

# 自己組織化臨界現象と 可換砂山模型

香取眞理

(中央大学工学部物理学科)

Katori, Makoto (Chuo University)

[スライド part 2/5]

青山学院大学 物理・数理学科 コロキウム

2015年度 第7回

2015年10月22日

青山学院大学 工学部 L棟6階 L603室

## 目次:

1. 砂山模型と自己組織化臨界現象
2. 可換砂山模型(ASM)の定義と基本的な性質
3. 可換砂山模型(ASM)と離散ポアソン方程式
4. 再帰的配置と全域木(spanning tree)との1対1対応
5. 砂山(ASP)-全域木(spanning tree)の局所-大域双対性
6. 可換砂山模型(ASM)研究が教えること

## 2. 可換砂山模型 (Abelian Sandpile Model: ASM) の定義と基本的な性質

## 2.1 「格子上のモデル」というよりは 「グラフそのもので与えられる」閾値ダイナミクス (toppling: 転倒)が基礎

- 離散モデル

空間：グラフの格子点  $\mathbf{z} \in \Lambda$  上に変数を置く（以下説明）。

時間：  $t \in \mathbb{N}_0 \equiv \{0, 1, 2, \dots\}$ （これは次節 1.2 で説明）。

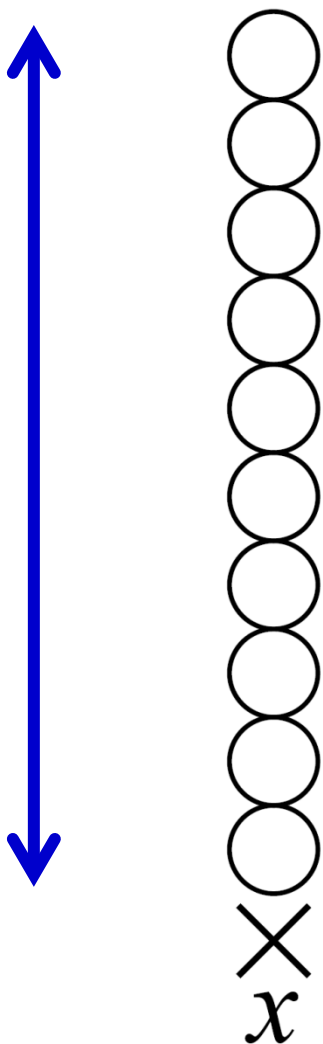
変数：砂山の高さ  $\eta_t(\mathbf{z}) \in \mathbb{N}_0 \equiv \{0, 1, 2, \dots\}$ （これは以下説明）。

(1) まず 1 つの点  $\mathbf{x}$  に注目する. 図では  $\times$  印で表した.

$\times$   
 $x$

(2) その上に砂粒 (砂山の単位 : a grain of sand) をいくつか重ねる.

重ねた砂粒の数 = 砂山の高さ =  $\eta_t(\mathbf{x}) \in \mathbb{N}_0 \equiv \{0, 1, 2, \dots\}$

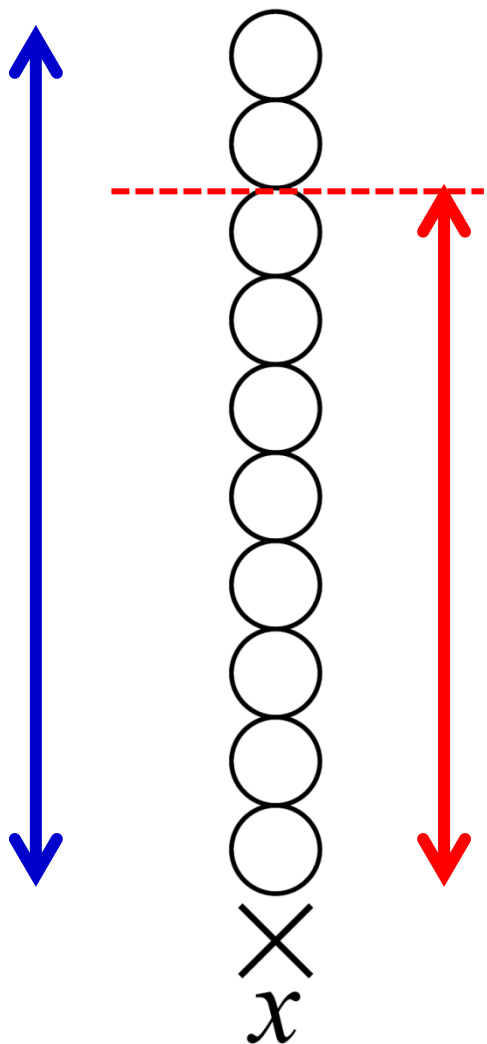


$\eta_t(\mathbf{x}) = 10$  の場合

(3) この点  $\mathbf{x}$  における砂山の安定な高さの閾値  $h_c(\mathbf{x}) \in \mathbb{N} \equiv \{1, 2, 3, \dots\}$  を定める.

$\eta_t(\mathbf{x}) < h_c(\mathbf{x}) \quad \dots \quad \mathbf{x}$  上の砂山は安定

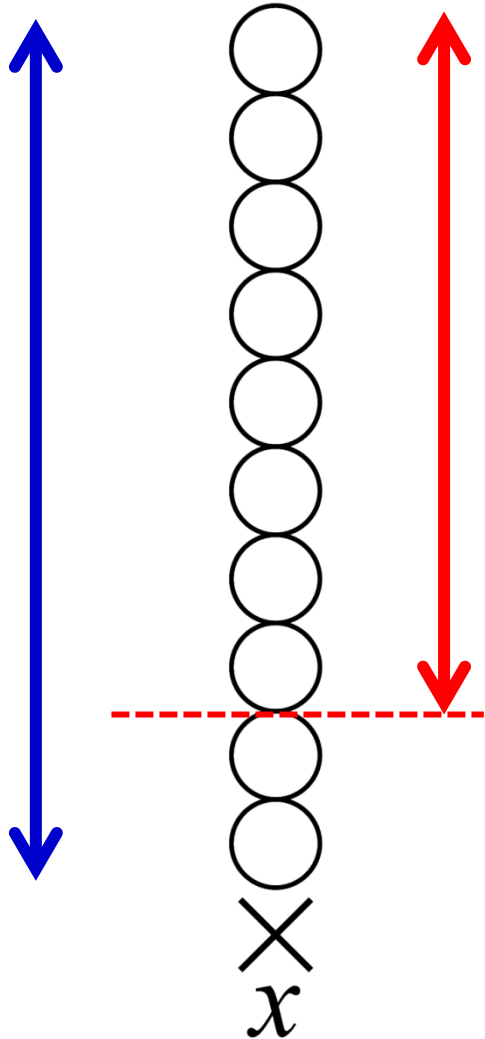
$\eta_t(\mathbf{x}) \geq h_c(\mathbf{x}) \quad \dots \quad \mathbf{x}$  上の砂山は不安定



ここでは  $h_c(\mathbf{x}) = 8$  とした.

$\eta_t(\mathbf{x}) = 10 \geq 8 = h_c(\mathbf{x})$  なので,  
この砂山は不安定である.

