Lattices generated by Chip Firing Game: characterizations and recognition algorithm

PHAN Thi Ha Duong

Institute of Mathematics
Vietnam Academy of Science and Technology

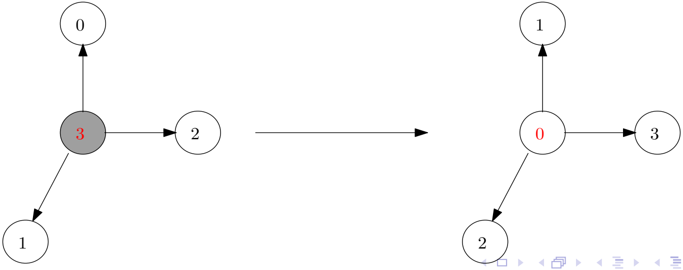
Journées SDA2 2017 : Systèmes Dynamiques, Automates and Algorithmes

Chip firing games (CFG) on a graph: Definition

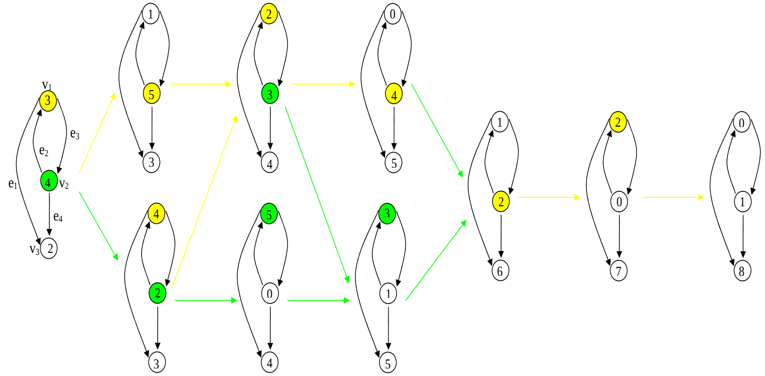
Given $G = (V, E)$ directed multigraph. A model of chip firing games is described by

(i) Configurations: each configuration is a distribution of chips on V.

(ii) Firing rule: One vertex can fire if its chips is at least its out-degree. When it fires, it gives one chip to each of its neighbors.

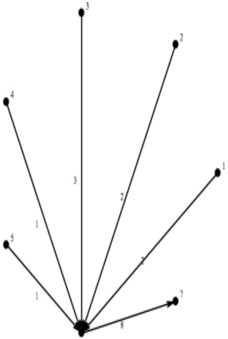


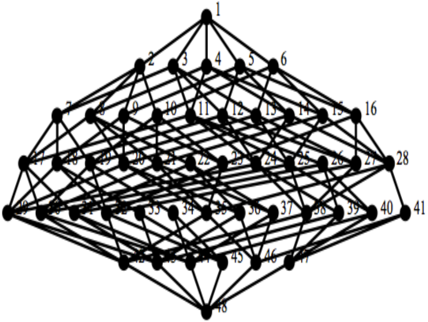
Small example of configuration space of CFG



Example of configuration space of CFG

CFG on a graph of 7 vertices





Origin of the Model

- Chip Firing Game is a discrete dynamical model defined by D. Dhar (1990) and by A. Björner, L. Lovász and W. Shor (1991).

Origin of the Model

- Chip Firing Game is a discrete dynamical model defined by D. Dhar (1990) and by A. Björner, L. Lovász and W. Shor (1991).
- This model has applications in physics, computer science, social science and pure mathematics, etc.

Origin of the Model

- Chip Firing Game is a discrete dynamical model defined by D. Dhar (1990) and by A. Björner, L. Lovász and W. Shor (1991).
- This model has applications in physics, computer science, social science and pure mathematics, etc.
- This model relates to combinatorial and algebraic objects such as lattice, matroid, spanning tree, group, etc.

Chip firing games as model to study dynamical systems

- Vectorial space: Spencer (1983)
- Automata: Durand-Lose (1999) Masson and Formenti (2006)
- Economy: Dollar game: N. Biggs (1999);
- Discrete dynamical systems: Self-Organized Criticality: Bak, Tang and Weisenfeld(1987), Latapy, Morvan and P. (1998, 2004), Goles and Kiwi (1993), ...

