A Novel Method for Counting Models on Grid Boolean Formulas

Carlos Guillén[1], Guillermo De Ita[1], and Aurelio López-López[2]

[1] Facultad de Ciencias de la Computación, BUAP
[2] Instituto Nacional de Astrofísica, Óptica y Electrónica
{cguillen,deita,allopez}@ccc.inaoep.mx

Abstract. We research on the possible orientations patterns of a grid graph G, and propose a method for counting certain combinatorial structures over the class of orientations of G. For example, our method can be applied for counting sink-free orientations of G, as well as it can be applied for solving the #2SAT problem for grid Boolean formulas.

Our proposal extends the classical transfer matrix method used for counting the number of independent sets in a grid.

Keywords: Grid Computing, Transfer Matrix Method, $\#2SAT$ Problem.

1 Introduction

Many important combinatorial problems are modelled as constraint satisfaction problems. Constraint satisfaction problems form a large class of combinatorial problems that contains many important 'real-world' problems. An instance of a constraint satisfaction problem consists of a set V of variables, a domain D, and a set F of constraints. For example, the domain may be $\{0, 1\}$, and the constraints may be clauses of a Boolean formula in Conjunctive Normal Form (CNF). The objective is to assign values in D to the variables in such a way that all constraints are satisfied.

In general, constraint satisfaction problems (CSP) are NP-hard; and considerable efforts, both practical and theoretical, have been made to identify tractable classes for that class of problems [7].

One application of the CSP has been for recognizing combinatorial patterns on graphs and to apply techniques developed for the CSP problem for solving different combinatorial problems on graphs. For example, given an undirected graph $G = (V, E)$, we can associate a monotone 2-CNF formula F_G with variables V, and where $F_G = \bigwedge_{(u,v)\in E}(u \vee v)$, a conjunctive normal form F_G is called *monotone* when each variable of F_G occurs with just one of its two signs.

A set $I \subseteq V$ is called an independent set if no two of its elements are joined by an edge. Let $S_I = \{v_j : j \in I\}$ be an independent set in G, then the assignment defined by $x_i = 0$ if $i \in I$ or $x_i = 1$ otherwise, satisfies F_G. The reason is that in every clause $(x_i \vee x_j)$ (representing the edge $\{v_i, v_j\}$ of G) at least one of

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 322–331, 2010.
© Springer-Verlag Berlin Heidelberg 2010

the variables is assigned to 1, since otherwise the nodes v_i and v_j are in the independent set S_I and then there are not an edge in G.

Other interesting problem to be modeled using 2-CNF's is about the orientation of a graph. Considering again, an undirected graph $G = (V, E)$, an orientation of G is an assignment of exactly one direction to each of the edges of G. Then, an orientation of $\{u, v\} \in E$ is (u, v) denoted as $u \rightarrow v$ or (v, u) denoted as $u \rightarrow v$. For an oriented edge $u \rightarrow v$, u is called the tail and v is called the head of the edge. The number of edges where u is a head is the in-degree of u and its out-degree is the number of edges where u is the tail. A node $u \in V$ with out-degree zero is called a sink of the graph.

An orientation O of a graph is *sink-free* if no node is a sink in O. There are important and classic problems related with recognize patterns and count combinatorial structures on the orientations, like: decision, construction, unique, listing, counting sink-free graph orientations and the acyclic orientations of the graph [5,3]. For example, the decision problem SFO on instance G is to determine whether G has a sink-free orientation, and the $\#SFO$ problem is to count the number of sink-free orientations of G.

Notice that an oriented edge $u \rightarrow v$ can be represented by the constraint $(\neg u \vee v)$ and then, problems related to oriented graphs can be considered as a restricted class of the CSP. In fact, Russ [11] has shown that the SFO problem is equivalent to determine the satisfiability of Boolean Conjunctive Formulas where each literal appears exactly once, problem knowing as Twice-SAT.

In this work, we consider a more general case of the patterns of orientations of an undirected edge. Given an undirected graph $G = (V, E)$, we associate to each edge $\{u, v\} \in E$ an ordered pair (s_1, s_2) of signs assigned as the labels of the edge. The signs s_1 and s_2 are related to the signs of the literals u and v respectively. For example, the clause $(\neg x \vee \neg y)$ determines the labeled edge: "$x \overset{-\;-}{\text{———}} y$".

Then, we have four different orientations for any edge $\{u, v\} \in E$; when $(s_1, s_2) = (-, +)$ the edge is type $u \rightarrow v$, if $(s_1, s_2) = (+, -)$ the edge is $v \rightarrow u$, both cases are called ordinary edges. The cases where $(s_1, s_2) = (+, +)$ denoted as $v \leftrightarrow u$, and where $(s_1, s_2) = (-, -)$ denoted as $v \rightarrow\leftarrow u$ are called skews edges.

