Sandpile Models and Lattices: A Comprehensive Survey.

Éric Goles [1], **Matthieu Latapy** [2,3],
Clémence Magnien [3]
Michel Morvan [3] **and Ha Duong Phan** [3].

Abstract: Starting from some studies of (linear) integer partitions, we noticed that the lattice structure is strongly related to a large variety of discrete dynamical models, in particular sandpile models and chip firing games. After giving an historical survey of the main results which appeared about this, we propose a unified framework to explain the strong relationship between these models and lattices. In particular, we show that the apparent complexity of these models can be reduced, by showing the possibility of symplifying them, and we show how the known lattice properties can be deduced from this.

Keywords: Sandpile Models, Chip Firing Games, Lattices, Integer Partitions, Discrete Dynamical Models.

1 Background.

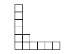
Given an integer n, a (*linear*) *partition* of n is a (weakly) decreasing sequence of positive integers, called the *parts* of the partition, such that the sum of all the parts is equal to n. A partition p of n is denoted by $(p_1, p_2, \ldots, p_k)$, where each p_i is a part, with $p_i \geq p_{i+1}$ for all i, $p_k > 0$, and $\sum_{i=0}^{k} p_i = n$. The integer k is called the *length* of the partition, and the integer p_1 is called the *height* of the partition. A partition of height at most h and length at most l is said to be included in a $h \times l$ *box*. Integer partitions are very classical objects of combinatorics, and many studies about their different aspects appeared [Mac16, And76, Sta97].

Given a partition p of n, there exists a classical representation of p called the *Ferrer diagram* of p: it consists in a series of columns of stacked squares such that the i-th column (from left to right) contains p_i squares, for each i. It is therefore a decreasing sequence of columns of stacked squares, which contains exactly n squares. For example, if one considers the two partitions $p = (4, 3, 3, 2)$

[1]Departamento de Ingeniería Matemática, Escuela de Ingniería, Universidad de Chile, Casilla 170-Correo 3, Santiago, Chile.

[2]INRIA Rocquencourt, B.P. 105 78153 Le Chesnay Cedex France, `Matthieu.Latapy@inria.fr`

[3]LIAFA, Université Paris 7, 2, place Jussieu, 75251 Paris Cedex 05, France. `(latapy,magnien,morvan,phan)@liafa.jussieu.fr`

and $q = (6, 2, 1, 1, 1, 1)$ of $n = 12$, then one obtains the diagrams  and  respectively. Notice that these diagrams can be viewed as (halves of) profiles of sand piles, which we will see is indeed confirmed by physical studies. Therefore, we will call each square is a *grain*, and we will say that sometimes one grain may *fall* from one column to another.

A binary relation $\leq$ over a set S is said to be an *order* if it is reflexive (for all x in S, $x \leq x$), transitive ($x \leq y$ and $y \leq z$ imply $x \leq z$) and anti-symmetric ($x \leq y$ and $y \leq x$ imply $y = x$). The set S together with the relation $\leq$ is then called a *partially ordered set*, or simply an *order*. If $x \leq y$ is an order, we say that y is *greater* than x, or equivalently that x is *smaller* than y. If $x \leq y$ and $x \neq y$ then we write $x < y$. An element x is *covered* by another element y if $x \leq y$ and if $x < z \leq y$ implies $y = z$. We then say that y is an *upper cover* of x, and x is a *lower cover* of y. In other words, y is strictly greater than x and there is no element in between. An order O is generally represented by a *Hasse diagram*: a point p_x of the plane is associated to each element of O, such that if $x \leq y$ then p_x is lower than p_y, and there is a line between p_x and p_y if and only if x is covered by y.

An ordered set L is a *lattice* if any two elements x and y of L have a greatest lower bound, called the *infimum* of x and y and denoted by $x \wedge y$, and a smallest greater bound, called the *supremum* of x and y and denoted by $x \vee y$. The infimum of x and y is nothing but the greatest element among the ones which are lower than both x and y. The supremum is defined dually. Notice that any finite lattice has a unique minimal and a unique maximal element. Indeed, if it contained two minimal elements, then they would not have an infimum and so the set could not be a lattice (the same holds for the maximal element). The study of lattices is an important part of order theory, and many results about them exist. In particular, various classes of lattices have been defined and appear in computer science, mathematics, physics, social sciences, and others. For more details about orders and lattices, we refer to [DP90].

A lattice L is *distributive* if it satisfies the two following distributivity relations:

$$\forall x, y, z \in L,\ x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

$$\forall x, y, z \in L,\ x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

A lattice is a *hypercube of dimension* n if it is isomorphic to the set of all the subsets of a set of n elements, ordered by inclusion. Hypercubes are also called *boolean lattices*. A lattice is *upper locally distributive* (denoted by *ULD* [Mon90]) if the interval between any element and the supremum of all its upper covers is a hypercube. *Lower locally distributive* (LLD) lattices are defined dually. Notice that a distributive lattice is a lattice that is at the same time upper and lower locally distributive: the intervals between any element and, on the one hand the supremum of all its upper covers, and on the other hand the infimum of all its lower covers, are both hypercubes. Distributive and ULD lattices have a great importance in the studies of the models we present in this paper, and in lattice theory in general.

Before entering in the core of this paper, let us give a precise definition of what we call a *discrete dynamical model*. At each (discrete) time step, such a model is in some state, which we call a *configuration*. Configurations are described by combinatorial objects, like graphs, integer partitions, and others, and we will not distinguish a configuration and its combinatorial description. A discrete dynamical model is then defined by an initial configuration and an evolution rule which says under which conditions the configuration may be changed, and which describes the new configurations one may obtain. This rule can generally be applied under a local condition, and it implies a local modification of the current configuration. Notice that in the general case the evolution rule can be applied in several places in a configuration, leading to several configurations. If a configuration c' can be obtained from a configuration c after one application of the evolution rule, we say that c' is a *successor* of c, or c is a *predecessor* of c', which is denoted by $c \longrightarrow c'$. We generally consider the set of all the reachable configurations of a given model, together with the predecessor relation, and we call it the *configuration space* of the considered model. If the model always reaches the same fixed point (configuration from which the evolution rule cannot be applied), we say that it is *convergent*.

Notice that, if there is no cycle in the configuration space, then the reflexive and transitive closure of the predecessor relation defines an order between the reachable configurations: c is smaller than c' if and only if c' can be obtained from c by a sequence of applications of the evolution rule. In this case we will use the Hasse diagram to represent the configuration space: the initial configuration is at the bottom of the diagram, and its successor are above it and linked to it by a line segment. The study of the orders induced over combinatorial objects by discrete dynamical models is an active area of research, which has already made it possible to obtain many results.

Note 1.1 Most of the works about discrete dynamical models and orders actually deal with the order induced by the *successor* relation instead of the one induced by the predecessor relation. This order is the *dual* of the one we use here, *i.e.* the order is flipped upside-down. Indeed, the classical convention in discrete dynamical studies is to put the initial configuration on the top of the drawing, and the final configuration on the bottom. We have chosen to do the opposite because it is more natural for the use of order theory. This does not change in any way the results presented here.

The fact that any finite lattice has a unique maximal element (as noticed above) implies directly that, if the configuration spaces of a discrete dynamical model are lattices then the model always reaches a unique final configuration (*i.e.* it *converges*). But the notion of convergence implied by lattices is stronger: the fact that a configuration space is a lattice not only implies that any configuration will lead to the same final configuration, but also that given any two configurations there is a unique first configuration reachable from both of them (which is their supremum). This notion of convergence gives in itself much information about the studied model, and completes the classical notions of convergence like *strong convergence* [Eri93].

Moreover, the fact that a configuration space is a lattice makes it possible to use the many codings and algorithms known about lattices and special classes of lattices [Ber98]. For example, there exists a generic algorithm which, given any distributive lattice, gives a random element of this set with the uniform distribution [Pro98]. Since most of the models we study are models of physical objects, the possibility of sampling a configuration with the uniform distribution is crucial: it makes it possible to study the entropy of the system, and it gives an idea of what the modelized object will look like in the nature.

In this paper, we give a survey of known results concerning the presence of lattices in the context of discrete dynamical models derived from studies of sandpiles. Indeed, during the last ten years, many results showing that a given model induces lattices appeared in the litterature. We show in the last section of this paper how some of these results can be unified in the framework of *simple* Chip Firing Games, and how some properties of this model explain the properties already noticed in the case of other discrete dynamical models.