Chip firing games as a tool to study structures of graphs, lattices, matroids

- Algebra: Sandpile group: Dhar et al. (1995), Biggs (1999), Cori, Le Borgne and Rossin (2000)
- Structures on graphs, Matroid: C. Merino (2001); S. Oh (2010).
- CFG and Rotor-router: Levine, Propp et al. (2007, 2008)
- CFG and random spanning trees: Baker and Shokrieh (2011)
- CFG and ULD lattices: Knauer and Felsner (2009 - 2011)

Some Questions

- Under which condition, the game stops after a finite steps ?
- Does the game converge or not? (From an initial configuration, the game reaches a unique stable configuration?)
- In the case of convergent game, what is the behavior of all possible evolutions ?
- Is there any characterization or algorithm to detect the structure of such a space configuration ? (Yes: Lattice structure !)

Some Questions (2)

- In the case of convergent game, how one can know which is the stable configuration of a given initial configuration.
- Reachabality problem: given two configurations, how one can know if one is reachable from another.
- From one initial configuration, the system converges to a stable configurations. How one can say about the set of all stabe configurations ?
- Critical configurations: special stable configurations (which can be recurrent under some condition).
- What is the structure of the set of all critical configuration.
- How one can deduce properties of a graphs from its critical configurations.

Two directions to study CFG

Vertically

- Fix an initial configuration, consider the set of all reachable configurations (the structure of configuration space).
- Study the order of this configuration space: theory of order, lattice, distributive lattice, etc.
- Find a characterization of class of lattices generated by CFG.
- Recognition algorithm of the class of lattices generated by CFG and related models.

Two directions to study CFG (in this course)

Horizontally

- Consider the set of all critical configurations.
- Group structure of the set of critical configurations: the Sandpile group.
- Relation between critical configurations and clasical objects on graphs: spanning trees, Graphic Matroid ...
- Enumeration of critical configurations: Tutte polynomial.

Convergent properties

- **Theorem.** (Björner, Lovász and Shor, '91)
  For a undirected connected graph with a sink, from an initial configuration, the game converges to a stable configuration.

Convergent properties

- **Theorem.** (Björner, Lovász and Shor, '91)
  For a undirected connected graph with a sink, from an initial configuration, the game converges to a stable configuration.
- **Theorem.** (Björner and Lovász, '92; Latapy and P, '01)
  For a directed connected graph without non trivial closed component, from an initial configuration, the game converges to a stable configuration.

Convergent properties

- **Theorem.** (Björner, Lovász and Shor, '91)
  For a undirected connected graph with a sink, from an initial configuration, the game converges to a stable configuration.
- **Theorem.** (Björner and Lovász, '92; Latapy and P, '01)
  For a directed connected graph without non trivial closed component, from an initial configuration, the game converges to a stable configuration.
- **Theorem.** (Holroyd et al., '08)
  For a directed connected graph with a global sink, from an initial configuration, the game converges to a stable configuration.

Order on configuration space of CFG

- Order on CFG: configuration b is greater than configuration a if b can be obtained from a by applying a sequence of firings. Note: This is a well defined order due from the convergent properties of CFG.

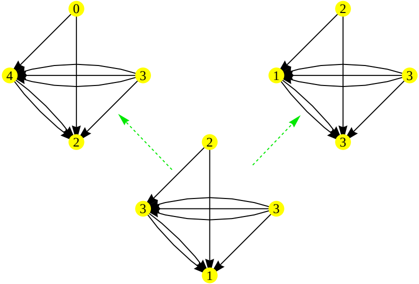
$\mathcal{L}(CFG)$: set of all $CFG(G, O)$ with all graphs G and all initial configurations O.

Order on configuration space of CFG

- Order on CFG: configuration b is greater than configuration a if b can be obtained from a by applying a sequence of firings. Note: This is a well defined order due from the convergent properties of CFG.
- Configuration space $CFG(G, O)$: the set of all reachable configurations from an initial one O.

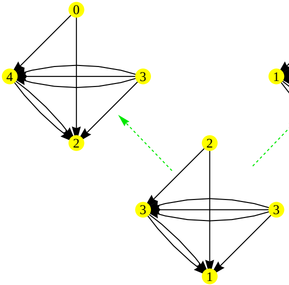
$\mathcal{L}(CFG)$: set of all $CFG(G, O)$ with all graphs G and all initial configurations O.