This type of orientations generalize the class of problems which could be modeled and solved through methods applied in the area of Constraint Satisfaction Problems. We present a new matrix method for recognizing and counting the number of sink-free orientations of a planar grid graph G under this class of orientations, for solving its related constraint satisfaction problem; to count the number of models of a 2-CNF on formulas whose constrained graph is a grid graph.

The constraint satisfaction problem has been a helpful language to model processing on Grid graph which is one of the most important physical graph topology for modeling parallel and distributed computing.

2 The Transfer Matrix Method and the #2SAT Problem

An undirected planar grid graph of size $m \times n$ is a graph $G_{m,n}$ with vertex set $V = \{0, ..., m\} \times \{0, ..., n\}$ and edge set $E = \{\{u, v\} : u, v \in V \wedge \|u - v\| = 1\}$. where $\|\cdot\|$ is the euclidean norm of R^2. Let $I(G_{m,n})$ be the number of independent sets of $G_{m,n}$. There is a large volume of literature devoted to recognize and count structures in a grid graph, e.g., spanning trees, Hamiltonian cycles, independent sets, acyclic orientations, k-coloring, and so on [1,4,8,6].

In other line research, the transfer matrix method is a general technique which has been used to find exact solutions for a great variety of problems. For example, Calkin [4] used this method for computing the number of independent sets over a grid graph $G_{m,n}$.

Shortly, we describe the method used by Calkin as follows. Let $\mathcal{C}_m$ be the set of all $(m + 1)$-vectors $\mathbf{v}$ of $0's$ and $1's$ without two consecutive $1's$ (the number of these vectors is F_{m+2}, the $m + 2$-th Fibonacci number). Let T_m be an $F_{m+2} \times F_{m+2}$ symmetric matrix of $0's$ and $1's$ whose rows and columns are indexed by the vectors of $\mathcal{C}_m$. The entry of T_m in position $(\mathbf{u}, \mathbf{v})$ is 1 if the vectors $\mathbf{u}, \mathbf{v}$ are orthogonal, and is 0 otherwise, T_m is called the transfer matrix for $G_{m,n}$. Then, $I(G_{m,n})$ is the sum of all entries of the n-th power matrix T_m^n, i.e., $I(G_{m,n}) = \mathbf{1}^t T_m^n \mathbf{1}$, where $\mathbf{1}$ is the (F_{m+2})-vector whose entries are all $1's$.

For example, if $m = 2$ and $n = 3$ we have that $\mathcal{C}_2 = \{(0, 0, 0), (1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 0, 1)\}$,

$$T_2 = \begin{pmatrix} 1&1&1&1&1 \\ 1&0&1&1&0 \\ 1&1&0&1&1 \\ 1&1&1&0&0 \\ 1&0&1&0&0 \end{pmatrix} \quad \textit{and} \quad T_2^3 = \begin{pmatrix} 17&12&13&12&9 \\ 12&7&10&8&5 \\ 13&10&9&10&8 \\ 12&8&10&7&5 \\ 9&5&8&5&3 \end{pmatrix}$$

therefore $I(G(2, 3)) = \mathbf{1}T_2^3\mathbf{1} = 227$.

The study of $I(G_{m,n})$ is closely related to the "hard-square model" used in statistical physics and, of particular interest is the so-called "hard-square entropy constant" defined as $\lim_{m,n\to\infty} I(G_{m,n})^{1/m \cdot n}$ [1]. Applications also include for instance tiling and efficient coding schemes in data storage [10].

Given a monotone Boolean formula F in 2-conjunctive normal form (2-CNF), we can associate an undirected signed graph $G_F = (V, E)$, called its constrained graph, where V is the set of variables of F and two vertices of V are connected by an edge in E if they belong to the same clause of F. We say that a 2-CNF formula is a *cycle, path, tree*, or *grid formula* if its constrained graph is a *cycle, path, tree*, or *grid* graph respectively.

It is known that the number of independent sets of G_F is the number of satisfying assignments (models) for monotone formulas F [9]. The number of models of a Boolean formula F is denoted as #SAT(F). The computation of #SAT(F) for formulas in 2-CNF is a classic #P-complete problem [3].

In order to extend the transfer matrix method for considering any kind of 2-CNF's, we have to deal with grid graphs with signed edges.

3 Oriented Grids

For each undirected edge $e = \{u, v\} \in E$ of an grid graph $G_{m,n}$, we consider four types of orientations for e: $(+,+)$, $(+,-)$, $(-,+)$ and $(-,-)$ (see figure 1).