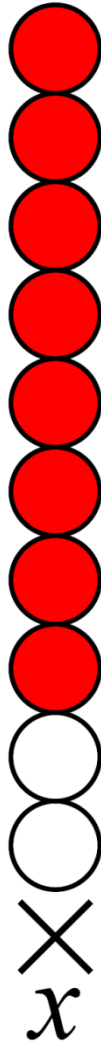
- (4)  $\eta_t(\mathbf{x}) \geq h_c(\mathbf{x})$  のとき, 砂山は topple (転倒) する.  
 $h_c(\mathbf{x})$  は単に閾値を与えるだけでなく,  
toppling の際に 周囲に飛び散る砂粒の総数 を指定する.



$$\eta_t(\mathbf{x}) = 10 \geq 8 = h_c(\mathbf{x})$$

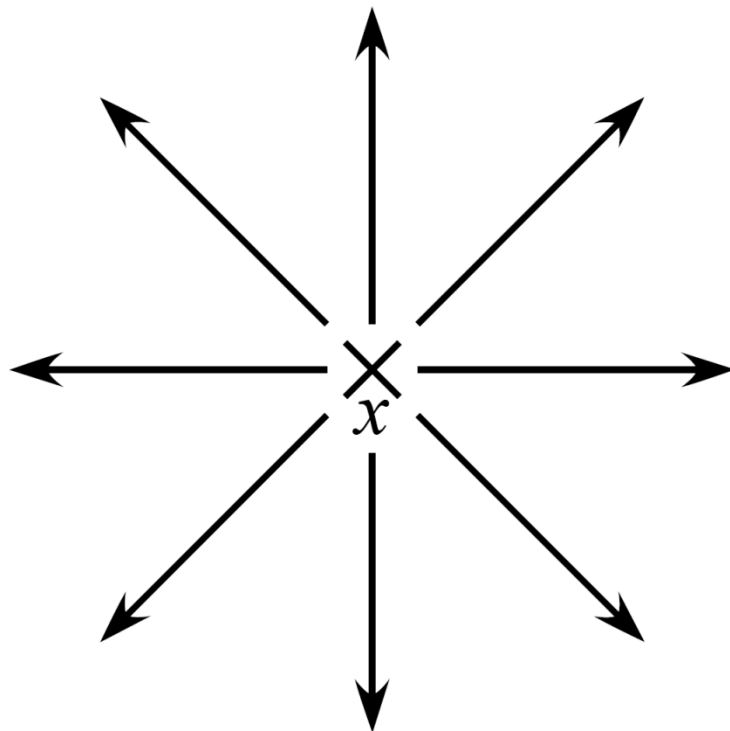


- (4)  $\eta_t(\mathbf{x}) \geq h_c(\mathbf{x})$  のとき, 砂山は topple (転倒) する.  
 $h_c(\mathbf{x})$  は単に閾値を与えるだけでなく,  
**toppling** の際に 周囲に飛び散る砂粒の総数 を指定する.

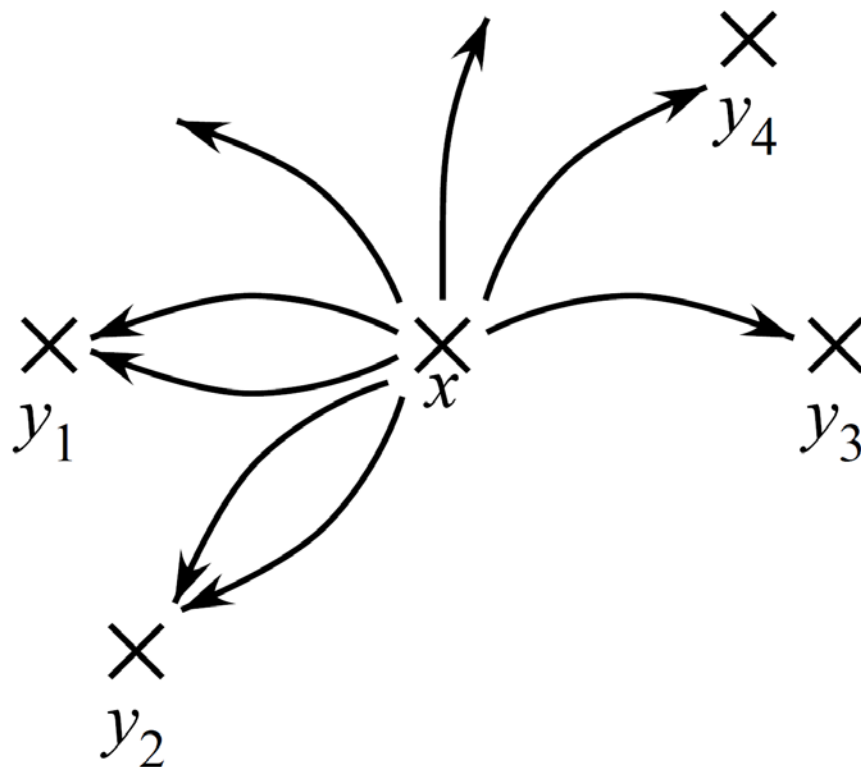


この例では,  $\eta_t(\mathbf{x}) = 10$  個の砂粒のうち  
 $h_c(\mathbf{x}) = 8$  個の砂粒 (赤い砂粒) が topple する.  
toppling の後, 点  $\mathbf{x}$  には砂粒は 2 個だけ残る.

- (5) 点  $\mathbf{x}$  から  $h_c(\mathbf{x})$  個の砂粒がどのように飛び散るかを矢印で指定したい.  
図の例では,  $\mathbf{x}$  を始点とする  $h_c(\mathbf{x}) = 8$  本の矢印を考えることになる.



