**Chip-Firing and Rotor-Routing on Directed Graphs**

*Alexander E. Holroyd, Lionel Levine, Karola Mészáros, Yuval Peres, James Propp and David B. Wilson*

arXiv:0801.3306v3

**Errata, questions and addenda (Darij Grinberg)**

- **Page 3, definition of the Laplacian:** You write:

$$\Delta_{ij} = \begin{cases} -a_{ij} & \text{for } i \neq j, \\ d_i - a_{ii} & \text{for } i = j \end{cases} .$$

It should be pointed out that $d_i$ is a shorthand notation for $d_{v_i}$.

- **Page 3, definition of "fire":** You write: "An active vertex $v$ can **fire**, resulting in a new chip configuration $\sigma'$ obtained by moving one chip along each of the $d_v$ edges emanating from $v$". At this point it would help to explicitly state that chips moving to the sink simply disappear (i. e., are forgotten). Otherwise readers might think that we keep track of how many chips went into the sink (this is a misunderstanding I had first) or that edges to the sink are not counted in the degree of a vertex (this is also a misunderstanding).

- **Page 3, Lemma 2.2:** In part 2 of the lemma, $\sigma_m$ should be $\sigma'_m$.

- **Page 3, Lemma 2.2:** One very pedantic remark: The $n$ in Lemma 2.2 has nothing to do with the $n$ on the upper half of page 3. The $n$ in Lemma 2.2 is the index of the last element of the sequence $\sigma_0, \sigma_1, ..., \sigma_n$, whereas the $n$ on the upper half of page 3 is the number of vertices of $G$. Some danger of confusion arises in the proof of Lemma 2.2 (and probably in a few other places in the paper), where you are speaking of a vertex $v_n$ which has nothing to do with the $v_n$ from the upper half of page 3. Maybe you should mention the ambiguity of $n$ in some footnote. (The same ambiguous use of $n$ appears in Lemma 3.9 and its proof.)

- **Page 4, proof of Lemma 2.2:** You write: "Then $v_i, v_1, v_2, ..., v_{i-1}, v_{i+1}, ..., v_n$ is a permissible firing sequence". I think it is permissible only if $i$ is defined as the *first* index $k$ satisfying $v_k = v'_1$ (and not just as some arbitrary index $k$ satisfying $v_k = v'_1$). Of course, this is easy to fix (just define $i$ this way).

- **Page 5, proof of Lemma 2.4:** You write: "and $d_{v_{r-1}}$ such chips will cause $v_{r-1}$ to fire once". Maybe this would be clearer if you replace the word "cause" by "allow" (at least it would be clearer this way to me), because until we know that each chip configuration stabilizes, we cannot speak of "causing" here (in fact, one could imagine that we fire some subset of vertices again and again, and never come to fire $v_{r-1}$ even though $d_{v_{r-1}}$ chips have accumulated at $v_{r-1}$).

- **Page 5:** What is the Theorem 4.8 you refer to when you write "see Theorem 4.8 for a better bound"?

- **Page 6, proof of Lemma 2.8:** I think this proof only shows that the order of $\mathcal{S}(G)$ is the *absolute value* of the determinant of $\Delta'(G)$. In fact, in order to have

the volume of any parallelepiped equal to the determinant of the matrix from its edge-vectors, we need our volumes to be *signed* - but the index of a lattice is the *unsigned* volume of its fundamental domain.
What is needed to fix the proof is an argument why $\det \Delta'(G) \geq 0$. I don't see how to prove this combinatorially. We can prove this by a mix of algebra and continuity arguments: The matrix $\Delta'(G)$ is (weakly) diagonally dominant, and the determinant of a (weakly) diagonally dominant matrix is always $\geq 0$ (this is a limiting case of the inequality proved in
http://www.artofproblemsolving.com/Forum/viewtopic.php?t=176225 ).

- **Page 6, Definition 2.11:** At this point, I believe it would be useful to make one rather trivial observation which you are tacitly using several times in the subsequent proofs (or at least I believe you use it).
There are two ways to interpret your definition of "accessible":
*First interpretation:* A chip configuration $\sigma$ is **accessible** if from any other chip configuration it is possible to obtain $\sigma$ by *first* adding some chips and *then* selectively firing active vertices.
*Second interpretation:* A chip configuration $\sigma$ is **accessible** if from any other chip configuration it is possible to obtain $\sigma$ by a sequence of operations, each of which is either an addition of a chip or a firing of an active vertex.
The observation is now that these two interpretations are equivalent.
*Proof.* If we *first* fire an active vertex in a chip configuration $\tau$, and *then* add some chip, then we could have just as well *first* added the chip to the configuration $\tau$ and *then* fired the vertex (because adding the chip can never make any active vertex non-active). Therefore, if we have a sequence of operations, each of which is either an addition of a chip or a firing of an active vertex, then we can move all the "addition-of-a-chip" operations to the front and all of the "firing-of-an-active-vertex" operations to the end of this sequence. Hence, the second interpretation is equivalent to the first interpretation.

- **Page 7:** What is the Lemma 4.6 you refer to when you write "cluster-firing and superstabilization as described in Definition 4.3 and Lemma 4.6"?