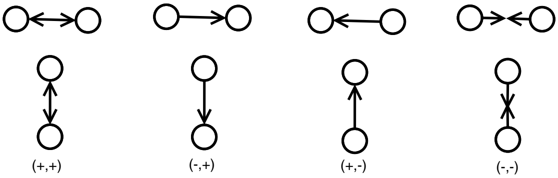

Fig. 1. Types of orientations

An oriented grid graph is a triplet $G = (V, E, \psi)$, where (V, E) is a graph (grid graph), and ψ is a function with domain E and range $\{+,-\} \times \{+,-\}$. The evaluation $\psi(e)$ is called the orientation of the edge $e \in E$.

Let $e = \{\mathbf{u}, \mathbf{v}\}$ be an edge of an oriented grid graph, if the vector $\mathbf{u} - \mathbf{v}$ is parallel to the vector $(0, 1)$, e is called column edge, if $\mathbf{u} - \mathbf{v}$ is parallel to the vector $(1, 0)$, e is called a row-edge.

A k-column of an oriented grid graph $G_{m,n}$ is the vertex-induced subgraph by the nodes

$$x_{k0}, ..., x_{km}$$

where $x_{ij} = (i, j)$. The vertex-induced subgraph by the nodes

$$x_{k0}, ..., x_{km}, x_{(k+1)0}, ..., x_{(k+1)m}$$

is denoted by $G_{m,k,k+1}$(see figure 2a).

From the k-columns and ℓ-rows of an oriented grid graph $G_{m,n}$ we define the vectors $\mathbf{s}_k, \mathbf{r}_k, \overrightarrow{\mathbf{r}}_k, \overleftarrow{\mathbf{r}}_k \in \{+,-\}^{2m}$ defined by

$$\mathbf{s}_k = (s'_{0k}, s_{1k}, s'_{1k}, ..., s_{m-1k}, s'_{m-1k}, s_{mk})$$
$$\mathbf{r}_k = (r'_{0k}, r_{1k}, r'_{1k}, ..., r_{m-1k}, r'_{m-1k}, r_{mk})$$
$$\overrightarrow{\mathbf{r}}_k = (r_{0k}, r_{1k}, r_{1k}, ..., r_{m-1k}, r_{m-1k}, r_{mk})$$
$$\overleftarrow{\mathbf{r}}_k = (r'_{0k}, r'_{1k}, r'_{1k}, ..., r'_{m-1k}, r'_{m-1k}, r_{mk})$$

The k-column nodes of $G_{m,n}$ induces a vector $\mathbf{x}_k \in \{0,1\}^{2m}$ given by

$$\mathbf{x}_k = (x'_{k0}, x_{k1}, x'_{k1}, ..., x_{km-1}, x'_{km-1}, x_{km})$$

(see figure 2b).

The vector $\mathbf{s}_k$ is called the k-oriented vector induced by the column nodes from $G_{m,n}$. $\mathbf{x}_k$ is called the k-vector induced by the column nodes from $G_{m,n}$ and the pair $\langle \mathbf{s}_k, \mathbf{s}_{k+1} \rangle$ is called the sign vectors induced by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$.

A valuation of the nodes x_{ij} is a function φ with domain $\{0, ..., m\} \times \{0, ..., n\}$ and range $\{0, 1\}$. We define the operation $\cdot : \{+,-\} \times \{0,1\} \rightarrow \{0,1\}$ as +0=0,+1=1,-0=1, and -1=0. The operation

$$\odot : \{+,-\}^{2m} \times \{0,1\}^{2m} \rightarrow \{0,1\}$$

is defined by $(s_i) \odot (x_i) = (s_i \cdot x_i)$.

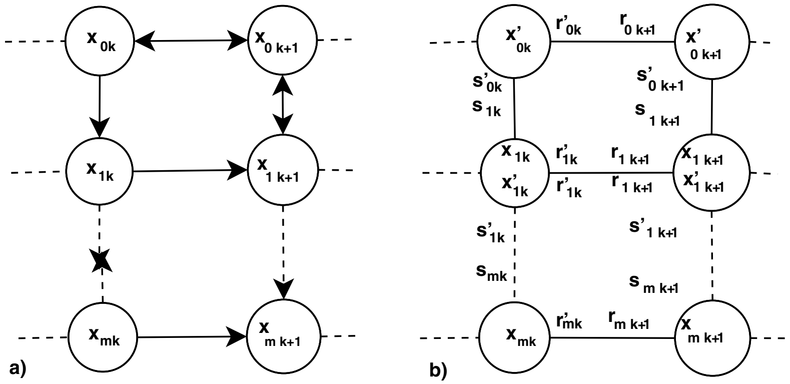

Fig. 2. a) Vertex-induced subgraph $G_{m,k,k+1}$ b) Vectors $\mathbf{s}_k, \mathbf{r}_k$