Examples of configuration space of CFG

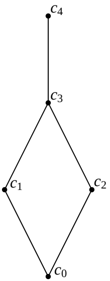


A step of evolution

Examples of configuration space of CFG

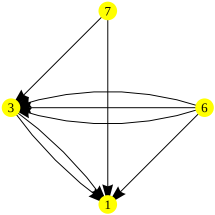


A step of evolution



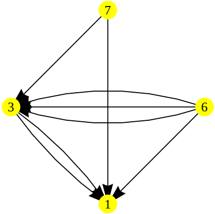
Generating lattice

Examples of configuration space of CFG (2)

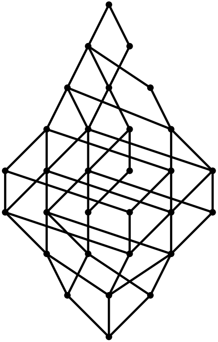


Changing initial configuration

Examples of configuration space of CFG (2)



Changing initial configuration



Generating lattice

Order, Lattice, Distributive lattice, ULD

- A partial order is a binary relation which is reflexive, antisymetric and transitive.

Order, Lattice, Distributive lattice, ULD

- A partial order is a binary relation which is reflexive, antisymetric and transitive.
- In a partial order set, for two elements a and b, the infimum of a and b, $inf(a, b)$ (if it exists) is the maximum element which is smaller than a and b. Duality for supremum of a and b: $sup(a, b)$.

Order, Lattice, Distributive lattice, ULD

- A partial order is a binary relation which is reflexive, antisymetric and transitive.
- In a partial order set, for two elements a and b, the infimum of a and b, $inf(a, b)$ (if it exists) is the maximum element which is smaller than a and b. Duality for supremum of a and b: $sup(a, b)$.
- A lattice is a partial order such that for every couple a and b, the $inf(a, b)$ and $sup(a, b)$ exist.

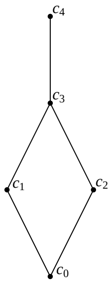
Order, Lattice, Distributive lattice, ULD

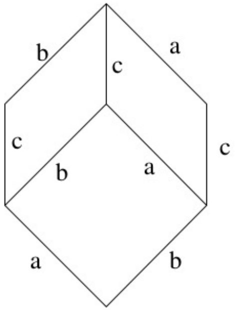
- A partial order is a binary relation which is reflexive, antisymetric and transitive.
- In a partial order set, for two elements a and b, the infimum of a and b, $inf(a, b)$ (if it exists) is the maximum element which is smaller than a and b. Duality for supremum of a and b: $sup(a, b)$.
- A lattice is a partial order such that for every couple a and b, the $inf(a, b)$ and $sup(a, b)$ exist.
- A distributive (D) lattice is a lattice such that two operators inf and sup have distributive property.

Order, Lattice, Distributive lattice, ULD

- A partial order is a binary relation which is reflexive, antisymetric and transitive.
- In a partial order set, for two elements a and b, the infimum of a and b, $inf(a, b)$ (if it exists) is the maximum element which is smaller than a and b. Duality for supremum of a and b: $sup(a, b)$.
- A lattice is a partial order such that for every couple a and b, the $inf(a, b)$ and $sup(a, b)$ exist.
- A distributive (D) lattice is a lattice such that two operators inf and sup have distributive property.
- A upper locally distributive (ULD) lattice is lattice such that: for any element a and if b is the sup of all upper covers of a, then the interval $[a, b]$ is a hypercube.

Example of Distributive and ULD lattice





Distributive lattices $\subsetneq$ ULD laticces

- **Distributive lattices**
  - Domino and lozenge tilings of a plane region Rémila ('04)
  - Planar spanning trees Gilmer and Litherland ('86)
  - Eulerian orientations of a planar graph Felsner ('04)
  - c-orientations of a graph Propp ('93)

Distributive lattices $\subsetneq$ ULD laticces

- **Distributive lattices**
  - Domino and lozenge tilings of a plane region Rémila ('04)
  - Planar spanning trees Gilmer and Litherland ('86)
  - Eulerian orientations of a planar graph Felsner ('04)
  - c-orientations of a graph Propp ('93)
- **Upper locally distributive lattices**
  - Subtrees of a tree Boulaye ('67)
  - Convex subsets of a poset Birkhoff and Bennett ('85)
  - Transitively oriented subgraphs of a transitively oriented digraph Björner ('85)
  - Feasible multi-sets of an antimatroid with repitition Björner and Ziegler ('92)

Lattice structure of Chip Firing Game

- Theorem (Björner and Lovász '92, Latapy and P. '01)
  For a given graph G and an initial configuration O, the configuration space $CFG(G, O)$ has a ULD lattice structure.