- **Page 7, proof of Lemma 2.13:** You write "$\beta \geq [d_{\max} + (-m)](\delta - \delta^\circ) \geq d_{\max}(\delta - \delta^\circ) \geq d_{\max}$". I think this is wrong, because $\beta = \alpha + [d_{\max} + (-m)](\delta - \delta^\circ)$ does not yield $\beta \geq [d_{\max} + (-m)](\delta - \delta^\circ)$. What you mean is that
$$\beta = \alpha + [d_{\max} + (-m)](\delta - \delta^\circ) = \alpha + (-m)\underbrace{(\delta - \delta^\circ)}_{\geq 1} + d_{\max}(\delta - \delta^\circ)$$
$$\geq \underbrace{\alpha + (-m)}_{\geq 0} + d_{\max}(\delta - \delta^\circ) \geq d_{\max}\underbrace{(\delta - \delta^\circ)}_{\geq 1} \geq d_{\max}.$$

- **Page 7, Lemma 2.14:** This lemma is nice, but I think that the actual fact you repeatedly use (for example, when you write "by Lemma 2.14" in the proof of Lemma 2.15) is the following strengthening:
**Lemma 2.14a:** Let $\epsilon = (2\delta) - (2\delta)^\circ$, where $\delta$ is given by $\delta(v) = d_v$ as before. If $\sigma$ is recurrent, and $k \geq 0$ is an integer, then $(\sigma + k\epsilon)^\circ = \sigma$.
*Proof.* Induct over $k$. The induction base ($k = 0$) is clear (since $\sigma$ is recurrent

and thus stable).
For the induction step, assume that $(\sigma + k\epsilon)^\circ = \sigma$, and try to prove that $(\sigma + (k+1)\epsilon)^\circ = \sigma$.
Indeed there is a fact which I call the "partial stabilization lemma", and which states that any two chip configurations $\alpha$ and $\beta$ satisfy $(\alpha+\beta)^\circ = (\alpha + \beta^\circ)^\circ$. Applied to $\alpha = \epsilon$ and $\beta = \sigma + k\epsilon$, this fact yields $(\epsilon + \sigma + k\epsilon)^\circ = (\epsilon + (\sigma + k\epsilon)^\circ)^\circ$. Since $(\sigma + k\epsilon)^\circ = \sigma$ and $\epsilon + \sigma + k\epsilon = \sigma + (k+1)\epsilon$, this rewrites as $(\sigma + (k+1)\epsilon)^\circ = (\epsilon + \sigma)^\circ$. Thus, $(\sigma + (k+1)\epsilon)^\circ = (\epsilon + \sigma)^\circ = (\sigma + \epsilon)^\circ = \sigma$ (by Lemma 2.14), completing the induction step. Lemma 2.14a is thus proven.

- **Page 7, Lemma 2.14:** I am just wondering: Why do you formulate this lemma for $\epsilon = (2\delta) - (2\delta)^\circ$ rather than for $\epsilon = \delta - \delta^\circ$ ? Unless I am mistaken, the proof of Lemma 2.14 works just as well if the definition of $\epsilon$ is replaced by $\epsilon = \delta - \delta^\circ$. The only difference I can see between $(2\delta) - (2\delta)^\circ$ and $\delta - \delta^\circ$ is that $(2\delta) - (2\delta)^\circ$ is accessible (while $\delta - \delta^\circ$ is not necessarily - at least I don't see a proof that it is). But you don't seem to ever use the accessibility of $(2\delta) - (2\delta)^\circ$ ...

- **Page 11, proof of Lemma 2.17:** You write: "If (4) holds, there is a chip configuration $\alpha$ such that $(\sigma + \alpha)^\circ = \sigma$ and $\alpha$ is nonzero on at least one vertex of each strongly connected component of $G$". Here you seem to be using the the following (not completely trivial) fact:
  **Lemma 2.14b:** If some stable chip configuration $\sigma$ and some chip configurations $\alpha_1, \alpha_2, ..., \alpha_k$ satisfy $(\sigma + \alpha_1)^\circ = (\sigma + \alpha_2)^\circ = ... = (\sigma + \alpha_k)^\circ = \sigma$, then $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_k)^\circ = \sigma$.
  *Proof.* We will prove that every $i \in \{0, 1, ..., k\}$ satisfies $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_i)^\circ = \sigma$.
  Indeed, let us prove this by induction over $i$: The induction base ($i = 0$) is clear (since $\sigma$ is stable).
  For the induction step, let $i \in \{1, 2, ..., k\}$. Assume that $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_{i-1})^\circ = \sigma$, and try to prove that $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_i)^\circ = \sigma$.
  Applying the "partial stabilization lemma" (which I quoted above in the proof of Lemma 2.14a) to $\alpha = \alpha_i$ and $\beta = \sigma + \alpha_1 + \alpha_2 + ... + \alpha_{i-1}$, we obtain $(\alpha_i + \sigma + \alpha_1 + \alpha_2 + ... + \alpha_{i-1})^\circ = (\alpha_i + (\sigma + \alpha_1 + \alpha_2 + ... + \alpha_{i-1})^\circ)^\circ$. Since $\alpha_i + \sigma + \alpha_1 + \alpha_2 + ... + \alpha_{i-1} = \sigma + \alpha_1 + \alpha_2 + ... + \alpha_i$ and $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_{i-1})^\circ = \sigma$, this rewrites as $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_i)^\circ = (\alpha_i + \sigma)^\circ$. Thus, $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_i)^\circ = (\alpha_i + \sigma)^\circ = (\sigma + \alpha_i)^\circ = \sigma$. This completes the induction step.
  We have thus proven by induction that every $i \in \{0, 1, ..., k\}$ satisfies $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_i)^\circ = \sigma$. Applied to $i = k$, this yields $(\sigma + \alpha_1 + \alpha_2 + ... + \alpha_k)^\circ = \sigma$. We have thus proven Lemma 2.14b.