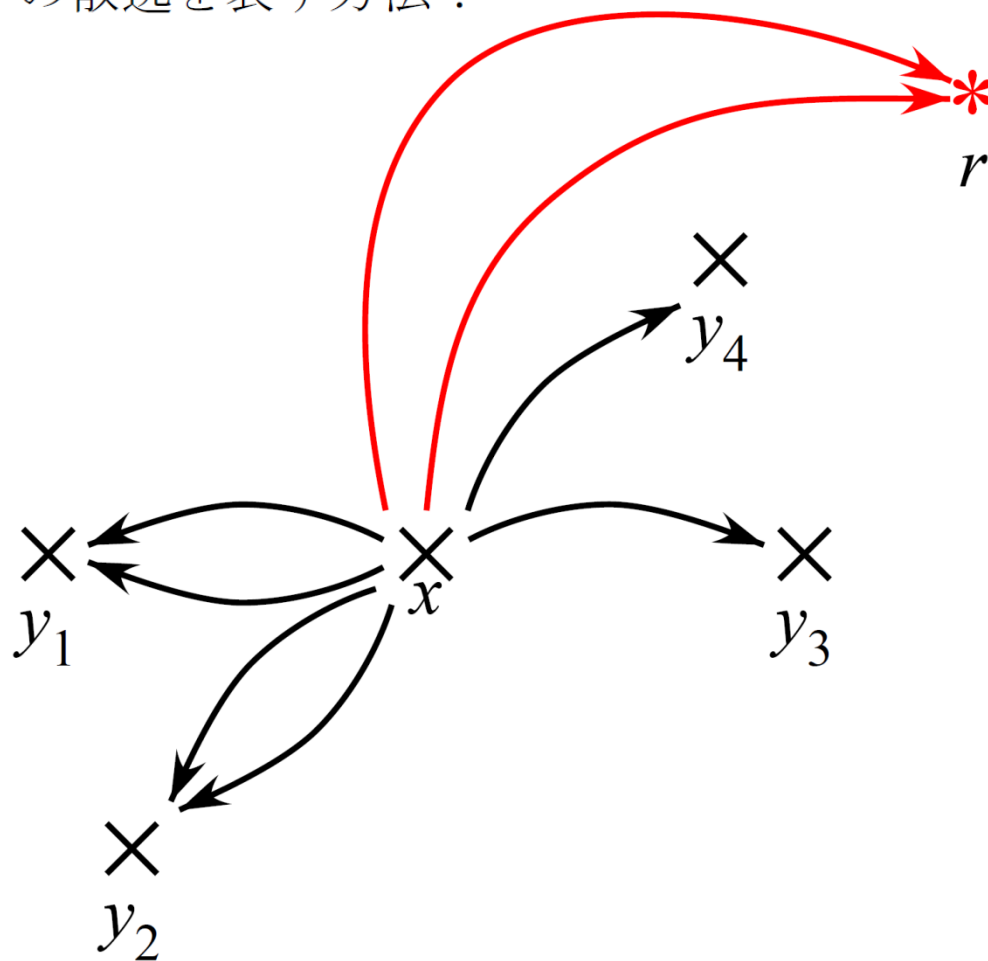
- (5) 点  $\mathbf{x}$  から  $h_c(\mathbf{x})$  個の砂粒がどのように飛び散るかを矢印で指定したい.  
図の例では,  $\mathbf{x}$  を始点とする  $h_c(\mathbf{x}) = 8$  本の矢印を考えることになる.



矢印の先に

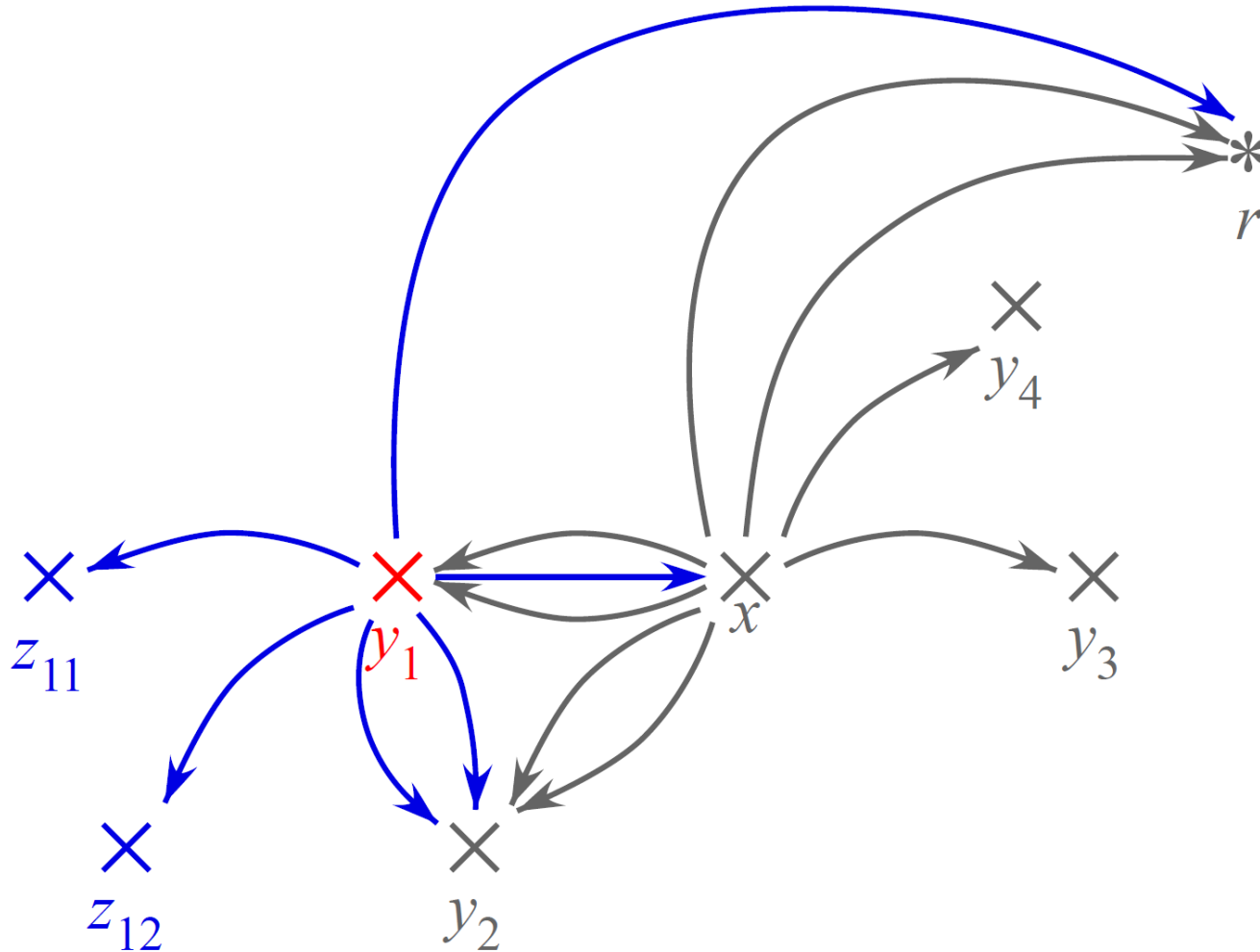
- 格子点がある場合 :  $\{y_1, y_2, y_3, y_4\} \subset \Lambda$  矢印 1 本あたり 1 個の砂粒が  $\mathbf{x} \rightarrow y_j$  へ流出する.  
この図の例は, 点  $\mathbf{x}$  の周りに 4 つの格子点があってそれらが格子  $\Lambda$  の一員になっている場合.
- 格子点がない場合 : 格子  $\Lambda$  の外に散逸してしまうと考える.