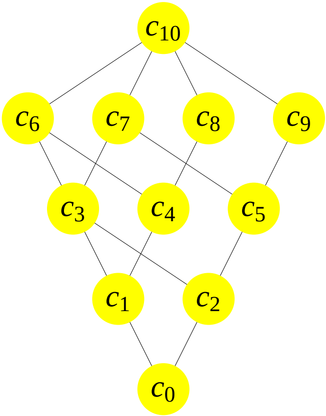
Lattice structure of Chip Firing Game

- Theorem (Björner and Lovász '92, Latapy and P. '01)
  For a given graph G and an initial configuration O, the configuration space $CFG(G, O)$ has a ULD lattice structure.
- Theorem (Magnien, P. and Vuillon '03)
  All distributive lattices can be genereted by CFG.

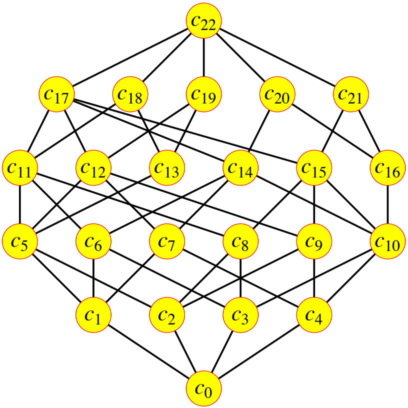
Lattice structure of Chip Firing Game

- Theorem (Björner and Lovász '92, Latapy and P. '01)
  For a given graph G and an initial configuration O, the configuration space $CFG(G, O)$ has a ULD lattice structure.
- Theorem (Magnien, P. and Vuillon '03)
  All distributive lattices can be genereted by CFG.
- Theorem (Magnien, P. and Vuillon '03)
  $D \subsetneq \mathcal{L}(CFG) \subsetneq ULD$

Example of a lattice of CFG which is not in D



Example of a ULD lattice which is not a lattices of CFG



The smallest ULD lattice which is not a lattices of CFG

CFG lattice vs ULD lattice

- Problem 1: Find a lattice characterization of ULD's that can be generated by CFGs.

CFG lattice vs ULD lattice

- Problem 1: Find a lattice characterization of ULD's that can be generated by CFGs.
- Problem 2: Find a generalization of CFGs generating the whole class of ULDs.

CFG lattice vs ULD lattice

- Problem 1: Find a lattice characterization of ULD's that can be generated by CFGs.
- Problem 2: Find a generalization of CFGs generating the whole class of ULDs.
- **Answer for Problem 2:**

CFG lattice vs ULD lattice

- Problem 1: Find a lattice characterization of ULD's that can be generated by CFGs.
- Problem 2: Find a generalization of CFGs generating the whole class of ULDs.
- **Answer for Problem 2:**

- **Theorem.** (Magnien, P. and Vuillon '01)
  The class of lattices generated by coloured CFG is exactly the class of ULD lattices.

CFG lattice vs ULD lattice

- Problem 1: Find a lattice characterization of ULD's that can be generated by CFGs.
- Problem 2: Find a generalization of CFGs generating the whole class of ULDs.
- **Answer for Problem 2:**

- **Theorem.** (Magnien, P. and Vuillon '01)

  The class of lattices generated by coloured CFG is exactly the class of ULD lattices.

- **Theorem.** (Knauer '10)

  Every ULD can be represented by a simple generilized CFG.

Characterization of ULD's that can be generated by CFGs.

- **Question: How to know if a given ULD lattice belongs to $\mathcal{L}(CFG)$ or not?**

Characterization of ULD's that can be generated by CFGs.

- **Question: How to know if a given ULD lattice belongs to $\mathcal{L}(CFG)$ or not?**
- **Our algorithm:**
  - Input: A ULD lattice L.
  - Output: Is L in $\mathcal{L}(CFG)$? If Yes then construct a corresponding CFG to generate L.

Characterization of ULD's that can be generated by CFGs.

- **Question: How to know if a given ULD lattice belongs to $\mathcal{L}(CFG)$ or not?**
- **Our algorithm:**
  - Input: A ULD lattice L.
  - Output: Is L in $\mathcal{L}(CFG)$? If Yes then construct a corresponding CFG to generate L.
  - Find full characterizations for $\mathcal{L}(CFG)$.