- **Page 11, proof of Lemma 2.17:** You write: "Moreover, $(\sigma + \beta)^\circ = \sigma$." It took me a while to figure out why this is so. Here is my proof:
  Since $\beta$ is obtained from $k\alpha$ by selective firing, we see that $\sigma + \beta$ is obtained from $\sigma + k\alpha$ by selective firing. Thus, $(\sigma + \beta)^\circ = (\sigma + k\alpha)^\circ$. Since Lemma 2.14b (applied to $\alpha_1 = \alpha$, $\alpha_2 = \alpha$, ..., $\alpha_k = \alpha$) yields $(\sigma + k\alpha)^\circ = \sigma$, we obtain $(\sigma + \beta)^\circ = (\sigma + k\alpha)^\circ = \sigma$.
  So much for my proof of $(\sigma + \beta)^\circ = \sigma$. Is this argument more complicated than what you had in mind? Because I find it strange that you leave this to the reader

(after all, you haven't even explicitly stated Lemma 2.14b).

- **Page 11:** The formula
$$\sum_{w} \Delta'_{v,w} f(w) = \sigma(v)$$
should be
$$\sum_{w} \Delta'_{v,w} f(w) = \sigma(v) \operatorname{mod} 1.$$

- **Page 13, proof of Lemma 3.4:** You write: "this same directed occurs within $\rho$". This should be "this same directed cycle occurs within $\rho$" (you forgot the word "cycle").

- **Page 14, proof of Lemma 3.7:** You write: "and hence the edge from $v_i$ to $v_{i+1}$ was traversed more recently than the edge from $v_{i-1}$ to $v_i$". Since both of these edges might have been traversed several times, I would find it clearer if you replaced this by "and hence the last time the chip was at $v_{i+1}$ is later than the last time it was at $v_i$" (or something similar - I don't think my sentence is good English, but I assume you get what I mean).

- **Page 14, proof of Lemma 3.9:** You write: "then some permutation of $v$ agrees with $v'$ in the first $i$ terms". I believe that, for the induction to work as intended, this should be amended to "then some *legal* permutation of $v$ agrees with $v'$ in the first $i$ terms", where a sequence of vertices of $G$ is said to be *legal* if it is possible to fire the vertices in the order of this sequence. Otherwise, it is not clear why "the vertices $v_1, v_2, ..., v_{i-1}, v_j, v_i, v_{i+1}, ..., v_{j-1}, v_{j+1}, ..., v_n$ can be fired in that order" (since we don't know whether the sequence $v$ itself was legal to begin with!).
Alternatively, you could also leave the "then some permutation of $v$ agrees with $v'$ in the first $i$ terms" part as it is, but then "the vertices $v_1, v_2, ..., v_{i-1}, v_j, v_i, v_{i+1}, ..., v_{j-1}, v_{j+1}, ..., v_n$ can be fired in that order" is not literally true, unless you allow vertices to host a negative number of chips.

- **Page 16, proof of Lemma 3.10:** You write: "where the induction hypothesis states that every path leads to either the sink or to the chip". I think it would be more precise to say that "where the induction hypothesis states that every sufficiently long path leads either to the sink or through the chip".

- **Page 17, Definition 3.11:** The formula
$$\sigma(\rho) = \left( \prod_{v \in V(G)} E_v^{\sigma(v)} \right) \rho$$
should be
$$\sigma(\rho) = \left( \prod_{v \in V(G) \setminus \{\text{sink}\}} E_v^{\sigma(v)} \right) \rho.$$

- **Page 19, proof of Lemma 3.16:** You write: "Since $\rho$ is reachable from $\rho'$ ". But why is $\rho$ reachable from $\rho'$ ? This is again something that took me a while to figure out. What you seem to be using here is the following lemma:
  **Lemma 3.16a.** Let $G$ be a digraph, and let $\rho$ be a rotor configuration. Let $\ell \in \mathbb{N}$. Let $\sigma_1$, $\sigma_2$, ..., $\sigma_\ell$ be chip configurations such that $\rho$ is reachable from $\sigma_i(\rho)$ for every $i \in \{1, 2, ..., \ell\}$. Then, $\rho$ is reachable from $(\sigma_1 + \sigma_2 + ... + \sigma_\ell)(\rho)$.
  *Proof.* For every $i \in \{1, 2, ..., \ell\}$, there exists some chip configuration $\tau_i$ such that $\tau_i(\sigma_i(\rho)) = \rho$ (since $\rho$ is reachable from $\sigma_i(\rho)$). Consider this $\tau_i$. Then, $(\tau_i + \sigma_i)(\rho) = \tau_i(\sigma_i(\rho)) = \rho$. Thus, $(\tau_\ell + \sigma_\ell)\rho = \rho$, $(\tau_{\ell-1} + \sigma_{\ell-1})\rho = \rho$, ..., $(\tau_1 + \sigma_1)\rho = \rho$. Now,
  $$\begin{aligned}
  &(\tau_1 + \tau_2 + ... + \tau_\ell)((\sigma_1 + \sigma_2 + ... + \sigma_\ell)(\rho)) \\
  &= \left(\underbrace{(\tau_1 + \tau_2 + ... + \tau_\ell) + (\sigma_1 + \sigma_2 + ... + \sigma_\ell)}_{=(\tau_1+\sigma_1)+(\tau_2+\sigma_2)+...+(\tau_\ell+\sigma_\ell)}\right)(\rho) \\
  &= ((\tau_1 + \sigma_1) + (\tau_2 + \sigma_2) + ... + (\tau_\ell + \sigma_\ell))(\rho) \\
  &= (\tau_1 + \sigma_1)\left((\tau_2 + \sigma_2)\left(...\left((\tau_{\ell-1} + \sigma_{\ell-1})\underbrace{((\tau_\ell + \sigma_\ell)\rho)}_{=\rho}\right)\right)\right) \\
  &= (\tau_1 + \sigma_1)\left((\tau_2 + \sigma_2)\left(...\left((\tau_{\ell-2} + \sigma_{\ell-2})\underbrace{((\tau_{\ell-1} + \sigma_{\ell-1})\rho)}_{=\rho}\right)\right)\right) \\
  &= ... \\
  &= (\tau_1 + \sigma_1)\rho = \rho.
  \end{aligned}$$
  Hence, $\rho$ is reachable from $(\sigma_1 + \sigma_2 + ... + \sigma_\ell)(\rho)$. This proves Lemma 3.16a.
  You apply Lemma 3.16a to $\sigma_j = \mathbf{1}_{v_j}$ (where $\mathbf{1}_{v_j}$ is the chip configuration consisting of a single chip at $v_j$), thus obtaining that $\rho$ is reachable from $(\mathbf{1}_{v_1} + \mathbf{1}_{v_2} + ... + \mathbf{1}_{v_\ell})(\rho)$. In other words, $\rho$ is reachable from $\left(\prod_i E_{v_i}\right)\rho$. Then you apply Lemma 3.16a to $k$ and $\prod_i E_{v_i}$ instead of $\ell$ and $\sigma_j$, thus concluding that $\rho$ is reachable from $\left(\prod_i E_{v_i}\right)^k \rho$. Since $\rho' = \left(\prod_i E_{v_i}\right)^k \rho$, this yields that $\rho$ is reachable from $\rho'$.
  Is this really the argument you intended?