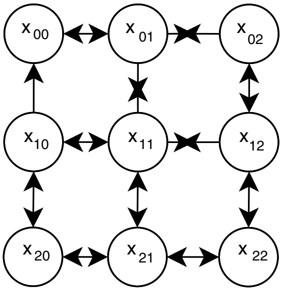

Fig. 3. Grid graph $G_{2,2}$

4 A Novel Matrix Method for Processing Grids

Let $\mathcal{F}_m$ be the set of all $2m$-vectors $\mathbf{v}$ of $0's$ and $1's$, and let $\mathcal{C}_m \subset \mathcal{F}_m$ be the set of all $2m$-vectors $\mathbf{v}$ of $0's$ and $1's$, such that $\mathbf{v}$ does not have two consecutive $1's$. The cardinality of $\mathcal{C}_m$ (denoted by $|\mathcal{C}_m|$) is F_{2m+1} (the $2m$+1-th Fibonacci number), while $|\mathcal{F}_m| = 2^{2m}$. Given $\mathbf{s} \in \{0,1\}^{2m}$, we define

$$\mathcal{F}_m^{\mathbf{s}} = \{\mathbf{e} \in \mathcal{F}_m : \mathbf{s} \odot \mathbf{e} \in \mathcal{C}_m\}$$

Following the idea proposed in [4], we define a matrix $T_k = T_{m,k}$, the transfer matrix of $G_{m,k,k+1}$ as follows. T_k is an $\mid \mathcal{F}_m^{\mathbf{s}_{k+1}} \mid \times \mid \mathcal{F}_m^{\mathbf{s}_k} \mid$ matrix of $0's$ and $1's$ whose rows and columns are indexed by vectors $(\mathbf{v}, \mathbf{u})$ of $\mathcal{F}_m^{\mathbf{s}^{k+1}} \times \mathcal{F}_m^{\mathbf{s}^{k}}$. The entry of T_k in position $(\mathbf{v}, \mathbf{u})$ is 1 if the vectors $\overrightarrow{\mathbf{r}}_k \odot \mathbf{u}$ and $\overleftarrow{\mathbf{r}}_{k+1} \odot \mathbf{v}$ are orthogonal, and is 0 otherwise.

Notice that if $\overrightarrow{\mathbf{r}}_k$ and $\overleftarrow{\mathbf{r}}_{k+1}$ have positive entries, then T_k is the transfer matrix used in the transfer method [4]. For example, if $G_{2,2}$ is the grid graph with labeled edges as illustrated in figure 3. For $G_{2,0,1}$, we have that $\mathbf{s}_0 = (+,-,+,+)$, $\mathbf{s}_1 = (-,-,+,+)$ and $\overrightarrow{\mathbf{r}}_0 = \overleftarrow{\mathbf{r}}_1 = (+,+,+,+)$, then $\mathcal{F}_2^{\mathbf{s}_0} = \{\mathbf{u}_1, \cdots, \mathbf{u}_4\}$ and $\mathcal{F}_2^{\mathbf{s}_1} = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4\}$, where $\mathbf{u}_1 = (0,0,0,0)$, $\mathbf{u}_2 = (0,1,1,0)$, $\mathbf{u}_3 = (0,0,0,1)$,

$\mathbf{u}_4 = (1,1,1,0)$, $\mathbf{v}_1 = (1,0,0,0)$, $\mathbf{v}_2 = (0,1,1,0)$, $\mathbf{v}_3 = (1,0,0,1)$ and $\mathbf{v}_4 = (1,1,1,0)$. The transfer matrix $T_0 = (a_{ij})_{4\times 4}$, is a 4×4 matrix determined, for $1 \leq i, j \leq 4$, as $a_{ij} = 1$, if $(\overleftarrow{\mathbf{r}}_1 \odot \mathbf{v}_i) \cdot (\overrightarrow{\mathbf{r}}_0 \odot \mathbf{u}_j) = 0$ and $a_{ij} = 0$ otherwise. Since $\overrightarrow{\mathbf{r}}_0 = \overleftarrow{\mathbf{r}}_1 = (+,+,+,+)$, then $\overleftarrow{\mathbf{r}}_1 \odot \mathbf{v}_i = \mathbf{v}_i$ and $\overrightarrow{\mathbf{r}}_0 \odot \mathbf{u}_j = \mathbf{u}_j$. Then,

$$T_0 = \begin{pmatrix} 1\,1\,1\,0 \\ 1\,0\,1\,0 \\ 1\,1\,0\,0 \\ 1\,0\,1\,0 \end{pmatrix} \tag{1}$$