(6) 格子  $\Lambda$  の外への散逸を表す方法：

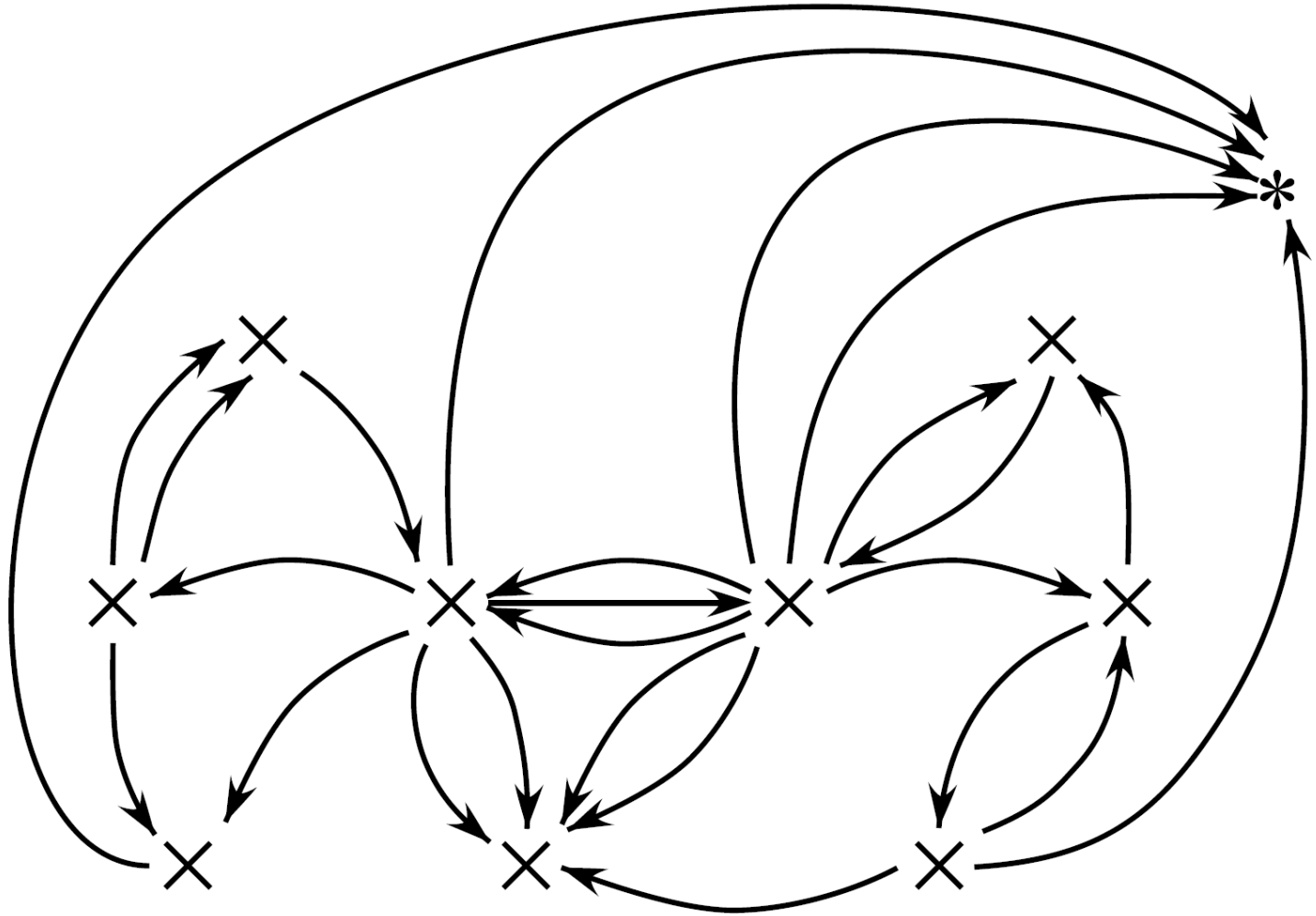


- 付加的な点  $\mathbf{r}$  を考え、そこに移動してしまったとする. ( $\mathbf{r}$  は考えている系の「外の世界」を表す.)
- 後で全域木 (spanning tree) との対応を考えるときには、これは根 (root) を表すことになるので  $\mathbf{r}$  と書くことにする. 図では  $*$  で表す.

- (7) 次に、点  $\mathbf{x}$  の最近接格子点の一つである  $\mathbf{y}_1$  上で同様の設定をする。  
 砂山の安定閾値  $h_c(\mathbf{y}_1)$  を定める。図の例では  $h_c(\mathbf{y}_1) = 6$  である。  
 よって、 $h_c(\mathbf{y}_1) = 6$  本の矢印を  $\mathbf{y}_1$  から放出させる。  
 これらの矢印の終点についての情報を指定する ( $\Lambda$  中の格子点か  $\mathbf{r}$  かを定める)。



- (8) この操作を  $\Lambda$  に含まれる格子点（下図の場合，9つの×印）すべてについて行う．すると，下図のような有向グラフ  $G = (V, E)$  が得られる．

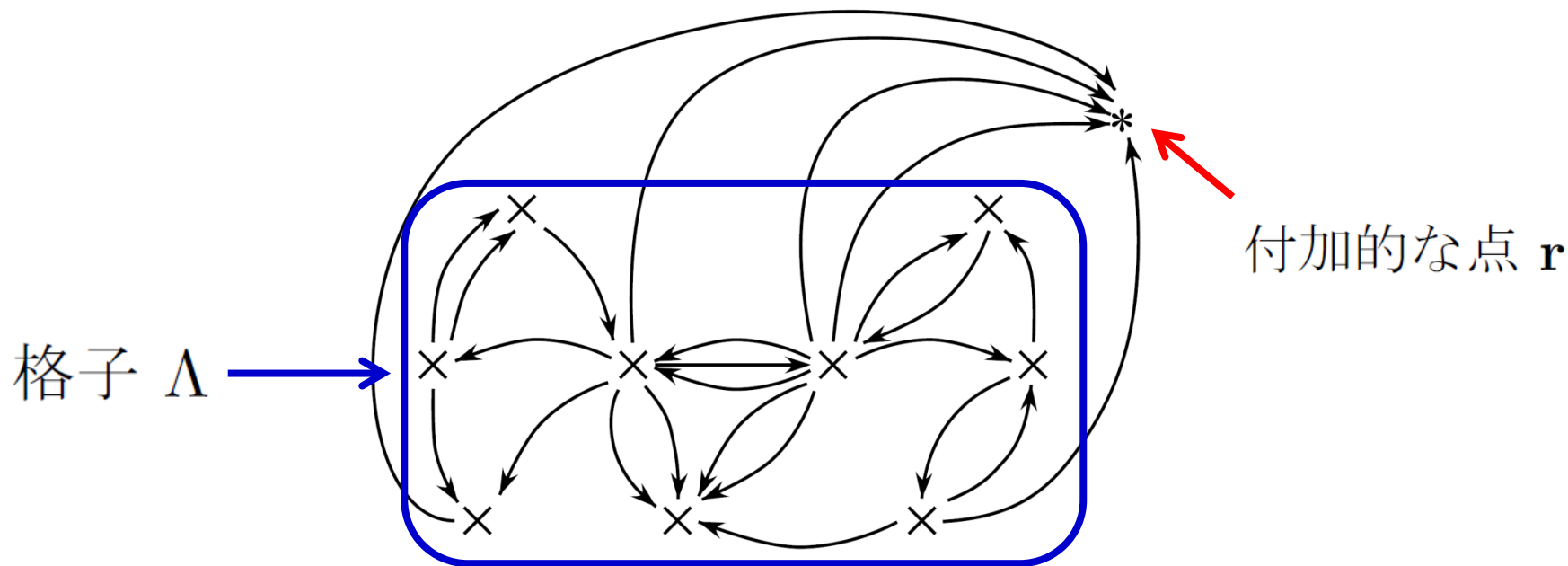


(8) この操作を  $\Lambda$  に含まれる格子点 (下図の場合, 9つの×印) すべてについて行う。すると, 下図のような有向グラフ  $G = (V, E)$  が得られる。