Characterization of ULD's that can be generated by CFGs.

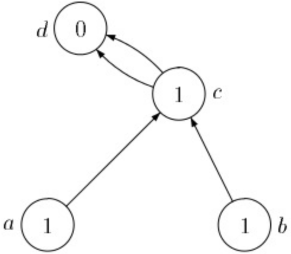
- **Question: How to know if a given ULD lattice belongs to $\mathcal{L}(CFG)$ or not?**
- **Our algorithm:**
  - Input: A ULD lattice L.
  - Output: Is L in $\mathcal{L}(CFG)$? If Yes then construct a corresponding CFG to generate L.
  - Find full characterizations for $\mathcal{L}(CFG)$.
  - Using CFG and related models, construct a filter from the class of distributive lattices to the class of ULD lattices.

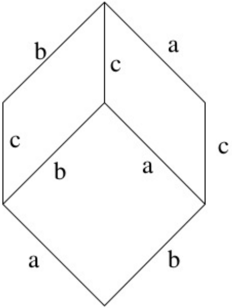
CFG vs Simple CFG

- A simple CFG is a CFG on which each vertex is fired at most once.
- Two convergent CFG are equivalent if their lattices of configuration space are isomorphic.

- **Theorem.** (Magnien, P. and Vuillon '01)
  Any convergent CFG is equivalent to a simple CFG.

Small example of CFG and ULD lattice with labeled cover relation

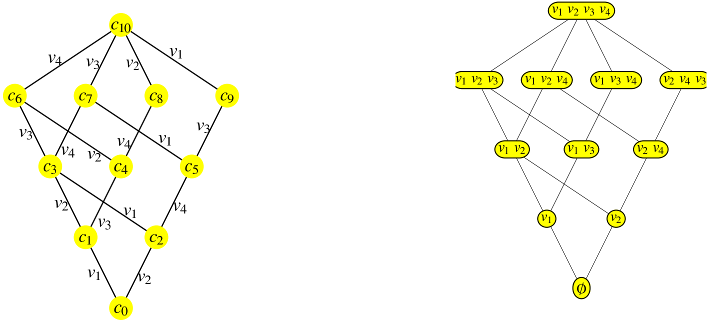




Coding each configuration of (simple) CFG by its set of firing vertices

Let L be the lattice generated by a $CFG(G, O)$:

- x of L coded by $V_x =$ firing vertices to obtained x from O.
- Cover $x \prec y$ (x obtained from y by firing v) is coded by v.



Meet irreducibles of D and ULD lattices

Definition

An element m in lattice L is called meet-irreducible if m has a unique upper cover.

Theorem. (Birkhoff '40)

A lattice is distributive if and only if it is isomorphic to the lattice of the ideals of the order induced by its meet-irreducibles.

Lemma (Caspard '98)

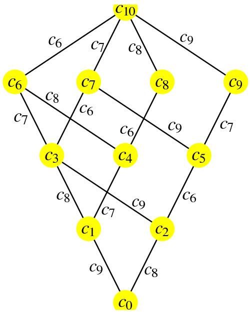
A lattice L is ULD if and only if for all $x, y \in L$,

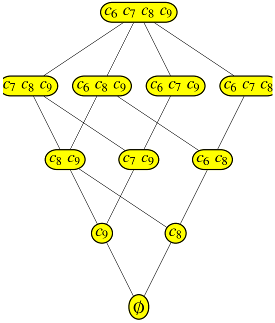
$$x \prec y \leftrightarrow M_y \subset M_x \text{ and } |M_y \backslash M_x| = 1.$$

(Here, $M_x = \{m \in M, m \geq x\}$).

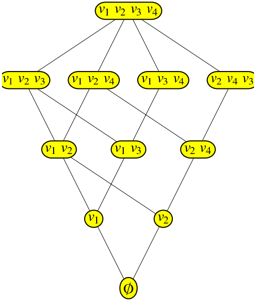
Coding each configuration of a ULD lattice a set of meet-irreducibles

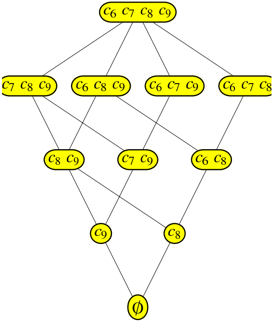
Let L be a ULD lattice, and M be its set of meet irreducibles.
- Each element x of L can be coded by $V_x = \{m \in M, m \not\geq x\}$.
- Each cover relation $x \prec y$ can be codes by $V_y \backslash V_x$.