We have $\mathbf{s}_1 = (-,-,+,+)$, $\mathbf{s}_2 = (+,+,+,+)$, $\overrightarrow{\mathbf{r}}_1 = (-,-,-,+)$ and $\overleftarrow{\mathbf{r}}_2 = (-,+,+,+)$, then $\mathcal{F}_2^{\mathbf{s}_1} = \{\boldsymbol{\mu}_1,...,\boldsymbol{\mu}_4\}$ and $\mathcal{F}_2^{\mathbf{s}_2} = \{\boldsymbol{\nu}_1,...,\boldsymbol{\nu}_5\}$, where $\boldsymbol{\mu}_1$=(1,0,0,0), $\boldsymbol{\mu}_2$=(0,1,1,0), $\boldsymbol{\mu}_3$=(1,0,0,1), $\boldsymbol{\mu}_4$=(1,1,1,0), $\boldsymbol{\nu}_1$=(0,0,0,0), $\boldsymbol{\nu}_2$=(1,0,0,0), $\boldsymbol{\nu}_3$=(0,1,1,0), $\boldsymbol{\nu}_4$=(0,0,0,1) and $\boldsymbol{\nu}_5$=(1,0,0,1). Then,

$$\{\overrightarrow{\mathbf{r}}_1 \odot \boldsymbol{\mu} : \boldsymbol{\mu} \in \mathcal{F}_2^{\mathbf{s}_1}\} = \{(0,1,1,0),(1,0,0,0),(0,1,1,1),(0,0,0,0)\}$$

and

$$\{\overleftarrow{\mathbf{r}}_2 \odot \boldsymbol{\nu} : \boldsymbol{\nu} \in \mathcal{F}_2^{\mathbf{s}_2}\} = \{(1,0,0,0),(0,0,0,0),(1,1,1,0),(1,0,0,1),(0,0,0,1)\}$$

The transfer matrix $T_1 = (b_{ij})_{5\times 4}$, is such that, for $1 \leq i \leq 5$ and $1 \leq j \leq 4$, $b_{ij} = 1$, if $(\overleftarrow{\mathbf{r}}_2 \odot \boldsymbol{\nu}_i) \cdot (\overrightarrow{\mathbf{r}}_1 \odot \boldsymbol{\mu}_j) = 0$ and $b_{ij} = 0$ otherwise. Then

$$T_1 = \begin{pmatrix} 1\,0\,1\,1 \\ 1\,1\,1\,1 \\ 0\,0\,0\,1 \\ 1\,0\,0\,1 \\ 1\,1\,0\,1 \end{pmatrix} \tag{2}$$

Remark 1. When we compare our method with the Calkin's method, if $\overrightarrow{\mathbf{r}}_k$ and $\overleftarrow{\mathbf{r}}_{k+1}$ have positive entries, then $T_k = T$ for all $k = 1, ..., n$, and then T will be the classic transfer matrix used in [4].

In the case, not necessarily monotone, of a formula F having a constrained an oriented grid graph $G_{m,n}$ and transfer matrices $T_0, \ldots, T_{n-1}$, is straightforward to conclude that the sum of all entries of the product matrix $T_{n-1} \cdots T_0$ is the number of satisfying assignment of F. This fact is expressed in the following theorem.

Theorem 1. *Let F be a grid formula such that its constrained graph is an oriented grid graph $G_{m,n}$ ($1 \leq n$), then the number of satisfying assignments of F is given by the sum of all of the entries of the product matrix $T_{n-1} \cdots T_0$, where T_k is the transfer matrix of $G_{m,k,k+1}$, $k = 0, ..., n-1$.*

Before detailing the proof, we consider the following example.
Example: Let $F = (x_0 \vee y_0) \wedge (\neg y_0 \vee \neg z_0) \wedge (z_0 \vee z_1) \wedge (z_1 \vee z_2) \wedge (z_2 \vee y_2) \wedge (y_2 \vee x_2) \wedge (x_2 \vee x_1) \wedge (\neg x_1 \vee x_0) \wedge (x_1 \vee y_1) \wedge (\neg y_1 \vee z_1) \wedge (\neg y_1 \vee \neg y_0) \wedge (y_1 \vee y_2)$. The constrained graph of F is the oriented grid graph $G_{2,2}$ with depicted in Figure

3. Then, from last example, T_0 and T_1 are the transfer matrices given in (2) and (3) respectively. Now, we have that the product matrix T_1T_0 is the following

$$T_1T_0 = \begin{pmatrix} 3\,2\,2\,0 \\ 4\,2\,3\,0 \\ 1\,0\,1\,0 \\ 2\,1\,2\,0 \\ 3\,1\,3\,0 \end{pmatrix}$$

therefore, $\#\mathrm{SAT}(F) = 30$.