$$G = (V, E)$$

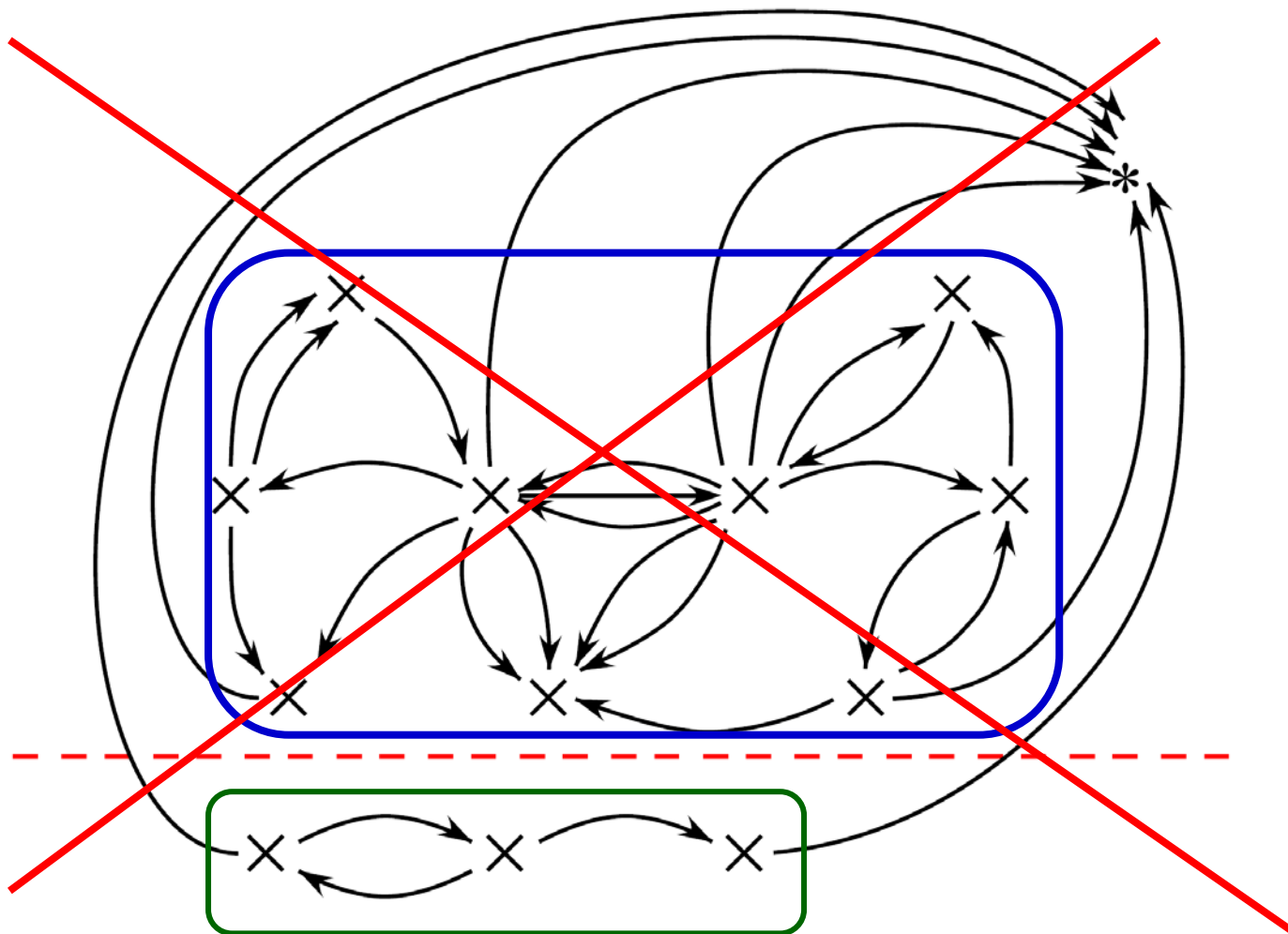
$$V = \text{点 (vertex) の集合} = \begin{cases} \text{元の格子 } \Lambda \text{ の格子点 } \{\mathbf{x}, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4, \mathbf{z}_{11}, \mathbf{z}_{12}, \dots\} \\ \text{付加的な点 } \mathbf{r} \end{cases}$$

$$E = \text{辺 (edge) の集合} = \begin{cases} \text{格子 } \Lambda \text{ に含まれている格子点同志を結ぶ有向辺} \\ \text{(向きがあることに注意. 一般には両方の向きがある.)} \\ \text{格子点から } \mathbf{r} \text{ への矢印 (散逸を表すので, 逆向きはない)} \end{cases}$$



- (9) 下のグラフは、 $r$  とそこへ至る矢印を除くと 2 つの部分に分かれてしまう (これを reducible という).

ここでは、このような分割はできない場合を考えることにする.  
これを irreducible graph とよぶ.





(i) 格子  $\Lambda$  に含まれる格子点 (×印) の総数を  $|\Lambda|$ , vertex の集合  $V$  に含まれる点の総数を  $|V|$  と記す. 当然,  $|V| = |\Lambda| + 1$  である.  $+1$  は  $\mathbf{r}$  の分である.

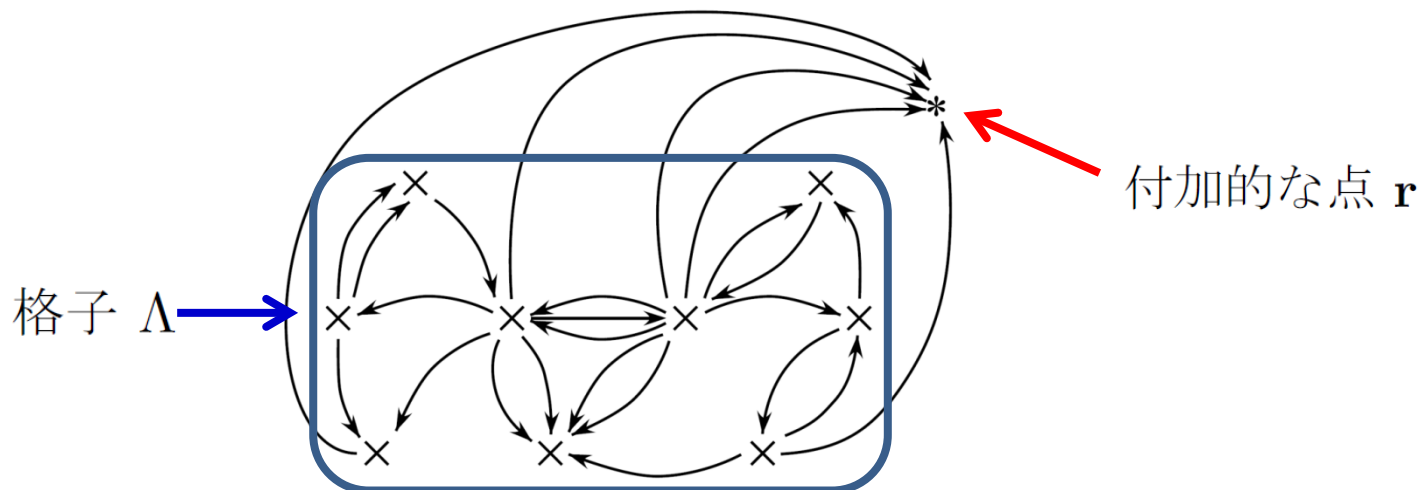
(ii) サイズ  $|V| = |\Lambda| + 1$  の実正方行列  $\Delta_*$  を考える:

$\mathbf{a}, \mathbf{b} \in V = \Lambda \cup \{\mathbf{r}\}$  に対して

$$\begin{cases} \Delta_*(\mathbf{a}, \mathbf{a}) = h_c(\mathbf{x}) = \text{「}\mathbf{x} \text{ を始点とする矢印の総数」}, & \mathbf{a} = \mathbf{x} \in \Lambda \text{ の場合} \\ = 0, & \mathbf{a} = \mathbf{r} \text{ の場合} \\ -\Delta_*(\mathbf{a}, \mathbf{b}) = \text{「}\mathbf{a} \text{ を始点とし } \mathbf{b} \text{ を終点とする矢印の本数」}, & \mathbf{a} \neq \mathbf{b} \text{ の場合} \end{cases}$$

この定義より,  $\mathbf{x} \in \Lambda$  に対しては

$$\begin{aligned} \Delta_*(\mathbf{x}, \mathbf{x}) - \sum_{\substack{\mathbf{y}: \mathbf{y} \neq \mathbf{x}, \\ \mathbf{y} \in \Lambda}} (-\Delta_*(\mathbf{x}, \mathbf{y})) &= \sum_{\mathbf{y}: \mathbf{y} \in \Lambda} \Delta_*(\mathbf{x}, \mathbf{y}) \\ &= -\Delta_*(\mathbf{x}, \mathbf{r}) \\ &= \mathbf{x} \text{ 上の砂山の 1 回の toppling で格子 } \Lambda \text{ から散逸してしまう砂粒の総数} \end{aligned}$$



(iii) 上の行列  $\Delta_*$  のうち  $\mathbf{r}$  の行と列を除いて得られるサイズ  $|\Lambda|$  の実正方行列を考える.

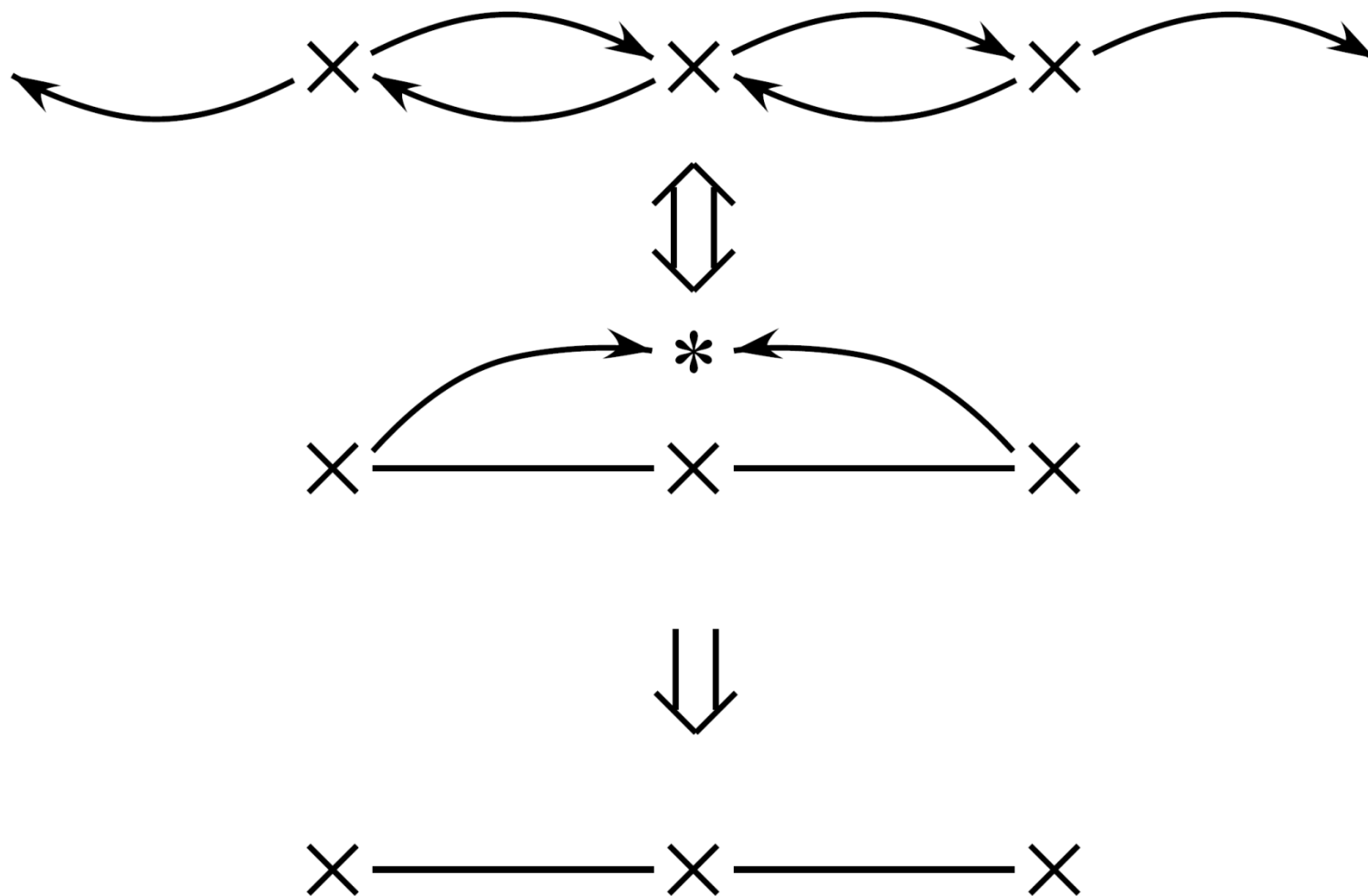
$$\Delta(\mathbf{x}, \mathbf{y}) = \Delta_*(\mathbf{x}, \mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \Lambda.$$

上の計算結果は次のように表される :

