to each other through wires. Roughly speaking, a balancer is a toggle mechanism alternately forwarding inputs to each of its output wires. It thus balances its inputs on its output wires. Figure 1 depicts a 3-balancer; we draw wires as horizontal lines with the balancer stretched vertically. The *type* of a balancer is the number of its output wires.

Aspnes *et al.* studied, in particular, *counting networks,* a subclass of balancing networks suitable for solving *counting problems,* where processors need to assign themselves successive values from a given range. They have presented constructions of counting networks built on two-input, two-output balancers, with layouts isomorphic to those of Batcher's *bitonic* sorting network [7], and the *periodic* sorting network of Dowd *et al.* [16], respectively. Subsequently, balancing networks in general, and counting networks in particular, received a lot of interest and attention. The study of balancing networks has focused on both constructions and impossibility results for such networks [2, 3, 19, 23, 26, 27, 28, 34], and analysis of their performance by both theoretical and experimental means [10, 17, 25].

In this work, we embark on a mathematical study of the combinatorial structure of balancing networks. We are interested in understanding how "external" properties of balancing networks come out as a result of "internal" combinatorial structure. Prime examples of properties of interest are the *step* property guaranteed by counting networks, and the *smoothing* property requiring outputs to come as close to each other as possible. The smoothing property has been associated with *smoothing networks* [5], a class of balancing networks suitable for solving *load balancing problems,* where processors need to assign themselves values from a given range that come as close to each other as possible. In addition to the step and smoothing properties, we consider various weaker versions of them, requiring the output to either possess a property weaker than the step or the smoothing property, or possess the step or the smoothing property on a restricted set of inputs. Accordingly, we talk about *block-output networks* or *block-input networks.* Constructions of networks possessing such properties have so far been presented [2, 5, 19, 23, 28]; however, the properties have been proven using rather ad-hoc, operational arguments, and their proofs have established little with respect to the cause of the properties. We attempt a careful and systematic study of the relation of properties of balancing networks to their combinatorial cause.

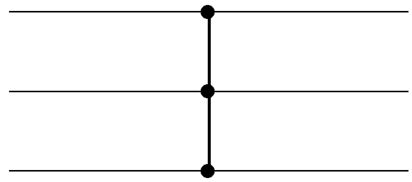

Figure 1: A symbolic representation of a 3-balancer

We introduce a novel matrix representation of a balancing network based on its relative interconnections. More specifically, we use the *incidence matrix* and the *order vector* to describe the relation between inputs and outputs for each of the balancers in a balancing network of depth one, called a *layer*. Roughly speaking, for each balancer, the incidence matrix describes which input is incident to which output, while the order vector assigns orders to all of its outputs; both the incidence matrix and the order vector make explicit use of balancers' types. We represent a balancing network as a collection of pairs of an incidence matrix and an order vector, one pair for each layer.

Our main mathematical instrument is a combinatorial result providing simple algebraic expressions for the outputs of a balancing network as a function of the inputs, depending on the types of balancers used, and the depth and topology of the network, as specified by its incidence matrices and order vectors. More specifically, we identify structural *transfer parameters* of a balancing network, called the *steady transfer matrix* and the *transient transfer function,* that are shown to shape its outputs. Transfer parameters, expressed in terms of incidence matrices and order vectors, are shown to enjoy rich combinatorial properties.

Our main combinatorial result proves invaluable in deriving combinatorial characterization results for block-output and block-input networks. These results are stated as necessary and sufficient conditions on the transfer parameters. For block-output networks, these conditions imply that, loosely speaking, the network uniformly distributes on its output wires the most significant part of the inputs, while the property is inherited down to the response of the network to the least significant part. For block-input networks, we have been able to show a necessary combinatorial property of the steady transfer matrix, implying uniform distribution of the most significant part of inputs.

We propose these combinatorial conditions to be "balancing analogs" of the well known *Zero-One principle* characterizing sorting networks (see, e.g., [15, Section 28.2]). Furthermore, we demonstrate that our combinatorial characterization results are not merely of pure mathematical interest by pursuing several applications of them to problems regarding balancing networks that have attracted significant interest in the past.

The first of these applications attempts to capture the relation between counting and sorting networks, an intriguing problem originally addressed by Aspnes *et al.* [5]. Abusing terminology, we say that a balancing network is a *sorting network* if its isomorphic comparator network is a sorting network. Aspnes *et al.* [5, Section 2.2] show that any counting network is a sorting network but not vice versa; their proof provides an explicit counterexample of a sorting network which is not a counting network, but it does not furnish any mathematical explanation for the observed separation.

We identify a necessary and sufficient condition on the transient transfer function of any sorting network; in contrast, for counting networks, a strictly stronger condition on the transient transfer function together with a combinatorial condition on the steady transfer matrix are shown to be both necessary and sufficient. This result provides a combinatorial interpretation of the separation between counting and sorting networks in terms of properties satisfied by transfer parameters of these networks.

An alternative interpretation of the separation between counting and sorting networks uses the *smooth counting constant* of a balancing network, which is the smallest integer such that if the network produces a step output whenever inputs come within that integer close, then the network "counts" all outputs, or infinite if no such integer exists. We establish that the smooth counting constants of counting and sorting networks, although both finite, are different. Our combinatorial results on the relation between counting and sorting networks provide precise delimiters of the boundary separating them.

We next turn to apply our combinatorial machinery in deriving impossibility results for block-output and block-input networks. We address the problem of constructing balancing networks of arbitrary width that have prescribed properties, originally posed by Aharonson and Attiya [2] and subsequently addressed in [19, 23, 34] for the special case of counting and smoothing networks. More precisely, what is the width of balancing networks with prescribed properties that can be constructed using a finite (but unbounded) number of balancers whose types are in the set $\{p_0, p_1, \ldots, p_{m-1}\}$? We show that a suitably defined factor of the width of a block-output or block-input network of depth d must be a divisor of P^d, where P is the least common multiple of $p_0, p_1, \ldots, p_{m-1}$. Our impossibility results strictly strengthen and significantly generalize previous ones in [2, 34] to arbitrary sets of balancer types and more general classes of networks. (A detailed comparison of these impossibility results to those in [2, 34] is deferred to Section 6.2.) Furtermore, we show lower bound results on several kinds of distances from input to output wires for both block-output and block-input networks.

In practice, it would be necessary to be able to mathematically verify that a given balancing network meets its specification as a module of a multiprocessor architecture; hence, the verification problem for balancing networks, originally addressed by Aspnes *et al.* [5, Section 7] for the special case of counting networks, is of extreme practical significance. Our next application derives the first formal verification algorithms for block-output networks and some special cases of block-input networks. These algorithms are simple, modular and easy to implement. (See Section 6.3 for a comparison of our verification algorithms to related results in [5].) Interestingly, our theory identifies for the first time a non-trivial property on balancing networks, namely the smoothing property, which allows for efficient verification, i.e., verification incurring time complexity that is polynomial in the size of the network.

Our final application identifies a methodology for designing smoothing networks. We argue that our combinatorial results allow the design of smoothing networks to be reduced to the combinatorial problem of determining a matrix chain with prescribed product. We present a pleliminary application of this idea for the case of smoothing networks whose width is a power of two.

The rest of this paper is organized as follows. In Section 2, we introduce a combinatorial framework for the study of balancing networks. In Section 3, we present a fundamental combinatorial result on the algebraic structure of balancing networks; this result introduces transfer parameters for balancing networks. Combinatorial properties of transfer parameters are shown in Section 4, while necessary and sufficient conditions involving them are formulated and shown, in Section 5, for several classes of balancing networks. A variety of applications of these conditions are presented in Section 6. We conclude, in Section 7, with a discussion of our results and some open problems.

2 Framework

In this Section, we introduce a combinatorial framework for the study of balancing networks. Within this framework, we present formal definitions and preliminary properties for balancing networks. We note that our presentation differs from that in [5], and those in [2, 17, 19, 23, 25, 26, 27, 28, 34] adopting it, in that it handles balancers as computing elements, rather than toggle mechanisms which asynchronously relay tokens from input to output wires. Consequently, our definitions are stated as conditions on computed outputs, rather than safety and liveness properties to hold in a quiescent state of an execution (cf. [28, Lemma 3.1]).

This Section is organized as follows. Section 2.1 introduces some useful notation and preliminary facts. Definitions for balancing networks appear in Section 2.2. Section 2.3 considers the related class of comparator networks.

2.1 Notation and Preliminaries

For any real number x, $\lceil x \rceil$ denotes the smallest integer no smaller than x, $\lfloor x \rfloor$ denotes the largest integer no larger than x, and $|x|$ denotes the absolute value of x. Fix throughout any integer $w \geq 2$. $\mathbf{X}^{(w)}$ denotes the vector $\langle x_0, x_1, \ldots, x_{w-1} \rangle^{\mathrm{T}}$, while $\lceil \mathbf{X}^{(w)} \rceil$ and $\lfloor \mathbf{X}^{(w)} \rfloor$ denote the integer vectors $\langle \lceil x_0 \rceil, \lceil x_1 \rceil, \ldots, \lceil x_{w-1} \rceil \rangle^{\mathrm{T}}$ and $\langle \lfloor x_0 \rfloor, \lfloor x_1 \rfloor, \ldots, \lfloor x_{w-1} \rfloor \rangle^{\mathrm{T}}$, respectively. We use $\mathbf{0}^{(w)}$ and $\mathbf{1}^{(w)}$ to denote the vectors $\langle 0, 0, \ldots, 0 \rangle^{\mathrm{T}}$ and $\langle 1, 1, \ldots, 1 \rangle^{\mathrm{T}}$, respectively, each with w entries. The definitions of $\mathbf{0}^{(w)}$ and $\mathbf{1}^{(w)}$ are extended to matrices with w rows and w columns in the natural way to yield $\mathbf{0}^{(w \times w)}$ and $\mathbf{1}^{(w \times w)}$, respectively. For any integer $p \geq 1$, denote $[p] = \{0, 1, \ldots, p-1\}$. We use $\mathcal{N}$ and $\Re$ to denote the sets of natural and real numbers, respectively. In all of our discussion, we will refer to a set $\mathcal{P} = \{p_0, p_1, \ldots, p_{m-1}\}$ of positive integers no less than two, and we will let P denote the least common multiple of integers in $\mathcal{P}$. Without loss of generality, assume p_0 is the largest integer in $\mathcal{P}$.

The *minimum norm* function $\|.\|_{\min} : \Re^w \rightarrow \Re$ is defined by $\|\mathbf{X}^{(w)}\|_{\min} = \min_{i \in [w]} |x_i|$. The *maximum norm* function $\|.\|_{\max} : \Re^w \rightarrow \Re$ is defined by $\|\mathbf{X}^{(w)}\|_{\max} = \max_{i \in [w]} |x_i|$. Both the minimum and the maximum norms can be extended from vectors to matrices in the natural way. We will also use an extension of the maximum norm function from vectors to vector functions $\mathbf{F} : \mathbf{D} \rightarrow \Re^w$ over any domain $\mathbf{D}$, which is defined by setting $\|\mathbf{F}\|_{\max} = \max_{x \in \mathbf{D}} \|\mathbf{F}(x)\|_{\max}$; that is, $\|\mathbf{F}\|_{\max}$ is the maximum value attained by a component of $\mathbf{F}$ over the domain $\mathbf{D}$ of $\mathbf{F}$. The *1-norm* function $\|.\|_1 : \Re^w \rightarrow \Re$ is defined by $\|\mathbf{X}^{(w)}\|_1 = \sum_{i=0}^{w-1} |x_i|$. Clearly, by definitions of maximum norm and 1-norm functions, it follows:

Proposition 2.1 *For any vector* $\mathbf{X}^{(w)} \in \Re^w$, $\|\mathbf{X}^{(w)}\|_1 \leq w \, \|\mathbf{X}^{(w)}\|_{\max}$.

Fix any integer $p \geq 2$, and let $\sum_{i \geq 0} x_i p^i$ denote the representation of the integer $x \geq 0$ in the p-ary arithmetic system, where, for each i, $x_i \in [p]$. For any integer $k \geq 1$, define $x \downarrow_p k = \sum_{0 \leq i \leq k-1} x_i p^i$ and $x \uparrow_p k = \sum_{i \geq k} x_i p^i$; that is, $x \downarrow_p k$ is the integer represented by the k least significant digits in the representation of x in the p-ary arithmetic system, while $x \uparrow_p k$ is the integer obtained from this representation by setting each of those digits to zero. Clearly, $x \downarrow_p k + x \uparrow_p k = x$. We continue with simple expressions for $x \downarrow_p k$ and $x \uparrow_p k$.

Lemma 2.2 *For any integers* $x \geq 0$, $p \geq 2$ *and* $k \geq 1$,

$$x \uparrow_p k \;=\; \left\lfloor \frac{x}{p^k} \right\rfloor p^k \, .$$

Proof: Clearly,

$$\begin{aligned} \left\lfloor \frac{x}{p^k} \right\rfloor p^k &= \left\lfloor \frac{\sum_{i \geq 0} x_i p^i}{p^k} \right\rfloor p^k = \left\lfloor \frac{\sum_{i \geq k} x_i p^i + \sum_{0 \leq i < k} x_i p^i}{p^k} \right\rfloor p^k \\ &= \left\lfloor \sum_{i \geq k} x_i p^{i-k} + \frac{\sum_{0 \leq i < k} x_i p^i}{p^k} \right\rfloor p^k \, . \end{aligned}$$

Notice that $\sum_{i \geq k} x_i p^{i-k}$ is an integer, while

$$\frac{\sum_{0 \leq i < k} x_i p^i}{p^k} \;\leq\; \frac{(p-1) \sum_{0 \leq i < k} p^i}{p^k} \;=\; \frac{p-1}{p^k} \cdot \frac{p^k - 1}{p-1} \;=\; 1 - \frac{1}{p^k} \;<\; 1 \, .$$

Hence,

$$\left\lfloor \sum_{i \geq k} x_i p^{i-k} + \frac{\sum_{0 \leq i < k} x_i p^i}{p^k} \right\rfloor \;=\; \sum_{i \geq k} x_i p^{i-k} \, ,$$

so that

$$\left\lfloor \frac{x}{p^k} \right\rfloor p^k \;=\; \sum_{i \geq k} x_i p^{i-k} \cdot p^k \;=\; \sum_{i \geq k} x_i p^i \;=\; x \uparrow_p k \, ,$$

as needed. ∎

Since $x \downarrow_p k + x \uparrow_p k = x$, Lemma 2.2 immediately implies:

Corollary 2.3 *For any integers $x \geq 0$, $p \geq 2$ and $k \geq 1$,*

$$x \downarrow_p k \;=\; x - \left\lfloor \frac{x}{p^k} \right\rfloor p^k .$$

Furthermore, define $x \updownarrow_p k = x_{k-1}p^{k-1}$; that is, $x \updownarrow_p k$ is the integer represented by the kth least significant p-ary digit of x.

By Lemma 2.2, for any integers m, k and d such that $k+m \geq 1$ and $d \geq 0$,

$$\frac{x \uparrow_p (k+m)}{p^d} \;=\; \left\lfloor \frac{x}{p^{k+m}} \right\rfloor p^{k+m-d} ,$$

which implies:

Lemma 2.4 *For any integers d, k and m such that $0 \leq d \leq k$ and $k+m \geq 1$, $x \uparrow_p (k+m)/p^d$ is an integer multiple of p^{k+m-d}.*

The definitions of $x \downarrow_p k$, $x \uparrow_p k$ and $x \updownarrow_p k$ involving the integer x can be extended component-wise to any vector $\mathbf{X}^{(w)}$ to yield $\mathbf{X}^{(w)} \downarrow_p k$, $\mathbf{X}^{(w)} \uparrow_p k$ and $\mathbf{X}^{(w)} \updownarrow_p k$ in the natural way.

Fix throughout any integer $g \geq 1$ and a *set partition* $\Pi = \{\pi_0, \pi_1, \ldots, \pi_{w-1}\}$ of $[wg]$ into *blocks* of size g; that is, Π is a collection of disjoint, non-empty subsets of $[wg]$, each of size g, whose union is $[wg]$. This set partition induces a *vector partition* of any vector $\mathbf{X}^{(wg)} \in \Re^{wg}$ into vectors $\mathbf{X}_0^{(g)}, \mathbf{X}_1^{(g)}, \ldots, \mathbf{X}_{w-1}^{(g)} \in \Re^g$ in the natural way: for each $j \in [w]$, $\mathbf{X}_j^{(g)} = \langle x_{j_0}, x_{j_1}, \ldots, x_{j_{g-1}} \rangle^{\mathrm{T}}$, where $\pi_j = \{j_0, j_1, \ldots, j_{g-1}\}$ and $j_0 < j_1 < \ldots < j_{g-1}$. Call $\mathbf{X}_j^{(g)}$, $0 \leq j \leq w-1$, a *block vector of* $\mathbf{X}^{(wg)}$ *under* Π, or a block vector of $\mathbf{X}^{(wg)}$ for short.

Furthermore, for any vector function $\mathbf{F} : \Re^{wg} \rightarrow \Re^{wg}$, the partition Π induces a *1-norm partition function* $\mathbf{F}_\Pi : \Re^{wg} \rightarrow \Re^w$, defined as follows:

$$\mathbf{F}_\Pi(\mathbf{X}^{(wg)}) \;=\; \langle \|(\mathbf{F}(\mathbf{X}^{(wg)}))_0^{(g)}\|_1, \|(\mathbf{F}(\mathbf{X}^{(wg)}))_1^{(g)}\|_1, \ldots, \|(\mathbf{F}(\mathbf{X}^{(wg)}))_{w-1}^{(g)}\|_1 \rangle^{\mathrm{T}} ;$$

that is, for each $\mathbf{X}^{(wg)} \in \Re^{wg}$, $\mathbf{F}_\Pi(\mathbf{X}^{(wg)})$ is the vector of 1-norms of the block vectors of $\mathbf{F}(\mathbf{X}^{(wg)})$. The next result establishes a consequence of the definition of $\mathbf{F}_\Pi$.

Lemma 2.5 $\|\mathbf{F}_\Pi\|_{\max} \leq g\,\|\mathbf{F}\|_{\max}$

Proof: We have:

$$\begin{aligned}
\|\mathbf{F}_\Pi\|_{\max} &= \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \|\mathbf{F}_\Pi(\mathbf{X}^{(wg)})\|_{\max} \\
&\quad \text{(by extension of maximum norm to a vector function)} \\
&= \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \max_{j\in[w]} \mathbf{F}_\Pi(\mathbf{X}^{(wg)})[j] \\
&\quad \text{(by definition of maximum norm)} \\
&= \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \max_{j\in[w]} \|(\mathbf{F}(\mathbf{X}^{(wg)}))_j^{(g)}\|_1 \\
&\quad \text{(by definition of } \mathbf{F}_\Pi\text{)} \\
&\leq \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \max_{j\in[w]} g\ \|(\mathbf{F}(\mathbf{X}^{(wg)}))_j^{(g)}\|_{\max} \\
&\quad \text{(by Proposition 2.1)} \\
&= g \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \max_{j\in[w]} \max_{r\in\pi_j} (\mathbf{F}(\mathbf{X}^{(wg)}))_j^{(g)}[r] \\
&\quad \text{(by definition of maximum norm)} \\
&= g \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \max_{j\in[wg]} \mathbf{F}(\mathbf{X}^{(wg)})[j] \\
&= g \max_{\mathbf{X}^{(wg)}\in\Re^{wg}} \|\mathbf{F}(\mathbf{X}^{(wg)})\|_{\max} \\
&\quad \text{(by definition of maximum norm)} \\
&= g\ \|\mathbf{F}\|_{\max}\,, \\
&\quad \text{(by extension of maximum norm to a vector function)}
\end{aligned}$$

as needed. ∎

2.2 Balancing Networks

Section 2.2.1 presents general definitions for balancing networks, while Section 2.2.2 introduces several classes of them.

2.2.1 General Definitions

Balancing networks are constructed from computing elements called balancers and wires much in the same way comparator networks are made up of comparators and wires (see Section 2.3).