If $F_{m,n}$ denotes a grid formula having as constrained graph a grid $G_{m,n}$, for $n > 0$, we can write

$$F_{m,n} = (\bigwedge_{i=0}^{n} C_i) \wedge (\bigwedge_{\ell=0}^{n-1} R_\ell) \tag{3}$$

where

$$C_i = \bigwedge_{k=0}^{m-1} (s'_{ki}x_{ki} \vee s_{k+1,i}x_{k+1,i}) \tag{4}$$

$s'_{ki}, s_{k+1,i} \in \{+,-\}$,

$$R_\ell = \bigwedge_{j=0}^{m} (r'_{j\ell}x_{j\ell} \vee r_{j,\ell+1}x_{j,\ell+1}) \tag{5}$$

$r'_{j\ell}, r_{j,\ell+1} \in \{+,-\}$. Here, the formulas C_i and R_ℓ are called *column-formula* and *row-formula* respectively. Notice that for $m, n > 0$

$$F_{m,n} = F_{m,n-1} \wedge C_n \wedge R_{n-1},\ F_{m,0} = C_0,\ F_{0,n} = R_0. \tag{6}$$

For $i = 0, ..., n-1$, we define

$$F_{m,i,i+1} = C_i \wedge C_{i+1} \wedge R_i \tag{7}$$

Note that

$$F_{m,n} = \bigwedge_{i=0}^{n-1} F_{m,i,i+1} \tag{8}$$

If $\phi : \{x_{0i}, \ldots, x_{mi}\} \rightarrow \{0,1\}$ is an assignment of values for the variables of C_i (partial assignments of the variables of $F_{m,n}$), is denoted by the $(m+1)$-vector $(\phi(x_{0i}), ..., \phi(x_{mi}))$. Also, observe that can be considered as a partial assignment on the nodes of a k column induced vector from the oriented grid graph $G_{m,n}$.

$$\mathbf{x}_k = (x'_{k0}, x_{k1}, x'_{k1}, ..., x'_{k,m-1}, x_{km})$$

where $\phi(x_{ki}) = \phi(x'_{ki})$ for $i = 1, ..., m-1$. That is, an assignment for the variables of C_i can be seen as a vector in $\{0,1\}^{2m}$.

To prove theorem 1, first, we characterize the partial assignments of the variables of $F_{m,n}$, such that satisfy each column-formula C_i (lemma 1). Second, we characterize the pairs of assignments that satisfy the formula (8), i.e. satisfy two consecutive column-formulas C_i, C_{i+1} and the respective row-formula R_i (lemma 2). Finally, we prove that all matrix of partial assignments derived from the lemmas 1 and 2, satisfies the formula $F_{m,n}$. Next, for simplicity, we omit the index i of $v_{ji}, x_{ji}, s_{ji}, r_{ji}, s'_{ji}$ and r'_{ji}.

Lemma 1. *The vector* $\mathbf{x} \in \{0,1\}^{2m}$ *satisfies the formula (4) if and only if* $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}}$*, where* $\overline{\mathbf{x}} = -\mathbf{x}$ *and* $\mathbf{s}$ *is the sign vector of* C_i*.*

Proof. If we assume that $\mathbf{x} = (x'_0, x_1, x'_1 ..., x'_{m-1} x_m)$ satisfies the formula (4) and $\mathbf{s} = (s'_0, s_1, s'_1, ..., s'_{m-1}, s_m)$, then $(s'_\ell x_\ell \vee s_{\ell+1} x_{\ell+1}) = 1$ for all $\ell \in \{0, ..., m-1\}$, that is equivalent to $(s'_\ell \overline{x}_\ell, s_{\ell+1}\overline{x}_{\ell+1}) \neq (1,1)$.

It is straightforward to verify that the vector

$$(s'_0\overline{x}_0, s_1\overline{x}_1, s'_1\overline{x'}_1, ..., s'_{m-1}\overline{x}_{m-1}, s'_m\overline{x}_m) = \mathbf{s} \odot \overline{\mathbf{x}}$$

does have no two consecutive 1's, since any case is implied by conditions $\phi(x_{ki}) = \phi(x'_{ki})$ for $i = 1, ..., m-1$ and $(s'_\ell\overline{x}_\ell, s_{\ell+1}\overline{x}_{\ell+1}) \neq (1,1)$. Therefore, $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}}$.