Coding by set of firing vertices and coding by set of meet-irreducibles





Map from a CFG to a ULD lattice

Idea of the algorithm:
If CFG on G corresponds to a lattice L then
$V(G)$ corresponds to the set of meet-irreducibles of L.

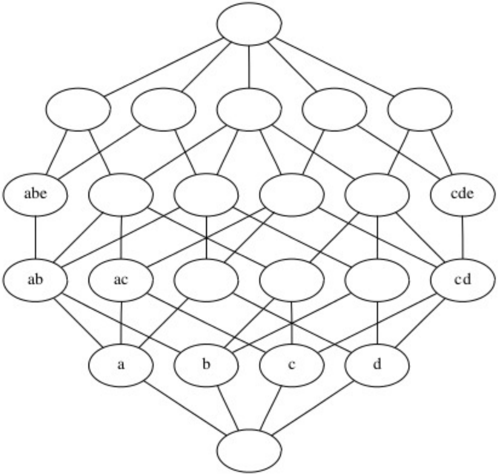
Construction of a CFG from a distributive lattice

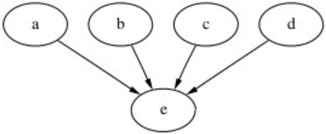
Let L be a lattice and M be its set of meet-irreducibles.

-
- The graph $G = (V, E)$ is contructed as follow:
- $V = M \cup s$
- E is convering edges in the order M, plus:
  - $indeg_M(v) - outdeg_M(v)$ edges from v to s (if it is positive)
  - one edge from v to s if v is isolated.
- The initial configuration is defined by: $c(v) = indeg_G(v) - outdeg_G(v)$

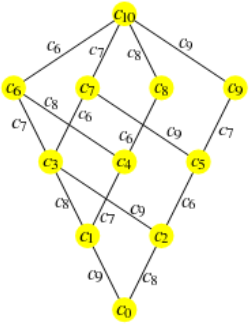
Theorem. (Magnien, P. and Vuillon '01)
$CFG(G, 0)$ is isomorphic with L.

Why a ULD can not be generated by a CFG ?



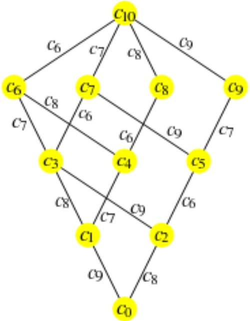


Example



In $\mathcal{E}(c_6)$, w is the number of chips needed to add to c_6 for firing it; and e_{c_8} is the number of edges from c_8 to c_6.

Example



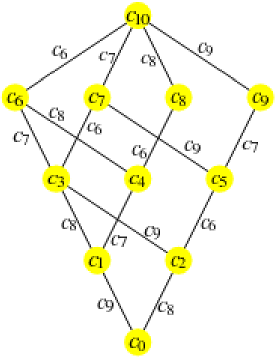
$$\mathcal{E}(c_6) = \begin{cases} w \leq e_{c_8} \\ w \leq e_{c_7} + e_{c_9} \\ e_{c_9} < w \end{cases}$$

$$\mathcal{E}(c_8) = \mathcal{E}(c_9) = \{w \geq 1\}$$

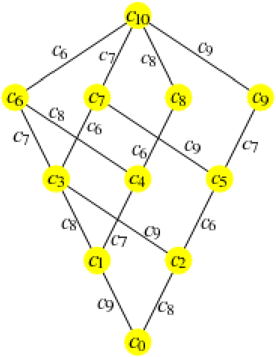
$$\mathcal{E}(c_7) = \begin{cases} w \leq e_{c_9} \\ w \leq e_{c_6} + e_{c_8} \\ e_{c_8} < w \end{cases}$$

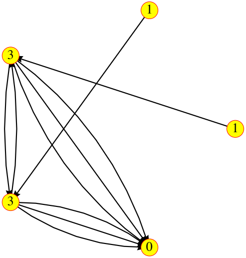
In $\mathcal{E}(c_6)$, w is the number of chips needed to add to c_6 for firing it; and e_{c_8} is the number of edges from c_8 to c_6.