For each integer $p \geq 2$, a *p-balancer* $b_p : \mathbf{X}^{(p)} \rightarrow \mathbf{Y}^{(p)}$, or *balancer* for short, is a computing element which receives non-negative, integer inputs $x_0, x_1, \ldots, x_{p-1}$ on input wires $0, 1, \ldots, p-1$, respectively, and computes non-negative integer outputs $y_0, y_1, \ldots, y_{p-1}$ on output wires $0, 1, \ldots, p-1$, respectively, such that (see Figure 2) for each j, $0 \leq j \leq p-1$,

$$y_j = \left\lceil \frac{\|\mathbf{X}^{(p)}\|_1 - j}{p} \right\rceil .$$

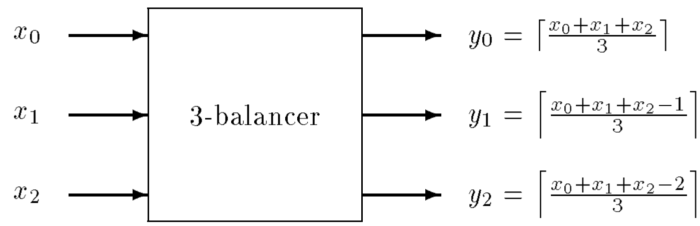

Figure 2: A 3-balancer

For each j, $0 \leq j \leq p-1$, the *order* of output wire j, denoted $\mathrm{ord}(j)$, is defined to be j/p. A *balancer over* $\mathcal{P}$ is a p-balancer for some $p \in \mathcal{P}$. We say that p is the *type* of a p-balancer. An immediate consequence of the definition of a p-balancer follows.

Proposition 2.6 *For a balancer* $b_p : \mathbf{X}^{(p)} \rightarrow \mathbf{Y}^{(p)}$, *for any* j *and* k, $0 \leq j < k \leq p-1$,

$$0 \;\leq\; y_j - y_k \;\leq\; 1\,.$$

Proposition 2.6 establishes the *step property* for the output vector of a p-balancer. The *sum preservation property* for a p-balancer is another immediate consequence of its definition.

Proposition 2.7 *For a balancer* $b_p : \mathbf{X}^{(p)} \rightarrow \mathbf{Y}^{(p)}$, $\|\mathbf{Y}^{(p)}\|_1 = \|\mathbf{X}^{(p)}\|_1$.

Proof: Set $\|\mathbf{X}^{(p)}\|_1 = lp + r$ for some integers l and r such that $l \geq 0$ and $0 \leq r \leq p-1$, so that, by definition of a p-balancer,

$$\|\mathbf{Y}^{(p)}\|_1 = \sum_{j=0}^{p-1} \left\lceil \frac{lp+r-j}{p} \right\rceil = \sum_{j=0}^{p-1} \left\lceil l + \frac{r-j}{p} \right\rceil = \sum_{j=0}^{p-1} \left(l + \left\lceil \frac{r-j}{p} \right\rceil\right) = lp + \sum_{j=0}^{p-1} \left\lceil \frac{r-j}{p} \right\rceil .$$

Notice that $\lceil (r-j)/p \rceil = 1$ if $r > j$ and 0 otherwise. Hence,

$$\|\mathbf{Y}^{(p)}\|_1 \;=\; lp + |\{j \in [p] : r > j\}| \;=\; lp + r \;=\; \|\mathbf{X}^{(p)}\|_1\,,$$

as needed. ∎

A *balancing network* $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *of width* w *over* $\mathcal{P}$ is a collection of balancers over $\mathcal{P}$, where output wires are connected to input wires, having w designated input wires $0, 1, \ldots, w-1$ (which are not connected to output wires of balancers), w designated output wires $0, 1, \ldots, w-1$ (similarly not connected to input wires of balancers), and containing no

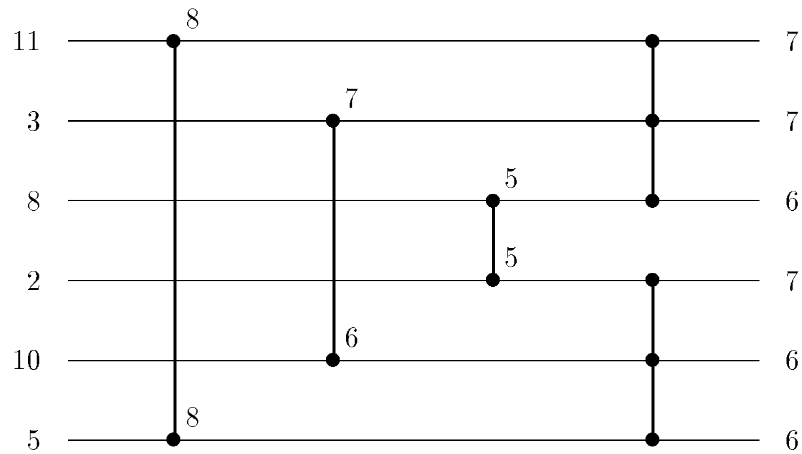

Figure 3: A balancing network

cycles.[1] Non-negative, integer inputs $x_0, x_1, \ldots, x_{w-1}$ are received on input wires $0, 1, \ldots, w-1$, respectively, and non-negative, integer outputs $y_0, y_1, \ldots, y_{w-1}$ are computed on output wires $0, 1, \ldots, w-1$, respectively, in the natural way; that is, each balancer computes its outputs as a function of its inputs and, in turn, the results of intermediate computations are fed into input wires of the next balancers and propagate through to arrive at the network's output wires. (Throughout the paper, we will sometimes abuse notation and use an input or output name as the name of the corresponding wire.) We depict a balancing network of width w as a collection of w horizontal lines with balancers stretched vertically. Figure 3 shows a balancing network, with outputs computed on output wires of all balancers for a specific input; notice that $w = 6$, $\mathbf{X}^{(6)} = \langle 11, 3, 8, 2, 10, 5\rangle^{\mathrm{T}}$ and $\mathbf{Y}^{(6)} = \langle 7, 7, 6, 7, 6, 6\rangle^{\mathrm{T}}$. Note that a line does not represent a single wire, but rather a sequence of distinct wires connecting various balancers.

The sum preservation property for a balancing network $\mathcal{B}$ follows naturally from its definition and the sum preservation property for a balancer (Proposition 2.7).

Proposition 2.8 *For a balancing network* $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$, $\|\mathbf{Y}^{(w)}\|_1 = \|\mathbf{X}^{(w)}\|_1$.

For a balancing network $\mathcal{B}$, the *depth* of $\mathcal{B}$, denoted $depth(\mathcal{B})$, is defined to be the maximal depth of any of its wires, where the depth of a wire is defined to be zero for an input wire of $\mathcal{B}$, and $\max_{l \in [p]} depth(x_l) + 1$ for an output wire of a p-balancer with input wires $x_0, x_1, \ldots, x_{p-1}$.

In case $depth(\mathcal{B}) = 1$, $\mathcal{B}$ will be called a *layer,* and it will be uniquely represented by a matrix $\mathbf{I}_{\mathcal{B}}$ with w rows and w columns, called the *incidence matrix,*[2] which determines incidences between input and output wires, and a vector $\mathbf{O}_{\mathcal{B}}$ with w rows, called the *order vector,* which determines the order of each output wire. Formally, we define:

[1] Balancing networks containing cycles have been considered by Aharonson and Attiya [2, Section 4].

[2] Our notion of an incidence matrix is different from all similar notions of incidence matrices encountered in combinatorial matrix theory (see, e.g., [9]) in that it explicitly uses node degrees.

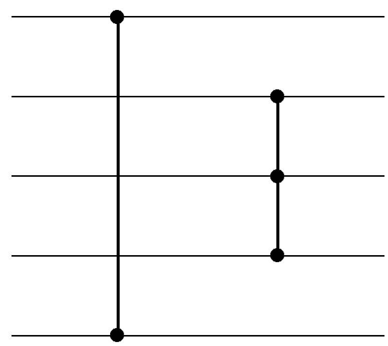

Figure 4: The layer $\mathcal{B}$

- for any i and j, $0 \leq i, j \leq w-1$, $\mathbf{I}_{\mathcal{B}}[ji] = 1/p$ if input wire i and output wire j are connected via a p-balancer, for some $p \in \mathcal{P}$, else $\mathbf{I}_{\mathcal{B}}[ji] = 1$ if output wire j coincides with input wire i, and 0 otherwise;
- for any j, $0 \leq j \leq w-1$, $\mathbf{O}_{\mathcal{B}}[j] = \mathrm{ord}(j)$ if output wire j is the output wire of a balancer, else $\mathbf{O}_{\mathcal{B}}[j] = 0$.

For example, for the layer $\mathcal{B}$ depicted in Figure 4 using the same conventions as for Figure 3, we have that

$$\mathbf{I}_{\mathcal{B}} \;=\; \begin{pmatrix} 1/2 & 0 & 0 & 0 & 1/2 \\ 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 0 & 0 & 0 & 1/2 \end{pmatrix},$$

and

$$\mathbf{O}_{\mathcal{B}} \;=\; \langle\; 0 \quad 0 \quad 1/3 \quad 2/3 \quad 1/2 \;\rangle^{\mathrm{T}}.$$

Apparently, by definitions of an incidence matrix and an order vector, we have:

Observation 2.1 *For a layer $\mathcal{B}$, $\lceil -\mathbf{O}_{\mathcal{B}} \rceil = \mathbf{0}^{(w)}$.*

From definitions of a balancer, an incidence matrix and an order vector, it follows:

Proposition 2.9 *For a layer $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$,*

$$\mathbf{Y}^{(w)} \;=\; \lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} - \mathbf{O}_{\mathcal{B}} \rceil \,.$$

We say that a matrix $\mathbf{I}$ is *doubly stochastic* if the entries of $\mathbf{I}$ are non-negative reals and all row and column sums are equal to one. We have:

Proposition 2.10 *For a layer $\mathcal{B}$, the incidence matrix $\mathbf{I}_{\mathcal{B}}$ is doubly stochastic.*

Proof: For the row (resp., column) sums, if j (resp., i) is not an output (resp., input) wire of a balancer, but coincides with input wire i_j (resp., output wire j_i), then $\sum_{i=0}^{w-1} \mathbf{I}_{\mathcal{B}}[ji] = \mathbf{I}_{\mathcal{B}}[ji_j] = 1$ (resp., $\sum_{j=0}^{w-1} \mathbf{I}_{\mathcal{B}}[ji] = \mathbf{I}_{\mathcal{B}}[j_i i] = 1$); else, assume j (resp., i) is an output (resp., input) wire of a p-balancer, for some $p \in \mathcal{P}$, in which case there are precisely p non-zero entries in row j (resp., column i), each being equal to $1/p$, that sum up to one, as needed. ∎

For a layer $\mathcal{B}$, we define a matrix $\mathbf{D}_{\mathcal{B}}$ with w rows and w columns, called the *distance matrix,* which determines the distance between input and output wires. Formally, for any i and j, $0 \leq i, j \leq w-1$, $\mathbf{D}_{\mathcal{B}}[ji] = 1$ if input wire i and output wire j are connected via a balancer, else $\mathbf{D}_{\mathcal{B}}[ji] = 0$ if input wire i and output wire j coincide, and ∞ otherwise. For example, for the layer $\mathcal{B}$ depicted in Figure 4, we have that

$$\mathbf{D}_{\mathcal{B}} = \begin{pmatrix} 1 & \infty & \infty & \infty & 1 \\ \infty & 1 & 1 & 1 & \infty \\ \infty & 1 & 1 & 1 & \infty \\ \infty & 1 & 1 & 1 & \infty \\ 1 & \infty & \infty & \infty & 1 \end{pmatrix} .$$

Notice that any incidence matrix entry determines the corresponding distance matrix entry, but not vice versa. However, it is easy to see that each of the whole of the distance and incidence matrices determines the other.

If $depth(\mathcal{B}) = d$ is greater than one, then $\mathcal{B}$ can be uniquely partitioned into layers $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_d$ from left to right in the obvious way. The incidence matrix $\mathbf{I}_{\mathcal{B}_i}$ and the order vector $\mathbf{O}_{\mathcal{B}_i}$ are associated with layer $\mathcal{B}_i$, $1 \leq i \leq d$. We represent $\mathcal{B}$ by the sequence of d pairs of an incidence matrix $\mathbf{I}_{\mathcal{B}_i}$ and an order vector $\mathbf{O}_{\mathcal{B}_i}$, $1 \leq i \leq d$.

We inductively extend the definition of the distance matrix from layers to arbitrary balancing networks as follows. Assume that $depth(\mathcal{B})$ is greater than one and let $\mathcal{B}'$ be the balancing network resulting from $\mathcal{B}$ by removing its rightmost layer $\mathcal{B}_d$. For any i and j, $0 \leq i, j \leq w-1$,

$$\mathbf{D}_{\mathcal{B}}[ji] = \min_{l \in [p]} (\mathbf{D}_{\mathcal{B}'}[jl] + \mathbf{D}_{\mathcal{B}_d}[li]) .$$

That is, the distance of input wire i and output wire j is determined by the minimum over all l of the sum of the distance of input wire i and output wire l in $\mathcal{B}'$ and the distance of input wire l and output wire j in $\mathcal{B}'$. We remark that the inductive step of the definition of the distance matrix corresponds to a generalized matrix multiplication over the closed semiring with the operations of min and $+$ on the set of non-negative integers extended to include infinity.[3]

It is possible to extend Lemma 2.4 to the product of a sequence of incidence matrices and an integer vector. Say that an integer vector $\mathbf{X}^{(w)}$ is a *vector multiple* of a scalar integer p if there exists an integer vector $\overline{\mathbf{X}}^{(w)}$ such that $\mathbf{X}^{(w)} = p\overline{\mathbf{X}}^{(w)}$; that is, each entry of $\mathbf{X}^{(w)}$ is an integer multiple of p. We show:

[3]See, e.g., [15, Chapter 26] for all-pairs shortest paths algorithms that employ such multiplications.

Lemma 2.11 *For any integer vector* $\mathbf{X}^{(w)}$ *and integers* d, k *and* m *such that* $1 \leq d \leq k$ *and* $k+m \geq 1$, *the product* $\mathbf{I}_1 \cdot \mathbf{I}_2 \cdot \ldots \cdot \mathbf{I}_d \cdot \mathbf{X}^{(w)} \uparrow_P (k+m)$ *of incidence matrices* $\mathbf{I}_1, \mathbf{I}_2, \ldots, \mathbf{I}_d$ *and the vector* $\mathbf{X}^{(w)} \uparrow_P (k+m)$ *is a vector multiple of* P^{k+m-d}.

Proof: By induction on d. For the base case where $d = 1$, we show that $\mathbf{I}_1 \cdot \mathbf{X}^{(w)} \uparrow_P (k+m)$ is a vector multiple of P^{k+m-1}. For any index $j \in [w]$, take the jth entry

$$\sum_{l=0}^{w-1} \mathbf{I}_1[jl] x_l \uparrow_P (k+m) \;=\; \sum_{l=0}^{w-1} P\, \mathbf{I}_1[jl] \cdot \frac{x_l \uparrow_P (k+m)}{P}$$

of $\mathbf{I}_1 \cdot \mathbf{X}^{(w)} \uparrow_P (k+m)$. Take any index $i \in [w]$ such that $\mathbf{I}_1[ji] \neq 0$. (Since, by Proposition 2.10, $\mathbf{I}_1$ is doubly stochastic, such an index exists.) By definition of incidence matrix, $\mathbf{I}_1[ji] = 1$ or $1/p$ for some $p \in \mathcal{P}$. It follows that $P\mathbf{I}_1[ji]$ is an integer. Also, by Lemma 2.4 (taking $d = 1$), $x_i \uparrow_P (k+m)/P$ is an integer multiple of P^{k+m-1}. Hence, $P\mathbf{I}_1[ji] \cdot x_i \uparrow_P (k+m)/P$ is an integer multiple of P^{k+m-1}. This implies that $\sum_{l=0}^{w-1} P\mathbf{I}_1[jl] \cdot x_l \uparrow_P (k+m)/P$ is an integer multiple of P^{k+m-1}, as needed.

Assume inductively that $\mathbf{I}_2 \cdot \mathbf{I}_3 \cdot \ldots \cdot \mathbf{I}_d \cdot \mathbf{X}^{(w)} \uparrow_P (k+m)$ is a vector multiple of $P^{k+m-(d-1)}$; that is, $\mathbf{I}_2 \cdot \mathbf{I}_3 \cdot \ldots \cdot \mathbf{I}_d \cdot \mathbf{X}^{(w)} \uparrow_P (k+m) = P^{k+m-(d-1)} \overline{\mathbf{X}}^{(w)}$ for some integer vector $\overline{\mathbf{X}}^{(w)}$. Hence,

$$\mathbf{I}_1 \cdot \mathbf{I}_2 \cdot \mathbf{I}_3 \cdot \ldots \cdot \mathbf{I}_d \cdot \mathbf{X}^{(w)} \uparrow_P (k+m) \;=\; P^{k+m-(d-1)} \mathbf{I}_1 \cdot \overline{\mathbf{X}}^{(w)} \;=\; P^{k+m-d}\, P\mathbf{I}_1 \cdot \overline{\mathbf{X}}^{(w)} .$$

It remains to show that $P\mathbf{I}_1 \cdot \overline{\mathbf{X}}^{(w)}$ is an integer vector. Take the jth entry $P \sum_{l=0}^{w-1} \mathbf{I}_1[jl]\, \overline{x}_l$ of $P\mathbf{C}_1 \cdot \overline{\mathbf{X}}^{(w)}$. Take any index $i \in [w]$ such that $\mathbf{I}_1[ji] \neq 0$. (Since, by Proposition 2.10, $\mathbf{I}_1$ is doubly stochastic, such an index exists.) Clearly, $\mathbf{I}_1[ji] = 1$ or $1/p$ for some $p \in \mathcal{P}$. It follows that $P\mathbf{I}_1[ji]$ is an integer. Hence, $P \sum_{l=0}^{w-1} \mathbf{I}_1[jl]\, x_l$ is an integer, as needed. ■

2.2.2 Classes

We proceed to provide definitions for several classes of balancing networks.

Counting and Smoothing Networks

Counting and smoothing networks are the most widely studied classes of balancing networks.

Definition 2.1 (Aspnes, Herlihy and Shavit [5]) *A* counting network over $\mathcal{P}$ *is a balancing network* $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *over* $\mathcal{P}$ *such that for any* j *and* k, $0 \leq j < k \leq w-1$,

$$0 \;\leq\; y_j - y_k \;\leq\; 1 \,.$$

That is, the output vector of a counting network has the *step property*. In [5, 25], counting networks have been shown suitable for implementing *shared counters* [18, 20, 38] and *producer/consumer buffers* for multiprocessor architectures [22]. Let $\mathbf{St}_w$ denote the class of counting networks of width w over $\mathcal{P}$.

Definition 2.2 (Aspnes, Herlihy and Shavit [5]) *For any integer $K \geq 1$, a* K-smoothing network over $\mathcal{P}$ *is a balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ over $\mathcal{P}$ such that for any j and k, $0 \leq j, k \leq w-1$,*

$$|y_j - y_k| \;\leq\; K\,.$$

That is, the output vector of a K-smoothing network has the *K-smoothing property*: each output is one of $K+1$ consecutive integers. Clearly, a counting network is also a K-smoothing network for every integer $K \geq 1$.

For a balancing network $\mathcal{B}$, define the *smoothing constant* of $\mathcal{B}$ to be the smallest integer K such that $\mathcal{B}$ is a K-smoothing network, or infinite if no such integer exists. A balancing network over $\mathcal{P}$ is a *smoothing network over* $\mathcal{P}$ if it is a K-smoothing network over $\mathcal{P}$ for some $K \geq 1$, i.e., if its smoothing constant is finite.

Smoothing networks are appropriate as hardware solutions to *load balancing problems* [5, 25, 36]. Let K-$\mathbf{Sm}_w$ and $\mathbf{Sm}_w$ denote the classes of K-smoothing networks and smoothing networks, respectively, of width w over $\mathcal{P}$.

It is possible to weaken Definitions 2.1 and 2.2 by either relaxing the requirements on the computed outputs or making assumptions on the inputs. Accordingly, we obtain several different sets of weaker definitions.

Block-Output Networks

Instead of requiring that the step or K-smoothing properties hold for individual lines of the output vector, we can require that these properties hold after blocks of output lines have been aggregated. Our formal definitions follow.

Definition 2.3 *A* $w \cdot g$ counting network over $\mathcal{P}$ *is a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ of width wg over $\mathcal{P}$ such that for any j and k, $0 \leq j < k \leq w-1$,*

$$0 \;\leq\; \|\mathbf{Y}_j^{(g)}\|_1 - \|\mathbf{Y}_k^{(g)}\|_1 \;\leq\; 1\,.$$

That is, the output vector is partitioned into block vectors, and the vector containing the sums of these block vectors has the step property. Notice that a $w \cdot 1$ counting network is a counting network. Let $\mathbf{St}_{w \cdot g}$ denote the class of $w \cdot g$ counting networks over $\mathcal{P}$.