- **Page 19, proof of Corollary 3.18:** You write: "The rows of $\Delta'(G)$ corresponding to vertices in $S$ sum to zero". This is ambiguous. The sum of the rows of $\Delta'(G)$ corresponding to vertices in $S$ is *not* the zero vector (unless the indegree of every vertex of $S$ happens to be equal to its outdegree). What you probably wanted to say is that each row of $\Delta'(G)$ corresponding to a vertex in $S$ has its entries adding up to 0, and its only entries are in the columns corresponding to vertices in $S$. (This yields that if we WLOG assume that the columns corresponding to the vertices in $S$ are the first $k$ columns of $\Delta'(G)$, and the rows corresponding to the rows in $S$ are the first $k$ rows of $\Delta'(G)$, then $\Delta'(G)$ is a lower block-triangular matrix, and its upper left part has determinant 0, so that the whole matrix $\Delta'(G)$ must also have determinant 0.)

- **Page 20, proof of Proposition 3.21:** You write: "Since $h$ is a harmonic function on $G$ ". I think that $h$ is harmonic on $V(G) \diagdown Z$ only. Thus you should only consider edges $e = (u, v)$ with $u \notin Z$. Can't we actually improve the inequality (1) by summing only over such edges $e$ on the right hand side? I mean, any other edge will neither ever be used by a random walk stopped on first hitting $Z$, nor will it ever be switched by a rotor-router walk stopped on first hitting $Z$.

- **Page 21, proof of Proposition 3.21:** You write: "We assign weight $\sum_v \operatorname{wt}(\rho(v))$ to a rotor configuration $\rho$". I would replace $\rho$ by $\tau$ here, since $\rho$ already stands for the (fixed!) initial rotor configuration (whose weight is 0 since $\operatorname{wt}(\rho(v)) = 0$ for every vertex $v$).

- **Page 21, the very first line of Section 4:** Replace $G = (E, V)$ by $G = (V, E)$.

- **Page 21, the fourth line of Section 4:** You claim that "equivalently, $G$ has a sink and every other vertex has out-degree that is at least as large as its in-degree". I don't think the word "equivalently" is appropriate here. Surely, if $G$ is an Eulerian digraph with sink, then $G$ has a sink and every other vertex has out-degree that is at least as large as its in-degree. But the converse doesn't hold: If $G$ has a sink and every other vertex has out-degree that is at least as large as its in-degree, then we can add some edges starting at the sink to ensure that every in-degree in the resulting graph becomes equal to the corresponding out-degree, but nothing guarantees us that the resulting graph is strongly connected.

- **Page 21, the sixth line of Section 4:** You write: "Such a tour exists if and only if $G$ is Eulerian." I am being really pedantic here, but no - such a tour can also exist if $G$ consists of an Eulerian graph and some isolated vertices ;)

- **Page 21, proof of Lemma 4.1:** You write: "By the "(4) $\Longrightarrow$ (1)" part of Lemma 2.17, if $(\sigma + \beta)^{\circ} = \sigma$, then $\sigma$ is recurrent." Are you claiming that every strongly connected component of $G$ contains some vertex $v$ such that $\beta(v) > 0$ ? This is wrong. Counterexample: Consider a directed cycle on 4 vertices. Remove one edge. You get an Eulerian digraph with sink. Each of its strongly connected components consists of one vertex only; in particular, the vertex opposite to the sink forms a strongly connected component, but its $\beta$-value is 0.
  Here is how you can fix this part of the proof:
  We need something slightly stronger than the "(4) $\Longrightarrow$ (1)" part of Lemma 2.17. We can get it by adding another equivalent assertion to Lemma 2.17:
  (5) There exists an epidemic chip configuration $\alpha$ such that $\sigma$ is reachable from $\sigma + \alpha$. Here, a chip configuration $\alpha$ is said to be *epidemic* if for every vertex $v$ of $G$, there exists a vertex $w$ of $G$ such that $\alpha(w) > 0$ and such that there exists a path from $w$ to $v$.
  The proof of (4) $\Longrightarrow$ (5) is obvious, and the proof of (5) $\Longrightarrow$ (1) is the same as the proof of (4) $\Longrightarrow$ (1) given in your paper.
  Now, we claim that in the situation of Lemma 4.1, the chip configuration $\beta$ is epidemic. In fact, let $v$ be a vertex of $G$. By the definition of an Eulerian digraph with sink, the digraph $G$ was obtained by taking an Eulerian digraph $G'$ and deleting all the edges from one vertex $s$. Consider such a $G'$. Since $G'$ is