Example



Example





System of linear inequalities of firing process

Let $L = (X, \leq)$ be a finite lattice. For each $m \in M$

- $\mathfrak{U}_m$: minimal elements on which m can be applied.

System of linear inequalities of firing process

Let $L = (X, \leq)$ be a finite lattice. For each $m \in M$

- $\mathfrak{U}_m$: minimal elements on which m can be applied.
- $\mathfrak{L}_m$: maximal elements on which m can not be applied.

System of linear inequalities of firing process

Let $L = (X, \leq)$ be a finite lattice. For each $m \in M$

- $\mathfrak{U}_m$: minimal elements on which m can be applied.
- $\mathfrak{L}_m$: maximal elements on which m can not be applied.
-
  - $\mathfrak{U}_m = \{j^- : j \in J \text{ and } j \downarrow m\}$ (J: join irreducibles of L).
  - $\mathfrak{L}_m =$ maximals of $\{X \setminus \bigcup_{a \in \mathfrak{U}_m} \{x \in L : a \leq x\}\}$

System of linear inequalities of firing process

Let $L = (X, \leq)$ be a finite lattice. For each $m \in M$

- $\mathfrak{U}_m$: minimal elements on which m can be applied.
- $\mathfrak{L}_m$: maximal elements on which m can not be applied.
-
  - $\mathfrak{U}_m = \{j^- : j \in J \text{ and } j \downarrow m\}$ (J: join irreducibles of L).
  - $\mathfrak{L}_m =$ maximals of $\{X \setminus \bigcup_{a \in \mathfrak{U}_m} \{x \in L : a \leq x\}\}$
- $\mathcal{E}(m) = \{ \sum_{x \in V_a} e_x < w : a \in \mathfrak{L}_m \}$

$$\cup \{w \leq \sum_{x \in V_a} e_x : a \in \mathfrak{U}_m\} \cup \{w \geq 1\}$$

(e_x is the number of edges from x to m.)

Main theorem

Theorem.
$L \in \mathcal{L}(CFG)$ if and only if for each meet- irreducible m, $\mathcal{E}(m)$ has non-negative solutions.

Main theorem

Theorem.
$L \in \mathcal{L}(CFG)$ if and only if for each meet- irreducible m, $\mathcal{E}(m)$ has non-negative solutions.

By using Karmarkar's algorithm, we can build an algorithm to run in time

$$O(|M|^{3.5} \times |J|^2 \times |L|^2 \times log(|L|)) \times log(log(|L|))$$

Algorithm

Input : A ULD lattice L which is input as a acyclic graph with the edges defined by the cover relation
Output: **Yes** if L is in $\mathrm{L(CFG)}$, **No** otherwise. If **Yes** then give a support graph G and an initial configuration c_0 on G so that $\mathrm{CFG(G, c_0)}$ is isomorphic to L

$V(G) := M \cup \{s\}$;
$E(G) := \emptyset$;
for $m \in M$ **do**
  Construct $\mathcal{E}(m)$;
  if $\mathcal{E}(m)$ *has no non-negative integral solutions* **then Reject**;
  else
    Let f_m be a non-negative integral solution of $\mathcal{E}(m)$;
    Let U_m be the collection of all variables in $\mathcal{E}(m)$;
    for $e_x \in U_m \backslash \{w\}$ **do**
      Add $f_m(e_x)$ edges (x, m) to G
    end
    Add $f_m(w) + \sum_{e_x \in U_m \backslash \{w\}} f_m(e_x)$ edges (m, s) to G
  end
end
Construct the initial configuration c_0 by

$$c_0(v) := \begin{cases} deg^+(v) & \text{if } deg^-(v) = 0 \\ deg^+(v) - f_v(w) & \text{if } deg^-(v) \neq 0 \text{ and } v \neq s \\ 0 & \text{if } v = s \end{cases}$$

Abelian Sandpile Model (ASM) and main result

- ASM is the CFG model which is restricted to undirected graphs

Abelian Sandpile Model (ASM) and main result

- ASM is the CFG model which is restricted to undirected graphs
- A vertex is chosen to be a sink (only receive chips, never fired during whole firing process)

Abelian Sandpile Model (ASM) and main result

- ASM is the CFG model which is restricted to undirected graphs
- A vertex is chosen to be a sink (only receive chips, never fired during whole firing process)