Definition 2.4 *For any integer $K \geq 1$, a $w \cdot g$* K-smoothing network over $\mathcal{P}$ *is a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ of width wg over $\mathcal{P}$ such that for any j and k, $0 \leq j\,, k \leq w-1$,*

$$|\|\mathbf{Y}_j^{(g)}\|_1 - \|\mathbf{Y}_k^{(g)}\|_1| \ \leq \ K\,.$$

That is, the output vector is partitioned into block vectors, and the vector containing the sums of these block vectors has the K-smoothing property. Notice that a $w \cdot 1$ K-smoothing network is a K-smoothing network, while a $w \cdot g$ counting network is also a $w \cdot g$ K-smoothing network for every integer $K \geq 1$. We note that Definition 2.4 generalizes [28, Definition 7.1].

For a balancing network $\mathcal{B}$ of width wg, define the *$w \cdot g$ smoothing constant* of $\mathcal{B}$ to be the smallest integer K such that $\mathcal{B}$ is a $w \cdot g$ K-smoothing network, or infinite if no such integer exists. A balancing network of width $w \cdot g$ over $\mathcal{P}$ is a *$w \cdot g$ smoothing network over $\mathcal{P}$* if it is a $w \cdot g$ K-smoothing network over $\mathcal{P}$ for some $K \geq 1$, i.e., if its $w \cdot g$ smoothing constant is finite.

Let K-$\mathbf{Sm}_{w \cdot g}$ and $\mathbf{Sm}_{w \cdot g}$ denote the classes of $w \cdot g$ K-smoothing networks and $w \cdot g$ smoothing networks, respectively, over $\mathcal{P}$.

Any of a $w \cdot g$ counting network, a $w \cdot g$ K-smoothing network, or a $w \cdot g$ smoothing network will be called a *block-output network*. This name reflects the fact that they are defined in terms of properties holding on the block vectors of their output vectors.

Block-Input Networks

An alternative way of weakening Definitions 2.1 and 2.2 is to relax the step or K-smoothing property of the output vector to hold for input vectors that have some kind of a step or K-smoothing property, rather than all input vectors. Our formal definitions follow.

Definition 2.5 *A $w \cdot g$* step counting network over $\mathcal{P}$ *is a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ over $\mathcal{P}$ such that if for every r, $0 \leq r \leq w-1$, for any j and k in π_r such that $j < k$,*

$$0 \ \leq \ x_j - x_k \ \leq \ 1\,,$$

then, for any j and k, $0 \leq j < k \leq wg-1$,

$$0 \ \leq \ y_j - y_k \ \leq \ 1\,.$$

That is, the output vector has the step property whenever each block vector of the input vector does. Let $\mathbf{St}_{w \cdot g} \rightsquigarrow \mathbf{St}_{wg}$ denote the class of $w \cdot g$ step counting networks over $\mathcal{P}$.

Definition 2.6 *For any integers K_1 and K such that $K_1 > K \geq 1$, a $w \cdot g$* K_1-smooth K-smoothing network over $\mathcal{P}$ *is a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ over $\mathcal{P}$ such that if for every r, $0 \leq r \leq w-1$, for any j and k in π_r,*

$$|x_j - x_k| \ \leq \ K_1\,,$$

then, for any j and k, $0 \leq j, k \leq wg - 1$,

$$|y_j - y_k| \ \leq \ K\,.$$

That is, the output vector has the K-smoothing property whenever each block vector of the input vector has the K_1-smoothing property. (The condition $K_1 > K$ has been imposed in order to outlaw trivial balancing networks from consideration.) Let $K_1\text{-}\mathbf{Sm}_{w \cdot g} \rightsquigarrow K\text{-}\mathbf{Sm}_{wg}$ denote the class of $w \cdot g$ K_1-smooth K-smoothing networks over $\mathcal{P}$.

A way of slightly strengthening both Definitions 2.5 and 2.6 is to require the step property for the output vector only if the input vector has some kind of a smoothing property.

Definition 2.7 *For any integer $K_1 \geq 1$, a $w \cdot g$ K_1-*smooth counting network *over $\mathcal{P}$ is a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ over $\mathcal{P}$ such that if for every r, $0 \leq r \leq w-1$, for any j and k in π_r,*

$$|x_j - x_k| \ \leq \ K_1\,,$$

then, for any j and k, $0 \leq j < k \leq wg - 1$,

$$0 \ \leq \ y_j - y_k \ \leq \ 1\,.$$

That is, the output vector has the step property whenever each block vector of the input vector has the K_1-smoothing property. Let $K_1\text{-}\mathbf{Sm}_{w \cdot g} \rightsquigarrow \mathbf{St}_{wg}$ denote the class of $w \cdot g$ K_1-smooth counting networks over $\mathcal{P}$.

Alternatively, a way of slightly weakening both Definitions 2.5 and 2.6 is to relax the K-smoothing property for the output vector to hold just in case the input vector has some kind of a step property.

Definition 2.8 *For any integer $K \geq 1$, a $w \cdot g$* step K-smoothing network *over $\mathcal{P}$ is a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ over $\mathcal{P}$ such that if for every r, $0 \leq r \leq w-1$, for any j and k in π_r such that $j < k$,*

$$0 \ \leq \ x_j - x_k \ \leq \ 1\,,$$

then, for any j and k, $0 \leq j, k \leq wg - 1$,

$$|y_j - y_k| \ \leq \ K\,.$$

That is, the output vector has the K-smoothing property whenever each block vector of the input vector has the step property. Let $\mathbf{St}_{w \cdot g} \rightsquigarrow K\text{-}\mathbf{Sm}_{wg}$ denote the class of $w \cdot g$ step K-smoothing networks over $\mathcal{P}$.

Any of a $w \cdot g$ step counting network, a $w \cdot g$ K_1-smooth K-smoothing network, a $w \cdot g$ K_1-smooth counting network, or a $w \cdot g$ step K-smoothing network will be called a *block-input network*. This name reflects the fact that they are defined in terms of conditions required to hold in case the block vectors of their input vectors have certain properties. Interesting special cases of Definitions 2.6 and 2.7 are obtained by setting $w = 1$; that is, the input vector is trivially partitioned into a single block vector.

Definition 2.9 *For any integers K_1 and K such that $K_1 > K \geq 1$, a* K_1-smooth K-smoothing network over $\mathcal{P}$ *is a $1 \cdot g$ K_1-smooth K-smoothing network over $\mathcal{P}$.*

That is, the output vector has the K-smoothing property whenever the input vector has the K_1-smoothing property. Intuitively, a K_1-smooth K-smoothing network amplifies the "smoothness" of its input vector.

Definition 2.10 (Klugerman and Plaxton [28]) *For any integer $K_1 \geq 1$, a* K_1-smooth counting network over $\mathcal{P}$ *is a $1 \cdot g$ K_1-smooth counting network over $\mathcal{P}$.*

That is, the output vector has the step property whenever the input vector has the K_1-smoothing property. In [28], a K_1-smooth counting network is called a K_1*-counter*.

For a balancing network $\mathcal{B}$, define the *smooth counting constant* of $\mathcal{B}$ to be the smallest integer K such that if $\mathcal{B}$ is a K-smooth counting network, then $\mathcal{B}$ is a counting network, or infinite if no such integer exists. Since a counting network is required to produce a step output on every input, it appears that the smooth counting constant of a counting network is infinite; however, we will later show that, surprisingly, it is finite.

Klugerman and Plaxton present an explicit construction of a 2-smooth counting network of width g and depth $\Theta(\lg g)$ over $\{2\}$ [28, Section 4.3]. Furthermore, they use this construction as a building block in constructing a "random" network with depth $\Theta(\lg g)$ over $\{2\}$ that counts with high probability [28, Section 5]. (Klugerman [27, Section 5.5] later derandomized this construction.)

2.3 Comparator Networks

Comparator networks are made up of computing elements called comparators and wires (see, e.g., [15, Chapter 28] or [29, Section 5.3.4]).

For any integer $p \geq 2$, a *p-comparator* $c_p : \mathbf{X}^{(p)} \rightarrow \mathbf{Y}^{(p)}$, or *comparator* for short, is a computing element which receives integer inputs $x_0, x_1, \ldots, x_{p-1}$ on input wires $0, 1, \ldots, p-1$,[4] respectively, and computes integer outputs $y_0, y_1, \ldots, y_{p-1}$ on output wires $0, 1, \ldots, p-1$, respectively, such that for each j, $0 \leq j \leq p-1$, y_j is the *jth order statistic* of the (multi)set $\{x_0, x_1, \ldots, x_{p-1}\}$, denoted $s_j(\mathbf{X}^{(p)})$; that is, y_j is equal to x_{i_j} where $x_{i_0} \geq x_{i_1} \geq \ldots \geq x_{i_{p-1}}$. A *comparator over $\mathcal{P}$* is a p-comparator for some $p \in \mathcal{P}$. The *sorting property* for the output vector of a p-comparator follows immediately from its definition.

Proposition 2.12 *For a comparator $c_p : \mathbf{X}^{(p)} \rightarrow \mathbf{Y}^{(p)}$, for any j and k, $0 \leq j < k \leq p-1$,*

$$0 \;\leq\; y_j - y_k \,.$$

[4] The assumption of integer inputs is made here in order to retain the same kind of inputs and outputs for comparator networks and balancing networks. In general, inputs and outputs of a comparator network may be drawn from any domain over which a total order is defined.

A *comparator network* $\mathcal{C} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *of width* w *over* $\mathcal{P}$ is a collection of comparators over $\mathcal{P}$, where output wires are connected to input wires, having w designated input wires $0, 1, \ldots, w-1$ (which are not connected to output wires of comparators), w designated output wires $0, 1, \ldots, w-1$ (similarly not connected to input wires of comparators), and containing no cycles. Integer inputs $x_0, x_1, \ldots, x_{w-1}$ are received on input wires $0, 1, \ldots, w-1$, respectively, and integer outputs $y_0, y_1, \ldots, y_{w-1}$ are computed on output wires $0, 1, \ldots, w-1$, respectively, in the natural way.

Definition 2.11 *A comparator network* $\mathcal{C} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *is a* sorting network *if for any* j *and* k*,* $0 \leq j < k \leq w-1$*,*

$$0 \leq y_j - y_k \,.$$

That is, the output vector of a sorting network has the sorting property. A classical result known as *Zero-One Principle* (see, e.g., [15, 29]) implies that the sorting property is algorithmically checkable for a given comparator network, albeit in time exponential in the width of the network.

Proposition 2.13 (Zero-One Principle) *A comparator network* $\mathcal{C} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *is a sorting network if (and only if)* $\mathbf{Y}^{(w)}$ *has the sorting property for every* $\mathbf{X}^{(w)}$ *such that* $\|\mathbf{X}^{(w)}\|_{\max} \leq 1$.

The *isomorphic comparator network* of a balancing network $\mathcal{B}$, denoted $C(\mathcal{B})$, is a comparator network obtained from $\mathcal{B}$ by substituting each p-balancer of $\mathcal{B}$ by a p-comparator. The following technical result indicates that a balancing network and its isomorphic comparator network compute identical outputs on a common binary input.

Lemma 2.14 *Consider a balancing network* $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *and its isomorphic comparator network* $C(\mathcal{B}) : \overline{\mathbf{X}}^{(w)} \rightarrow \overline{\mathbf{Y}}^{(w)}$*. If* $\mathbf{X}^{(w)} = \overline{\mathbf{X}}^{(w)} \in \{0,1\}^w$*, then* $\mathbf{Y}^{(w)} = \overline{\mathbf{Y}}^{(w)}$*.*

Proof: The claim follows naturally from the corresponding claim for a p-balancer, for any $p \in \mathcal{P}$, which we show. Consider any index $j \in [p]$. Since $\mathbf{X}^{(p)} \in \{0,1\}^p$, $0 \leq \|\mathbf{X}^{(p)}\|_1 \leq p$, so that $-(p-1) \leq \|\mathbf{X}^{(p)}\|_1 - j \leq p$, i.e.,

$$-1 + \frac{1}{p} \;\leq\; \frac{\|\mathbf{X}^{(p)}\|_1 - j}{p} \;\leq\; 1\,.$$

Hence, by definition of a p-balancer,

$$y_j \;=\; \left\lceil \frac{\|\mathbf{X}^{(p)}\|_1 - j}{p} \right\rceil \;=\; \begin{cases} 1, & j < \|\mathbf{X}^{(p)}\|_1 \\ 0, & j \geq \|\mathbf{X}^{(p)}\|_1 \end{cases}$$

On the other hand,

$$\begin{aligned}\overline{y}_j &= s_j(\overline{\mathbf{X}}^{(p)}) && \text{(by definition of a } p\text{-comparator)}\\ &= s_j(\mathbf{X}^{(p)}) && \text{(since } \mathbf{X}^{(p)} = \overline{\mathbf{X}}^{(p)})\\ &= \begin{cases} 1, & j < \|\mathbf{X}^{(p)}\|_1 \\ 0, & j \geq \|\mathbf{X}^{(p)}\|_1 \end{cases} .\end{aligned}$$

It follows that $y_j = \overline{y}_j$, as needed. ■

We will sometimes abuse terminology and say that the balancing network $\mathcal{B}$ is a *sorting network* if the comparator network $C(\mathcal{B})$ is a sorting network. Let $\mathbf{Srt}_w$ denote the class of sorting networks of width w over $\mathcal{P}$. We next show the coincidence of the classes of sorting networks and 1-smooth counting networks.

Proposition 2.15 *The class of sorting networks is identical to the class of 1-smooth counting networks.*

Proof: It is shown in [5, Theorem 2.6] that any sorting network is a 1-smooth counting network. To show the opposite inclusion, consider a 1-smooth counting network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$. In order to prove that $C(\mathcal{B}) : \overline{\mathbf{X}}^{(w)} \rightarrow \overline{\mathbf{Y}}^{(w)}$ is a sorting network, it suffices, by Proposition 2.13, to show that for any input vector $\overline{\mathbf{X}}^{(w)} \in \{0,1\}^w$, the output vector $\overline{\mathbf{Y}}^{(w)}$ has the sorting property. Set $\mathbf{X}^{(w)} = \overline{\mathbf{X}}^{(w)} \in \{0,1\}^w$. By Lemma 2.14, $\mathbf{Y}^{(w)} = \overline{\mathbf{Y}}^{(w)}$. Clearly, $\mathbf{X}^{(w)} \in \{0,1\}^w$ has the 1-smoothing property; since $\mathcal{B}$ is a 1-smooth counting network, it follows that $\mathbf{Y}^{(w)}$ has the step property. Hence, it follows that $\overline{\mathbf{Y}}^{(w)}$ has the sorting property, as needed. ■

Proposition 2.15 provides an interesting complement to the "Smoothing + Sorting = Counting" principle shown by Aspnes *et al.* [5, Theorem 2.6]. It implies that sorting networks are *unique* in having the property that their cascading with a 1-smoothing network results in a counting network. Our final Proposition represents a natural generalization of the Zero-One Principle, stated for sorting networks, *viz.* 1-smooth counting networks (by Proposition 2.15), to K_1-smooth counting networks.

Proposition 2.16 (Generalized Zero-One Principle) *A balancing network $\mathcal{B} : \mathbf{X}^{(g)} \rightarrow \mathbf{Y}^{(g)}$ is a K_1-smooth counting network if (and only if) $\mathbf{Y}^{(g)}$ has the step property for every $\mathbf{X}^{(g)}$ such that $\|\mathbf{X}^{(g)}\|_{\max} \leq K_1$.*

Proof: We establish that $\mathcal{B}$ is a K_1-smooth counting network (Definition 2.10) by showing that $\mathbf{Y}^{(g)}$ has the step property on any K_1-smooth input vector $\mathbf{X}^{(g)}$. Write $\mathbf{X}^{(g)} = x\,\mathbf{1}^{(g)} + \overline{\mathbf{X}}^{(g)}$, where $\|\overline{\mathbf{X}}^{(g)}\|_{\max} \leq K_1$. Let $\overline{\mathbf{Y}}^{(g)}$ be the output of $\mathcal{B}$ on input $\overline{\mathbf{X}}^{(g)}$. By assumption, $\overline{\mathbf{Y}}^{(g)}$ has the step property. We prove:

Claim 2.17 $\mathbf{Y}^{(g)} = x\,\mathbf{1}^{(g)} + \overline{\mathbf{Y}}^{(g)}$

Proof: The claim follows naturally from the corresponding claim for a p-balancer, for any balancer over $\mathcal{P}$ $b_p : \mathbf{X}^{(p)} \to \mathbf{Y}^{(p)}$, which we show. Clearly,

$$\begin{aligned}
\mathbf{Y}^{(p)} &= \left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(p)} - \mathbf{O}_{\mathcal{B}} \right\rceil && \text{(by Proposition 2.9)} \\
&= \left\lceil \mathbf{I}_{\mathcal{B}} \cdot (x\,\mathbf{1}^{(p)} + \overline{\mathbf{X}}^{(p)} - \mathbf{O}_{\mathcal{B}} \right\rceil && \\
&= \left\lceil x\,\mathbf{I}_{\mathcal{B}} \cdot \mathbf{1}^{(p)} + \mathbf{I}_{\mathcal{B}} \cdot \overline{\mathbf{X}}^{(p)} - \mathbf{O}_{\mathcal{B}} \right\rceil && \\
&= \left\lceil x\,\mathbf{1}^{(p)} + \mathbf{I}_{\mathcal{B}} \cdot \overline{\mathbf{X}}^{(p)} - \mathbf{O}_{\mathcal{B}} \right\rceil && \text{(by Proposition 2.10)} \\
&= x\,\mathbf{1}^{(p)} + \left\lceil \mathbf{I}_{\mathcal{B}} \cdot \overline{\mathbf{X}}^{(p)} - \mathbf{O}_{\mathcal{B}} \right\rceil && \text{(since } x \text{ is an integer)} \\
&= x\,\mathbf{1}^{(p)} + \overline{\mathbf{Y}}^{(p)} && \text{(by Proposition 2.9)},
\end{aligned}$$

as needed. ∎

Since $\overline{\mathbf{Y}}^{(g)}$ has the step property and $x\,\mathbf{1}^{(g)}$ is a constant vector, it follows, by Claim 2.17, that $\mathbf{Y}^{(g)}$ has the step property, as needed. ∎

Klugerman and Plaxton have independently shown the generalized Zero-One Principle [28, Lemma 4.3] using a similar but less transparent proof.

3 The Algebraic Structure of Balancing Networks

We show that for any balancing network, the output vector takes a particular algebraic form as a function of the input vector, depending on the types of balancers used, and the depth and topology of the network.[5]

Theorem 3.1 *Let $\mathcal{B} : \mathbf{X}^{(w)} \to \mathbf{Y}^{(w)}$ be a balancing network of depth d over $\mathcal{P}$ with associated incidence matrices $\mathbf{I}_{\mathcal{B}_1}, \mathbf{I}_{\mathcal{B}_2}, \ldots, \mathbf{I}_{\mathcal{B}_d}$, and order vectors $\mathbf{O}_{\mathcal{B}_1}, \mathbf{O}_{\mathcal{B}_2}, \ldots, \mathbf{O}_{\mathcal{B}_d}$, respectively. Then,*

$$\mathbf{Y}^{(w)} \;=\; \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \uparrow_P d + \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)} \downarrow_P d)\,,$$

for some matrix $\mathbf{I}_{\mathcal{B}}$ and vector function $\mathbf{F}_{\mathcal{B}} : [P^d]^w \to \mathcal{N}^w$, where

(1) $\mathbf{I}_{\mathcal{B}} \;=\; \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{I}_{\mathcal{B}_{d-1}} \cdot \ldots \cdot \mathbf{I}_{\mathcal{B}_1}$, *and*

[5] We note that Aharonson and Attiya [2, Lemma 3.1] provide expressions for the outputs of a balancing network in the special case where $x_0 = w\prod_{p\in\mathcal{P}} p$ and $x_i = 0$ for $i \neq 0$.