Eulerian, there exists an Eulerian tour of $G'$. Let us walk *backwards* along this tour, starting at the vertex $v$, until we reach the vertex $s$ for the first time (with one exception: if $v = s$, then we walk backwards until we reach the vertex $s$ for the *second* time). Then, go a step forward again. Let $w$ be the vertex in which we land. Then, $w$ is the endpoint of an edge from $s$, so that $\beta(w) > 0$, but on the other hand, there exists a path from $w$ to $v$ in $G$ (namely, we can reach $v$ from $w$ by walking forward along the Eulerian tour, without ever using an edge from $s$). This shows that $\beta$ is epidemic, and we can apply the "(5) $\Longrightarrow$ (1)" part of Lemma 2.17, qed.

- **Page 21, proof of Lemma 4.1:** You write: "The rows of $\Delta'$ are linearly independent". Do you have a quick proof for this seemingly obvious statement? The best proof I can come up with is nontrivial: Since $G$ is an Eulerian digraph with sink, its sink is a global sink, and thus it has an oriented spanning tree rooted at the sink. Therefore, Corollary 3.18 yields that $\det \Delta'(G) \geq 1$, so that $\Delta' = \Delta'(G)$ is nonsingular, i. e., the rows of $\Delta'$ are linearly independent. But this looks like an overkill argument...

- **Page 22, second line of the page:** You write: "Then $\sigma$ is recurrent if and only if every non-sink vertex fires in the stabilization of $\sigma + \beta$." I don't think this is true (take $\sigma$ to be a configuration consisting of very big numbers), but there we can make it true either by replacing "fires" by "fires exactly once" or by requiring $\sigma$ to be stable. Here are my proofs of this:
  *Statement 1:* If $\sigma$ is a chip configuration such that every non-sink vertex fires exactly once in the stabilization of $\sigma + \beta$, then $\sigma$ is recurrent.
  *Statement 2:* If $\sigma$ is a stable chip configuration such that every non-sink vertex fires at least once in the stabilization of $\sigma + \beta$, then $\sigma$ is recurrent.
  *Proof of Statement 1:* Let $\sigma$ be a chip configuration such that every non-sink vertex fires exactly once in the stabilization of $\sigma+\beta$. Then, $(\sigma + \beta)^\circ = \sigma + \beta - \sum\limits_{i=1}^{n-1} \Delta'_i$ (using the notations of the proof of Lemma 4.1). Since $\beta = \sum\limits_{i=1}^{n-1} \Delta'_i$, this yields $(\sigma + \beta)^\circ = \sigma$, and thus $\sigma$ is recurrent (this is shown just as in the proof of Lemma 4.1). This proves Statement 1.
  Before proving Statement 2, we show something stronger:
  *Statement 3:* If $\sigma$ is a stable chip configuration, then every non-sink vertex fires at most once in the stabilization of $\sigma + \beta$.
  *Proof of Statement 3:* Assume the contrary. Then, there exists a stable chip configuration $\sigma$ such that some non-sink vertex fires more than once in the stabilization of $\sigma + \beta$. Consider such a $\sigma$. Let $v_1, v_2, ..., v_n$ be the vertices fired in the stabilization of $\sigma + \beta$ (in this order). Then, there exists some $j$ such that some $i < j$ satisfies $v_j = v_i$ (because some non-sink vertex fires more than once in the stabilization of $\sigma + \beta$). Consider the smallest such $j$, and consider the corresponding $i$. Clearly, no two of the vertices $v_1, v_2, ..., v_{j-1}$ are equal (else, our $j$ wouldn't be the *smallest* $j$ such that some $i < j$ satisfies $v_j = v_i$). Just before the vertex $v_i = v_j$ fired the first time (at step $i$), it had $(\sigma + \beta)(v_i) + \sum\limits_{k=1}^{i-1} a_{v_k, v_i}$ chips (since it had $(\sigma + \beta)(v_i)$ chips at the beginning, and at step $k$ it ob-

tained $a_{v_k,v_i}$ chips from firing $v_k$). After it fired the first time, it therefore had $(\sigma+\beta)(v_i)+\sum\limits_{k=1}^{i-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) + a_{v_i,v_i}$ chips. Just before it fired the second time (at step $j$), it thus had $(\sigma+\beta)(v_i)+\sum\limits_{k=1}^{i-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) + a_{v_i,v_i} + \sum\limits_{k=i+1}^{j-1} a_{v_k,v_i}$ chips (since at step $k$ it obtained $a_{v_k,v_i}$ chips from firing $v_k$). This number of chips must be $\geq \operatorname{outdeg}(v_i)$ (since firing the vertex $v_i$ at step $j$ is allowed). Thus,

$$(\sigma+\beta)(v_i)+\sum_{k=1}^{i-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) + a_{v_i,v_i} + \sum_{k=i+1}^{j-1} a_{v_k,v_i} \geq \operatorname{outdeg}(v_i).$$