2 Historical context.

A very classical family of lattices in combinatorics is the Young lattices family. Given two integers h and l, the Young lattice $L(h, l)$ is the set of all the partitions included in the $h \times l$ box, ordered componentwise: $p \leq q$ in $L(h, l)$ if and only if for all i, $p_i \leq q_i$. This ordered set is a (distributive) lattice [Ber71], the infimum of two partitions p and q being the partition r defined by $r_i = \min(p_i, q_i)$, and the supremum being s defined by $s_i = \max(p_i, q_i)$. Moreover, $L(h, l)$ can be viewed as the configuration space of the following discrete dynamical model: the initial configuration is the empty partition (), which is included in the $h \times l$ box for any h and l. The successors of a partition p are the partitions obtained from p by adding one grain on one column, under the condition that we still obtain a partition, and that it remains included in the $h \times l$ box. See Figure 1 for an example. Notice that this is equivalent to the Dyck lattice, *i.e.* the lattice of the paths from $(0, 0)$ to (l, h) on a planar grid, with only vertical and horizontal steps. These lattices have been widely studied, and can be generalized to other kinds of integer partitions, as shown for example in [Lat00]. They are also related to some special kinds of

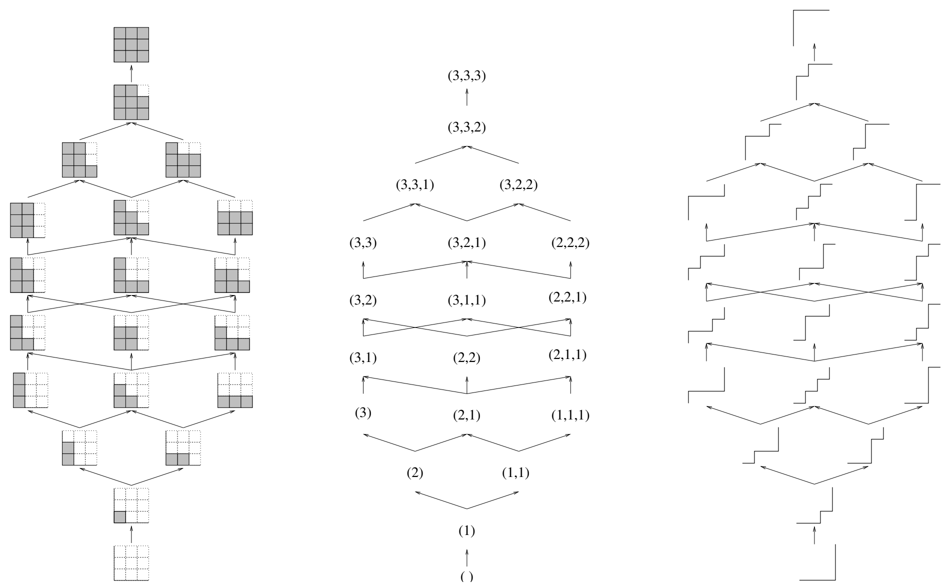


Figure 1: The (distributive) lattice $L(3,3)$. From left to right: the representation by Ferrer diagrams, the representation by k-uplets, and the Dyck paths equivalent.

tilings, but this is outside the scope of this paper.

In 1973, Brylawski studied the set of partitions of an integer n together with the following order, known as the *dominance ordering*:

$$p \geq q \text{ if and only if } \sum_{j=1}^{i} p_j \leq \sum_{j=1}^{i} q_j \text{ for all } i.$$

In other words, a partition p is greater than a partition q if the i-th prefix sum of p is smaller than the i-th prefix sum of q for all i. In [Bry73], Brylawski proved that this order is a lattice, denoted by $L_B(n)$. Moreover, he proved that the lattice $L_B(n)$ can be viewed as the configuration space of a discrete dynamical model defined as follows. The configurations of the model are (the Ferrer diagrams of) the partitions of n, the initial one being the partition (n) (or equivalently a stack of n grains). The model has two evolution rules: the vertical and the horizontal one.

- **Vertical rule:** a grain can fall from column i to column $i+1$ if the height difference between the i-th column and the $(i+1)$-th one is at least two. In other words, $p \longrightarrow q$ if and only if there exists an integer i such that $p_i - p_{i+1} \geq 2$, $q_i = p_i - 1$, $q_{i+1} = p_{i+1} + 1$, and for all $k \notin \{i, i+1\}$, $q_k = p_k$. Notice that this is equivalent to say that a grain can fall from column i to column $i+1$ if the series of columns remains (weakly) decreasing.

- **Horizontal rule:** a grain can slip from column i to column j if $i < j$ and the height difference between these two columns is exactly 2, and the height difference between the i-th and each of the columns between the i-th and the j-th is exactly 1. In other words, $p \longrightarrow q$ if and only if there exists an integer i and an integer j such that for all $i < k < j$, $p_k = p_i - 1 = p_j + 1$, $q_i = q_k = q_j = p_k$, and for all $k \notin \{i, j\}$, $q_k = p_k$.

These evolution rules are described in Figure 2, and the configuration space $L_B(7)$ is shown in Figure 3. Brylawski proved that any partition of n can be obtained from (n) by iterating these rules, and that the order induced by the evolution rule is nothing but the dominance ordering. Moreover, he gave an explicit formula for the supremum:

$$\sup(p, q) = r \text{ if and only if for all } j\text{: } \sum_{i=1}^{j} r_i = min(\sum_{i=1}^{j} p_i, \sum_{i=1}^{j} q_i).$$

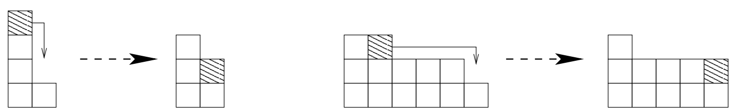


Figure 2: *The rules of the Brylawski model. Left: the vertical rule. Right: the horizontal rule.*

An important restriction of the model of Brylawski has been introduced later [GK93]. This model, called Sand Pile Model (SPM), is defined exactly like the Brylawski model, except that the horizontal rule is not allowed. The configuration space obtained starting from a column of n grains is denoted by $SPM(n)$. An example is shown in Figure 3. SPM appeared as a paradigm for the physical phenomenon called Self-Organized Criticality (SOC) [Jen98, Tan93]. It has been used to study avalanches (the size of real avalanches obeys the same laws as the avalanches in SPM [Tan93]), and profiles of dunes [Bak97]. It is also related to distributed computing problems, as shown in [DKVW95]. Here, we will only consider SPM as an abstract model, its configurations being integer partitions. In [GK93] it was proved that $SPM(n)$ is always a sub-order of $L_B(n)$. Therefore, the order relation between the partitions in $SPM(n)$ is nothing but the dominance ordering defined above. Goles and Kiwi proved in [GK93] that $SPM(n)$ is a lattice, and that the formula for the supremum is the same as the one for $L_B(n)$, given above. Moreover, a characterization of the elements of $SPM(n)$ is given in [GMP98b]. One may notice that $SPM(n)$ and $L_B(n)$ share a large set of properties. However, they also have many differences. We will detail these later, but we can already notice that it is proved in [GMP98b] that all the sequences of applications of the rules from the initial configuration to the final one have the same length in $SPM(n)$, which is clearly not true for $L_B(n)$ (see Figure 3). Some other

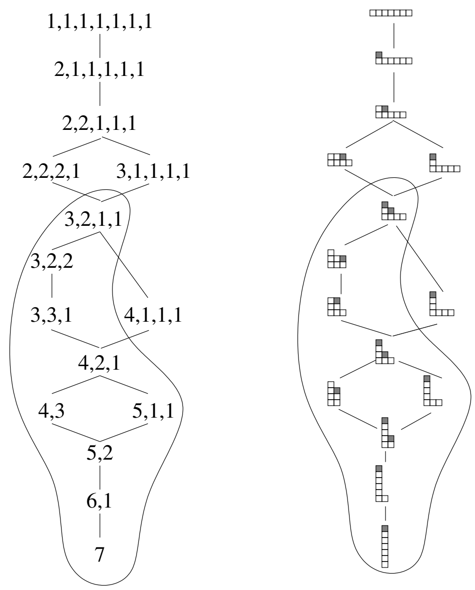


Figure 3: *The lattice of all the partitions of 7, namely $L_B(7)$. If we restrict the model to the vertical rule, we obtain the outlined part, which is nothing but $SPM(7)$.*

works gave more informations on the structure of these lattices. In particular, it is shown in [LMMP01] and [LP99] that both $SPM(n)$ and $L_B(n)$ have a self-similar structure and that a tree can be associated to these sets. Recursive formulae are given for the cardinals of these lattices, as well as infinite extensions of the model (leading to infinite lattices).

The surprising fact that all the configuration spaces of the Young model, the Brylawski model and SPM all are lattices was then noticed and the question of how much one can modify these models without breaking this property arised. A series of variations of these models has then been introduced to answer this question. The first of them was the Ice Pile Model: a grain can slip from a column i to the column j like in the Brylawski model, but only if $j - i$ is below a given value k (the length of the horizontal moves is bounded by k) [GMP98b]. The configuration space of the model started with a column of n stacked grains is then denoted by $IPM(n, k)$. An example is shown in Figure 4. In [GMP98b], it is proved that $IPM(n, k)$ is always a sub-order of $L_B(n)$. Again, the model induces a lattice [GMP98b]. This model can be viewed as a generalization of the Brylawski model as well as a generalization of SPM: $L_B(n)$ is nothing but $IPM(n, n)$, and $SPM(n)$ is nothing but $IPM(n, 0)$. Another generalization of SPM has then been introduced: $L(n, \theta)$ is the configuration space obtained from a column of n grains when a grain can move from column i to column $i + 1$ if the height difference between the two columns is at least equal to θ [GMP98b]. Therefore, $SPM(n)$ is nothing but $L(n,2)$. Notice that θ may be negative, which makes it possible for the grains to go up (in this case, we do not obtain partitions of n anymore, but *compositions* of n, the length of which is restricted to n to avoid infinite moves on the right). An example is given in Figure 4. Again, the sets $L(n,\theta)$ are lattices for any n and θ [GMP98b].