(2) $\mathbf{F}_{\mathcal{B}} = \mathbf{F}_{\mathcal{B}_d}$, *where the vector functions* $\mathbf{F}_{\mathcal{B}_l} : [P^l]^w \rightarrow \mathbf{N}^w$, $1 \leq l \leq d$, *are defined inductively as follows:*

$$
\begin{aligned}
&\mathbf{F}_{\mathcal{B}_l}(\mathbf{X}^{(w)} \downarrow_P l) \\
= &\begin{cases} \left\lceil \mathbf{I}_{\mathcal{B}_l} \cdot \mathbf{I}_{\mathcal{B}_{l-1}} \cdot \ldots \cdot \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P l + \mathbf{I}_{\mathcal{B}_l} \cdot \mathbf{F}_{\mathcal{B}_{l-1}}(\mathbf{X}^{(w)} \downarrow_P (l-1)) - \mathbf{O}_{\mathcal{B}_l} \right\rceil, & l > 1 \\[2ex] \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P 1 - \mathbf{O}_{\mathcal{B}_1} \right\rceil, & l = 1 \end{cases}
\end{aligned}
$$

Proof: By induction on the depth d of the network. For the base case where $d = 1$, notice that

$$
\begin{aligned}
\mathbf{Y}^{(w)} &= \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} - \mathbf{O}_{\mathcal{B}_1} \right\rceil & \text{(by Proposition 2.9)} \\
&= \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot (\mathbf{X}^{(w)} \uparrow_P 1 + \mathbf{X}^{(w)} \updownarrow_P 1) - \mathbf{O}_{\mathcal{B}_1} \right\rceil \\
&= \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \uparrow_P 1 + \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P 1 - \mathbf{O}_{\mathcal{B}_1} \right\rceil .
\end{aligned}
$$

An application of Lemma 2.11 with $d = k = 1$ and $m = 0$ yields that $\mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \uparrow_P 1$ is a vector multiple of $P^{1+0-1} = 1$, i.e., an integer vector. It follows that

$$
\mathbf{Y}^{(w)} = \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \uparrow_P 1 + \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P 1 - \mathbf{O}_{\mathcal{B}_1} \right\rceil ,
$$

as needed.

Assume inductively that the claim holds for all balancing networks of depth $d-1$ over $\mathcal{P}$. Let $\mathcal{B}' : \mathbf{X}^{(w)} \rightarrow \mathbf{Z}^{(w)}$ be the balancing network of depth $d-1$ obtained from $\mathcal{B}$ by removing its rightmost layer. By induction hypothesis,

$$
\mathbf{Z}^{(w)} = \mathbf{I}_{\mathcal{B}_{d-1}} \cdot \mathbf{I}_{\mathcal{B}_{d-2}} \cdot \ldots \cdot \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \uparrow_P (d-1) + \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{(w)} \downarrow_P (d-1)) ,
$$

where the vector functions $\mathbf{F}_{\mathcal{B}_l} : [P^l]^w \rightarrow \mathbf{N}^w$, $1 \leq l \leq d-1$, are defined inductively as follows:

$$
\begin{aligned}
&\mathbf{F}_{\mathcal{B}_l}(\mathbf{X}^{(w)} \downarrow_P l) \\
= &\begin{cases} \left\lceil \mathbf{I}_{\mathcal{B}_l} \cdot \mathbf{I}_{\mathcal{B}_{l-1}} \cdot \ldots \cdot \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P l + \mathbf{I}_{\mathcal{B}_l} \cdot \mathbf{F}_{\mathcal{B}_{l-1}}(\mathbf{X}^{(w)} \downarrow_P (l-1)) - \mathbf{O}_{\mathcal{B}_l} \right\rceil, & l > 1 \\[2ex] \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P 1 - \mathbf{O}_{\mathcal{B}_1} \right\rceil, & l = 1 \end{cases}
\end{aligned}
$$

Hence,

$$
\begin{aligned}
\mathbf{Y}^{(w)} &= \left\lceil \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{Z}^{(w)} - \mathbf{O}_{\mathcal{B}_d} \right\rceil & \text{(by Proposition 2.9)} \\
&= \left\lceil \mathbf{I}_{\mathcal{B}_d} \cdot (\mathbf{I}_{\mathcal{B}_{d-1}} \cdot \mathbf{I}_{\mathcal{B}_{d-2}} \cdot \ldots \cdot \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \uparrow_P (d-1) + \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{(w)} \downarrow_P (d-1))) - \mathbf{O}_{\mathcal{B}_d} \right\rceil
\end{aligned}
$$

(by induction hypothesis)

$$
\begin{aligned}
&= \left\lceil \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{I}_{\mathcal{B}_{d-1}} \cdot \ldots \cdot \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \uparrow_P (d-1) + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{(w)} \downarrow_P (d-1)) - \mathbf{O}_{\mathcal{B}_d} \right\rceil \\
&= \left\lceil \mathbf{I}_{\mathcal{B}} \cdot (\mathbf{X}^{(w)} \uparrow_P d + \mathbf{X}^{(w)} \updownarrow_P d) + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{w} \downarrow_P (d-1)) - \mathbf{O}_{\mathcal{B}_d} \right\rceil \\
&= \left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \uparrow_P d + \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \updownarrow_P d + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{w} \downarrow_P (d-1)) - \mathbf{O}_{\mathcal{B}_d} \right\rceil
\end{aligned}
$$

An application of Lemma 2.11 with $k = d$ and $m = 0$ yields that $\mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \uparrow_P d$ is a vector multiple of $P^{d+0-d} = 1$, i.e., an integer vector. Hence,

$$
\begin{aligned}
\mathbf{Y}^{(w)} &= \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \uparrow_P d + \left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \updownarrow_P d + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{(w)} \downarrow_P (d-1))) - \mathbf{O}_{\mathcal{B}_d} \right\rceil \\
&= \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \uparrow_P d + \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)} \downarrow_P d) \ ,
\end{aligned}
$$

as needed. ■

Call the matrix $\mathbf{I}_{\mathcal{B}}$ the *steady transfer matrix* of $\mathcal{B}$. Call the vector function $\mathbf{F}_{\mathcal{B}}$ the *transient transfer function* of $\mathcal{B}$. Call $\mathbf{I}_{\mathcal{B}}$ and $\mathbf{F}_{\mathcal{B}}$ the *transfer parameters* of $\mathcal{B}$.

Theorem 3.1 shows that the output vector of a balancing network is the sum of two terms. The first term $\mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{w} \uparrow_P d$, called the *steady output* term, involves the most significant part $\mathbf{X}^{(w)} \uparrow_P d$ of the input vector; this part is obtained by setting the d least significant P-ary digits of each entry of the input vector to zero. The steady output term is a *linear* transformation, defined by the steady transfer matrix $\mathbf{I}_{\mathcal{B}}$, of the most significant part of the input vector. The second term $\left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \updownarrow_P d + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{(w)} \downarrow_P (d-1)) - \mathbf{O}_{\mathcal{B}_d} \right\rceil$, called the *transient output* term, involves the least significant part $\mathbf{X}^{(w)} \downarrow_P d$ of the input vector; this part corresponds to the d least significant P-ary digits of each entry of the input vector. The transient output term is the image, under the transient transfer function $\mathbf{F}_{\mathcal{B}}$ of $\mathcal{B}$, of the least significant part of the input vector; apparently, the least significant part of the input vector undergoes a *non-linear* transformation defined by $\mathbf{F}_{\mathcal{B}}$.

Thus, the steady transfer matrix $\mathbf{I}_{\mathcal{B}}$ is determined by the relative incidences in the network and shapes the steady output term, while the transient transfer function $\mathbf{F}_{\mathcal{B}}$ is determined by both the relative incidences in the network and the relative order of outputs for each balancer, and shapes the transient output term.

4 Transfer Parameters

In this Section, we present several interesting combinatorial and algebraic properties of the transfer parameters of a balancing network. These properties will be useful in showing several combinatorial characterization results that will follow in Section 5, while their proofs rely heavily on Theorem 3.1. The reader may prefer to skip this Section for now, returning to it later when its results are required. Sections 4.1 and 4.2 consider the steady transfer matrix and the transient transfer function, respectively.

4.1 The Steady Transfer Matrix

Since the product of doubly stochastic matrices is doubly stochastic (see, e.g., [4, Corollary 8.40]), Theorem 3.1(1) immediately implies:

Proposition 4.1 *For any balancing network $\mathcal{B}$, the steady transfer matrix $\mathbf{I}_{\mathcal{B}}$ is doubly stochastic.*

Our next combinatorial result establishes an interesting integrality restriction on the steady transfer matrix.

Proposition 4.2 *Assume $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ is a balancing network of depth d over $\mathcal{P}$. Then, $P^d\, \mathbf{I}_{\mathcal{B}}$ is an integer matrix.*

Proof: Fix any indices i and j, $0 \leq i, j \leq w-1$; we show that $P^d\, \mathbf{I}_{\mathcal{B}}[ji]$ is an integer. By Theorem 3.1, the steady transfer matrix $\mathbf{I}_{\mathcal{B}}$ is the product of d incidence matrices; hence, an application of Lemma 2.11 with $k = d$ and $m = 0$ yields that for any integer vector $\mathbf{X}^{(w)}$, the vector $\mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \uparrow_P d$ is a vector multiple of $P^{d+0-d} = 1$, i.e., an integer vector; so, the jth entry $\sum_{l=0}^{w-1} \mathbf{I}_{\mathcal{B}}[jl]\, x_l \uparrow_P d$ is an integer. In particular, this holds for an input vector $\mathbf{X}^{(w)}$ in which $x_i \uparrow_P d = P^d$ and $x_l \uparrow_P d = 0$ for $l \neq i$. Hence, $P^d\, \mathbf{I}_{\mathcal{B}}[ji]$ is an integer, as needed. ■

We continue to show a general lower bound on each entry of the steady transfer matrix.

Proposition 4.3 *Assume $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ is a balancing network over $\mathcal{P}$. Then, for any indices i and j, $0 \leq i, j \leq w-1$,*

$$\mathbf{I}_{\mathcal{B}}[ji] \;\geq\; \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}}[ji]}}\,.$$

Proof: By induction on the depth d of the network $\mathcal{B}$. For the base case where $d = 1$, we proceed by case analysis. There are three cases for input wire i and output wire j: *(i)* they are connected via a p-balancer, for some $p \in \mathcal{P}$, so that $\mathbf{I}_{\mathcal{B}}[ji] = 1/p \geq 1/p_0$ and $\mathbf{D}_{\mathcal{B}}[ji] = 1$; *(ii)* they coincide so that $\mathbf{I}_{\mathcal{B}}[ji] = 1$ and $\mathbf{D}_{\mathcal{B}}[ji] = 0$; *(iii)* neither of the above holds, so that $\mathbf{I}_{\mathcal{B}}[ji] = 0$ and $\mathbf{D}_{\mathcal{B}}[ji] = \infty$. Clearly, in all cases, $\mathbf{I}_{\mathcal{B}}[ji] \geq 1/p_0^{\mathbf{D}_{\mathcal{B}}[ji]}$, as needed.

Assume inductively that for some $d \geq 2$, the claim holds for all balancing networks of depth $d-1$ or less over $\mathcal{P}$. Take a balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ of depth d over $\mathcal{P}$. Let $\mathcal{B}' : \mathbf{X}^{(w)} \rightarrow \mathbf{Z}^{(w)}$ be the balancing network obtained from $\mathcal{B}$ by removing its rightmost layer $\mathcal{B}_d$. It follows, by Theorem 3.1(1), that $\mathbf{I}_{\mathcal{B}} = \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{I}_{\mathcal{B}'}$, so that

$$\mathbf{I}_{\mathcal{B}}[ji] \;=\; \sum_{l=0}^{w-1} \mathbf{I}_{\mathcal{B}_d}[jl] \cdot \mathbf{I}_{\mathcal{B}'}[li]$$

$$
\begin{aligned}
&\geq \sum_{l=0}^{w-1} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}_d}[jl]}} \cdot \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}'}[li]}} && \text{(by induction hypothesis)} \\
&= \sum_{l=0}^{w-1} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}_d}[jl]+\mathbf{D}_{\mathcal{B}'}[li]}} \\
&\geq \max_{0\leq l\leq w-1} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}_d}[jl]+\mathbf{D}_{\mathcal{B}'}[li]}} \\
&= \frac{1}{p_0^{\min_{0\leq l\leq w-1}(\mathbf{D}_{\mathcal{B}_d}[jl]+\mathbf{D}_{\mathcal{B}'}[li])}} \; = \; \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}}[ji]}},
\end{aligned}
$$

as needed. ■

In particular, Proposition 4.3 implies a general lower bound on the smallest distance from an input wire to an output wire in any balancing network.

Corollary 4.4 *For any balancing network $\mathcal{B}$ over $\mathcal{P}$,*

$$
\|\mathbf{D}_{\mathcal{B}}\|_{\min} \;\geq\; \log_{p_0} \frac{1}{\|\mathbf{I}_{\mathcal{B}}\|_{\max}}.
$$

We continue with a generalization of Proposition 4.3.

Proposition 4.5 *Assume $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ is a balancing network over $\mathcal{P}$. Then,*

(1) *for any indices $i \in [wg]$ and $j \in [w]$,*

$$
\sum_{r\in\pi_j} \mathbf{I}_{\mathcal{B}}[ri] \;\geq\; \frac{g}{p_0^{\max_{r\in\pi_j} \mathbf{D}_{\mathcal{B}}[ri]}} \text{ and}
$$

(2) *for any indices $i \in [w]$ and $j \in [wg]$,*

$$
\sum_{r\in\pi_i} \mathbf{I}_{\mathcal{B}}[jr] \;\geq\; \frac{g}{p_0^{\max_{r\in\pi_i} \mathbf{D}_{\mathcal{B}}[jr]}}.
$$

Proof: By Proposition 4.3, for any indices $i \in [wg]$ and $j \in [w]$,

$$
\sum_{r\in\pi_j} \mathbf{I}_{\mathcal{B}}[ri] \;\geq\; \sum_{r\in\pi_j} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}}[ri]}} \;\geq\; \sum_{r\in\pi_j} \min_{r\in\pi_j} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}}[ri]}} \;=\; \sum_{r\in\pi_j} \frac{1}{p_0^{\max_{r\in\pi_j} \mathbf{D}_{\mathcal{B}}[ri]}} \;=\; \frac{g}{p_0^{\max_{r\in\pi_j} \mathbf{D}_{\mathcal{B}}[ri]}},
$$

and for any indices $i \in [w]$ and $j \in [wg]$,

$$
\sum_{r\in\pi_i} \mathbf{I}_{\mathcal{B}}[jr] \;\geq\; \sum_{r\in\pi_i} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}}[jr]}} \;\geq\; \sum_{r\in\pi_i} \min_{r\in\pi_i} \frac{1}{p_0^{\mathbf{D}_{\mathcal{B}}[jr]}} \;=\; \sum_{r\in\pi_i} \frac{1}{p_0^{\max_{r\in\pi_i} \mathbf{D}_{\mathcal{B}}[jr]}} \;=\; \frac{g}{p_0^{\max_{r\in\pi_i} \mathbf{D}_{\mathcal{B}}[jr]}},
$$

as needed. ■

In particular, Proposition 4.5 implies an interesting min-max result: a general lower bound on the smallest, over all input (resp., output) wires, among the largest, over all blocks of outputs (resp., inputs), distances from an input wire (resp., an input wire in a given block of inputs) to an output wire in a given block of outputs (resp., an output wire).

Corollary 4.6 *For any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ over $\mathcal{P}$,*

(1)

$$\min_{i \in [wg]} \max_{r \in \pi_j} \mathbf{D}_{\mathcal{B}}[ri] \;\geq\; \log_{p_0} \frac{g}{\max_{i \in [wg]} \sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri]} \; \textit{and}$$

(2)

$$\min_{j \in [wg]} \max_{r \in \pi_i} \mathbf{D}_{\mathcal{B}}[jr] \;\geq\; \log_{p_0} \frac{g}{\max_{j \in [wg]} \sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr]} \,.$$

4.2 The Transient Transfer Function

We start by showing affinity.

Proposition 4.7 *For any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ over $\mathcal{P}$, $\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(w)}) \;=\; \mathbf{0}^{(w)}$.*

Proof: By induction on the depth d of the network $\mathcal{B}$. For the base case where $d = 1$, Theorem 3.1(2) implies that

$$\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(w)}) \;=\; \mathbf{F}_{\mathcal{B}_1}(\mathbf{0}^{(w)}) \;=\; \left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{0}^{(w)} - \mathbf{O}_{\mathcal{B}_1} \right\rceil \;=\; \lceil -\mathbf{O}_{\mathcal{B}_1} \rceil \;=\; \mathbf{0}^{(w)} \,,$$

by Observation 2.1, as needed.

Assume inductively that for some $d \geq 2$, for any balancing network $\mathcal{B}'$ of depth $d-1$ over $\mathcal{P}$, $\mathbf{F}_{\mathcal{B}'}(\mathbf{0}^{(w)}) = \mathbf{0}^{(w)}$. Take any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ of depth d over $\mathcal{P}$. Since $\mathbf{0}^{(w)} = \mathbf{0}^{(w)} \downarrow_P d$, Theorem 3.1(2) implies that

$$\begin{aligned}
\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(w)}) &= \mathbf{F}_{\mathcal{B}_d}(\mathbf{0}^{(w)} \downarrow_P d) \\
&= \left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{0}^{(w)} \updownarrow_P d + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{0}^{(w)} \downarrow_P (d-1)) - \mathbf{O}_{\mathcal{B}_d} \right\rceil \\
&= \left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{0}^{(w)} + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{0}^{(w)}) - \mathbf{O}_{\mathcal{B}_d} \right\rceil \\
&= \left\lceil \mathbf{0}^{(w)} + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{0}^{(w)} - \mathbf{O}_{\mathcal{B}_d} \right\rceil & \text{(by induction hypothesis)} \\
&= \lceil -\mathbf{O}_{\mathcal{B}_d} \rceil \;=\; \mathbf{0}^{(w)} & \text{(by Observation 2.1),}
\end{aligned}$$

as needed. ■

We next establish an upper bound of $P^d - 1$ on $\|\mathbf{F}_{\mathcal{B}}\|_{\max}$.

Proposition 4.8 *For any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ of depth d over $\mathcal{P}$,*

$$\|\mathbf{F}_{\mathcal{B}}\|_{\max} \;\leq\; P^d - 1\,.$$

Proof: By induction on the depth d of the network $\mathcal{B}$. For the base case where $d = 1$,

$$\begin{aligned}
\left\|\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)} \downarrow_P 1)\right\|_{\max} &= \left\|\mathbf{F}_{\mathcal{B}_1}(\mathbf{X}^{(w)} \downarrow_P 1)\right\|_{\max} \\
&= \left\|\left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot \mathbf{X}^{(w)} \updownarrow_P 1 - \mathbf{O}_{\mathcal{B}_1} \right\rceil\right\|_{\max} && \text{(by Theorem 3.1(2))} \\
&\leq \left\|\left\lceil \mathbf{I}_{\mathcal{B}_1} \cdot (P-1)\mathbf{1}^{(w)} - \mathbf{O}_{\mathcal{B}_1} \right\rceil\right\|_{\max} && \text{(since } \mathbf{X}^{(w)} \updownarrow_P 1 \in [P]^w) \\
&= \left\|\left\lceil (P-1)\mathbf{1}^{(w)} - \mathbf{O}_{\mathcal{B}_1} \right\rceil\right\|_{\max} && \text{(by Proposition 2.10)} \\
&= \left\|(P-1)\mathbf{1}^{(w)} + \lceil -\mathbf{O}_{\mathcal{B}_1} \rceil\right\|_{\max} \\
&= \left\|(P-1)\mathbf{1}^{(w)} + \mathbf{0}^{(w)}\right\|_{\max} && \text{(by Observation 2.1)} \\
&= \left\|(P-1)\mathbf{1}^{(w)}\right\|_{\max} \;=\; P - 1\,,
\end{aligned}$$

so that

$$\|\mathbf{F}_{\mathcal{B}}\|_{\max} \;=\; \max_{\mathbf{X}^{(w)} \in [P]^w} \|\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)})\|_{\max} \;\leq\; P - 1 \;=\; P^1 - 1\,,$$

as needed.