$\mathbf{x} \in \Lambda$  に対して

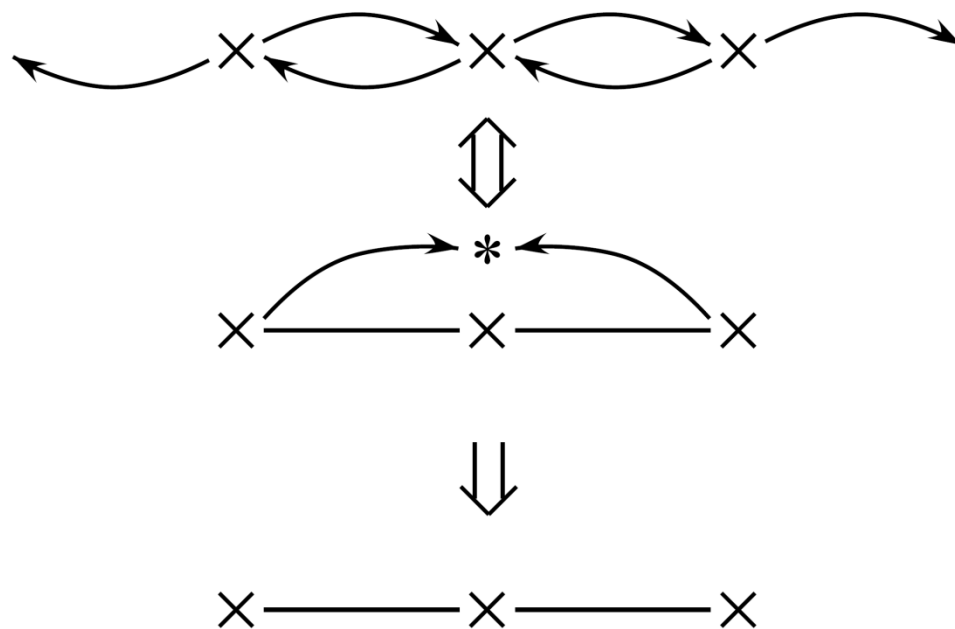
$$\begin{aligned} \sum_{\mathbf{y}: \mathbf{y} \in \Lambda} \Delta(\mathbf{x}, \mathbf{y}) &= -\Delta_*(\mathbf{x}, \mathbf{r}) \\ &= \mathbf{x} \text{ 上の砂山の 1 回の toppling で格子 } \Lambda \text{ から散逸してしまう砂粒の総数} \end{aligned}$$

サイズ  $L = 3$  の 1 次元格子 (直線) 上のモデル

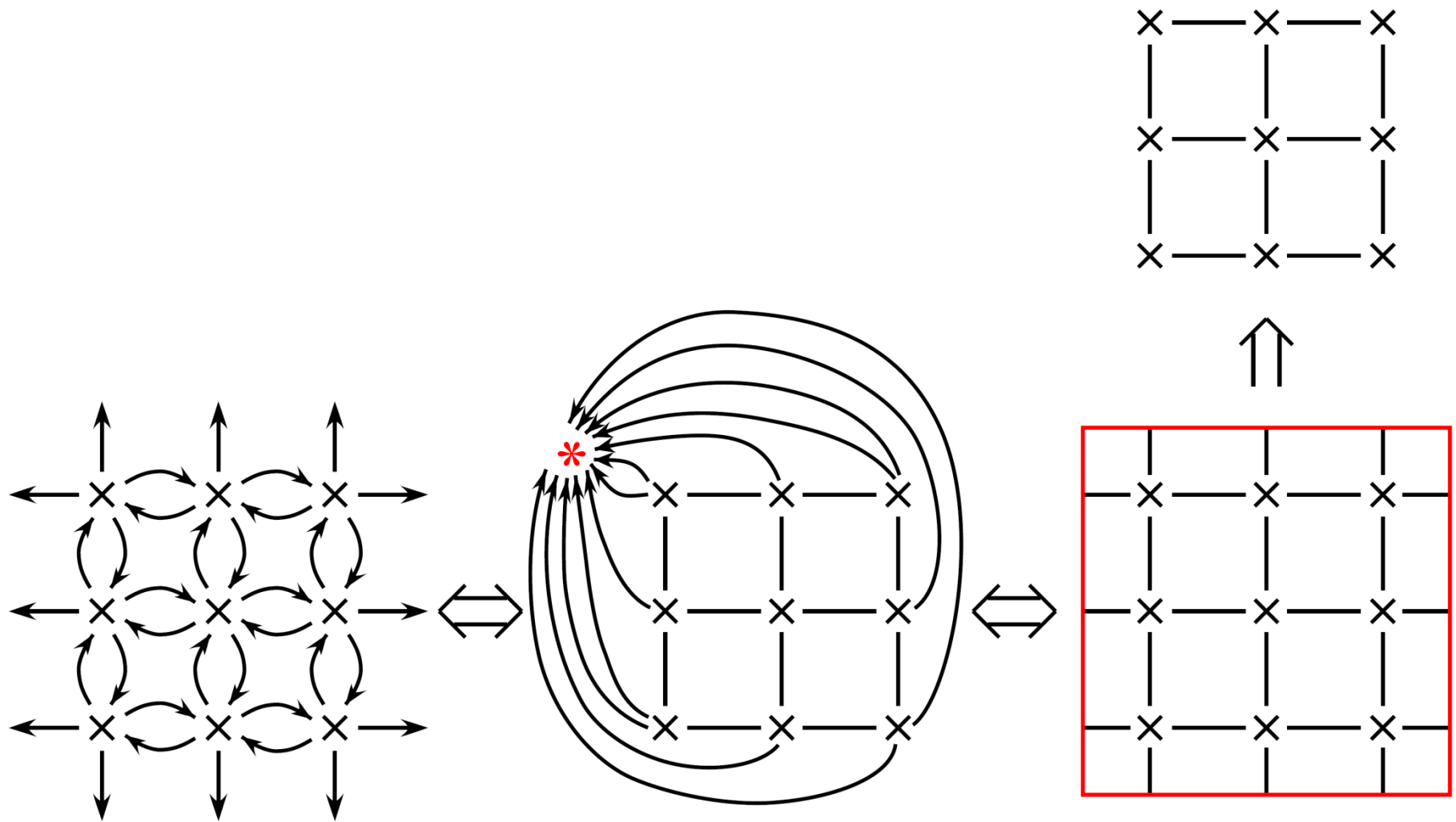


# サイズ $L = 3$ の 1 次元格子（直線）上のモデル

$$\Delta(x, y) = \begin{cases} 2, & x = y \text{ のとき} \\ -1, & |x - y| = 1 \text{ のとき} \\ 0, & \text{それ以外の場合} \end{cases}$$