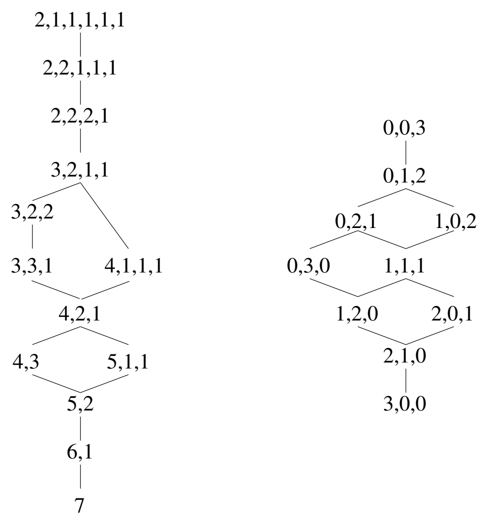


Figure 4: *The configuration spaces $IPM(7,2)$ (left), and $L(3,-1)$ (right). Notice that the order $IPM(7,2)$ is a sub-order of $L_B(7)$ shown in Figure 3, which is the case for any n and k.*

These two models were natural extensions of the Brylawski model and of SPM. They were more general, but the lattice property was still preserved. Therefore, the investigation continued with stronger modifications of the models. The first idea has been to allow multiple grains to fall at each time step, leading to the model $CFG(n,m)$: starting from an initial column of n grains, m grains can fall from column i to columns $i+1$, $i+2$, $\ldots$, $i+m$ (each of them receiving one grain) if the height difference between column i and $i+1$ is strictly greater than m. See Figure 5 for an illustration. Clearly, $SPM(n)$ is nothing but $CFG(n,1)$. Again, the obtained configuration spaces are lattices [GMP98c]. An example is given in Figure 6 (left).

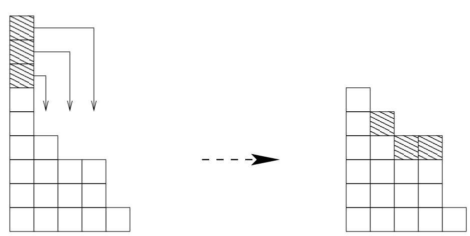


Figure 5: *The evolution rule used to obtain $CFG(n,m)$, when $m=3$.*

Another idea to modify the behaviour of the models was to consider that the grains move on a ring (rather than on a line): they can fall from the n-th column to the first one. Such a variation of SPM, called the *Game of Cards*, has been introduced in [DKVW95] and studied from the lattice point of view in [GMP98a]. The game is very simple: it is composed of k players disposed around a table, and each player can give a card to his/her right neighbour if he/she has more cards than him/her. Initially, one player has all the cards. An example is given in Figure 6 (right). It is shown that, when the model is convergent, it generates a lattice, and the initial configurations which make it convergent are characterized. Moreover, it is shown that, when the model does not converge, the lattice structure is still present under a slightly modified form [GMP98a].

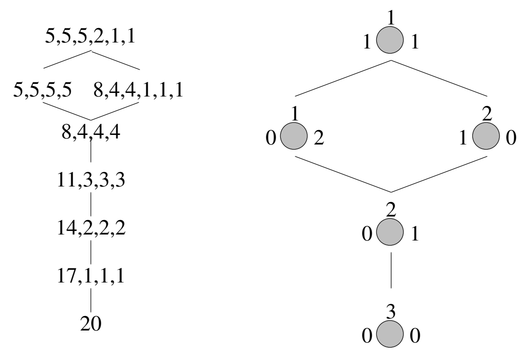


Figure 6: *Left: the configuration space $CFG(n, m)$ when $n = 20$ and $m = 3$. Right: an example of Game of Cards, with 3 players and 3 cards (the shaded disk represents the table).*

Another similar model was introduced in [Lat01] to study some other kinds of integer partitions: given two integers n and b, a b-ary partition of n is a k-uplet $(p_0, p_1, \ldots, p_{k-1})$ such that $\sum_{i=0}^{k-1} p_i \cdot b^i = n$. The configurations of this model are the b-ary partitions of n, and the evolution rule says that a b-ary partition can be transformed into another one by decreasing its i-th component by b (if it is at least equal to b) and increasing its right neighbour by 1. The obtained configuration space is denoted by $R_b(n)$, and it is a (distributive) lattice [Lat01]. See Figure 7 for some examples.

It appeared in these studies that the fact that the considered discrete dynamical models induce lattice structures over their configuration spaces is a very stable property. Notice however that some natural ideas to extend SPM and the Brylawski model do not preserve the lattice structure. In particular, two dimensionnal generalizations (the grains move on a planar grid), which seem interesting for the study of planar partitions, do not preserve the lattice structure. Therefore, we wondered if one could define a general model having this properties, which would explain how and when it appears. The first step to answer this question was to explore the other models defined in the litterature which induce the lattice structure,

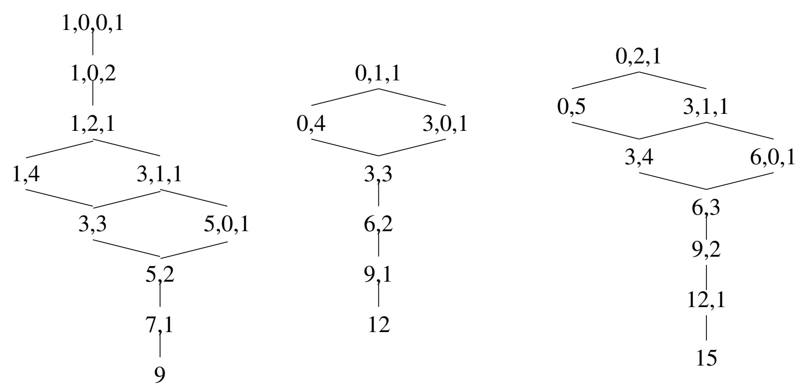


Figure 7: *Examples of (distributive) lattices of the b-ary partitions of an integer. From left to Right:* $R_2(9)$, $R_3(12)$ *and* $R_3(15)$.

and then try to determine some general characteristics which may be responsible for this property. We will now present shortly the variety of models known in the litterature, and the next sections will be devoted to the explanation of these properties.

The Edge Firing Game (EFG), also called the source reversal game, has been defined in various contexts [MKM78, Pre86a, Pre86b]. Given an undirected graph $G = (V, E)$, one defines an *orientation* of G as a directed graph $G' = (V', E')$ such that $V' = V$ and $\{v, v'\} \in E$ implies either $(v, v') \in E'$ or $(v', v) \in E'$. The configurations of an EFG are orientations of a given graph with a distinguished vertex, and the evolution rule is the following: if a vertex that is not the distinguished vertex has no incoming edge, then we can reverse all its (outgoing) edges. Again, it is shown in [Pro93] that the configuration space of any EFG is a (distributive) lattice. See Figure 8 for an example.

During the same period, the physicists studied the Abelian Sandpile Model (ASM) [DM90, DRSV95] introduced in [BTW87]: the model is defined over a finite two-dimensional grid, each cell containing a number of grains. The evolution rule then says that a cell which contains at least four grains can give one of them to each of its four neighbours. Therefore, its number of grains is decreased by four. If the cell is on the border of the grid, then some grains may fall to the exterior, which simply stores the grains it receives. See Figure 9 (left) for an example. This model has many important properties, and has mainly been studied from the algebraic point of view [DRSV95]. It has been extended by Cori and Rossin in [CR00]: a number of gains is associated to each vertex of a given undirected connected graph with a special vertex called the *sink*. Any vertex except the sink can give a grain to each of its neighbours if it contains sufficiently many grains (*i.e.* at least as many grains as its degree). See Figure 9 (right) for an example. The algebraic properties of the original model are preserved, and this generalization received much attention since then. For a survey of the different studies concerning the algebraic properties

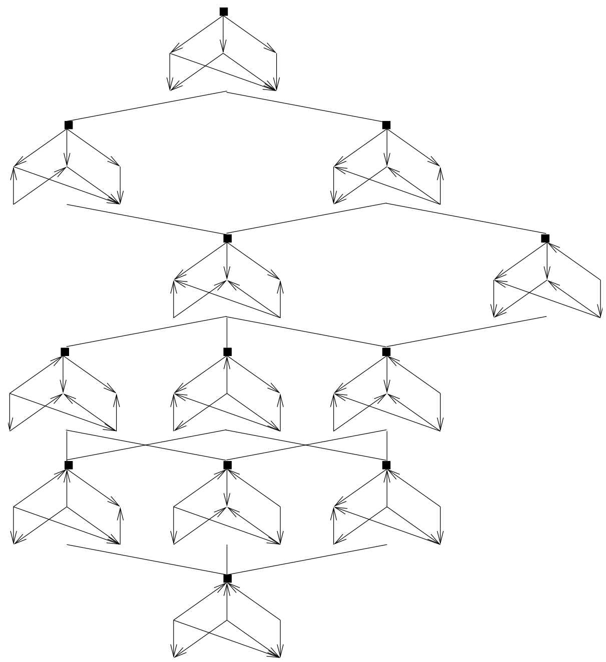


Figure 8: *An example of Edge Firing Game. The distinguished vertex is marked with a black square.*

of ASMs, we refer to [Dha98] and [IP98]. A directed extension, very close to the Chip Firing Game defined below, has been studied in social science by Biggs [Big97, Big99, Heu99]. The same kind of algebraic studies have been done on this model, showing similar properties.