In other words,

$$\begin{aligned}
0 &\leq (\sigma+\beta)(v_i)+\sum_{k=1}^{i-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) + a_{v_i,v_i} + \sum_{k=i+1}^{j-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) \\
&= \underbrace{(\sigma+\beta)(v_i)}_{=\sigma(v_i)+\beta(v_i)} + \underbrace{\sum_{k=1}^{i-1} a_{v_k,v_i} + a_{v_i,v_i} + \sum_{k=i+1}^{j-1} a_{v_k,v_i}}_{=\sum_{k=1}^{j-1} a_{v_k,v_i}} - \operatorname{outdeg}(v_i) - \operatorname{outdeg}(v_i) \\
&= \underbrace{\sigma(v_i)}_{\substack{<\operatorname{outdeg}(v_i) \\ (\text{since } \sigma \text{ is stable})}} + \underbrace{\beta(v_i)}_{=\operatorname{outdeg}(v_i)-\operatorname{indeg}(v_i)} + \sum_{k=1}^{j-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) - \operatorname{outdeg}(v_i) \\
&< \operatorname{outdeg}(v_i) + \operatorname{outdeg}(v_i) - \operatorname{indeg}(v_i) + \sum_{k=1}^{j-1} a_{v_k,v_i} - \operatorname{outdeg}(v_i) - \operatorname{outdeg}(v_i) \\
&= \sum_{k=1}^{j-1} a_{v_k,v_i} - \operatorname{indeg}(v_i).
\end{aligned}$$

But for every $k \in \{1, 2, ..., j-1\}$, the number $a_{v_k,v_i}$ counts all edges from $v_k$ to $v_i$. Hence, the sum $\sum\limits_{k=1}^{j-1} a_{v_k,v_i}$ counts all edges from vertices in $\{v_1, v_2, ..., v_{j-1}\}$ to $v_i$, each of them only once (since no two of the vertices $v_1, v_2, ..., v_{j-1}$ are equal), and thus $\sum\limits_{k=1}^{j-1} a_{v_k,v_i} \leq \operatorname{indeg}(v_i)$. Hence,

$$0 < \underbrace{\sum_{k=1}^{j-1} a_{v_k,v_i}}_{\leq \operatorname{indeg}(v_i)} - \operatorname{indeg}(v_i) \leq \operatorname{indeg}(v_i) - \operatorname{indeg}(v_i) = 0,$$

which is a contradiction. Our assumption must therefore be wrong, and Statement 3 is proven.

*Proof of Statement 2:* Let $\sigma$ be a stable chip configuration such that every non-sink vertex fires at least once in the stabilization of $\sigma+\beta$. By Statement 3, every non-sink vertex must have fired at most once in the stabilization of $\sigma+\beta$. Thus, every non-sink vertex must have fired *exactly* once in the stabilization of $\sigma+\beta$. Statement 1 now yields that $\sigma$ is recurrent, so that Statement 2 is proven.

- **Page 22, proof of Lemma 4.2:** The proof ends with the words "which implies $\sigma$ is recurrent by Lemma 2.17." I guess you mean Lemma 4.1 rather than Lemma 2.17 here.

- **Page 23, discussion of "multicluster-firing":** You write: "Since the digraph is Eulerian, $C_m$ may be cluster-fired". It took me half a day to realize why this is true, and my proof is not trivial:
  Denote by $\sigma$ the initial chip configuration, before cluster-firing. We have to show that the configuration $\sigma-\sum\limits_{i\in C_m}\Delta'_i$ is nonnegative. In order to do this, it is enough to show that $\left(\sigma-\sum\limits_{i\in C_m}\Delta'_i\right)(v)\geq 0$ for every $v\in V(G)$. Since this is obvious when $v\notin C_m$, we can WLOG assume $v\in C_m$. Since we can "multicluster-fire" the multiset union of $C_m$, $C_{m-1}$, ..., $C_1$, we have $\left(\sigma-\sum\limits_{k=1}^{m}\sum\limits_{i\in C_k}\Delta'_i\right)(v)\geq 0$. This rewrites as $\sigma(v)-\sum\limits_{k=1}^{m}\sum\limits_{i\in C_k}\Delta'_i(v)\geq 0$.
  We will now show that $\sum\limits_{i\in C_k}\Delta'_i(v)\leq 0$ for every $k\in\{1,2,...,m-1\}$. In fact, let $k\in\{1,2,...,m-1\}$ be arbitrary. Then, $v\in C_m\subseteq C_k$, so that

$$\sum_{i\in C_k}\Delta'_i(v)=\underbrace{\Delta'_v(v)}_{=-\operatorname{outdeg}v+a_{v,v}}+\sum_{\substack{i\in C_k;\\ i\neq v}}\underbrace{\Delta'_i(v)}_{\substack{=a_{i,v}\\ (\text{since } i\neq v)}}=(-\operatorname{outdeg}v+a_{v,v})+\sum_{\substack{i\in C_k;\\ i\neq v}}a_{i,v}$$
$$=\underbrace{a_{v,v}+\sum_{\substack{i\in C_k;\\ i\neq v}}a_{i,v}}_{=\sum\limits_{i\in C_k}a_{i,v}\leq\sum\limits_{i\in V(G)}a_{i,v}=\operatorname{indeg}v}-\underbrace{\operatorname{outdeg}v}_{\substack{\geq\operatorname{indeg}v\\ (\text{since } G \text{ is an Eulerian}\\ \text{digraph with sink})}}\leq\operatorname{indeg}v-\operatorname{indeg}v=0.$$

  Now,

$$\left(\sigma-\sum_{i\in C_m}\Delta'_i\right)(v)=\sigma(v)-\sum_{i\in C_m}\Delta'_i(v)$$
$$=\left(\sigma(v)-\sum_{k=1}^{m}\sum_{i\in C_k}\Delta'_i(v)\right)+\underbrace{\left(\sum_{k=1}^{m}\sum_{i\in C_k}\Delta'_i(v)-\sum_{i\in C_m}\Delta'_i(v)\right)}_{=\sum\limits_{k=1}^{m-1}\sum\limits_{i\in C_k}\Delta'_i(v)}$$
$$=\underbrace{\left(\sigma(v)-\sum_{k=1}^{m}\sum_{i\in C_k}\Delta'_i(v)\right)}_{\geq 0}+\sum_{k=1}^{m-1}\underbrace{\sum_{i\in C_k}\Delta'_i(v)}_{\geq 0}\geq 0.$$

We have thus proven that every vertex $v$ of $G$ satisfies $\left(\sigma - \sum_{i \in C_m} \Delta'_i\right)(v) \geq 0$. Thus, the configuration $\sigma - \sum_{i \in C_m} \Delta'_i$ is nonnegative, so that $C_m$ can be cluster-fired from $\sigma$.