Assume inductively that for some $d \geq 2$, for any balancing network $\mathcal{B}'$ of depth $d-1$ over $\mathcal{P}$, $\|\mathbf{F}_{\mathcal{B}'}\|_{\infty} \leq P^{d-1} - 1$. Take any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ of depth d over $\mathcal{P}$. We have:

$$\begin{aligned}
\left\|\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)} \downarrow_P d)\right\|_{\max} &= \left\|\mathbf{F}_{\mathcal{B}_d}(\mathbf{X}^{(w)} \downarrow_P d)\right\|_{\max} \\
&= \left\|\left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \updownarrow_P d + \mathbf{I}_{\mathcal{B}_d} \cdot \mathbf{F}_{\mathcal{B}_{d-1}}(\mathbf{X}^{(w)} \downarrow_P (d-1)) - \mathbf{O}_{\mathcal{B}_d} \right\rceil\right\|_{\max} \\
&\quad \text{(by Theorem 3.1(2))} \\
&\leq \left\|\left\lceil \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(w)} \updownarrow_P d + \mathbf{I}_{\mathcal{B}_d} \cdot (P^{d-1} - 1)\mathbf{1}^{(w)} - \mathbf{O}_{\mathcal{B}_d} \right\rceil\right\|_{\max} \\
&\quad \text{(by induction hypothesis)} \\
&\leq \left\|\left\lceil \mathbf{I}_{\mathcal{B}} \cdot (P-1)P^{d-1}\mathbf{1}^{(w)} + \mathbf{I}_{\mathcal{B}_d} \cdot (P^{d-1} - 1)\mathbf{1}^{(w)} - \mathbf{O}_{\mathcal{B}_d} \right\rceil\right\|_{\max} \\
&\quad \text{(since } \mathbf{X}^{(w)} \updownarrow_P d \in P^{d-1}[P]^w) \\
&= \left\|\left\lceil (P-1)P^{d-1}\mathbf{1}^{(w)} + (P^{d-1} - 1)\mathbf{1}^{(w)} - \mathbf{O}_{\mathcal{B}_d} \right\rceil\right\|_{\max} \\
&\quad \text{(by Proposition 2.10)} \\
&= \left\|\left\lceil ((P-1)P^{d-1} + P^{d-1} - 1)\mathbf{1}^{(w)} - \mathbf{O}_{\mathcal{B}_d} \right\rceil\right\|_{\max}
\end{aligned}$$

$$
\begin{aligned}
&= \left\| (P^d - 1)\mathbf{1}^{(w)} + \lceil -\mathbf{O}_{\mathcal{B}_d} \rceil \right\|_{\max} \\
&= P^d - 1 + \| \lceil -\mathbf{O}_{\mathcal{B}_d} \rceil \|_{\max} \\
&= P^d - 1 + \left\| \mathbf{0}^{(w)} \right\|_{\max} \qquad \text{(by Observation 2.1)} \\
&= P^d - 1\,,
\end{aligned}
$$

so that

$$
\|\mathbf{F}_{\mathcal{B}}\|_{\max} \;=\; \max_{\mathbf{X}^{(w)} \in [P^d]^w} \|\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)})\|_{\max} \;\leq\; P^d - 1\,,
$$

as needed. ■

We finally show that, under an appropriate assumption on the steady transfer matrix, a relation holds between 1-norms of block output vectors and corresponding entries of the block transient transfer function.

Proposition 4.9 *Assume $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$. Then,*

$$
\|\mathbf{Y}_k^{(g)}\|_1 - \|\mathbf{Y}_l^{(g)}\|_1 \;=\; \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)} \downarrow_P d)[k] - \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)} \downarrow_P d)[l]\,,
$$

for all k and l, $0 \leq k, l \leq w-1$.

Proof: Clearly, for any k and l, $0 \leq k, l \leq w-1$,

$$
\begin{aligned}
\|\mathbf{Y}_k^{(g)}\|_1 - \|\mathbf{Y}_l^{(g)}\|_1 &= \sum_{r \in \pi_k} y_r - \sum_{r \in \pi_l} y_r \\
&= \sum_{r \in \pi_k} \Big(\sum_{m=0}^{wg-1} \mathbf{I}_{\mathcal{B}}[rm]\, x_m \uparrow_P d + \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(wg)} \downarrow_P d)[r] \Big) \\
&\quad - \sum_{r \in \pi_l} \Big(\sum_{m=0}^{wg-1} \mathbf{I}_{\mathcal{B}}[rm]\, x_m \uparrow_P d + \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(wg)} \downarrow_P d)[r] \Big) \\
&\quad \text{(by Theorem 3.1)} \\
&= \sum_{m=0}^{wg-1} \sum_{r \in \pi_k} \mathbf{I}_{\mathcal{B}}[rm]\, x_m \uparrow_P d - \sum_{r \in \pi_k} \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(wg)} \downarrow_P d)[r] \\
&\quad - \sum_{m=0}^{wg-1} \sum_{r \in \pi_l} \mathbf{I}_{\mathcal{B}}[rm]\, x_m \uparrow_P d + \sum_{r \in \pi_l} \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(wg)} \downarrow_P d)[r] \\
&\quad \text{(changing the order of summation)} \\
&= \sum_{m=0}^{wg-1} \frac{1}{w}\, x_m \uparrow_P d + \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)} \downarrow_P d)[k] \\
&\quad - \sum_{m=0}^{wg-1} \frac{1}{w}\, x_m \uparrow_P d - \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)} \downarrow_P d)[l] \\
&= \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)} \downarrow_P d)[k] - \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)} \downarrow_P d)[l]\,,
\end{aligned}
$$

as needed. ■

Proposition 4.9 specializes in the case $g = 1$ to yield:

Corollary 4.10 *Assume $\mathbf{I}_{\mathcal{B}} = (1/w)\ \mathbf{1}^{(w \times w)}$. Then, for all indices j and k, $0 \leq j, k \leq w - 1$.*

$$y_j - y_k \;=\; \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)} \downarrow_P d)[j] - \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(w)} \downarrow_P d)[k] \,.$$

5 Combinatorial Characterizations

Necessary and sufficient conditions are presented for a balancing network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ of depth d over $\mathcal{P}$ to belong to each of the classes of balancing networks introduced in Section 2.2.2. These conditions are expressed in terms of the transfer parameters of $\mathcal{B}$.

5.1 Block-Output Networks

Recall that Definitions 2.3 and 2.4 for $w \cdot g$ counting networks and $w \cdot g$ K-smoothing networks, respectively, require that certain properties hold on the differences between 1-norms of block vectors of the output vector corresponding to any input vector. On the other hand, Proposition 4.9 implies that, under an appropriate combinatorial condition on $\mathbf{I}_{\mathcal{B}}$, these properties are equivalent to corresponding properties of the block transient transfer function $\mathbf{F}_{\mathcal{B}/\Pi}$ to hold for all (input) vectors in the domain of $\mathbf{F}_{\mathcal{B}/\Pi}$.

To formalize this equivalence, say that a vector function $\mathbf{F} : \mathcal{N}^{wg} \rightarrow \mathcal{N}^{w}$ is *step on* $\mathbf{D} \subseteq \mathcal{N}^{wg}$ (resp., *K-smooth on* $\mathbf{D} \subseteq \mathcal{N}^{wg}$) if the vector $\mathbf{F}(\mathbf{X}^{(wg)})$ has the step (resp., K-smoothing) property for every vector $\mathbf{X}^{(wg)} \in \mathbf{D}$. So, Proposition 4.9 immediately implies necessary and sufficient conditions for $w \cdot g$ counting networks and $w \cdot g$ K-smoothing networks, holding under a certain combinatorial assumption on the steady transfer matrix.

Proposition 5.1 *For a network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$, assume that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$. Then, $\mathcal{B}$ is a $w \cdot g$ counting network (resp., $w \cdot g$ K-smoothing network) if and only if $\mathbf{F}_{\mathcal{B}/\Pi}$ is step (resp., K-smooth) on $[P^d]^{wg}$.*

Our first major combinatorial characterization result establishes that the condition assumed in Proposition 5.1 is, in fact, necessary for $w \cdot g$ counting networks.

Theorem 5.2 *The network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ is a $w \cdot g$ counting network if and only if*

(1) $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] \;=\; 1/w$ *for all $i \in [wg]$ and $j \in [w]$, and*

(2) $\mathbf{F}_{\mathcal{B}/\Pi}$ *is step on $[P^d]^{wg}$.*

Proof: Assume first that $\mathcal{B}$ is a $w \cdot g$ counting network. By Proposition 5.1, it suffices to show that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$.

Fix any indices $i \in [wg]$ and $j \in [w]$, and set $\mathbf{X}^{(wg)} \downarrow_P (d+1) = \mathbf{0}^{(wg)}$, $x_i \uparrow_P (d+1) \neq 0$ and $x_l \uparrow_P (d+1) = 0$ for $l \neq i$. It follows that $\mathbf{X}^{(wg)} \uparrow_P d = \mathbf{X}^{(wg)} \uparrow_P (d+1)$, $\mathbf{X}^{(wg)} \downarrow_P d = \mathbf{0}^{(wg)}$, and $\|\mathbf{X}^{(wg)}\|_1 = x_i$. Since, by Proposition 4.7, $\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(wg)}) = \mathbf{0}^{(wg)}$, Theorem 3.1 implies that

$$\mathbf{Y}^{(wg)} \;=\; \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} \uparrow_P (d+1)\,.$$

An application of Lemma 2.11 with $k = d$ and $m = 1$ yields that $\mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} \uparrow_P (d+1)$ is a vector multiple of $P^{d+1-d} = P$. Hence, for each j, $0 \leq j \leq w-1$, $\|\mathbf{Y}_j^{(g)}\|_1$ is a multiple of P. Since $\mathcal{B}$ is a $w \cdot g$ counting network, $0 \leq \|\mathbf{Y}_j^{(g)}\|_1 - \|\mathbf{Y}_k^{(g)}\|_1 \leq 1$ for all j and k, $0 \leq j < k \leq w-1$. It follows that $\|\mathbf{Y}_j^{(g)}\|_1 = \|\mathbf{Y}_k^{(g)}\|_1$ for all j and k, $0 \leq j, k \leq w-1$.

By Proposition 2.8, $\|\mathbf{Y}^{(wg)}\|_1 = \|\mathbf{X}^{(wg)}\|_1 = x_i$, so that $\sum_{j=0}^{w-1} \|\mathbf{Y}_j^{(g)}\|_1 = x_i$. It follows that for each $j \in [w]$,

$$\|\mathbf{Y}_j^{(g)}\|_1 \;=\; \frac{1}{w}\, x_i\,.$$

Notice that

$$\|\mathbf{Y}_j^{(g)}\|_1 \;=\; \sum_{r \in \pi_j} \Big(\sum_{l=0}^{wg-1} \mathbf{I}_{\mathcal{B}}[rl]\, x_l \uparrow_P (d+1) \Big) \;=\; \sum_{r \in \pi_j} (\mathbf{I}_{\mathcal{B}}[ri]\, x_i) \;=\; x_i \sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri]\,.$$

It follows that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$, as needed. Assume now that conditions (1) and (2) hold. Proposition 5.1 immediately implies that $\mathcal{B}$ is a $w \cdot g$ counting network. ■

Theorem 5.2 specializes in the case $g = 1$ to yield:

Corollary 5.3 *The network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ is a counting network if and only if*

(1) $\mathbf{I}_{\mathcal{B}} \;=\; (1/w)\, \mathbf{1}^{(w \times w)}$, *and*

(2) $\mathbf{F}_{\mathcal{B}}$ *is step on* $[P^d]^w$.

Corollary 5.3 indicates that, for a counting network, the steady output term is a constant vector each of whose entries is a scalar multiple of the 1-norm of the most significant part of the input vector, the scaling factor being the reciprocal of the network's width; loosely speaking, this establishes that the contribution due to the most significant parts of inputs is fairly shared among the w output wires. Moreover, the step property is inherited down to the transient output term. We proceed to show a corresponding combinatorial characterization result for $w \cdot g$ K-smoothing networks.

Theorem 5.4 *The network $\mathcal{B} : \mathbf{X}^{(wg)} \to \mathbf{Y}^{(wg)}$ is a $w \cdot g$ K-smoothing network if and only if*

(1) $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ *for all* $i \in [wg]$ *and* $j \in [w]$, *and*

(2) $\mathbf{F}_{\mathcal{B}/\Pi}$ *is* K*-smooth on* $[P^d]^{wg}$.

Proof: Assume first that $\mathcal{B}$ is a $w \cdot g$ K-smoothing network. By Proposition 5.1, it suffices to show that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$.

Fix any indices $i \in [wg]$ and $j \in [w]$, and set $\mathbf{X}^{(wg)} \downarrow_P (d + \lfloor \log_P K + 1 \rfloor) = \mathbf{0}^{(wg)}$, $x_i \uparrow_P (d+1) \neq 0$ and $x_l \uparrow_P (d+1) = 0$ for $l \neq i$. It follows that $\mathbf{X}^{(wg)} \downarrow_P d = \mathbf{0}^{(wg)}$ and $\|\mathbf{X}^{(wg)}\|_1 = x_i$. Since $P \geq 2$ and $K \geq 1$, $\log_P K \geq 0$, which implies that $\lfloor \log_P K + 1 \rfloor \geq 1$; hence, $\mathbf{X}^{(wg)} \uparrow_P d = \mathbf{X}^{(wg)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor)$. Since, by Proposition 4.7, $\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(wg)}) = \mathbf{0}^{(wg)}$, Theorem 3.1 implies that

$$\mathbf{Y}^{(wg)} = \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor)\,.$$

An application of Lemma 2.11 with $k = d$ and $m = \lfloor \log_P K + 1 \rfloor$ yields that $\mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor)$ is a vector multiple of $P^{d + \lfloor \log_P K + 1 \rfloor - d} = P^{\lfloor \log_P K + 1 \rfloor}$. It follows that for each $j \in [w]$, $\|\mathbf{Y}_j^{(g)}\|_1$ is a multiple of $P^{\lfloor \log_P K + 1 \rfloor} > P^{\log_P K} = K$. Since $\mathcal{B}$ is a $w \cdot g$ K-smoothing network, $|\|\mathbf{Y}_j^{(g)}\|_1 - \|\mathbf{Y}_k^{(g)}\|_1| \leq K$ for all j and k, $0 \leq j, k \leq w-1$. It follows that $\|\mathbf{Y}_j^{(g)}\|_1 = \|\mathbf{Y}_k^{(g)}\|_1$ for all j and k, $0 \leq j, k \leq w-1$.

By Proposition 2.8, $\|\mathbf{Y}^{(wg)}\|_1 = \|\mathbf{X}^{(wg)}\|_1 = x_i$, so that $\sum_{j=0}^{w-1} \|\mathbf{Y}_j^{(g)}\|_1 = x_i$. It follows that for each $j \in [w]$,

$$\|\mathbf{Y}_j^{(g)}\|_1 = \frac{1}{w}\, x_i\,.$$

Notice that

$$\|\mathbf{Y}_j^{(g)}\|_1 = \sum_{r \in \pi_j} \Big(\sum_{l=0}^{wg-1} \mathbf{I}_{\mathcal{B}}[rl]\, x_l \uparrow_P (d + \lfloor \log_P K + 1 \rfloor) \Big) = \sum_{r \in \pi_j} (\mathbf{I}_{\mathcal{B}}[ri]\, x_i) = x_i \sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri]\,.$$

It follows that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$, as needed. Assume now that conditions (1) and (2) hold. Proposition 5.1 immediately implies that $\mathcal{B}$ is a $w \cdot g$ K-smoothing network. ∎

Theorem 5.4 specializes in the case $g = 1$ to yield:

Corollary 5.5 *The network $\mathcal{B} : \mathbf{X}^{(w)} \to \mathbf{Y}^{(w)}$ is a K-smoothing network if and only if*

(1) $\mathbf{I}_{\mathcal{B}} = (1/w)\, \mathbf{1}^{(w \times w)}$, *and*

(2) $\mathbf{F}_{\mathcal{B}}$ *is* K*-smooth on* $[P^d]^w$.

As in counting networks, Corollary 5.5 indicates that, for a K-smoothing network, the steady output term is a constant vector each of whose entries is a scalar multiple of the 1-norm of the most significant part of the input vector, the scaling factor being the reciprocal of the network width. Loosely speaking, this establishes that the contribution due to the most significant parts of inputs is fairly shared among the w output wires. Moreover, the K-smoothing property is inherited down to the transient output term.

We next establish that the latter in the pair of necessary and sufficient conditions for a $w \cdot g$ K-smoothing network established in Theorem 5.4 can be relaxed if K is sufficiently large.

Proposition 5.6 *Assume $K \geq g(P^d - 1)$. Then, the network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ is a $w \cdot g$ K-smoothing network if and only if*

$$\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] \;=\; \frac{1}{w}$$

for all $i \in [wg]$ and $j \in [w]$.

Proof: Assume first that $\mathcal{B}$ is a $w \cdot g$ K-smoothing network. By Theorem 5.4, $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$. Assume now that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$. Fix any $\mathbf{X}^{(wg)} \in [P^d]^{wg}$. Then, for any indices j and k, $0 \leq j, k \leq w-1$,

$$\begin{aligned}
|\mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)})[j] - \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)})[k]| \;&\leq\; |\mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)})[j]| \\
&\leq\; \|\mathbf{F}_{\mathcal{B}/\Pi}\|_{\max} && \text{(by definition of maximum norm)} \\
&\leq\; g\,\|\mathbf{F}_{\mathcal{B}}\|_{\max} && \text{(by Lemma 2.5)} \\
&\leq\; g(P^d - 1) && \text{(by Proposition 4.8)} \\
&\leq\; K\,.
\end{aligned}$$

So, $\mathbf{F}_{\mathcal{B}/\Pi}$ is K-smooth on $[P^d]^{wg}$. It follows, by Theorem 5.4, that $\mathcal{B}$ is a $w \cdot g$ K-smoothing network, as needed. ∎

Theorem 5.6 implies that for sufficiently large K, the $w \cdot g$ K-smoothing property is intrinsic to the incidence pattern of a balancing network. Theorem 5.6 specializes in the case $g = 1$ to yield:

Corollary 5.7 *Assume $K \geq P^d - 1$. Then, the network $\mathcal{B}$ is a K-smoothing network if and only if $\mathbf{I}_{\mathcal{B}} = (1/w)\,\mathbf{1}^{(w \times w)}$.*

Our next major result provides a combinatorial characterization of $w \cdot g$ smoothing networks.

Theorem 5.8 *The network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ is a $w \cdot g$ smoothing network if and only if*

$$\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] \;=\; \frac{1}{w}\,,$$

for all $i \in [wg]$ and $j \in [w]$.

Proof: Assume first that $\mathcal{B}$ is a $w \cdot g$ smoothing network, i.e., a $w \cdot g$ K-smoothing network for any integer $K \geq 1$. By Theorem 5.4, $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$.

Assume now that $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$. Fix any integer $K \geq g(P^d - 1)$. Theorem 5.6 implies that $\mathcal{B}$ is a $w \cdot g$ K-smoothing network, hence a $w \cdot g$ smoothing network, as needed. ∎

Theorem 5.8 specializes in the case $g = 1$ to yield:

Corollary 5.9 *The network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ is a smoothing network if and only if $\mathbf{I}_{\mathcal{B}} = (1/w)\, \mathbf{1}^{(w \times w)}$.*

We finally establish a combinatorial property of the $w \cdot g$ smoothing constant.

Proposition 5.10 *Assume $\mathcal{B}$ is a $w \cdot g$ smoothing network. Then, $\mathcal{B}$ is a $w \cdot g$ $g(P^d - 1)$-smoothing network.*

Proof: By Theorem 5.8, $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$ for all $i \in [wg]$ and $j \in [w]$. Fix any $\mathbf{X}^{(wg)} \in [P^d]^{wg}$. Then, for any indices j and k, $0 \leq j, k \leq w - 1$,

$$\begin{aligned}
|\mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)})[j] - \mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)})[k]| &\leq |\mathbf{F}_{\mathcal{B}/\Pi}(\mathbf{X}^{(wg)})[j]| \\
&\leq \|\mathbf{F}_{\mathcal{B}/\Pi}\|_{\max} && \text{(by definition of maximum norm)} \\
&\leq g\, \|\mathbf{F}_{\mathcal{B}}\|_{\max} && \text{(by Lemma 2.5)} \\
&\leq g(P^d - 1) && \text{(by Proposition 4.8)}.
\end{aligned}$$

So, $\mathbf{F}_{\mathcal{B}/\Pi}$ is $g(P^d - 1)$-smooth on $[P^d]^{wg}$. It follows, by Theorem 5.4, that $\mathcal{B}$ is a $w \cdot g$ $g(P^d - 1)$-smoothing network, as needed. ∎

Proposition 5.10 represents an interesting Zero-One Law for the $w \cdot g$ K-smoothing property. It establishes that $g(P^d - 1)$ is a "threshold" value for the $w \cdot g$ smoothing constant of a balancing network: the $w \cdot g$ smoothing constant is either infinite (i.e., the network is not K-smoothing for any integer K), or at most $g(P^d - 1)$. For $g = 1$, Proposition 5.10 specializes to yield:

Corollary 5.11 *Assume $\mathcal{B}$ is a K-smoothing network. Then, $\mathcal{B}$ is a $(P^d - 1)$-smoothing network.*

5.2 Block-Input Networks

Recall that all definitions for block-input networks require certain properties to hold, over appropriate domains of input vectors, on the outputs' differences. On the other hand, Corollary 4.10 implies that if the steady transfer matrix is a constant matrix, then these properties are equivalent to corresponding properties for the block transient transfer function to hold over the "low-order projections" of those domains.