Independently, Björner, Lovász and Shor introduced the Chip Firing Game (CFG) in [BLS91, BL92]. It is defined over a directed (multi)graph as follows: a *configuration* of the game is a distribution of chips on the vertices of the graph, and a configuration can be transformed into another one by transferring a chip from one vertex along each of its outgoing edges, if it contains at least as many chips as its outgoing degree. See Figure 10 for an example. Convergence conditions (involving the number of chips or the structure of the graph) are given in [BLS91, BL92, LP01], as well as different proofs of the fact that the configuration space of any convergent CFG is a lattice. Notice that the ASM can be viewed as a special case of the CFG (concerning the configuration spaces), which implies that any ASM induces a lattice. Actually, we will see in the next section that most of the models we have presented here are special cases of CFG, and we will explain in Section 4 how the lattice property can be understood as a consequence of a stronger property of Chip Firing Games.

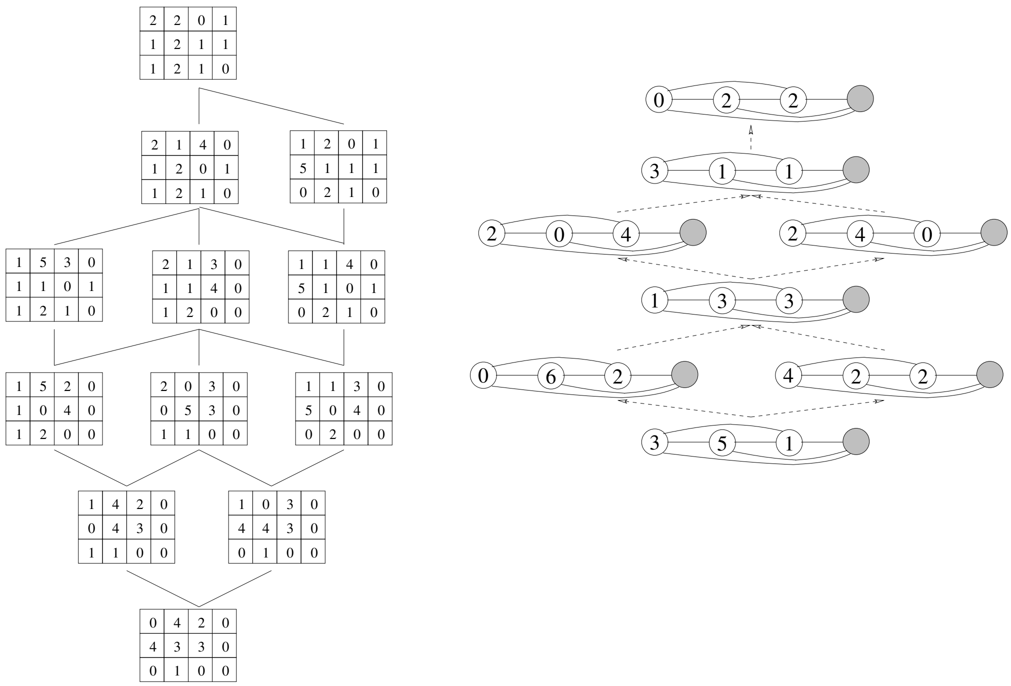


Figure 9: *Left: an example of the original Abelian Sandpile Model on a* 4×3 *grid. Right: an example of the generalized Abelian Sandpile Model on a graph (the sink is the shaded vertex).*

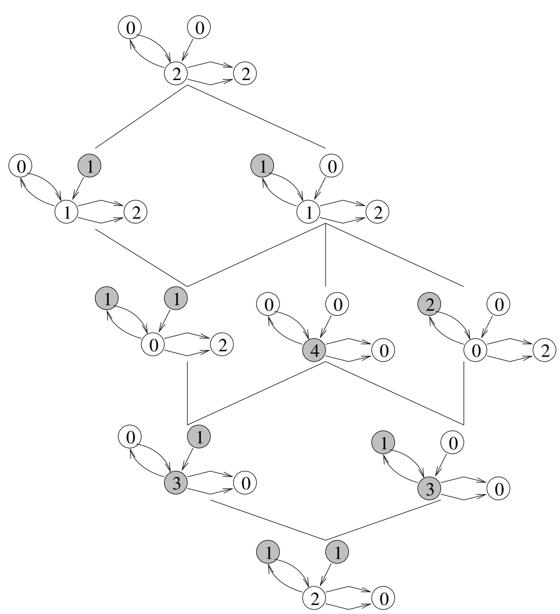


Figure 10: *The configuration space of a CFG.*

3 The Chip Firing Game as a general model.

In this section, we show how most of the models presented in the previous section are actually special cases of Chip Firing Games, which implies that some of their properties (in particular the fact that their configuration spaces are lattices) can be deduced from properties of Chip Firing Games. To achieve this, we will give for each instance of a model an instance of a Chip Firing Game such that its configuration space is isomorphic to the one of the original model. We will not give the details of the proofs of these isomorphisms: they are obvious from the construction of each simulation. Since it is known from [BLS91, BL92, LP01] that the configuration space of any convergent CFG is a lattice, and even an Upper Locally Distributive (ULD) lattice, we obtain as corollaries the known results about the lattice structures of the configuration spaces of all these models, adding the fact that they are *ULD* lattices. This makes it possible to understand the fact that a large variety of models induce lattices as a consequence both of the expressivity power of CFG (many models can be simulated by a CFG), and of some strong properties of CFG (they always induce ULD lattices).

The Young lattice $L(h, l)$ can be obtained as the configuration space of the CFG defined over $G = (V, E)$ with $V = \{1, 2, \ldots, l\}$ and $E = \{(i, i + 1) \mid 1 \leq i \leq l - 1\}$. To a partition p in $L(h, l)$, we associate the configuration of the CFG where vertex i contains $p_{i+1} - p_i$ chips (see Figure 11). Notice that this model can also be simulated by an EFG as follows: let us consider the decreasing boundary of the Ferrer diagram of a partition p in $L(h, l)$. This boundary contains exactly l horizontal step and h vertical ones. Now, let us replace each horizontal step by an edge directed from left to right, and each vertical step by an edge directed from right to left. See Figure 11 for an example. One can easily check that running this EFG is equivalent to the Young model we started with.

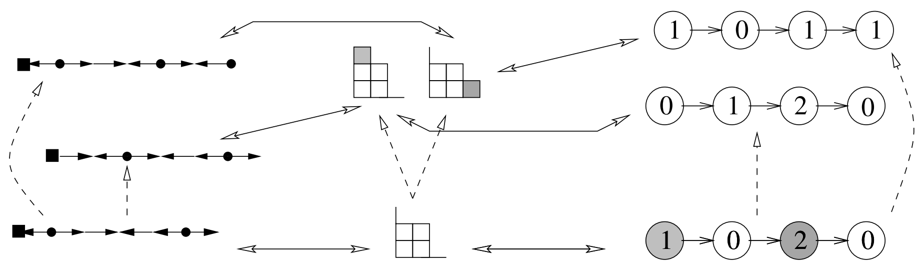


Figure 11: *In the middle, a transition in the Young model, simulated left by an EFG and right by a CFG.*

SPM can be encoded as a CFG in the following way: let n be the number of grains in the system. Then, consider the graph $G = (V, E)$ where $V = \{0, 1, \ldots, n\}$ and $E = \{(i, i + 1) | 1 \leq i \leq n - 1\} \cup \{(i, i - 1) | 1 \leq i \leq n\}$. We associate to each partition p in $SPM(n)$ the following repartition of chips on this graph, denoted by $\pi(p)$: the vertex number i contains $p_i - p_{i+1}$ chips. Now, if we play the CFG defined over G with configuration $\pi(p)$ for a given p in $SPM(n)$, it is clear that the successors of this configuration are the elements of $\{\pi(p'),\, p \longrightarrow p' \text{ in } SPM(n)\}$. See Figure 12 for an illustration of this. Therefore, if we play the CFG on this graph starting from the configuration $\pi((n))$, we obtain a configuration space isomorphic to $SPM(n)$. This coding was first developed in [GK93]. Notice that it is easy to reconstruct a configuration p in $SPM(n)$ from a configuration of the CFG.

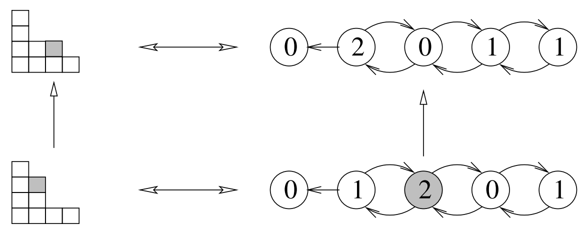


Figure 12: *Coding of the Sand Pile Model with a Chip Firing Game.*

$L(n,\theta)$ can be encoded as a CFG in the same way, except that each vertex of the CFG contains $p_i - p_{i+1} - \theta + 2$ chips if p is the corresponding configuration of $L(n,\theta)$ (see Figure 13).