- **Page 24, definition of superstabilization:** You write: "We call the configuration $\sigma_n$ in Corollary 4.6 the **superstabilization** of $\sigma_0$." But why does every chip configuration actually have a superstabilization?
  I think this follows from an analogue of Lemma 2.4:
  **Lemma.** For any Eulerian digraph $G$ with sink, every chip configuration on $G$ can be superstabilized. Moreover, *any* sequence of allowed cluster-firings eventually has to stop.
  *Proof.* For every chip configuration $\sigma$ on $G$, let $c(\sigma)$ denote the total number of chips in $\sigma$. This number is $c(\sigma) = \sum_{v \in V \setminus \{s\}} \sigma(v)$, where $V = V(G)$ and where $s$ is the sink of $G$.
  For every vertex $v$ of $G$, let $t(v)$ be the length of the shortest path from $v$ to $s$ in $G$. (Note that $t(s) = 0$.)
  Let $v_1, v_2, ..., v_{n-1}$ be an ordering of $V \setminus \{s\}$ such that $t(v_1) \leq t(v_2) \leq ... \leq t(v_{n-1})$.
  For every chip configuration $\sigma$ on $G$, let $u(\sigma)$ denote the sequence $(c(\sigma), -\sigma(v_1), -\sigma(v_2), ..., -\sigma(v_{n-1}))$.
  It is easy to see that whenever we perform an allowed cluster-firing on a chip configuration $\sigma$, then the resulting chip configuration $\sigma'$ satisfies $u(\sigma') < u(\sigma)$ with respect to the lexicographic ordering.[1]
  Now, let us start with a chip configuration $\sigma$ and iteratively cluster-fire nonempty subsets of $V \setminus \{s\}$, as long as we can find subsets we are allowed to cluster-fire. Then, each cluster-firing lowers the value of $u(\sigma)$ in the lexicographic ordering. But there are only finitely many possible values of $u(\sigma)$ that can occur during our iterative cluster-firing (because iterative cluster-firing can only decrease but never increase $c(\sigma)$, and for every fixed $c(\sigma)$ there are only finitely many possibilities for $\sigma(v_1)$, $\sigma(v_2)$, ..., $\sigma(v_{n-1})$). Thus, our iterative cluster-firing has to come to an end at some point (i. e., after sufficiently many cluster-firings our

[1] *Proof.* In fact, let $A$ be the subset we use for cluster-firing. There exists some $i \in \{1, 2, ..., n-1\}$ such that $v_i \in A$ (since $A$ is a nonempty subset of $V \setminus \{s\} = \{v_1, v_2, ..., v_{n-1}\}$). Take the *smallest* such $i$. Let $w$ be the second vertex on the shortest path from $v_i$ to $s$. Then, there exists an edge from $v_i$ to $w$, and we have $t(w) < t(v_i)$. We distinguish between two cases:
*Case 1:* The vertex $w$ is the sink $s$. In this case, cluster-firing $A$ results in at least one chip (namely, one of the chips at $v_i$) being moved into the sink (and thus disappearing). Hence, $c(\sigma') < c(\sigma)$ and thus $u(\sigma') < u(\sigma)$ (by the definition of lexicographic ordering).
*Case 2:* The vertex $w$ is not the sink $s$. Then, $w = v_j$ for some $j \in \{1, 2, ..., n-1\}$. Consider this $j$. Then, $t(w) = t(v_j)$. Thus, $t(w) < t(v_i)$ becomes $t(v_j) < t(v_i)$. Hence, $j < i$ (because $t(v_1) \leq t(v_2) \leq ... \leq t(v_{n-1})$). Consequently, the vertex $v_j$ is not in $A$ (since $i$ was defined as the *smallest* $i \in \{1, 2, ..., n-1\}$ such that $v_i \in A$). Hence, cluster-firing $A$ increases the number of chips at $v_j$ by at least 1 (since at least one chip gets fired from $v_i$ to $w = v_j$, whereas $v_j$ itself does not fire). In other words, $\sigma'(v_j) \geq \sigma(v_j) + 1$. Moreover, for every $k \in \{1, 2, ..., i-1\}$, cluster-firing $A$ does not decrease the number of chips at $v_k$ (because $k < i$, so that $v_k \notin A$ (since $i$ was defined as the *smallest* $i \in \{1, 2, ..., n-1\}$ such that $v_i \in A$)). In other words, $\sigma'(v_k) \geq \sigma(v_k)$ for every $k \in \{1, 2, ..., i-1\}$. Combined with the fact that $\sigma'(v_j) \geq \sigma(v_j) + 1$ and that $c(\sigma') \leq c(\sigma)$ (because cluster-firing does not increase the number of chips), this shows that $u(\sigma') < u(\sigma)$.
We have thus showed that $u(\sigma') < u(\sigma)$ in both cases.

chip configuration becomes superstable). The Lemma is proven.
Note that the Lemma just proven is a generalization of Lemma 2.4, since every allowed chip-firing operation is an allowed cluster-firing operation (with the cluster being a one-element subset of $V(G)$). Thus the above proof of the Lemma can also be seen as an alternative proof of Lemma 2.4.