To formalize this equivalence, denote by *blockstep*($\mathcal{N}^{wg}$) (resp., *block-K_1-smooth*($\mathcal{N}^{wg}$)) the set of all integer vectors $\mathbf{X}^{(wg)}$ such that each block vector $\mathbf{X}_j^{(g)}$, $0 \leq j \leq w-1$, has the step property (resp., K_1-smoothing property). Call any vector $\mathbf{X}^{(wg)} \in$ *blockstep*($\mathcal{N}^{wg}$) (resp., $\mathbf{X}^{(wg)} \in$ *block-K_1-smooth*($\mathcal{N}^{wg}$)) a *block-step* vector (resp., a *block-K_1-smooth* vector).

For each integer $k \geq 1$, denote by (*blockstep*($\mathcal{N}^{wg}$)) $\downarrow_P k$ (resp., (*block-K_1-smooth*($\mathcal{N}^{wg}$)) $\downarrow_P k$) the set of all integer vectors $\mathbf{X}^{(wg)}$ such that $\mathbf{X}^{(wg)} = \mathbf{Y}^{(wg)} \downarrow_P k$ for some block-step (resp., block-K_1-smooth) vector $\mathbf{Y}^{(wg)}$; that is, (*blockstep*($\mathcal{N}^{wg}$)) $\downarrow_P k$ is the set of "restrictions" of block-step (resp., block-K_1-smooth) vectors with wg entries to their k least significant P-ary digits. Notice that (*blockstep*($\mathcal{N}^{wg}$)) $\downarrow_P k \subseteq [P^k]^{wg}$ (resp., (*block-K_1-smooth*($\mathcal{N}^{wg}$)) $\downarrow_P k \subseteq [P^k]^{wg}$).

So, by definitions of block-input networks, Corollary 4.10 immediately implies necessary and sufficient conditions for block-input networks, holding under the assumption that the steady transfer matrix is a constant matrix.

Proposition 5.12 *For a network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ of depth d over $\mathcal{P}$, assume that $\mathbf{I}_{\mathcal{B}} = (1/wg)\mathbf{1}^{(wg \times wg)}$. Then, $\mathcal{B}$ is a $w \cdot g$ step (resp., $w \cdot g$ K_1-smooth) counting network if and only if $\mathbf{F}_{\mathcal{B}}$ is step on (blockstep($\mathcal{N}^{wg}$)) $\downarrow_P d$ (resp., (block-K_1-smooth($\mathcal{N}^{wg}$)) $\downarrow_P d$).*

Proposition 5.13 *For a network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ of depth d over $\mathcal{P}$, assume that $\mathbf{I}_{\mathcal{B}} = (1/wg)\mathbf{1}^{(wg \times wg)}$. Then, $\mathcal{B}$ is a $w \cdot g$ step (resp., $w \cdot g$ K_1-smooth) K-smoothing network if and only if $\mathbf{F}_{\mathcal{B}}$ is K-smooth on (blockstep($\mathcal{N}^{wg}$)) $\downarrow_P d$ (resp., (block-K_1-smooth($\mathcal{N}^{wg}$)) $\downarrow_P d$).*

Clearly, for any $d \geq 1$, *blockstep*($\mathcal{N}^{wg}$) $\subset$ (*blockstep*($\mathcal{N}^{wg}$)) $\downarrow_P d$. Hence, any $w \cdot g$ step counting network that satisfies the combinatorial condition on the steady transfer matrix assumed in Proposition 5.12 satisfies a property actually stronger than the $w \cdot g$ step counting property: any input vector in ((*blockstep*($\mathcal{N}^{wg}$)) $\downarrow_P d) \setminus$ *blockstep*($\mathcal{N}^{wg}$), though not block-step, will result in a step output vector when filtered through such a network.[6] Similar conclusions hold for the rest of the classes of block-input networks.

Unlike the case of block-output networks (Proposition 5.1), we have not been able to show that the assumption we made for each of Propositions 5.12 and 5.13 is also necessary. Instead, we have established the necessity of a combinatorial property of $\mathbf{I}_{\mathcal{B}}$ that is strictly weaker than that assumption. Our next combinatorial result formally shows this necessity for the case of $w \cdot g$ step K-smoothing networks.

Theorem 5.14 *Assume $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ is a $w \cdot g$ step K-smoothing network. Then,*

$$\sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr] \ = \ \frac{1}{w} \, ,$$

[6] Busch and Mavronicolas show that the *bitonic merging* network [7] is an example of such a network.

for all $i \in [w]$ and $j \in [wg]$.

Proof: Assume $depth(\mathcal{B}) = d$. Fix any indices $i \in [w]$ and $j \in [wg]$, and define a block-step vector $\mathbf{X}^{(wg)}$ such that $\mathbf{X}^{(wg)} \downarrow_P (d + \lfloor \log_P K + 1 \rfloor) = \mathbf{0}^{(wg)}$, $\mathbf{X}_i^{(g)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor) = x\, \mathbf{1}^{(g)}$, where $x \neq 0$, and $\mathbf{X}_l^{(g)} \uparrow_P (d+1) = \mathbf{0}^{(g)}$ for $l \neq i$. It follows that $\mathbf{X}^{(wg)} \downarrow_P d = \mathbf{0}^{(wg)}$ and $\|\mathbf{X}^{(wg)}\|_1 = g\, x$. Since $P \geq 2$ and $K \geq 1$, $\log_P K \geq 0$, which implies that $\lfloor \log_P K + 1 \rfloor \geq 1$; hence, $\mathbf{X}^{(wg)} \uparrow_P d = \mathbf{X}^{(wg)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor) = \mathbf{X}^{(wg)}$. Since, by Proposition 4.7, $\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(wg)}) = \mathbf{0}^{(wg)}$, Theorem 3.1 implies that

$$\mathbf{Y}^{(wg)} \;=\; \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor) \;=\; \mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} .$$

An application of Lemma 2.11 with $k = d$ and $m = \lfloor \log_P K + 1 \rfloor$ yields that $\mathbf{I}_{\mathcal{B}} \cdot \mathbf{X}^{(wg)} \uparrow_P (d + \lfloor \log_P K + 1 \rfloor)$ is a vector multiple of $P^{d + \lfloor \log_P K + 1 \rfloor - d} = P^{\lfloor \log_P K + 1 \rfloor}$. It follows that for each $j \in [wg]$, y_j is a multiple of $P^{\lfloor \log_P K + 1 \rfloor} > P^{\log_P K} = K$. Since $\mathcal{B}$ is a $w \cdot g$ step K-smoothing network and $\mathbf{X}^{(wg)}$ is block-step, $\mathbf{Y}^{(wg)}$ has the K-smoothing property, i.e., $|y_j - y_k| \leq K$ for all j and k, $0 \leq j, k \leq wg - 1$. It follows that $y_j = y_k$ for all j and k, $0 \leq j, k \leq wg - 1$.

By Proposition 2.8, $\|\mathbf{Y}^{(wg)}\|_1 = \|\mathbf{X}^{(wg)}\|_1 = g\, x$. It follows that for each $j \in [w]$,

$$y_j \;=\; \frac{1}{wg}\, g\, x \;=\; \frac{1}{w}\, x .$$

Notice that

$$y_j \;=\; \sum_{l=0}^{wg-1} \mathbf{I}_{\mathcal{B}}[jl]\, x_l \;=\; \sum_{l=0}^{w-1} \sum_{r \in \pi_l} \mathbf{I}_{\mathcal{B}}[jr]\, x_r \;=\; \sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr]\, x_r \;=\; x \sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr] .$$

It follows that $\sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr] = 1/w$, as needed. ∎

Since a $w \cdot g$ K_1-smooth counting network is also both a $w \cdot g$ step counting network and a $w \cdot g$ K_1-smooth K-smoothing network, while each of the latter networks is also a $w \cdot g$ K_1-smooth counting network, Theorem 5.14 immediately implies:

Corollary 5.15 *Assume $\mathcal{B} : \mathbf{X}^{(wg)} \to \mathbf{Y}^{(wg)}$ is a $w \cdot g$ step counting network, a K_1-smooth, K-smoothing network, or a K_1-smooth counting network. Then,*

$$\sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr] \;=\; \frac{1}{w} ,$$

for all $i \in [w]$ and $j \in [wg]$.

We remark that the (necessary) combinatorial condition on the steady transfer matrix shown in Theorem 5.14 and Corollary 5.15 for block-input networks is "dual" to the (necessary and sufficient) corresponding conditions shown in Theorems 5.2 and 5.4 for block-output networks: by matrix transposition, one yields the other. Recall that outputs and inputs are associated with rows and columns, respectively, of the steady transfer matrix. Hence, given that the primary difference between the definitions of block-output and block-input networks is with respect to outputs and inputs, this "duality" is perhaps not too surprising.

6 Applications

Sections 6.1, 6.2, 6.3 and 6.4 contain results on the relation between counting and sorting, impossibility results, verification algorithms, and a methodology for the design of smoothing networks, respectively.

6.1 Counting Versus Sorting

Recall that, by Proposition 2.15, the class of sorting networks is identical to the class of 1-smooth counting networks. Thus, in order to obtain combinatorial characterization results for sorting networks, we proceed to obtain such results for K_1-smooth counting networks, a restricted class of block-input networks, and specialize them to the case where $K_1 = 1$. We show:

Proposition 6.1 *Assume $K_1 \leq P^d - 1$. Then, the network $\mathcal{B} : \mathbf{X}^{(g)} \rightarrow \mathbf{Y}^{(g)}$ of depth d over $\mathcal{P}$ is a K_1-smooth counting network if and only if $\mathbf{F}_{\mathcal{B}}$ is step on $[K_1 + 1]^g$.*

Proof: Assume first that $\mathcal{B}$ is a K_1-smooth counting network. Take any vector $\mathbf{X}^{(g)} \in [K_1 + 1]^g$. We show that $\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$ has the step property. Since $\|\mathbf{X}^{(g)}\|_{\max} \leq K_1$ and $K_1 \leq P^d - 1$, it follows that $\mathbf{X}^{(g)} \uparrow_P d = \mathbf{0}^{(g)}$. Hence, by Theorem 3.1, $\mathbf{Y}^{(g)} = \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$. Since $\mathbf{X}^{(g)}$ is K_1-smooth and $\mathcal{B}$ is a K_1-smooth counting network, $\mathbf{Y}^{(g)}$ has the step property. It follows that $\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$ has the step property, as needed.

Assume now that $\mathbf{F}_{\mathcal{B}}$ is step on $[K_1 + 1]^g$. Take any K_1-smooth input vector $\mathbf{X}^{(g)}$. We show that $\mathbf{Y}^{(g)}$ has the step property. By assumption, $\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$ has the step property. Since $\|\mathbf{X}^{(g)}\|_{\max} \leq K_1$ and $K_1 \leq P^d - 1$, it follows that $\mathbf{X}^{(g)} \uparrow_P d = \mathbf{0}^{(g)}$. Hence, by Theorem 3.1, $\mathbf{Y}^{(g)} = \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$. It follows that $\mathbf{Y}^{(g)}$ has the step property, as needed. ■

Since $P \geq 2$ and $d \geq 1$, $P^d - 1 \geq 1$. Hence, a combinatorial characterization of sorting networks, *viz.* 1-smooth counting networks (by Proposition 2.15), follows immediately from Proposition 6.1 by setting K_1 to one.

Corollary 6.2 *The network $\mathcal{B} : \mathbf{X}^{(g)} \rightarrow \mathbf{Y}^{(g)}$ is a sorting network if and only if $\mathbf{F}_{\mathcal{B}}$ is step on $\{0,1\}^g$.*

Corollary 6.2 implies that the sorting property is solely determined by the transient output term; more specifically, a network sorts if and only if its transient transfer function has the step property on the "binary" part of its domain. In contrast, by Corollary 5.3, a network "counts" if and only if its steady transfer matrix is constant and its transient transfer function has the step property on its *whole* domain. As Aspnes *et al.* remark [5, p. 1026], "There is a sense in which constructing counting networks is "harder" than constructing sorting networks."

Thus, Corollaries 5.3 and 6.2 furnish a mathematical explanation for this hardness by establishing combinatorial conditions on the transfer parameters which a counting network must, in addition, satisfy.

We next consider K_1-smooth counting networks with $K_1 > P^d - 1$, in order to completely settle the question of combinatorially characterizing K_1-smooth counting networks.

Proposition 6.3 *Assume $K_1 > P^d - 1$. Then, the network $\mathcal{B} : \mathbf{X}^{(g)} \rightarrow \mathbf{Y}^{(g)}$ of depth d over $\mathcal{P}$ is a K_1-smooth counting network if and only if*

(1) $\mathbf{I}_{\mathcal{B}} \ = \ (1/g)\,\mathbf{1}^{(g\times g)}$, *and*

(2) $\mathbf{F}_{\mathcal{B}}$ *is step on* $[P^d]^g$.

Proof: Assume first that $\mathcal{B}$ is a K_1-smooth counting network. We start by showing that $\mathbf{I}_{\mathcal{B}} = (1/g)\,\mathbf{1}^{(g\times g)}$.

Fix any indices i and j, $0 \leq i, j \leq g-1$. We show that $\mathbf{I}_{\mathcal{B}}[ji] = 1/g$. Consider an input vector $\mathbf{X}^{(g)}$ such that $\mathbf{X}^{(g)} \downarrow_P d = \mathbf{0}^{(g)}$, $x_i \uparrow_P d = P^d$ and $x_l \uparrow_P d = 0$ for $l \neq i$, so that $\|\mathbf{X}^{(g)}\|_1 = P^d$. Since, by Proposition 4.7, $\mathbf{F}_{\mathcal{B}}(\mathbf{0}^{(g)}) = \mathbf{0}^{(g)}$, Theorem 3.1 implies that

$$y_j \ = \ \mathbf{I}_{\mathcal{B}}[ji]\, P^d .$$

Since $P^d \leq K_1$, the input vector $\mathbf{X}^{(g)}$ is K_1-smooth. Hence, since $\mathcal{B}$ is a K_1-smooth counting network, $\mathbf{Y}^{(g)}$ has the step property so that

$$y_j \ = \ \left\lceil \frac{\|\mathbf{X}^{(g)}\|_1 - j}{g} \right\rceil \ = \ \left\lceil \frac{P^d - j}{g} \right\rceil .$$

It follows that

$$\mathbf{I}_{\mathcal{B}}[ji] \ = \ \frac{1}{P^d} \left\lceil \frac{P^d - j}{g} \right\rceil .$$

Thus, $\mathbf{I}_{\mathcal{B}}[ji]$ is independent of i so that $\sum_{l=0}^{g-1} \mathbf{I}_{\mathcal{B}}[jl] \ = g\, \mathbf{I}_{\mathcal{B}}[ji]$. However, by Proposition 4.1, $\sum_{l=0}^{g-1} \mathbf{I}_{\mathcal{B}}[jl] = 1$. It follows that $\mathbf{I}_{\mathcal{B}}[ji] = 1/g$, as needed.

We continue to show that $\mathbf{F}_{\mathcal{B}}$ is step on $[P^d]^g$. Take any vector $\mathbf{X}^{(g)} \in [P^d]^g$. We show that $\mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$ has the step property. Clearly, $\mathbf{X}^{(g)} \uparrow_P d = \mathbf{0}^{(g)}$ and $\mathbf{X}^{(g)} \downarrow_P d = \mathbf{X}^{(g)}$, so that, by Theorem 3.1, $\mathbf{Y}^{(g)} = \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$. Since $\|\mathbf{X}^{(g)}\|_{\max} \leq P^d$ and $P^d \leq K_1$, $\mathbf{X}^{(g)}$ is K_1-smooth; so, since $\mathcal{B}$ is a K_1-smooth counting network, it follows that $\mathbf{Y}^{(g)} = \mathbf{F}_{\mathcal{B}}(\mathbf{X}^{(g)})$ has the step property, as needed.

Assume now that conditions (1) and (2) hold. Corollary 5.3 implies that $\mathcal{B}$ is a counting network. Since a counting network is also a K_1-smooth counting network for every integer $K_1 \geq 1$, it follows that $\mathcal{B}$ is a K_1-smooth counting network, as needed. ■

Appealing to Corollary 5.3, Proposition 6.3 establishes that the class of counting networks of depth d over $\mathcal{P}$ is precisely the class of K_1-smooth counting networks with $K_1 > P^d - 1$. It follows that the smooth counting constant of a counting network of depth d over $\mathcal{P}$ is equal to P^d. Thus, Proposition 6.3 identifies a second instance of a Zero-One Law for a "smoothing-like" property, namely, the K_1-smooth counting property. Specifically, Proposition 6.3 implies that the smooth counting constant of a balancing network is at most P^d.

We remark that Propositions 2.15 and 6.3 identify 1-smooth counting networks and P^d-smooth counting networks as those classes of K_1-smooth counting networks that coincide with the classes of sorting and counting networks, respectively. This furnishes an alternative mathematical explanation of the separation between sorting and counting networks, as being due to the difference between their smooth counting constants.

Klugerman [27, Section 2.2] addresses the related problem of separating sorting networks and smoothing networks. Klugerman [27, Lemmas 2.2.3 and 2.2.4] provides explicit counterexamples of a smoothing but not sorting network, and a sorting but not smoothing (or counting) network, and he writes:

> "Our understanding of the relationship between counting networks and sorting networks far exceeds our understanding of the relationship between smoothing networks and a sorting network. Perhaps there is no strong connection. What is true is that one does not necessarily imply the other."

We feel that our results significantly contribute to the understanding of the relationship between smoothing networks and sorting networks. Notice that Corollaries 5.9 and 6.2 characterizing smoothing and sorting networks, respectively, are "incomparable", since they establish conditions on different combinatorial objects, namely the steady transfer matrix and the transient transfer function. Thus, Corollaries 5.9 and 6.2 provide a mathematical explanation of the separations observed by Klugerman [27] as being due to "incomparable" combinatorial conditions on transfer parameters of smoothing networks and sorting networks.

6.2 Impossibility Results

Inconstructibility results and lower bounds appear in Sections 6.2.1 and 6.2.2, respectively.

6.2.1 Inconstructible Widths

For block-output networks, we show:

Proposition 6.4 *Assume $\mathcal{B} : \mathbf{X}^{(wg)} \to \mathbf{Y}^{(wg)}$ is a block-output network of depth d over $\mathcal{P}$. Then, w divides P^d.*

Proof: Proposition 4.2 implies that for any $i \in [wg]$ and $j \in [w]$, $P^d \sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri]$ is an integer. By Theorems 5.2, 5.4 and 5.8, $\sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri] = 1/w$. It follows that w divides P^d, as needed. ■

Proposition 6.4 specializes in the case $g = 1$ to yield:

Corollary 6.5 *Assume $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ is a counting network or a K-smoothing network of depth d over $\mathcal{P}$. Then, w divides P^d.*

As for K-smoothing networks, Corollary 6.5 strictly strengthens [2, Theorem 3.5] showing a corresponding width limitation, namely that for each prime factor w_i of w, there exists some integer $p \in \mathcal{P}$ such that w_i divides p. We argue that Corollary 6.5 is indeed strictly stronger. Let the unique prime factorization of w be $w = \prod w_i^{l_i}$. Since P is the least common multiple of integers in $\mathcal{P}$, the unique prime factorization of P is $P = \prod p_m^{\max_p l_m(p)}$, where the product is taken over all prime factors p_m of integers $p \in \mathcal{P}$, and $l_m(p)$ is the degree of p_m in the unique prime factorization of p. Assume first that w divides P^d. Then, clearly, each prime factor of w is equal to some p_m in the unique prime factorization of P; hence, this prime factor divides each $p \in \mathcal{P}$ such that p_m is a prime factor of p. Assume now that for each prime factor w_i of w, there exists some $p \in \mathcal{P}$ such that w_i divides p. Clearly, w_i is equal to some p_m in the unique prime factorization of P. However, w may not divide P^d if $l_m > d \max_p l_m(p)$. A corresponding impossibility result for K-smoothing networks over $\{2\}$, namely that w divides 2^d, has been claimed in [34]; Corollary 6.5 is the generalization of this impossibility result to an arbitrary set of balancer types.