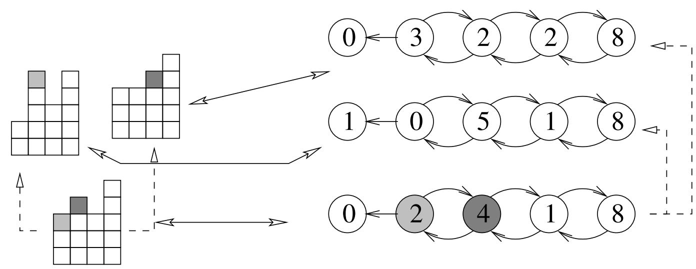


Figure 13: *Coding of $L(n,\theta)$ with a Chip Firing Game when $\theta = -1$.*

The underlying (multi-)graph of the CFG that simulates $CFG(n,m)$ is different: $V = \{0, 1, \ldots, n\}$ and each vertex i has m outgoing edges $(i, i-1)$ and another outgoing edge $(i, i+m)$. See Figure 14 for an illustration. A configuration p of $CFG(n,m)$ is then equivalent to a configuration of the CFG where vertex i contains $p_i - p_{i+1}$ chips for each i.

The Game of Cards can be simulated by the following CFG. Its graph is a ring of k vertices: the i-th vertex has an outgoing edge to vertex $i+1$ modulus k and another one to $i-1$ modulus k. Then, a configuration c of the game is encoded by a configuration of the CFG where vertex i contains as many chips as the difference between the number of cards of player i and the number of cards of its right neighbour plus 1. Notice that this coding is quite different from the previous ones, since the graph of the obtained CFG is a cycle.

To obtain a configuration space isomorphic to $R_b(n)$, one has simply to consider the CFG defined over the following multigraph. The vertex set is $V = \{0, 1, \ldots, n\}$,

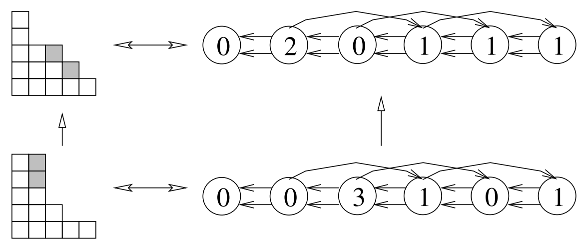


Figure 14: *Coding of $CFG(n,m)$ with a Chip Firing Game when $m = 2$.*

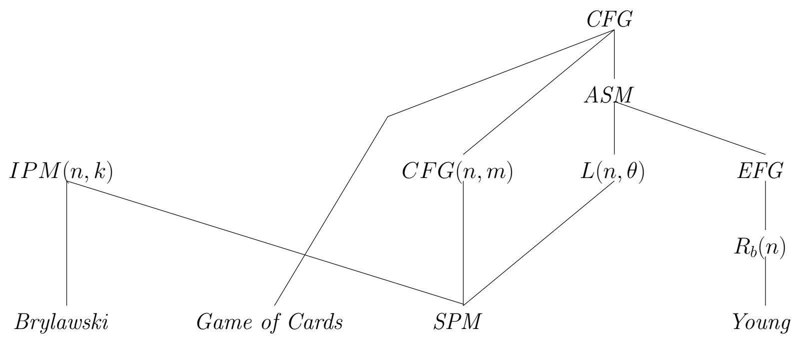


Figure 15: The diagram of the simulations between the models we have discussed. The most general models are on the top, while the more specific ones are on the bottom. Notice that almost all the models we have presented can be simulated by a CFG.

vertex i for $1 \leq i \leq n-1$ having $b-1$ outgoing edges to vertex 0 and one outgoing edge to $i+1$. If one starts this CFG from the configuration where vertex 1 contains n chips, all the other ones being empty, then it is clear that the obtained configuration space is isomorphic to $R_b(n)$.

As already noticed, any ASM can be simulated by a CFG. The simulation of an Edge Firing Game with a CFG is less obvious. Let us consider an EFG defined over the undirected graph $G = (V, E)$ with distinguished vertex ν, and with the initial orientation O. It is clear that the configuration space of the following ASM is isomorphic to the one of the EFG: the ASM is defined over G with sink ν, and its initial configuration is the one where each vertex v contains as many grains as the number of outgoing edges it has in O. Since any ASM can be simulated by a CFG, any EFG can itself be simulated by a CFG.

We can summarize the simulations results given in this section by the diagram of Figure 15. This is the diagram of the order over the models we have cited above, defined as follows: a given model is smaller than another if the former can be simulated by the latter. Notice that almost all the models we have presented can be simulated by a CFG.

On the other hand, let us emphasize on the fact that the general results on

CFGs can be used to prove that a given set is a lattice: it suffices to give a CFG such that its configuration space is isomorphic to the considered set. Likewise, one can prove that a given set is a distributive lattice by proving an isomorphism with the configuration space of an EFG. This technique has for example been applied in [BL01] in the context of tilings. This is a new and original proof technique, which is very interesting for the order theoretical point of view.

Notice that not all models presented in the previous section can be encoded as special CFGs. This can easily be seen because models like L_B induce lattices which are not ULD, but this can also be understood by studying the proof techniques used to show that these models induce lattices. On the one hand, the proofs that the Chip Firing Games and the models which can be encoded as CFGs induce lattices is based on the notion of *shot-vector*: for a CFG with vertex set $\{v_1, \ldots, v_n\}$, the shot-vector of a firing sequence s is the vector $(a_1, \ldots, a_n)$ such that, for all i, a_i is the number of times the vertex v_i is fired during the sequence s. It is proved in [LP01] that the configurations of a CFG and the shot-vectors of its firing sequences are in one-to-one correspondence, and that the order on the configurations corresponds to the componentwise order on the shot-vectors. This is the fundamental property which makes it possible to prove that the configuration spaces of these models are ULD lattices.

On the other hand, for models like L_B or IPM, the proof that they induce lattices uses an explicit formula for the upper bound of two given configurations. The lattices induced by these models are less structured than ULD lattices, but it is possible to give an explicit formula for the final configuration, as well as a characterization of all elements of the configuration space, and the length of the longest path from the initial to the final configuration.

4 The Simple Chip Firing Game

We have seen in the previous section that the Chip Firing Game can be viewed as a generalization of many other models. Therefore the study of CFGs takes a special importance, because any of its property is shared by these models, and a good understanding of CFGs will help understand the other ones. In this section we introduce a new notion about CFG, the *simple* CFG. We will see that any CFG is equivalent (in terms of configuration space) to a simple CFG. We use this result to give a new proof of the fact that the configuration space of any CFG is a ULD lattice in a natural and straightforward way. This shows how a good understanding of the CFG allows to state natural proofs about the model. Most of the results exposed in this part can be found in [MPV01].

Definition 4.1 *A convergent CFG is* simple *if each of its vertices is fired at most once during any firing sequence that, starting from the initial configuration, reaches* *the final configuration.*

Notice that any simple CFG is necessarily convergent. We will say that two CFGs are *equivalent* if their configuration spaces are isomorphic. In the sequel, we will denote by $L(C)$ the configuration space of any convergent CFG C. The next theorem states that any convergent CFG is equivalent to a simple one. This will allow the study of CFGs through the use of *simple* CFGs, without loss of generality.

Theorem 4.2 *Any convergent CFG is equivalent to a simple CFG.*

Proof : The idea of the proof is the following: if a CFG is not simple, then it contains a vertex a which is fired more than once between the initial and final configuration. We will replace a by two vertices a_1 and a_2. They will be fired alternatively, first a_1, then a_2, and so on, and one of them will be fired each time a was fired. Each of the new vertices a_1 and a_2 will be fired less often than a between the initial and the final configuration. Therefore, by iterating this process, we will eventually obtain a simple CFG.

Before giving the formal description of this transformation, we will explain two things: how vertex a can be replaced by two different vertices that will play its role, and how we can guarantee that the two vertices a_1 and a_2 will be fired alternatively. The way to replace a by two vertices is to split all the chips that are in a in the initial configuration, or will arrive in a through incoming edges, into two halves, and put one half in each vertex a_1 and a_2. This means that the initial configuration of a_1 and a_2 will be half of the initial configuration of a, and there will be half as much edges coming in a_1 and a_2 as in a. Of course this cannot always be done immediately because a might have an odd number of incoming edges (or contain initially an odd number of chips). Our first step is therefore to double everything in our CFG: chips and edges. We obtain then a new CFG, which we will call the double of the original CFG. It is clearly equivalent to our first CFG, and all the number of edges and chips are even. We can then distribute evenly the chips and incoming edges of a on a_1 and a_2.