Suppose that $\mathbf{s} \odot \mathbf{x} \in \mathcal{C}_m$, for $\ell = 0, ..., m$ then $(s'_\ell\overline{x}_\ell, s_{\ell+1}\overline{x}_{\ell+1})$ does not have two consecutive $1's$. The vector $\mathbf{x}$ satisfies the column-formula C_i (equation (4)), otherwise, there is $\ell \in \{0, ..., m-1\}$ such that $s'_\ell x_\ell \vee s_{\ell+1}x_{\ell+1} = 0$, then $s'_\ell\overline{x}_\ell = 1$ and $s_{\ell+1}\overline{x}_{\ell+1} = 1$ (contradiction). □

Lemma 2. *The pair* $(\mathbf{x}, \mathbf{y}) \in \{0,1\}^{4m}$ *satisfies* $F_{m,i,i+1}$ *if and only if* $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \mathcal{F}_m^{\mathbf{s}_i} \times \mathcal{F}_m^{\mathbf{s}_{i+1}}$ *and* $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$*.*

Proof. Suppose that $\mathbf{x} = (x'_0, x_1, x'_1 ..., x'_{m-1}, x_m)$ and $\mathbf{y} = (y'_0, y_1, y'_1 ..., y'_{m-1}, y_m)$ are such that $(\mathbf{x}, \mathbf{y})$ satisfies $F_{m,i,i+1}$. From lemma 1, $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}_i}$ and $\overline{\mathbf{y}} \in \mathcal{F}_m^{\mathbf{s}_{i+1}}$, we must prove that $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$.

By hypothesis $(r'_{ij}x_j \vee r_{i+1,j}y_j) = 1$ for all $j = 0, ..., m$, then $r'_{ij}\overline{x}_j \wedge r_{i+1,j}\overline{y}_j = 0$ for all $j = 0, ..., m$, therefore $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$.

If $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}_i}$ and $\overline{\mathbf{y}} \in \mathcal{F}_m^{\mathbf{s}_{i+1}}$, from lemma 1, $\mathbf{x}$ satisfies C_i and $\mathbf{y}$ satisfies C_{i+1}. Now, if $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$, then $r'_{ij}\overline{x}_j \cdot r_{i+1,j}\overline{y}_j = 0$ for all $j = 0, ..., m$, hence $(r'_{ij}x_j \vee r_{i+1,j}y_j) = 1$ for all $j = 0, ..., m$. Therefore $(\mathbf{x}, \mathbf{y})$ satisfies the row-formula R_j (equation (5)) for $j = 0, ..., m$. □

Remark 2. Notice that if $\mathcal{F}_m^{\mathbf{s}_i} = \{\mathbf{x}_0^i, ..., \mathbf{x}_{r_i}^i\}$, then the sum of the entries of the matrix $T_i = (a_{kl}^i)_{r_{i+1} \times r_i}$ where $a_{kl}^i = 1$ if $(\overrightarrow{\mathbf{r}}_{i+1} \odot \mathbf{x}_k^{i+1}) \cdot (\overleftarrow{\mathbf{r}}_i \odot \mathbf{x}_l^i) = 0$ and $a_{kl}^i = 0$ otherwise, is $\#SAT(F_{m,i,i+1})$, that is, there is a bijection between the set of non zero entries of T_i and the set of satisfying assignments of $F_{m,i,i+1}$. Therefore, from previous lemma we have that $\mathbf{1}^t T_i \mathbf{1} = \#SAT(F_{m,i,i+1})$, where T_i is the transfer matrix of the column i to the column $i+1$ of $G_{m,n}$ (the constrained graph of $F_{m,n}$). Finally, we prove the theorem 1.

Proof (Theorem 1). From equation (8), it is clear that the vector $(\mathbf{x}_0, ..., \mathbf{x}_n)$ $\in \{0,1\}^{2m(n+1)}$ satisfies the formula $F_{m,n}$ if and only if $(\mathbf{x}_i, \mathbf{x}_{i+1})$ satisfies $F_{m,i,i+1}$ for $i = 0, ..., n-1$. By lemma 2, $(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_{i+1}) \in \mathcal{F}_m^{\mathbf{s}_i} \times \mathcal{F}_m^{\mathbf{s}_{i+1}}$ and

$$(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$$

for $i = 0, ..., n-1$. Let $a^i_{l_{i+1}l_i}$ be the entry of the transfer matrix T_i in position

$$(\bar{\mathbf{x}}_{i+1}, \bar{\mathbf{x}}_i) \in \mathcal{F}_m^{\mathbf{s}_{i+1}} \times \mathcal{F}_m^{\mathbf{s}_i}$$