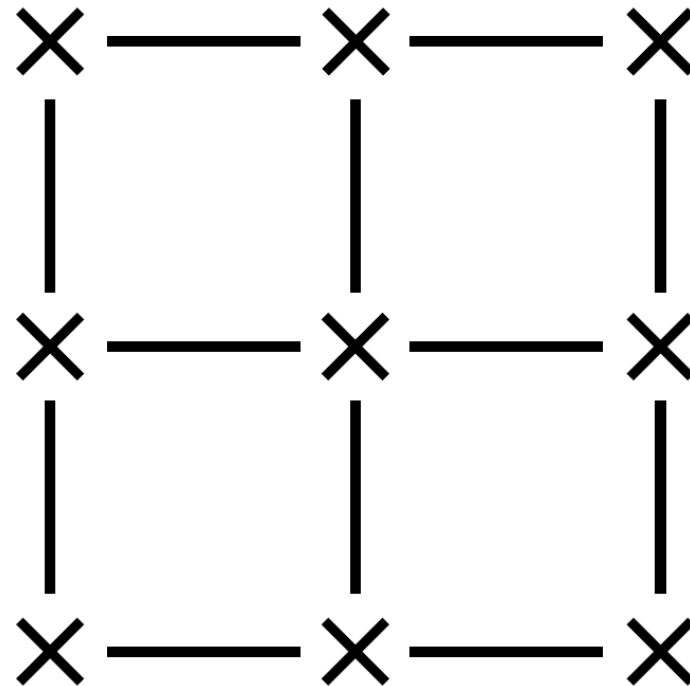


# サイズ $L = 3$ の 2 次元格子（正方格子）上のモデル



サイズ  $L = 3$  の 2 次元格子（正方格子）上のモデル

$$\Delta(\mathbf{x}, \mathbf{y}) = \begin{cases} 4, & \mathbf{x} = \mathbf{y} \text{ のとき} \\ -1, & |\mathbf{x} - \mathbf{y}| = 1 \text{ のとき} \\ 0, & \text{それ以外} \text{ のとき} \end{cases}$$



$d$  次元 **BTW** 模型 (**Bak-Tang-Wiesenfeld model**) :

$\Lambda = B_L = \{1, 2, \dots, L\}^d$  (サイズ  $L$  の  $d$  次元立方格子) の上,  
次のルール行列で定義される.

$$\Delta(\mathbf{x}, \mathbf{y}) = \begin{cases} 2d, & \mathbf{x} = \mathbf{y} \text{ のとき} \\ -1, & |\mathbf{x} - \mathbf{y}| = 1 \text{ のとき} \\ 0, & \text{それ以外} \text{ のとき} \end{cases}$$

## 2.2 時空上での局所的な摂動と 大域的な「なだれ」(avalanche)

- これまでの定義では  $\eta_t(\mathbf{x}) \in \mathbb{N}_0 \equiv \{0, 1, 2, \dots\}, \mathbf{x} \in \Lambda$ .
- このうち、安定な砂山高さ配置は  $h_t(\mathbf{x}) \in \{0, 1, 2, \dots, h_c(\mathbf{x}) - 1\}, \forall \mathbf{x} \in \Lambda$ .



$h_t = (h_t(\mathbf{x}))_{\mathbf{x} \in \Lambda} \in \mathcal{S} \equiv$  安定配置全体の集合

- $\mathcal{S}$  内の配置の間を遷移するプロセスを考えたい.

$$(h_t)_{t \in \mathbb{N}_0} = \left( (h_t(\mathbf{x}))_{\mathbf{x} \in \Lambda} \right)_{t \in \mathbb{N}_0}.$$



- (1) ある時刻  $t \in \mathbb{N}_0$  で  $h_t \in \mathcal{S}$  とする. これに局所的な摂動を加える.  
例えば,  $\Lambda$  中の 1 つの格子点  $\mathbf{x} \in \Lambda$  を選び, 砂粒を 1 つ加える.

$$\eta_{(1)}^{\mathbf{x}}(\mathbf{z}) = h_t(\mathbf{z}) + \delta_{\mathbf{z}\mathbf{x}}, \quad \forall \mathbf{z} \in \Lambda.$$

- (2) もしも,  $\eta_{(1)}^{\mathbf{x}} \in \mathcal{S}$  なら (つまり,  $\eta_{(1)}^{\mathbf{x}}(\mathbf{x}) < h_c(\mathbf{x})$  なら)

$$h_{t+1} = \eta_{(1)}^{\mathbf{x}}.$$

- (2)' 別の (あるいは同じ) 格子点を選んで, そこに砂粒を 1 つ加える. 以下, 同様.

(3) もしも、 $\eta_{(1)}^{\mathbf{x}} \notin \mathcal{S}$  なら (つまり、 $\eta_{(1)}^{\mathbf{x}}(\mathbf{x}) \geq h_c(\mathbf{x})$  なら) ,  $\mathbf{x}$  上の砂山を topple させる.



この toppling の結果, 周囲の格子点  $\mathbf{y}$  上の砂山が不安定化することがある.



toppling を連鎖的に起こす.



再び安定配置になるまで繰り返す.

この一連の toppling をなだれ (avalanche) とよぶ.

- 一連の toppling の記述方法 :

- $\eta_{(0)}^{\mathbf{x}} \equiv h_t$  として, 不安定配置の列  $\{\eta_{(1)}^{\mathbf{x}}, \eta_{(2)}^{\mathbf{x}}, \dots, \eta_{(\ell)}^{\mathbf{x}}, \dots\}$  を以下のように構成する.
- $\eta_{(\ell)}^{\mathbf{x}} \notin \mathcal{S}, \ell \geq 1 \implies A_{(\ell)}^{\mathbf{x}}(h_t) = \{\mathbf{z} \in \Lambda : \eta_{(\ell)}^{\mathbf{x}}(\mathbf{z}) \geq h_c(\mathbf{z})\} \neq \emptyset.$
- toppling は次式のように, ルール行列  $\Delta$  で与えられる :

$$\eta_{(\ell+1)}^{\mathbf{x}}(\mathbf{z}) = \eta_{(\ell)}^{\mathbf{x}}(\mathbf{z}) - \sum_{\mathbf{y}: \mathbf{y} \in A_{(\ell)}^{\mathbf{x}}(h_t)} \Delta(\mathbf{y}, \mathbf{z}), \quad \mathbf{z} \in \Lambda.$$

- なだれの継続時間  $\tau$ : 上の連鎖の結果,  $\eta_{(\ell+1)}^{\mathbf{x}} \in \mathcal{S}$  となったら,  $\tau = \ell + 1.$

$\tau = \tau(\mathbf{x}, h_t)$  : なだれ前の配置  $h_t$  と摂動の位置  $\mathbf{x}$  の関数

- そして,  $h_{t+1} = \eta_{(\tau)}^{\mathbf{x}}$  とする.

なだれの継続時間はいつも有限  $\tau < \infty$  でなければ, プロセスは定義できない.

必ず散逸がなければならない.

- 有限格子  $\Lambda = B_L$  (サイズ  $L$  の立方格子) 上の BTW 模型  
散逸は  $B_L$  の境界のみで起こる.  
空間的に非一様な系.  
システムサイズ  $L \rightarrow \infty$  で一様な熱力学的極限を得ることを証明することに困難.  
(バルク極限  $\neq$  熱力学的極限)

- **dissipative ASM**

Tsuchiya, T., Katori, M.: Phys. Rev. E **61**, 1183 (2000).

Katori, M.: MI lecture note (Kyushu University) **63**, pp.58-91 (2015);  
arXiv:math-ph/1505.00334.

空間的に一様な系 (トーラス  $T_L$  上で定義)

各格子点で一定の率で散逸あり.

熱力学的極限の存在が証明できる.

(その後, 散逸率  $\rightarrow 0$  の極限をとる : off-critical  $\rightarrow$  critical )

Schmidt, K., Verbitskiy, E.: Commun. Math. Phys. **292**, 721-750 (2009).

Járai, A., Redig, F., Saada, E.: J. Stat. Phys. **159**, 1369-1407 (2015).