Now for the outgoing edges: each firing of a_1 or a_2 must play the role of a firing of a for the other vertices. Therefore a_1 and a_2 must have as much outgoing edges as a (in the doubled CFG). This can seem to create a lack of chips in a_1 and a_2 (each of them has as much outgoing edges as a, but only half as much incoming edges), but this will be corrected by the process that guarantees that the two vertices are fired alternatively: let d be the initial outdegree of a, and let N be twice the number of chips in the original CFG. We place $N-d$ edges from a_1 to a_2 and as many from a_2 to a_1. We also place in the initial configuration N more chips in a_1 than in a_2. This guarantees that a_2 cannot be fired before a_1: because of the large number of edges from a_2 to a_1, there are not enough chips in the game to gather enough chips in a_2 if a_1 keeps its initial number of chips. When a_1 is fired, it sends $2d$ chips to the successors of a, and $N-d$ chips to a_2. a_1 has lost

$N + d$ chips, therefore it contains now as much chips as a in the corresponding configuration of C, and a_2 has gained $N - d$ chips, therefore it contains N more chips than a in the corresponding configuration. This takes care of the apparent lack of chips we spoke of above. Now, it will not be possible to fire a_1 again before a_2 is fired, for the same reason that it was not possible to fire a_2 before a_1 in the first place. This sketch is incomplete, because it is not correct in the case where there are loops on a. Now we give the formal description of the transformation (which is correct in all cases).

Let C be a non simple CFG, defined on a graph $G = (V, E)$, and with initial configuration σ, and let a be a vertex that is fired twice or more between the initial and final configuration in C. For a vertex v, we denote by $l(v)$ the number of loops on v. We denote by $d^{>}{}_G(v)$ the number of edges going out of v that are not loops (*i.e.* $d^{>}{}_G(v) = d_G{}^{+}(v) - l(v)$), and we define dually $d^{<}{}_G(v)$. The CFG C', defined on the multi-graph $G' = (V', E')$ and initial configuration σ' is defined in the following way. Let N be twice the number of chips in C. Let $V' = V \backslash \{a\} \cup \{a_1, a_2\}$, with $a_1, a_2 \notin V$. E' is defined by:

- for each $v, w \in V \backslash \{a\}$, if there are n edges(v, w) in E, then there are $2n$ edges (v, w) in E'.
- for each edge (v, a), $v \neq a$ in E, there is one edge (v, a_1) and one edge (v, a_2) in E'
- for each edge (a, v), $v \neq a$ in E, there are two edges (a_1, v) and two edges (a_2, v) in E'
- for each loop (a, a) in E, there is one loop (a_1, a_1) and one loop (a_2, a_2) in E'
- there are $N - d_G{}^{>}(a)$ edges both from a_1 to a_2 and from a_2 to a_1.

Moreover, for all $v \neq a$, $\sigma'(v) = 2\sigma(v)$, $\sigma'(a_1) = \sigma(a) + N$, and $\sigma'(a_2) = \sigma(a)$.

Figure 16 illustrates the construction. We will prove the following property: every configuration of C' is such that either a_1 contains N chips more than a_2, or a_2 contains N chips more than a_1. This is true for the initial configuration. Since for each $v \neq a_1, a_2$, there is the same number of edges from v to a_1 as from v to a_2, the firing of any other vertex that one of the a_i does not change this property. Let us suppose now that we can fire one of the vertices a_i, for instance a_1. Let x be the number of chips in a_2. The fact that a_1 can be fired implies that a_1 is the vertex that contains N chips more than the other, therefore there are $N + x$ chips in a_1. After the firing of a_1, there are $N + x - 2 \cdot d_G{}^{>}(a) - (N - d_G{}^{>}(a)) = x - d_G{}^{>}(a)$ chips in a_1, and $N + x - d_G{}^{>}(a)$ in a_2. Therefore the property is verified.

To prove that $L(C')$ is isomorphic to $L(C)$, the only thing that remains to show is that one of the vertices a_1 or a_2 can be fired in C' if and only if a can be fired in the double of C. We recall that always one of the vertices a_1 or a_2 contain N

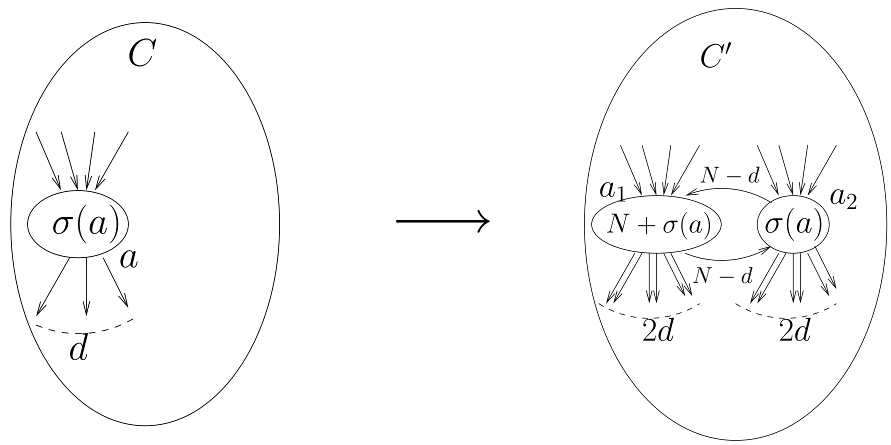


Figure 16: Simplification of a CFG

chips more than the other. This vertex can be fired if and only if it contains more than ${d_G}^+(a)+N$ chips. Then the sum of the number of chips in a_1 and a_2 is more than $N+2\cdot d_G^+(a)$ chips. In the corresponding configuration of the double of C, a contains then more than $2\cdot d_G^+(a)$ chips, which means that a can be fired.

By this method we obtain a CFG C' where the vertices a_1 and a_2 are each fired less often than in the initial CFG. By iterating this procedure, we eventually obtain a simple CFG equivalent to C. □

This theorem makes it possible to only consider simple CFGs in the following. Notice however that in [Eri89] it is shown that a convergent CFG may need an exponential number of firings with respect to the number of its vertices to reach its stable configuration. Therefore, given a non-simple CFG C, the number of vertices of an equivalent simple CFG can be exponential in the number of vertices of C. The purpose of introducing simple CFGs is not to be algorithmically efficient, but to introduce simple and natural proofs.

Given any simple CFG, we can associate to each firing sequence the set of vertices fired during the sequence. Then, it is obvious that if two sequences starting from the same configuration σ have the same set of vertices, then they lead to the same configuration σ'. The following theorem shows that the converse is also true.

Theorem 4.3 *Given a simple CFG C, if, starting from the same configuration, two sequences of firings s and t lead to the same configuration, then the set of vertices fired during s and t are the same.*

Proof : Let C be a simple CFG with support graph $G=(V,E)$, and let s and t be two firing sequences leading from a configuration σ to a configuration σ'. Let X and Y be the sets of vertices fired in s and t respectively, and suppose $X\neq Y$. We can suppose without loss of generality that $X\setminus Y$ is not empty. The sequence

s begins by a (possibly empty) sequence s_1 of vertices in $X \cap Y$, followed by the occurence of a vertex $v \in X \setminus Y$. This means that, after the firing of all the vertices of s_1, v contains more chips than its outdegree. Now if we go from σ to σ' following the sequence t, all the vertices of $X \cap Y$ are fired in the process, therefore all the vertices of s_1 are fired. From this we conclude that, after the firing of all the vertices of Y, the vertex v can be fired, which means that v can be fired in configuration σ'. Since configuration σ' can be obtained after the firing of all vertices of X (including v), and since v can be fired in configuration σ', we conclude that v can be fired at least twice. This is impossible, because C is simple. Therefore we must have $X = Y$. $\square$

This allows us to define the *shot-set* $s(\sigma)$ of a configuration σ as the set of the vertices fired to reach σ from the initial configuration. We will say that a subset X of the vertex set of a CFG is a *valid* shot-set if its vertices can be ordered as a valid firing sequence. The configurations and the valid shot-sets of any CFG are in a one-to-one correspondence: a valid shot-set corresponds to a unique configuration. In the next lemma we show that this correspondence induces in fact an isomorphism.

Lemma 4.4 *The configuration space of a simple CFG is isomorphic to the set of its shot-sets, ordered by inclusion.*

Proof : Let C be a simple CFG, and let σ and σ' be two configurations such that σ' can be reached from σ by a firing sequence using the vertices $v_1, \ldots, v_n$. Then we have $s(\sigma') = s(\sigma) \cup \{v_1, \ldots, v_n\}$. On the other hand, if we have $s(\sigma) \subseteq s(\sigma')$, then there exists a sequence of firings leading from σ to σ': the vertices of $s(\sigma)$ can be fired first because $s(\sigma)$ is a valid shot-set, then the vertices of $s(\sigma') \setminus s(\sigma)$ contain at least as many chips as before, and so they can be fired in the order in which they appear in any firing sequence that reach σ' starting from the initial configuration. $\square$

This is a very helpful result, because many results can be proved much more simply if we work on the shot-sets instead than on the configurations themselves. An example of this approach can be seen in the next theorem:

Theorem 4.5 *The configuration space of a simple CFG is a ULD lattice.*

Proof : We recall that any set of sets ordered by inclusion having a unique minimal element, and closed under union, is a lattice. We will prove that the set of the shot-sets of any simple CFG is closed under union: let X and Y, $X \neq Y$, be two valid shot-sets of a simple CFG C. We can suppose without loss of generality that $X \setminus Y$ is not empty. Let s and t be two valid firing sequences using all the vertices respectively of X and Y. These sequences have a common beginning s_1, possibly empty. After s_1, the sequence s is continued with a vertex $x \in X \setminus Y$. We claim that $Y \cup \{x\}$ is a valid shot-set of C: indeed, since x is not fired during t, the number of chips it contains does not decrease during this sequence, and since x can be fired after the sequence s_1, it can still be fired after the whole sequence t. Therefore $Y \cup \{x\}$ is a valid shot-set of C, and we can extend this reasonning to show that $X \cup Y$ is a valid shot-set. Since the set of the shot-sets of a convergent CFG has a unique minimal element (the empty set, corresponding to the initial configuration), and is closed under union, it is a lattice.