We next turn to block-input networks and show:

Proposition 6.6 *Assume $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ is a block-input network of depth d over $\mathcal{P}$. Then, w divides P^d.*

Proof: Proposition 4.2 implies that for any $i \in [w]$ and $j \in [wg]$, $P^d \sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr]$ is an integer. By Theorem 5.14 and Corollary 5.15, $\sum_{r \in \pi_i} \mathbf{I}_{\mathcal{B}}[jr] = 1/w$. It follows that w divides P^d, as needed. ■

In conclusion, we remark that the proofs of Propositions 6.4 and 6.6 followed from simple properties of the steady transfer matrix that are necessary for block-output networks and block-input networks (Theorems 5.2, 5.4 and 5.8, and Theorem 5.14 and Corollary 5.15, respectively), but did not use any property of the transient transfer function. Since properties of the steady transfer matrix are associated with properties of the steady output term, this appears to suggest that, in general, width limitations on block-output networks and block-input networks are due to requirements on their steady behavior. Notice also that the width limitations shown in Propositions 6.4 and 6.6 involve none of the parameters g or K that are used in the definitions of block-output networks and block-input networks.

6.2.2 Lower Bounds

An immediate implication of Propositions 6.4 and 6.6 is a lower bound on depth for block-output and block-input networks.

Proposition 6.7 *Assume* $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ *is a block-output or block-input network over* $\mathcal{P}$*. Then, depth*$(\mathcal{B}) \geq \log_P w$.

Moran and Taubenfeld [34, Section 5] show that the depth of any K-smoothing network of width w over $\mathcal{P}$ is at least $\log_2 w$. Clearly, this result is but the special case where $g = 1$ and $\mathcal{P} = \{2\}$ of the implication of Proposition 6.7 for $w \cdot g$ K-smoothing networks.

For the special case where $g = 1$, we can substantially strengthen Proposition 6.7. Recall that by Corollaries 5.3 and 5.5, for any counting or K-smoothing network $\mathcal{B}$ of width w over $\mathcal{P}$, $\mathbf{I}_{\mathcal{B}} = (1/w)\, \mathbf{1}^{(w \times w)}$. Hence, Corollary 4.4 immediately implies:

Proposition 6.8 *Assume* $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ *is a counting or* K*-smoothing network over* $\mathcal{P}$*. Then,*

$$\|\mathbf{D}_{\mathcal{B}}\|_{\min} \;\geq\; \log_{p_0} w\,.$$

Proposition 6.8 implies that for counting or K-smoothing networks over $\mathcal{P}$, *every* path from an input wire to an output wire must have length at least $\log_{p_0} w$. In [5, Corollary 2.5] and [34, Section 5], it is shown that the depth of any counting network or K-smoothing network, respectively, of width w over $\{2\}$ is at least $\log_2 w$. Proposition 6.8 strictly strengthens and generalizes to an arbitrary set of balancer types each of these results.

We continue with more specific results for block-output networks and block-input networks.

Proposition 6.9 *For a block-output network* $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ *over* $\mathcal{P}$*,*

$$\min_{i \in [wg]} \max_{r \in \pi_j} \mathbf{D}_{\mathcal{B}}[ri] \;\geq\; \log_{p_0} wg\,.$$

Proof: By Corollary 4.6(1),

$$\min_{i \in [wg]} \max_{r \in \pi_j} \mathbf{D}_{\mathcal{B}}[ri] \;\geq\; \log_{p_0} \frac{g}{\max_{i \in [wg]} \sum_{r \in \pi_j} \mathbf{I}_{\mathcal{B}}[ri]} \;=\; \log_{p_0} \frac{g}{\max_{i \in [wg]} 1/w} \;=\; \log_{p_0} wg\,,$$

by Theorems 5.2 and 5.4, as needed. ■

Proposition 6.9 implies that for block-output networks, there exists, for each output block, at least one "long" path from an input wire to an output wire in the block. A corresponding result can be shown for block-input networks by using Corollary 4.6(2), Theorem 5.14 and Corollary 5.15.

Proposition 6.10 *For a block-output network* $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ *over* $\mathcal{P}$*,*

$$\min_{j \in [wg]} \max_{r \in \pi_i} \mathbf{D}_{\mathcal{B}}[jr] \;\geq\; \log_{p_0} wg\,.$$

6.3 Verification Algorithms

Let $\mathbf{\Phi}$ be a *property* on balancing networks over $\mathcal{P}$, identified with the class of networks satisfying it.

Definition 6.1 *$\mathbf{\Phi}$ is a* finite property *if for any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$, there exists a condition $C = C(\mathcal{B}, \mathbf{X}^{(w)})$ such that the network $\mathcal{B}$ possesses the property $\mathbf{\Phi}$ if and only if it satisfies $C(\mathcal{B}, \mathbf{X}^{(w)})$ for all $\mathbf{X}^{(w)}$ with $\|\mathbf{X}^{(w)}\|_{\max} \leq \lambda_C$, for some integer $\lambda_C = \lambda_C(\mathcal{B})$.*

That is, $\mathbf{\Phi}$ is a finite property if, for any given balancing network, there exists a condition C, formulated in terms of parameters of the networks and the input vector, such that the network possesses the property if and only if the condition holds when each individual input is no more than a "threshold" value, possibly dependent on parameters of the network. Call C a *finiteness condition for $\mathbf{\Phi}$*, and call λ_C the *threshold size for C*.

Finite properties allow for effective verification through checking, on a given network, that a finiteness condition C holds for all $\mathbf{X}^{(w)}$ with $\|\mathbf{X}^{(w)}\|_{\max} \leq \lambda_C$. The corresponding time complexity of verification is equal to $(\lambda_C + 1)^w$ times the time complexity of verifying, on a single input vector, that C holds.

For a finite property $\mathbf{\Phi}$, define the *threshold size of $\mathbf{\Phi}$*, denoted $\lambda_{\mathbf{\Phi}} = \lambda_{\mathbf{\Phi}}(\mathcal{B})$, to be the least possible threshold size over all finiteness conditions for $\mathbf{\Phi}$. We proceed to establish upper bounds on the threshold sizes of various properties of interest. For the sorting property, Proposition 2.13 immediately implies:

Proposition 6.11 *For any network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ over $\mathcal{P}$, $\lambda_{\mathbf{Srt}_w}(\mathcal{B}) = 1$*

For the $w \cdot g$ step property and the $w \cdot g$ K-smoothing property, Theorems 5.2 and 5.4 imply:

Proposition 6.12 *For any network $\mathcal{B} : \mathbf{X}^{(wg)} \rightarrow \mathbf{Y}^{(wg)}$ of depth d over $\mathcal{P}$, $\lambda_{\mathbf{St}_{w \cdot g}}(\mathcal{B}) \leq P^d - 1$ and $\lambda_{K\text{-}\mathbf{Sm}_{w \cdot g}}(\mathcal{B}) \leq P^d - 1$.*

Since counting and K-smoothing networks are special cases of $w \cdot g$ counting networks and $w \cdot g$ K-smoothing networks, respectively, Proposition 6.12 immediately implies:

Corollary 6.13 *For any network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ of depth d over $\mathcal{P}$, $\lambda_{\mathbf{St}_w}(\mathcal{B}) \leq P^d - 1$ and $\lambda_{K\text{-}\mathbf{Sm}_w}(\mathcal{B}) \leq P^d - 1$.*

We can make no conclusions regarding threshold sizes of block-input networks, in general, since we have not been able to show any conditions that are both necessary and sufficient for them. However, for the special case of the K_1-smooth counting property, Proposition 2.16 implies an upper bound of K_1 on its threshold size. Our next result establishes an upper bound of $\min\{K_1, P^d - 1\}$ on this threshold size, which may sometimes be better than K_1.

Proposition 6.14 *For any network $\mathcal{B} : \mathbf{X}^{(g)} \to \mathbf{Y}^{(g)}$ of depth d over $\mathcal{P}$, $\lambda_{K_1\text{-}\mathbf{Sm}_{1\cdot g} \rightsquigarrow \mathbf{St}_g} \leq \min\{K_1, P^d - 1\}$.*

Proof: If $K_1 \leq P^d - 1$, the claim follows immediately from Proposition 2.16. So assume $K_1 > P^d - 1$ so that $\min\{K_1, P^d - 1\} = P^d - 1$. Recall that in this case, appealing to Proposition 6.3, $\mathcal{B}$ is a K_1-smooth counting network if and only if it is a counting network, so that

$$\lambda_{K_1\text{-}\mathbf{Sm}_{1\cdot g} \rightsquigarrow \mathbf{St}_g} \;\leq\; \lambda_{\mathbf{St}_g} \;\leq\; P^d - 1 \;=\; \min\{K_1, P^d - 1\}\,,$$

as needed. ■

By way of example, we describe an algorithm for verifying the step property; that is, we describe an algorithm which, given a network $\mathcal{B}$ of depth d over $\mathcal{P}$, checks whether or not $\mathcal{B}$ is a counting network.

> Compute $\mathbf{I}_{\mathcal{B}}$ and $\mathbf{F}_{\mathcal{B}}$ from $\mathbf{I}_{\mathcal{B}_1}, \mathbf{I}_{\mathcal{B}_2}, \ldots, \mathbf{I}_{\mathcal{B}_d}$ and $\mathbf{O}_{\mathcal{B}_1}, \mathbf{O}_{\mathcal{B}_2}, \ldots, \mathbf{O}_{\mathcal{B}_d}$, using their inductive definitions in Theorem 3.1. Check whether or not $\mathbf{I}_{\mathcal{B}}$ and $\mathbf{F}_{\mathcal{B}}$ satisfy conditions (1) and (2) in Corollary 5.3.

For condition (1), $d-1$ matrix multiplications suffice, incurring a time complexity polynomial in w and d. For condition (2), there are P^{dw} input vectors on which $\mathbf{F}_{\mathcal{B}}$ needs to be evaluated, and each evaluation incurs a time complexity proportional to $size(\mathcal{B})$, for a total time complexity of $O(P^{dw} size(\mathcal{B}))$, which is exponential in w. Hence, this last time complexity is the dominating one for verifying the step property.

Similar verification algorithms for the K-smoothing property, the $w \cdot g$ step property, the $w \cdot g$ K-smoothing property and the K_1-smooth step property follow from Corollary 5.5, and Theorems 5.2 and 5.4, respectively. A drawback of all these verification algorithms is that they incur exponential time complexity, since they involve evaluating the transient transfer function on exponentially many input vectors. At this point, it is natural to ask whether there are (finite) properties allowing for *efficient* (polynomial time) verification. To capture such properties, a refinement of Definition 6.1 is called for.

Definition 6.2 *$\mathbf{\Phi}$ is a* constant property *if for any balancing network $\mathcal{B} : \mathbf{X}^{(w)} \to \mathbf{Y}^{(w)}$, there exists a condition $C = C(\mathcal{B})$ such that the network $\mathcal{B}$ possesses the property $\mathbf{\Phi}$ if and only if $C(\mathcal{B})$ holds.*

Clearly, a constant property is a finite property with zero threshold size. Consequently, verification of a constant property is reduced to a single check of a finiteness condition. Theorem 5.8 identifies a constant property.

Proposition 6.15 *The $w \cdot g$ smoothing property is a constant property.*

Proposition 6.15 specializes in the case $g = 1$ to yield:

Corollary 6.16 *The smoothing property is a constant property.*

In [5, Section 7], the problem of verifying that a network "counts" is also studied; it is shown there that a balancing network of size m over $\{2\}$ is a counting network if its output vector is step in all sequential executions in which at most 2^m tokens traverse the network. We compare this result to Corollary 6.13. Our result, being more general in dealing with networks over arbitrary sets of balancer types, specializes in the case $\mathcal{P} = \{2\}$ to imply that, beyond the steady transfer matrix being constant, it suffices to consider executions in which at most 2^d tokens enter on each input wire, where d is the network's depth; obviously, these are executions in which at most 2^{dw} tokens traverse the network in total. Since, in general, $m \in \Theta(dw)$, both results provide finiteness conditions for the step property of essentially identical threshold sizes in the special case $\mathcal{P} = \{2\}$. Furthermore, it is shown in [5, Section 7] that this threshold size is the smallest possible. However, our result has given rise to an effective algorithm (sketched above) for verifying the step property; this algorithm incurs a $\Theta(dw2^{dw})$ time complexity. To the best of our understanding, it is not clear how the result in [5] can be translated into a corresponding verification algorithm of comparable time complexity. (This is so because 2^m tokens need to be assigned to input wires and traverse the network asynchronously in all possible combinations, and there are $\binom{2^m+w-1}{2^m} \in \Theta((2^m)^{2^m+w}) = \Theta(2^{dw})^{2^{dw}+w}$ ways of even distributing 2^m tokens into w input wires.)

In conclusion, we remark that the threshold size of the sorting property depends neither on the types of balancers used nor the depth d of a network. In contrast, an analysis in [5, Section 7] implies that the dependence on d of the threshold size of the step property is necessary in the case where $\mathcal{P} = \{2\}$.

6.4 A Methodology to Design Smoothing Networks

Appealing to Theorem 3.1(1), Corollary 5.9 suggests a methodology to design a smoothing network of width w and a given depth: construct a matrix product chain of length equal to the given depth that results in a matrix with all entries equal to $1/w$. We present an application of this general methodology for the special case where w is a power of two.

Proposition 6.17 *For any integer w that is a power of two, there exists a smoothing network $\mathcal{B} : \mathbf{X}^{(w)} \rightarrow \mathbf{Y}^{(w)}$ with depth $\lg w$.*

Proof: We construct a sequence of $\lg w$ incidence matrices $\mathbf{I}_0^{(w\times w)}, \mathbf{I}_1^{(w\times w)}, \ldots, \mathbf{I}_{\lg w-1}^{(w\times w)}$ such that $\mathbf{I}_{\lg w-1}^{(w\times w)} \cdot \ldots \cdot \mathbf{I}_1^{(w\times w)} \cdot \mathbf{I}_0^{(w\times w)} = (1/w)\, \mathbf{1}^{(w\times w)}$.

We first introduce some notation. For any integer $w \geq 2$, denote by $\mathbf{Dg}^{(w \times w)}$ the square matrix that has all of its diagonal entries equal to $1/2$, and all other entries equal to 0.

We proceed by induction on the size w of an incidence matrix. For the base case where $w = 2$,

$$\mathbf{I}_0^{(2\times 2)} \;=\; \frac{1}{2}\,\mathbf{1}^{(2\times 2)} \;=\; \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}.$$

Assume inductively that for some $w \geq 4$, we have constructed $\mathbf{I}_k^{(w/2\times w/2)}$ for all k such that $0 \leq k \leq \lg w - 2$. We show how to construct $\mathbf{I}_k^{(w\times w)}$ for all k such that $0 \leq k \leq \lg w - 1$:

$$\mathbf{I}_k^{(w\times w)} \;=\; \begin{cases} \begin{pmatrix} \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \\ \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \end{pmatrix}, & k = 0 \\ \begin{pmatrix} \mathbf{I}_{k-1}^{(w/2\times w/2)} & \mathbf{0}^{(w/2\times w/2)} \\ \mathbf{0}^{(w/2\times w/2)} & \mathbf{I}_{k-1}^{(w/2\times w/2)} \end{pmatrix}, & 0 < k \leq \lg w - 1 \end{cases}$$

By way of example, we consider the case where $w = 16$ and exhibit the incidence matrices $\mathbf{I}_0^{(16\times 16)}$, $\mathbf{I}_1^{(16\times 16)}$, $\mathbf{I}_2^{(16\times 16)}$ and $\mathbf{I}_3^{(16\times 16)}$:

$$\mathbf{I}_0^{(16\times 16)} = \left(\begin{array}{cccccccc|cccccccc}
1/2&0&0&0&0&0&0&0&1/2&0&0&0&0&0&0&0\\
0&1/2&0&0&0&0&0&0&0&1/2&0&0&0&0&0&0\\
0&0&1/2&0&0&0&0&0&0&0&1/2&0&0&0&0&0\\
0&0&0&1/2&0&0&0&0&0&0&0&1/2&0&0&0&0\\
0&0&0&0&1/2&0&0&0&0&0&0&0&1/2&0&0&0\\
0&0&0&0&0&1/2&0&0&0&0&0&0&0&1/2&0&0\\
0&0&0&0&0&0&1/2&0&0&0&0&0&0&0&1/2&0\\
0&0&0&0&0&0&0&1/2&0&0&0&0&0&0&0&1/2\\
\hline
1/2&0&0&0&0&0&0&0&1/2&0&0&0&0&0&0&0\\
0&1/2&0&0&0&0&0&0&0&1/2&0&0&0&0&0&0\\
0&0&1/2&0&0&0&0&0&0&0&1/2&0&0&0&0&0\\
0&0&0&1/2&0&0&0&0&0&0&0&1/2&0&0&0&0\\
0&0&0&0&1/2&0&0&0&0&0&0&0&1/2&0&0&0\\
0&0&0&0&0&1/2&0&0&0&0&0&0&0&1/2&0&0\\
0&0&0&0&0&0&1/2&0&0&0&0&0&0&0&1/2&0\\
0&0&0&0&0&0&0&1/2&0&0&0&0&0&0&0&1/2
\end{array}\right),$$

$$\mathbf{I}_1^{(16\times 16)} = \left(\begin{array}{cccccccc|cccccccc}
1/2&0&0&0&1/2&0&0&0&0&0&0&0&0&0&0&0\\
0&1/2&0&0&0&1/2&0&0&0&0&0&0&0&0&0&0\\
0&0&1/2&0&0&0&1/2&0&0&0&0&0&0&0&0&0\\
0&0&0&1/2&0&0&0&1/2&0&0&0&0&0&0&0&0\\
\cline{1-8}
1/2&0&0&0&1/2&0&0&0&0&0&0&0&0&0&0&0\\
0&1/2&0&0&0&1/2&0&0&0&0&0&0&0&0&0&0\\
0&0&1/2&0&0&0&1/2&0&0&0&0&0&0&0&0&0\\
0&0&0&1/2&0&0&0&1/2&0&0&0&0&0&0&0&0\\
\hline
0&0&0&0&0&0&0&0&1/2&0&0&0&1/2&0&0&0\\
0&0&0&0&0&0&0&0&0&1/2&0&0&0&1/2&0&0\\
0&0&0&0&0&0&0&0&0&0&1/2&0&0&0&1/2&0\\
0&0&0&0&0&0&0&0&0&0&0&1/2&0&0&0&1/2\\
\cline{9-16}
0&0&0&0&0&0&0&0&1/2&0&0&0&1/2&0&0&0\\
0&0&0&0&0&0&0&0&0&1/2&0&0&0&1/2&0&0\\
0&0&0&0&0&0&0&0&0&0&1/2&0&0&0&1/2&0\\
0&0&0&0&0&0&0&0&0&0&0&1/2&0&0&0&1/2
\end{array}\right),$$

$$
\mathbf{I}_2^{(16\times 16)} = \left(\begin{array}{cccccccc|cccccccc}
1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2
\end{array}\right),
$$

$$
\mathbf{I}_3^{(16\times 16)} = \left(\begin{array}{cccccccc|cccccccc}
1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2
\end{array}\right).
$$

By construction,

$$
\begin{aligned}
& \mathbf{I}_{\lg w-1}^{(w\times w)} \cdot \ldots \cdot \mathbf{I}_1^{(w\times w)} \cdot \mathbf{I}_0^{(w\times w)} \\
= & \begin{pmatrix} \mathbf{I}_{\lg w-2}^{(w/2\times w/2)} & \mathbf{0}^{(w/2\times w/2)} \\ \mathbf{0}^{(w/2\times w/2)} & \mathbf{I}_{\lg w-2}^{(w/2\times w/2)} \end{pmatrix} \cdot \ldots \cdot \begin{pmatrix} \mathbf{I}_0^{(w/2\times w/2)} & \mathbf{0}^{(w/2\times w/2)} \\ \mathbf{0}^{(w/2\times w/2)} & \mathbf{I}_0^{(w/2\times w/2)} \end{pmatrix} \cdot \\
& \begin{pmatrix} \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \\ \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \end{pmatrix} \\
= & \begin{pmatrix} \mathbf{I}_{\lg w-2}^{(w/2\times w/2)} \cdot \ldots \cdot \mathbf{I}_0^{(w/2\times w/2)} & \mathbf{0}^{(w/2\times w/2)} \\ \mathbf{0}^{(w/2\times w/2)} & \mathbf{I}_{\lg w-2}^{(w/2\times w/2)} \cdot \ldots \cdot \mathbf{I}_0^{(w/2\times w/2)} \end{pmatrix} \cdot \\
& \begin{pmatrix} \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \\ \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \end{pmatrix} \\
= & \begin{pmatrix} (2/w)\,\mathbf{1}^{(w/2\times w/2)} & \mathbf{0}^{(w/2\times w/2)} \\ \mathbf{0}^{(w/2\times w/2)} & (2/w)\,\mathbf{1}^{(w/2\times w/2)} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \\ \mathbf{Dg}^{(w/2\times w/2)} & \mathbf{Dg}^{(w/2\times w/2)} \end{pmatrix} \\
& \text{(by induction hypothesis)} \\
= & (1/w)\,\mathbf{1}^{(w\times w)}.
\end{aligned}
$$

Take a balancing network $\mathcal{B}$ such that each of $\mathbf{I}_0^{(w\times w)}, \mathbf{I}_1^{(w\times w)}, \ldots, \mathbf{I}_{\lg w-1}^{(w\times w)}$ is the incidence matrix of the corresponding layer of $\mathcal{B}$, scanning layers from left to right. By Theorem 3.1(1),

$\mathbf{I}_{\mathcal{B}} = \mathbf{I}_{\lg w-1}^{(w\times w)} \cdot \ldots \cdot \mathbf{I}_{1}^{(w\times w)} \cdot \mathbf{I}_{0}^{(w\times w)} = (1/w)\,\mathbf{1}^{(w\times w)}$. Hence, by Theorem 5.9, $\mathcal{B}$ is a smoothing network, as needed. ∎

Busch and Mavronicolas show [12, Proposition 6.2] that the *bitonic merging* network of Batcher [7] with width $\lg w$ provides another instance of a balancing network that has depth $\lg w$ and all entries of its steady transfer matrix equal to $1/w$, assuming that w is a power of two. Hence, by Theorem 5.9, the bitonic merging network is a smoothing network, too. At present, we do not know the value of the smoothing constant of either the bitonic merging network or the network in Proposition 6.17.