- **Page 26, proof of Lemma 4.9:** You write: "Since $G$ is strongly connected, we may apply this same argument to every state in the future history of the system, with every vertex of $G$ playing the role of $w$." It would be better (for people with a small memory, like me) to state why this holds (namely, because of Lemma 3.6).

- **Page 26, Corollary 4.10:** Replace $\mathcal{T}(G,v)$ by $\mathcal{T}(G,w)$.

- **Page 26, proof of Corollary 4.10:** On the first line of this proof, $\prod\limits_{u\in V}$ should be $\prod\limits_{v\in V}$.

- **Page 27, proof of Corollary 4.11:** On the fourth line of this proof, $\rho(v)=e^{-}$ should be replaced by $\rho^{-}(v)=e^{-}$. Besides, on the third line of the proof, I would replace "obtained from $\rho$" by "obtained from $\rho\mid_{G'}$" (to make clear that "the edge immediately preceding $e$ in the cyclic ordering of the edges emanating from $v$" does not mean an edge pointing outside of $\mathcal{C}$).

- **Page 27, proof of Lemma 4.12:** The claim that the sandpile group $\mathcal{S}(G_v)$ is "also isomorphic to $\mathbb{Z}^n\diagup\mathbb{Z}^n\Delta(G)$" is wrong, unless you mean something different by $\Delta(G)$ than the Laplacian of $G$ you defined on page 3. (The group $\mathbb{Z}^n\diagup\mathbb{Z}^n\Delta(G)$ is actually infinite, since every element of $\mathbb{Z}^n\Delta(G)$ has the sum of its coordinates equal to 0.) In reality, the sandpile group $\mathcal{S}(G_v)$ is isomorphic to $\mathbb{Z}_0^n\diagup\mathbb{Z}^n\Delta(G)$, where $\mathbb{Z}_0^n$ is the subgroup

$$\{x\in\mathbb{Z}^n \mid \text{the sum of the coordinates of } x \text{ is } 0\}$$

of $\mathbb{Z}^n$.
To be honest I would explain this proof differently (with less words and more algebra):
Let $i:\mathbb{Z}^{n-1}\to\mathbb{Z}_0^n$ (where $\mathbb{Z}_0^n$ is defined as above) be the $\mathbb{Z}$-module isomorphism which maps every vector $(x_1,x_2,...,x_{n-1})$ to $(x_1,x_2,...,x_{n-1},-x_1-x_2-...-x_{n-1})$. If $d$ denotes any row of $\Delta'(G)$, then $i(d)$ is the corresponding row of $\Delta(G)$ (because every row of $\Delta(G)$ has its entries sum up to 0), so that $i(d)\in\mathbb{Z}^n\Delta(G)$. Thus, $i(\mathbb{Z}^{n-1}\Delta'(G))\subseteq\mathbb{Z}^n\Delta(G)$.
On the other hand, $\mathbb{Z}^n\Delta(G)$ is the $\mathbb{Z}$-module generated by the rows of $\Delta(G)$. Since the last row of $\Delta(G)$ equals minus the sum of all other rows of $\Delta(G)$, this yields that $\mathbb{Z}^n\Delta(G)$ is the $\mathbb{Z}$-module generated by the first $n-1$ rows of $\Delta(G)$. But since each of these first $n-1$ rows of $\Delta(G)$ lies in $i(\mathbb{Z}^{n-1}\Delta'(G))$ (in fact, if $d$ is one of the first $n-1$ rows of $\Delta(G)$, then $d=i(d')$, where $d'$ is the corresponding row of $\Delta'(G)$), this yields that $\mathbb{Z}^n\Delta(G)\subseteq i(\mathbb{Z}^{n-1}\Delta'(G))$. Combined with $i(\mathbb{Z}^{n-1}\Delta'(G))\subseteq\mathbb{Z}^n\Delta(G)$, this yields $i(\mathbb{Z}^{n-1}\Delta'(G))=\mathbb{Z}^n\Delta(G)$. Hence, $i$ induces a $\mathbb{Z}$-module isomorphism $\mathbb{Z}^{n-1}\diagup\mathbb{Z}^{n-1}\Delta'(G)\to\mathbb{Z}_0^n\diagup\mathbb{Z}^n\Delta(G)$.

Thus, $\mathbb{Z}^{n-1} \diagup \mathbb{Z}^{n-1} \Delta' (G) \cong \mathbb{Z}_0^n \diagup \mathbb{Z}^n \Delta (G)$, which is precisely what we wanted to show.

- **Page 28, about Lemma 5.3:** You write: "The next lemma uses cycle-popping to give a constructive proof of the injectivity of the chip addition operators $E_v$ on acyclic stack configurations. In the case of periodic rotor stacks, we gave a non-constructive proof in Lemma 3.10."
  But doesn't Lemma 5.3 rather prove *surjectivity*? (And surjectivity was already constructively proven - in the case of periodic rotor stacks - in Lemma 3.10.)

- **Page 29, after the proof of Lemma 5.3:** You write: "then the unique path in $\rho_0$ from $v$ to the sink is the **loop-erasure** of the path taken by rotor-router walk started at $v$ with initial configuration $\rho'$." First, what does "loop-erasure" mean? (In general, if we have some directed path with loops, then we can remove them one by one but the outcome depends on the order in which they were removed. Thus, I think, a precise definition of "loop-erasure" would be necessary.) Second, how is this proven?

- **References:** Judging by Sportiello's website ( http://wwwteor.mi.infn.it/~sportiel/ ), the preprint [CPS07] is now avaliable on arXiv as arXiv:0809.3416.