Now we show that the configuration space of any convergent CFG is a ULD lattice: if in a given configuration σ, with shot-set s, n different vertices $v_1, \dots, v_n$ can be fired, then the firing of one of them does not impede the firing of the others. From this we conclude that any subset of $s \cup \{v_1, \dots, v_n\}$ is a valid shot-set. The shot-set of the supremum of all the upper covers of σ is $s \cup \{v_1, \dots, v_n\}$. Therefore the interval between σ and the supremum of its upper covers is a hypercube of dimension n. This is the definition of ULD lattices. $\square$

Since any convergent CFG is equivalent to a simple CFG, we have immediately the following corollary:

Corollary 4.6 *The configuration space of any convergent CFG is a ULD lattice.*

Notice that the bijection between the configurations and the shot-sets is very convenient, because it does not only provide a simple way to prove that the configuration space of a CFG is a lattice, it also provides a simple formula for the upper bound. Indeed, for any two configurations a and b of a CFG, we have:

$$s(a \vee b) = s(a) \cup s(b).$$

5 Conclusion and perspectives

We have presented in this paper the study of the structure of the configuration spaces of some models which generate lattices. This study started with the study of some sandpile models and the two simple evolution rules of the Brylawski model. It has then been continued for some time with the models obtained by making modifications of these rules. This has given rise to the models SPM, IPM, $L(n, \theta)$ and $CFG(n, m)$, which also generate lattices. This shows that the lattice structure is inherant to these models, and cannot be broken easily by changing the rules.

One other model which also is a representation of some sand piles phenomena, the Chip Firing Game, was studied with the same idea. It was proved that it generates lattices, and that it is a generalization of SPM, $L(n, \theta)$, $CFG(n, m)$ and others: these models can be encoded as special CFGs. This has given to the CFG a special importance among all these models, and it was studied in the attempt to determine why lattices appear in this context, and which properties they share. During this study a very special class of CFG has arisen, the *simple* CFGs. These are the CFGs such that the evolution rule is applied only once to each vertex between the initial and final configurations. It was proved that any convergent CFG is equivalent to a simple CFG. This makes it possible to study the lattice structure of these models much more easily. The original proof that CFGs generate lattices used the same kind of techiniques as the proofs previously made for other models. With the simple CFGs, a new proof was devised, which was more natural and more in agreement with the structure of CFGs. This gives a better understanding of why the CFGs, and at the same time all the models that can be encoded as CFGs, induce lattices.

There are many directions of research for further work. We present them now, including some which have already been the subject of some attention.

Different classes of lattices

We have seen that some models can be encoded as special CFGs. However, this cannot be done for the Brylawski model: all the lattices induced by CFGs are ranked, *i.e.* all the paths from the minimal to the maximal element have the same length, whereas lattices induced by the Brylawski model are not. Therefore some attempts have been made on the one hand to characterize exactly which lattices can be obtained by CFGs. Such a characterization can help to decide whether a given model is a particular case of the CFG or not: if not all the configuration spaces it induces are in the class $L(CFG)$ of lattices induced by CFG, then we know that we cannot find an encoding of this model as a CFG. In [MPV01] it has been proved that $L(CFG)$ is not the whole ULD class (*i.e.* there exists a ULD lattice which is the configuration space of no CFG), but contains the class D of distributive lattices. This is an interesting result from the lattice theory point of view, since the distributive and ULD lattices classes are very close to one another, and there is no known lattice class between these two. As already discussed, the Abelian Sandpile Model can be seen as a particular case of the Chip Firing Game, therefore the class $L(ASM)$ of lattices induced by ASM is included in $L(CFG)$. In [Mag01] some attempts have been made to define this class more precisely, and it has been proved that $L(ASM)$ is another class between the distributive and the ULD lattices. To summarize these results, we have the following relations:

$$D \subsetneq L(ASM) \subsetneq L(CFG) \subsetneq ULD.$$

We have seen that, among the other models presented in this paper, some of them are generalizations of others, which implies some inclusion relations between the classes of lattices they induce. Figure 17 summarizes these relations. It emphasizes the complexity of the characterization problems in lattice theory. The two classes $L(CFG)$ and $L(ASM)$ have not been characterized exactly. Finding an algorithm

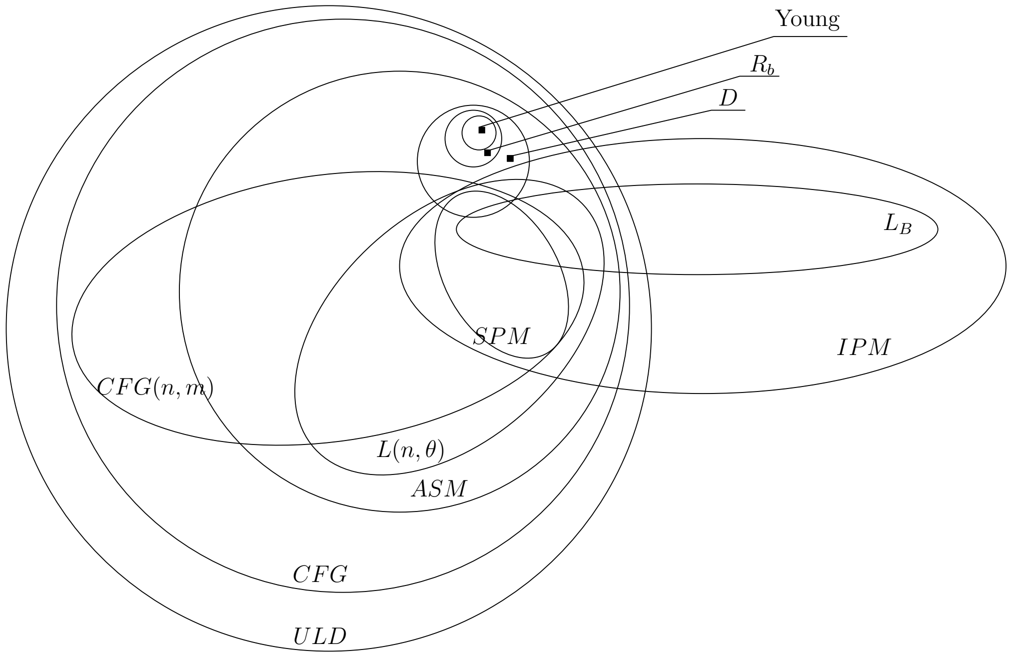


Figure 17: The classes of lattices induced by various models

to decide if a given lattice is induced by one or more of these models is a challenge both for the study of discrete dynamical models and for lattice theory.

Generalizations of the models

Another direction of research is the extension of the models we have studied to a more general model (in the same manner as the CFG is itself an extension of SPM). The CFG is for the time being the most general of the models we have studied, therefore it makes sense to try to start from it to obtain a generalization of the Brylawski model. Indeed, SPM and L_B are very close to one another in their definition, and the study of a model that represents them both would help to understand their specificities better. In [MPV01] a generalization of the CFG, the *coloured* Chip Firing Game, has been presented. It generates exactly the ULD class. Therefore it cannot simulate the Brylawski model (since the lattices $L_B(n)$ are not ULD), and the model needs to be extended further.

Infinite extensions

Another natural idea to extend the model is to consider that there is an infinity of grains. Some work has been done about this in [LP99, LMMP01, Lat01], where SPM, L_B and R_b are started with an infinite first column. The configuration spaces of such models are ordered as infinite lattices. It has also been proved that they can be represented by inifinite trees, which emphasizes their strong self-similarity. This work has only been done with linear models, and the same kind of study on more complex models like the Chip Firing Game or the Abelian Sandpile Model may lead to interesting results.

Some of the models we have presented are always convergent (mainly the linear models), and some are not. CFGs, for instance, may have cycles in their configuration spaces, and therefore they may stay in the cycle forever. It is shown in [LP01] that the configuration spaces of such models can be seen as infinite lattices, which share the same main properties as in the convergent case. For instance, infinite lattices induced by non-convergent CFGs are also ULD. The study of the configuration spaces of non-convergent models has not been deepened further, and would be a natural complement of the study of convergent ones. Another idea is to consider models with inifinite configurations, for instance CFGs on infinite graphs.