7 Discussion and Directions for Further Research

In this Section, we provide a review of our results, a survey of follow-up work, and directions for further research.

7.1 Review and Follow-Up Work

We have presented a mathematical framework for the study of the combinatorial properties of balancing networks. Within this framework, we derived a combinatorial result expressing the outputs of a balancing network as a function of the inputs, dependent on the type of balancers used, and the depth and topology of the network, as defined by certain structural matrices. This single result has been our main mathematical instrument in deriving combinatorial characterization results for various classes of balancing networks that have recently been the focus of intensive research, such as counting networks and smoothing networks. In turn, these characterization results have implied corresponding impossibility results and verification algorithms for these classes of networks. Our proofs have been non-trivial, yet elementary in nature. We summarize in Table 1 the combinatorial characterization results shown in this paper. We feel that our results have made substantial progress towards improving our understanding of the mathematical features and combinatorial properties of balancing networks. Beyond the specific results we have derived, an important contribution of our work is, in our opinion, the development of an alternative, combinatorial framework within which further research on balancing networks can be pursued.

Following the original presentation of our work [11], some follow-up work has appeared. Most closely, Busch and Mavronicolas [13] consider the class of *threshold networks,* originally introduced by Aspnes *et al.* [5, Section 5.3], and use techniques similar to the present ones to show combinatorial characterization results for this class; in turn, these results imply corresponding impossibility results and a verification algorithm for threshold networks. (Preliminary versions of these results had been included in [11].)

Busch and Mavronicolas [12] develop a paradigmatic methodology for proving correctness for balancing networks as another application of the combinatorial theory presented in this

Network class	Combinatorial conditions	
	Necessary	Sufficient
$\mathbf{St}_{w\cdot g}$	$\sum_{r\in\pi_j}\mathbf{I}_{\mathcal{B}}[ri]=1/w$, $i\in[wg]$ and $j\in[w]$, & $\mathbf{F}_{\mathcal{B}/\Pi}$ step on $[P^d]^{wg}$	
K-$\mathbf{Sm}_{w\cdot g}$	$\sum_{r\in\pi_j}\mathbf{I}_{\mathcal{B}}[ri]=1/w$, $i\in[wg]$ and $j\in[w]$, & $\mathbf{F}_{\mathcal{B}/\Pi}$ K-smooth on $[P^d]^{wg}$	
$\mathbf{Sm}_{w\cdot g}$	$\sum_{r\in\pi_j}\mathbf{I}_{\mathcal{B}}[ri]=1/w$, $i\in[wg]$ and $j\in[w]$,	
$\mathbf{St}_{w}$	$\mathbf{I}_{\mathcal{B}}=(1/w)\,\mathbf{1}^{(w\times w)}$ & $\mathbf{F}_{\mathcal{B}}$ step on $[P^d]^w$	
K-$\mathbf{Sm}_{w}$	$\mathbf{I}_{\mathcal{B}}=(1/w)\,\mathbf{1}^{(w\times w)}$ & $\mathbf{F}_{\mathcal{B}}$ K-smooth on $[P^d]^w$	
$\mathbf{Sm}_{w}$	$\mathbf{I}_{\mathcal{B}}=(1/w)\,\mathbf{1}^{(w\times w)}$	
$\mathbf{St}_{w\cdot g}\leadsto\mathbf{St}_{wg}$	$\sum_{r\in\pi_i}\mathbf{I}_{\mathcal{B}}[jr]=1/w$, $i\in[w]$ and $j\in[wg]$	$\mathbf{F}_{\mathcal{B}}$ step on $(\mathit{blockstep}(\mathcal{N}^{wg}))\downarrow_P d$, if $\mathbf{I}_{\mathcal{B}}[ji]=1/wg$, $i\in[wg]$
K_1-$\mathbf{Sm}_{w\cdot g}\leadsto K$-$\mathbf{Sm}_{wg}$		$\mathbf{F}_{\mathcal{B}}$ K-smooth on $(\mathit{block}\text{-}K_1\text{-}\mathit{smooth}(\mathcal{N}^{wg}))\downarrow_P d$, if $\mathbf{I}_{\mathcal{B}}[ji]=1/wg$, $i\in[wg]$
K_1-$\mathbf{Sm}_{w\cdot g}\leadsto$-$\mathbf{St}_{wg}$		$\mathbf{F}_{\mathcal{B}}$ step on $(\mathit{block}\text{-}K_1\text{-}\mathit{smooth}(\mathcal{N}^{wg}))\downarrow_P d$, if $\mathbf{I}_{\mathcal{B}}[ji]=1/wg$, $i\in[wg]$
$\mathbf{St}_{w\cdot g}\leadsto K$-$\mathbf{Sm}_{wg}$		$\mathbf{F}_{\mathcal{B}}$ K-smooth on $(\mathit{blockstep}(\mathcal{N}^{wg}))\downarrow_P d$, if $\mathbf{I}_{\mathcal{B}}[ji]=1/wg$, $i\in[wg]$
K_1-$\mathbf{Sm}_{1\cdot g}\leadsto\mathbf{St}_g$, with $K_1>P^d-1$	$\mathbf{I}_{\mathcal{B}}=(1/g)\,\mathbf{1}^{(g\times g)}$ & $\mathbf{F}_{\mathcal{B}}$ step on $[P^d]^g$	
K_1-$\mathbf{Sm}_{1\cdot g}\leadsto\mathbf{St}_g$ with $K_1\leq P^d-1$	$\mathbf{F}_{\mathcal{B}}$ step on $[K_1+1]^g$	
$\mathbf{Srt}_g$	$\mathbf{F}_{\mathcal{B}}$ step on $\{0,1\}^g$	

Table 1: Summary of combinatorial characterization results

paper. This methodology has been applied to yield transparent correctness proofs for the *bitonic counting* network [12], a generalization of the bitonic counting network with width $p2^k$ for any integers $p\geq 2$ and $k\geq 0$ [10], and a new *odd-even counting* network [14] with layout building on that of the classical, *odd-even sorting* network of Batcher [7]. All of these proofs are simple and modular, and yield significant insights into the combinatorial structure of the respective networks; such insights had not been provided by earlier correctness proofs that relied on rather ad-hoc, operational arguments. Furthermore, significant fragments of these proofs consist of a very detailed and structured case analysis; it would be interesting to check if the routine parts of these proofs can be automated using a standard theorem prover like, e.g., the Larch Prover [21].

Brit, Moran and Taubenfeld [8] consider a class of counting protocols, called *static counters* and made up of smaller *atomic counters,* that generalize counting networks while still supporting *weak increment* and *weak read* operations. A generalization of Corollary 6.5 is shown in [8, Lemma 2.4] determining the integers modulo which a static counter can count, given a set of integers modulo which its building atomic counters count. We conjecture that similar impossibility results hold for classes of protocols generalizing corresponding weaker classes of balancing networks, like, e.g., smoothing networks.

7.2 Further Research

Our work raises many new interesting questions. Most obviously, we are still lacking a general combinatorial characterization of block-input networks. What would be necessary and sufficient conditions for a balancing network to be a *linearizable* counting network [26]? There are comparison networks, e.g., *Odd-Even* or *Insertion* [15, 29], that are sorting networks, but whose isomorphic balancing networks are known *not* to be counting networks [5]. This implies that the transfer parameters of these networks satisfy the conditions in Proposition 6.2, but not those in Corollary 5.3. What are the *tightest* conditions satisfied by these parameters? In other words, do Odd-Even and Insertion Sort networks actually do something more than just sorting? Since both sorting networks and counting networks have been found to be special cases of K_1-smooth counting networks for appropriate values of the parameter K_1, the question of precisely determining the computational power of these networks may also be stated as follows: What is the largest K_1, $1 \leq K_1 < P^d - 1$, for each of Odd-Even and Insertion Sort networks to be a K_1-smooth counting network? Results in these last directions related to the Odd-Even network have already been obtained in [14].

Our proofs of the combinatorial characterization and impossibility results have heavily relied on using inputs as large as P^d for networks of depth d. If the conditions on network outputs were only required to hold for inputs bounded above by some integer strictly less than P^d, these proofs would be invalidated. (A similar observation has been made in [2, Section 7] regarding the proofs of the less strong impossibility results presented there.) In practice, one may anticipate uses of balancing networks on multiprocessor architectures supporting a bounded number of processors, or required to "balance" a bounded number of jobs. Thus, it would be interesting to see whether the width inconstructibility results we showed could be overcome for networks with bounded inputs. In our opinion, the main gap left in our work is a characterization of balancing networks handling "small" inputs. We note, however, that none of the previous studies on balancing networks has addressed this special case.

Aiello *et al.* [3] introduce the notion of a *randomized balancer,* which is a balancer with two input and two output wires such that the output values are within one of each other, but the output wire with excess value, instead of being the top one, is chosen at random. Aiello *et al.* present constructions of *randomized* balancing networks made up of both (usual) balancers and randomized balancers; these constructions guarantee, up to a certain probability, variants of the counting or smoothing property. It would be extremely interesting to extend our present theory to randomized balancing networks; what would the probabilistic analogs of the transfer parameters and the combinatorial characterization results shown in this work be? Can any kind of limitations on network width be shown for randomized counting or smoothing networks? Making balancing networks resilient to failures is an important research direction to which we believe that our combinatorial framework can contribute too.

Lorys *et al.* [31] present a technique, called *periodification scheme,* for transforming any sorting network into a *periodic* sorting network of a constant period.[7] We believe that our

[7] Roughly speaking, a sorting network $\mathcal{B}$ is periodic if it is used repeatedly: while the output produced by $\mathcal{B}$

combinatorial framework may provide an excellent host for such transformations. We view any such transormation as consisting of identifying, given all incidence matrices and order vectors of a sorting network, a *constant* number of pairs of an incidence matrix and an order vector which "preserves", through Theorem 3.1, the transfer parameters of the original network. In particular, it would be interesting to check if the correctness proof of the periodification scheme in [31] can be expressed in our combinatorial framework; if so, this might enable one to prove even stronger properties of this scheme, like, e.g., whether or not it transforms any *counting* network into a periodic counting network of a constant period.

Finally, beyond some exponenential and polynomial upper bounds established in this paper, the precise computational complexity of verification for block-output networks and block-input networks remains as yet unknown. Parberry [33] shows that the problem of verifying that a given network does *not* sort is $\mathcal{NP}$-complete even when restricted to networks of depth close to optimal. It would be interesting to determine the computational complexity of corresponding problems for counting and smoothing networks.

Acknowledgments:

We are indebted to Maurice Herlihy for many enlightening discussions and encouragement during all stages of our research. We have had enriching discussions on the topic of our work with Cynthia Dwork and Orli Waarts. Hagit Attiya, Nikos Hardavellas and Isaac Saias read earlier drafts and provided numerous helpful comments and criticisms. We would also like to thank two anonymous referees for their constructive suggestions. Special thanks are due to the designers of Internet, which hosted most of our conversations on this research in an almost always convenient way.

is not sorted, it is fed back into it. The number of layers traversed during a single run of $\mathcal{B}$ is called its *period*; the *depth* of $\mathcal{B}$ is equal to the number of runs of $\mathcal{B}$ to produce a sorted output times its period. Periodic sorting networks of constant period are easy to implement in VLSI technology [16].

References

[1] A. Agarwal and M. Cherian, "Adaptive Backoff Synchronization Techniques," *Proceedings of the 16th Annual ACM-IEEE International Symposium on Computer Architecture,* pp. 396–406, June 1989.

[2] E. Aharonson and H. Attiya, "Counting Networks with Arbitrary Fan-Out," *Distributed Computing,* Vol. 8, pp. 163–169, 1995.

[3] W. Aiello, R. Venkatesan and M. Yung, "Coins, Weights and Contention in Balancing Networks," *Proceedings of the 13th Annual ACM Symposium on Principles of Distributed Computing,* pp. 193–205, August 1994.

[4] M. Aigner, *Combinatorial Theory,* Springer-Verlag, 1979.

[5] J. Aspnes, M. Herlihy and N. Shavit, "Counting Networks," *Journal of the ACM,* Vol. 41, No. 5, pp. 1020–1048, September 1994.

Preliminary version: "Counting Networks and Multi-Processor Coordination," *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing,* pp. 348–358, May 1991.

[6] A. Barak and A. Shiloh, "A Distributed Load Balancing Policy for a Multicomputer," *Software Practice and Experience,* Vol. 15, No. 9, pp. 901–913, September 1985.

[7] K. E. Batcher, "Sorting Networks and their Applications," *Proceedings of AFIPS Spring Joint Computer Conference,* Vol. 32, pp. 307–314, 1968.

[8] H. Brit, S. Moran and G. Taubenfeld, "Public Data Structures: Counters as a Special Case," *Proceedings of the 3rd Israel Symposium on Theory of Computing and Systems,* pp. 98–110, January 1995.

[9] R. A. Brualdi and H. J. Ryser, Combinatorial Matrix Theory, *Encyclopedia of Mathematics and its Applications,* Vol. 21, Cambridge University Press, 1991.

[10] C. Busch, N. Hardavellas and M. Mavronicolas, "Contention in Counting Networks," *Proceedings of the 13th Annual ACM Symposium on Principles of Distributed Computing,* pp. 404, August 1994.

[11] C. Busch and M. Mavronicolas, "A Combinatorial Treatment of Balancing Networks," *Proceedings of the 13th Annual ACM Symposium on Principles of Distributed Computing,* pp. 206–215, August 1994.

[12] C. Busch and M. Mavronicolas, "Proving Correctness for Balancing Networks," *DIMACS Series on Discrete Mathematics and Theoretical Computer Science,* Vol. 22 ("Parallel Processing of Discrete Optimization Problems," P. M. Pardalos, K. G. Ramakrishnan and M. G. C. Resende eds.), pp. 1–32, American Mathematical Society, July 1995.

[13] C. Busch and M. Mavronicolas, "Impossibility Results for Threshold Networks," Technical Report FORTH-ICS/TR-137, Institute of Computer Science, Foundation for Research and Technology – Hellas, September 1995. Submitted for publication.

[14] C. Busch and M. Mavronicolas, "Comparator Versus Balancer Merging Networks," in preparation.

[15] T. Cormen, C. Leiserson and R. Rivest, *Introduction to Algorithms,* Mc-Graw Hill and MIT Press, 1990.

[16] M. Dowd, Y. Perl, L. Rudolph and M. Saks, "The Periodic Balanced Sorting Network," *Journal of the ACM,* Vol. 36, No. 4, pp. 738–757, October 1989.

[17] C. Dwork, M. Herlihy and O. Waarts, "Contention in Shared Memory Algorithms," *Proceedings of the 25th Annual ACM Symposium on Theory of Computing,* pp. 174–183, May 1993.

[18] C. S. Ellis and T. J. Olson, "Algorithms for Parallel Memory Allocation," *Journal of Parallel Programming,* Vol. 17, No. 4, pp. 303–345, August 1988.

[19] E. W. Felten, A. LaMarca and R. Ladner, "Building Counting Networks from Larger Balancers," Technical Report 93-04-09, Department of Computer Science and Engineering, University of Washington, April 1993.

[20] E. Freudenthal and A. Gottlieb, "Process Coordination with Fetch-and-Increment," *Proceedings of the 4th International Conference on Architectural Support for Programming Languages and Operating Systems,* 1991.

[21] S. J. Garland and J. V. Guttag, "A Guide to LP, the Larch Prover," Technical Report # 82, DEC Systems Research Center, December 1991.

[22] A. Gottlieb, B. D. Lubachevsky and L. Rudolph, "Basic Techniques for the Efficient Coordination of Very Large Numbers of Cooperating Sequential Processors," *ACM Transactions on Programming Languages and Systems,* Vol. 5, No. 2, pp. 164–189, April 1983.

[23] N. Hardavellas, D. Karakos and M. Mavronicolas, "Notes on Sorting and Counting Networks," *Proceedings of the 7th International Workshop on Distributed Algorithms (WDAG-93),* Lecture Notes in Computer Science, Vol. # 725 (A. Schiper, ed.), Springer-Verlag, pp. 234–248, Lausanne, Switzerland, September 1993.

[24] D. Hensgen, R. Finkel and U. Manber, "Two Algorithms for Barrier Synchronization," *International Journal of Parallel Programming,* Vol. 17, No. 1, pp. 1–17, 1988.

[25] M. Herlihy, B.-C. Lim and N. Shavit, "Low Contention Load Balancing on Large-Scale Multiprocessors," *Proceedings of the 4th Annual ACM Symposium on Parallel Algorithms and Architectures,* pp. 219–227, July 1992.

[26] M. Herlihy, N. Shavit and O. Waarts, "Low Contention Linearizable Counting Networks," *Proceedings of the 32nd Annual IEEE Symposium on Foundations of Computer Science,* pp. 526–535, October 1991.

[27] M. Klugerman, "Small-Depth Counting Networks and Related Topics," Ph.D. Thesis, Department of Mathematics, Massachusetts Institute of Technology, September 1994.

[28] M. Klugerman and C. Plaxton, "Small-Depth Counting Networks," *Proceedings of the 24th Annual ACM Symposium on Theory of Computing,* pp. 417–428, May 1992.

[29] D. Knuth, *The Art of Computer Programming,* Volume 3 (*Sorting and Searching*), Addison-Wesley, 1973.

[30] C. P. Kruskal, L. Rudolph and M. Snir, "Efficient Synchronization on Multiprocessors with Shared Memory," *Proceedings of the 5th Annual ACM Symposium on Principles of Distributed Computing,* pp. 218–228, August 1986.

[31] K. Lorys, M. Kutylowski, B. Oesterdiekhoff and R. Wanka, "Fast and Feasible Periodic Sorting Networks of Constant Depth," *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science,* pp. 369–380, October 1994.

[32] J. M. Mellor-Crummey and M. L. Scott, "Algorithms for Scalable Synchronization on Shared Memory Multiprocessors," Technical Report 342, Department of Computer Science, University of Rochester, April 1990.

[33] I. Parberry, "Single-Exception Sorting Networks and the Computational Complexity of Optimal Sorting Network Verification," *Mathematical Systems Theory,* Vol. 23, pp. 81–93, 1990.

[34] S. Moran and G. Taubenfeld, "A Lower Bound on Wait-Free Counting," *Proceedings of the 12th Annual ACM Symposium on Principles of Distributed Computing,* pp. 251–259, August 1993.

[35] L. M. Ni, C. W. Xu and T. B. Gendreau, "A Distributed Drafting Algorithm for Load Balancing," *IEEE Transactions on Software Engineering,* SE-11(10), October 1985.

[36] D. Peleg and E. Upfal, "The Token Distribution Problem," *SIAM Journal on Computing,* Vol. 18, pp. 229–241, 1989.

[37] L. Rudolph, M. Slivkin and E. Upfal, "A Simple Load Balancing Scheme for Task Allocation in Parallel Machines," *Proceedings of the 3rd Annual ACM Symposium on Parallel Algorithms and Architectures,* pp. 237–245, July 1991.

[38] H. S. Stone, "Database Applications of the Fetch-and-Add Instruction," *IEEE Transactions on Computers,* Vol. C-33, No. 7, pp. 604–612, July 1984.

[39] Y.-T. Wang and R. J. Morris, "Load Sharing in Distributed Systems," *IEEE Transactions on Computers,* C-34(3), March 1985.