For each $m \in M$, replace each variable e_x in $\mathcal{E}(m)$ by $e_{x,m}$ and keep the linear inequalities, we obtain a new system $\mathcal{E}'(m)$

Abelian Sandpile Model (ASM) and main result

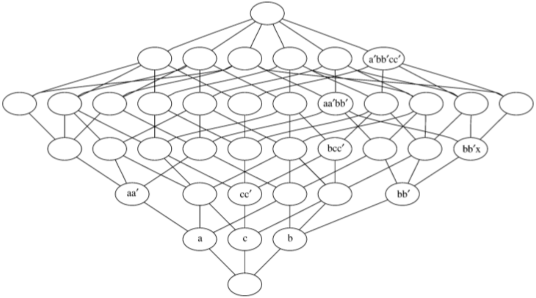
- ASM is the CFG model which is restricted to undirected graphs
- A vertex is chosen to be a sink (only receive chips, never fired during whole firing process)

For each $m \in M$, replace each variable e_x in $\mathcal{E}(m)$ by $e_{x,m}$ and keep the linear inequalities, we obtain a new system $\mathcal{E}'(m)$

Theorem.

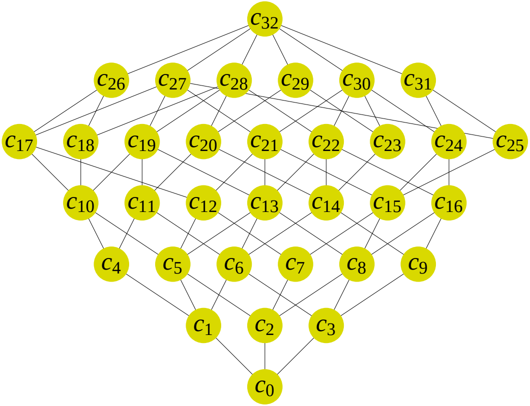
$L \in \mathcal{L}(ASM)$ if and only if $\{e_{m_1,m_2} = e_{m_2,m_1} | m_1, m_2 \in M\} \cup \bigcup_{m \in M} \mathcal{E}'(m)$ has non-negative solutions

First example of a lattice in $\mathcal{L}(CFG)$ but not in $\mathcal{L}(ASM)$



[Magnien '03]

Smallest example of a lattice in $\mathcal{L}(CFG)$ but not in $\mathcal{L}(ASM)$



CFGs on DAGs and main theorem

Theorem.

Any CFG on an acyclic directed graph is equivalent to a ASM.
Therefore $\mathcal{L}(ACFG) \subsetneq \mathcal{L}(ASM)$.

Inclusion on classes of lattices

$$D \subsetneq \mathcal{L}(ACFG) \subsetneq \mathcal{L}(ASM) \subsetneq \mathcal{L}(CFG) \subsetneq ULD$$

Next problems

- What about $\mathcal{L}(ECFG)$? (CFG on Eulerian graphs)
- A characterization of classes of graphs which are closed by simple CFGs
- Given a pair (G, L), is L generated by a CFG on G for some initial configuration?

Main references

K. Meszaros Y. Peres J. Propp A. E. Holroyd, L. Levin and D. B. Wilson.
Chip firing and rotor-routing on directed graphs.
In and Out of Equilibrium II. Progress in Propability (Birkhauser 2008), 60.

A. Bjorner, L. Lovász, and W. Shor.
Chip-firing games on graphes.
E.J. Combinatorics, 12:283–291, 1991.

Phan Thi Ha Duong and Tran Thi Thu Huong.
On the stability of sand piles model.
Theoret. Comput. Sci., 411(3):594–601, 2010.

E. Goles and M.A. Kiwi.
Games on line graphes and sand piles.
Theoret. Comput. Sci., 115:321–349, 1993.

M. Latapy and H.D. Phan.
The lattice structure of chip firing games.
Physica D, 115:69–82, 2001.

C. Magnien, H. D. Phan, and L. Vuillon.
Characterisation of lattice induced by (extended) chip firing game.
Discrete Math. Theoret. Comput. Sci., AA:229–244, 2001.

Thi Ha Duong Phan.
Two sided sand piles model and unimodal sequences.
ITA, 42(3):631–646, 2008.

Trung Van Pham and Thi Ha Duong Phan.
Lattices generated by chip firing game models: characterizations and recognition algorithm

Thank you for your attention!