Algebraic properties

In all the studies presented above, the configuration space of the models and its structure were studied. No special interest was given to the configurations of the models themselves. For the Abelian Sandpile Model it is known [DRSV95] that some special stable configurations, called the *recurrent* configurations, form an abelian group. This algebraic aspect of the model has given rise to many interesting studies [CR00, Dha98, IP98]. However, these studies are entirely independent of the studies of the configuration spaces we presented here. Combining these two aspects would surely give a much better understanding of the models, and is probably one of the most important directions for further work.

Tilings problems

Finally, some other kinds of discrete dynamical models appear in the context of tiling theory: for some classes of tiling problems, one can define a local rearrangement of tiles, called *flip*, which transforms a tiling of a given region into another tiling of the same region. In some cases (mainly tilings with dominoes or with three lozenges [Rem99, BL01]), it has been proved that the flip relation gives the distributive lattice structure to the set of all possible tilings of a given region. In [BL01] a notion of tiling on graphs is introduced as a generalization for these problems. These tilings of graphs have the particularity that the set of all possible tilings is ordered as a union of distributive lattices by the flip relation. The proof of this uses height functions, like the original proofs for the particular cases. In [BL01] it is also proved that height functions can be viewed as special Edge Firing Games. This proves that the study of discrete dynamical models exposed in this paper can have applications in a great scope of seemingly unrelated problems. We are only at the beginning of this study with a goal of generalization to other problems in mind.

References

[And76] George E. Andrews. *The Theory of Partitions*, volume 2 of *Encyclopedia of Mathematics and its applications.* Addison-Wesley Publishing Company, 1976.

[Bak97] Per Bak. *How nature works - The science of SOC.* Oxford university press, 1997.

[Ber71] Claude Berge. *Principles of Combinatorics*, volume 72 of *Mathematics in science and engineering.* Academic Press, 1971.

[Ber98] Karell Bertet. *Sur quelques aspects algorithmiques et structurels des treillis.* PhD thesis, Université Paris VII, 1998.

[Big97] N. Biggs. Algebraic potential theory on graphs. *Bull. London Math. Soc.*, 29:641–682, 1997.

[Big99] N. Biggs. Chip firing and the critical group of a graph. *Journal of Algebraic Combinatorics*, 9:25–45, 1999.

[BL92] A. Björner and L. Lovász. Chip-firing games on directed graphs. *Journal of Algebraic Combinatorics*, 1:304–328, 1992.

[BL01] O. Bodini and M. Latapy. Generalized tilings with height functions. 2001. Submitted.

[BLS91] A. Björner, L. Lovász, and W. Shor. Chip-firing games on graphs. *European Journal of Combinatorics*, 12:283–291, 1991.

[Bry73] T. Brylawski. The lattice of integer partitions. *Discrete Mathematics*, 6:210–219, 1973.

[BTW87] P. Bak, C. Tang, and K. Wiesenfeld. Self-organized criticality: an explanation of the $1/f$ noise. *Physics Review Letters*, 59(4):381–384, 1987.

[CR00] Robert Cori and Dominique Rossin. On the sandpile group of a graph. *European Journal of Combinatorics*, 21(4):447–459, 2000.

[Dha98] Deepak Dhar. The abelian sandpile model and related models. 1998. Preprint available on Arxiv, at `http://arxiv.org/abs/cond-mat/9808047`

[DKVW95] J. Desel, E. Kindler, T. Vesper, and R. Walter. A simplified proof of the self-stabilizing protocol: A game of cards. *Information Processing Letters*, 54:327–328, 1995.

[DM90] D. Dhar and S.N. Majumbar. Abelian sandpile model on the Bethe lattice. *Journal of Physics*, A 23:4333–4350, 1990.

[DP90] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Orders*. Cambridge university press, 1990.

[DRSV95] Deepak Dhar, P. Ruelle, S. Sen, and D. Verma. Algebraic aspects of sandpile models. *Journal of Physics*, A 28:805–831, 1995.

[Eri89] Kimmo Eriksson. No polynomial bound for the chip firing game on directed graphs. *Proceedings of the American Mathematical Society*, 112:1203–1205, 1989.

[Eri93] Kimmo Eriksson. *Strongly Convergent Games and Coxeter Groups*. PhD thesis, Kungl Tekniska Hogskolan, Sweden, 1993.

[GK93] E. Goles and M.A. Kiwi. Games on line graphs and sand piles. *Theoretical Computer Science*, 115:321–349, 1993.

[GMP98a] E. Goles, M. Morvan, and H.D. Phan. Lattice structure and convergence of a game of cards. 1998. To appear in Annals of Combinatorics.

[GMP98b] E. Goles, M. Morvan, and H.D. Phan. Sand piles and order structure of integer partitions. 1998. To appear in Discrete Applied Mathematics. Preprint available at `http://www.liafa.jussieu.fr/~phan/`.

[GMP98c] E. Goles, M. Morvan, and H.D. Phan. The structure of linear chip firing games and related models. 1998. To appear in Theoretical Computer Science. Preprint available at `http://www.liafa.jussieu.fr/~phan/`.

[Heu99] Jan van den Heuvel. Algorithmic aspects of a chip firing game. *London School of Economics, CDAM Research Reports.*, 1999. Preprint available at `http://www.cdam.lse.ac.uk/Reports/reports99.html`.

[IP98] E.V. Ivashkevich and V.V. Priezzhev. Introduction to the sandpile model. *Physica A*, 254:97–116, 1998.

[Jen98] Henrik Jeldtoft Jensen. *Self-Organised Criticality*. Cambridge university press, 1998.

[Lat00] Matthieu Latapy. Generalized integer partitions, tilings of zonotopes and lattices. In A.A.Mikhalev D.Krob and E.V.Mikhalev, editors, *Proceedings of the 12-th international conference Formal Power Series and Algebraic Combinatorics (FPSAC'00)*, pages 256–267. Springer, June 2000. Preprint available at `http://www.liafa.jussieu.fr/~latapy/`.

[Lat01] Matthieu Latapy. Partitions of an integer into powers. In *Discrete Mathematics and Theoretical Computer Science, Proceedings of the 1-st international conference Discrete Models: Combinatorics, Computation, and Geometry (DM-CCG'01)*, pages 215–228. MIMD, July 2001. Preprint available at `http://www.liafa.jussieu.fr/~latapy/`.

[LMMP01] Matthieu Latapy, Roberto Mantaci, Michel Morvan, and Ha Duong Phan. Structure of some sand piles model. *Theoretical Computer Science*, 262:525–556, 2001. Preprint available at `http://www.liafa.jussieu.fr/~latapy/`.

[LP99] M. Latapy and H.D. Phan. The lattice of integer partitions and its infinite extension. 1999. To appear in DMTCS, special issue, proceedings of ORDAL'99. Preprint available at `http://www.liafa.jussieu.fr/~latapy/`.

[LP01] Matthieu Latapy and Ha Duong Phan. The lattice structure of chip firing games. *Physica D*, 115:69–82, 2001. Preprint available at `http://www.liafa.jussieu.fr/~latapy/`.

[Mac16] P.A. MacMahon. *Combinatory Analysis*. Cambridge University Press, 1916.

[Mag01] Clémence Magnien. Lattices induced by chip firing games and related models. 2001. Submitted. Preprint available at `http://www.liafa.jussieu.fr/~magnien/`.

[MKM78] V.M. Malhotra, M.P. Kumar, and S.N. Maheshwari. An $o(|v|^2)$ algorithm for finding maximal flows in networks. *Inform. Process. Letters*, 7:277–278, 1978.

[Mon90] Bernard Monjardet. The consequences of dilworth's work on lattices with unique irreductible decompositions. In K. P. Bogart, R. Freese, and J. Kung, editors, *The Dilworth theorems Selected papers of Robert P. Dilworth*, pages 192–201. Birkhauser, Boston, 1990.

[MPV01] Clémence Magnien, Ha Duong Phan, and Laurent Vuillon. Characterization of lattices induced by (extended) chip firing games. In *Discrete Mathematics and Theoretical Computer Science, Proceedings of the 1-st international conference Discrete Models: Combinatorics, Computation, and Geometry (DM-CCG'01)*, pages 229–244. MIMD, July 2001. Preprint available at `http://www.liafa.jussieu.fr/~magnien/`.

[Pre86a] O. Pretzel. On reorienting graphs by pushing down maximal vertices. *Order*, 3:135–153, 1986.

[Pre86b] O. Pretzel. Orientations and reorientations of graphs. *Contemp. Math.*, 57:103–125, 1986.

[Pro93] James Propp. Lattice structure of orientations of graphs. 1993. Preprint, available at `http://www.math.wisc.edu/~propp/orient.html`.

[Pro98] James Propp. Generating random elements of finite distributive lattices. *Electronic Journal of Combinatorics*, 4, 1998.

[Rem99] Eric Remila. The lattice structure of the set of domino tilings of a polygon. *LIP - ENS Lyon - Research Report 1999-25*, 1999. To appear in DMTCS, special issue, proceedings of ORDAL'99.

[Sta97] Richard.P. Stanley. *Enumerative combinatorics*, volume 1. Cambridge University Press, 1997. Cambridge Studies in Advanced Mathematics 49.

[Tan93] C. Tang. Self-organised criticality. *OCPA Newsletter*, 1993.