Then, by definition of T_i and previous analysis,

$$(\mathbf{x}_0, ..., \mathbf{x}_n) \in \{0,1\}^{(2m)(n+1)}$$

satisfies the formula $F_{m,n}$ if and only if

$$(\bar{\mathbf{x}}_0, ..., \bar{\mathbf{x}}_n) \in \mathcal{F}_m^{\mathbf{s}_0} \times \cdots \times \mathcal{F}_m^{\mathbf{s}_n} \texttt{ and } a^{n-1}_{l_n l_{n-1}} \cdots a^0_{l_1 l_0} = 1$$

Therefore $\#SAT(F_{m,n})$ is the cardinality of the set

$$\{(\bar{\mathbf{x}}_0, \cdots, \bar{\mathbf{x}}_n) \in \mathcal{F}_m^{\mathbf{s}_0} \times \cdots \times \mathcal{F}_m^{\mathbf{s}_n} : a^{n-1}_{l_n l_{n-1}} \cdots a^0_{l_1 l_0} = 1\}$$

Taking into account all the terms $a^{n-1}_{l_n l_{n-1}} \cdots a^0_{l_1 l_0} = 0$, we obtain

$$\#SAT(F_{m,n}) = \sum_{(l_0,...,l_n) \in I_0 \times \cdots \times I_n} a^{n-1}_{l_n l_{n-1}} \cdots a^1_{l_2 l_1} \cdot a^0_{l_1 l_0} = \mathbf{1}^t T_{n-1} \cdots T_0 \mathbf{1}$$

where $I_k = \{0, ..., t_k\}$, $t_k = \mid \mathcal{F}_m^{\mathbf{s}_k} \mid$ for $k = 0, ..., n$. □

In [2] we show a possible application of our method for modeling a distributed work among a team formed by 16 remote collaborators, and which is represented by a network grid. Some constraints are defined on the orientations which distribute the flow of tasks among the members of the team. Our method is applied for counting the number of different paths of the oriented grid where such paths violate the defined constraints.

5 Conclusions

We have considered the different pattern orientations of an undirected edge, and applying those pattern on a grid graph G, we develop a new matrix method for recognizing and counting the number of sink-free orientations of G for solving its related constraint satisfaction problem; to count the number of models of a 2-CNF on formulas whose constrained graph is a grid graph.

The type of orientations considered here generalize the class of problems which could be modeled and solved through methods applied in the area of Constraint Satisfaction Problems.

References

1. Baxter, R.J.: Planar Lattice Gases with Nearest-Neighbour Exclusion. Annals of Combinatorics 3, 191–203 (1999)
2. Guillén, C., Vera, E., López-López, A., De Ita, G.: Applying the Transfer Matrix Method for Supervising Lines in a Network Grid. In: Proc. REV 2008 (2008),`www.rev-conference.org`

3. Barbosa, V.C., Ferreira, R.G.: On the phase transitions of graph coloring and independent sets. Physica A: Statistical Mechanics and its Applications 343, 401–423 (2004)
4. Calkin, N.J.: The Number of Independent Sets in a Grid Graph. SIAM Journal on Discrete Mathematics 11(1), 54–60 (1998)
5. Gärtner, B., Morris, W., Rüst, L.: Unique Sink Orientations of Grids. Algorithmica 51(2), 200–235 (2008)
6. Golin, M.J., Leung, Y.C., Wang, Y., Yong, X.: Counting Structures in Grid Graphs, Cylinders and Tori Transfer Matrices: Survey and New Results. In: ALENEX/ANALCO, pp. 250–258 (2005)
7. Grohe, M., Marx, D.: Constraint Solving via Fractional Edge Covers. In: Soda'06, Miami (2006)
8. Reinhardt, E.: The Fibonacci Number of a Grid Graph and a New Class of Integer Sequences. JIS Journal of Integer Sequences 88(2), 1–16 (2005) Article 05.2.6
9. Roth, D.: On the Hardness of Approximate Reasoning. Artificial Intelligence, 273–302 (1996)
10. Roth, R.M., Siegel, P.H., Wolf, J.K.: Efficient Coding Schemes for the Hard-Square Model. IEEE Trans. Inform. Theory 47, 1166–1176 (2001)
11. Russ, B.: Randomized Algorithms: Approximation, Generation, and Counting, Distinguished dissertations. Springer, Heidelberg (2001)
12. Vadhan Salil, P.: The complexity of Counting in Sparse, Regular, and Planar Graphs. SIAM Journal on Computing 31(2), 398–427 (2001)
13. Valiant, L.G.: The complexity of enumeration and reliability problems. SIAM J. Comput. 8(3) (1979)