NCDL: Supplementary Material

Joshua J. Horacsek
Department of Computer Science
University of Calgary
Calgary, Alberta
j.horacsek@ncdl.ai

Usman R. Alim
Department of Computer Science
University of Calgary
Calgary, Alberta
ualim@ucalgary.ca

Abstract

This supplementary material provides proofs for the assertions in the main paper, as well as some additional helpful figures to illustrate concepts behind the algorithms we develop. The full implementation for these is also available and serves as further reference, and is available at `https://www.ncdl.ai`

1 Proofs

Proposition 3.1 (κ-index)**.** *Given i, j with $0 \leq i, j < C$, for any $\mathbf{n} \in \mathcal{L}_{\mathcal{R}}, \mathbf{m} \in \mathcal{L}_{\mathcal{S}}$ with $\iota(\mathbf{n}) = i$ and $\iota(\mathbf{m}) = j$, it must be that both $\iota(\mathbf{n}-\mathbf{m})$ and $\iota(\mathbf{n}+\mathbf{m})$ are constant. We define $\kappa_{\pm}(i,j) := \iota(\mathbf{n} \pm \mathbf{m})$.*

Proof. Suppose we have $\mathbf{n}, \mathbf{n}' \in \mathcal{L}_{\mathcal{R}}$ and $\mathbf{m}, \mathbf{m}' \in \mathcal{L}_{\mathcal{S}}$ with $\iota(\mathbf{n}) = \iota(\mathbf{n}') = i$ and $\iota(\mathbf{m}) = \iota(\mathbf{m}') = j$. Then, further suppose for a contradiction that $k := \iota(\mathbf{n} - \mathbf{m})$ and $l := \iota(\mathbf{n}' - \mathbf{m}')$ and $k \neq l$. By definition, we have $\mathbf{n} = \mathbf{v}_i^{\mathcal{R}} + \mathbf{D}\mathbf{a}$, $\mathbf{n}' = \mathbf{v}_i^{\mathcal{R}} + \mathbf{D}\mathbf{a}'$, $\mathbf{m} = \mathbf{v}_j^{\mathcal{S}} + \mathbf{D}\mathbf{b}$, $\mathbf{m}' = \mathbf{v}_j^{\mathcal{S}} + \mathbf{D}\mathbf{b}'$ for some $\mathbf{a}, \mathbf{b}, \mathbf{a}', \mathbf{b}' \in \mathbb{Z}^s$. By assumption it must also be that $\mathbf{n} - \mathbf{m} = \mathbf{v}_k + \mathbf{D}\mathbf{x}$ and $\mathbf{n}' - \mathbf{m}' = \mathbf{v}_l + \mathbf{D}\mathbf{y}$ for $\mathbf{x}, \mathbf{y} \in \mathbb{Z}^s$. Rearranging these shows that both $\mathbf{v}_j - \mathbf{v}_i \in \mathbf{v}_k + \mathbf{D}\mathbb{Z}^s$ and $\mathbf{v}_j - \mathbf{v}_i \in \mathbf{v}_l + \mathbf{D}\mathbb{Z}^s$, which is impossible since our cosets are mutually exclusive. Thus $k = l$. A similar argument holds for the case of addition. $\square$

Convolution Figure 1, helps provide some intuition for the following proposition

Proposition 3.2. *For a lattice tensor $a_{\mathcal{L}_{\mathcal{R}}}$ and filter $f_{\mathcal{L}_{\mathcal{P}}}$ where the region $\mathcal{P}$ is strictly positive, the result of the convolution $o_{b,k,\mathcal{L}_{\mathcal{S}}} := (a_{b,c,\mathcal{L}_{\mathcal{R}}} \star f_{c,k,\mathcal{L}_{\mathcal{P}}})$ can be written in terms of its output cosets as*

$$\tilde{o}_i = \sum_{j=0}^{C-1} \tilde{a}_{\kappa_+(i,j)} \left[\cdot + \delta(i,j)\right] * \tilde{f}_j \tag{1}$$

where $*$ is the traditional Cartesian convolution operator, and $\delta(i,j) := \mathbf{D}^{-1}(\mathbf{v}^{\mathcal{R}}_{\kappa_+(i,j)} - \mathbf{v}^{\mathcal{R}}_i - \mathbf{v}^{\mathcal{S}}_j)$ is a constant by which we shift the appropriate coset of $a_{\mathcal{L}_{\mathcal{R}}}$.

Proof. We start with the definition

$$o_{b,c,\mathcal{L}_{\mathcal{S}}}[n, l, \mathbf{p}] := \sum_{j=0}^{c-1} \sum_{s \in \text{supp}(f)} a[n, j, \mathbf{p} + \mathbf{s}] f[l, j, \mathbf{s}] \tag{2}$$

we restrict to the output on the i^{th} coset, which gives

$$\tilde{o}_i[n, l, \mathbf{p}] = \sum_{j=0}^{c-1} \sum_{s \in \text{supp}(f)} a[n, j, \mathbf{D}\mathbf{p} + \mathbf{v}^{\mathcal{R}}_i + \mathbf{s}] f[l, j, \mathbf{s}]. \tag{3}$$

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

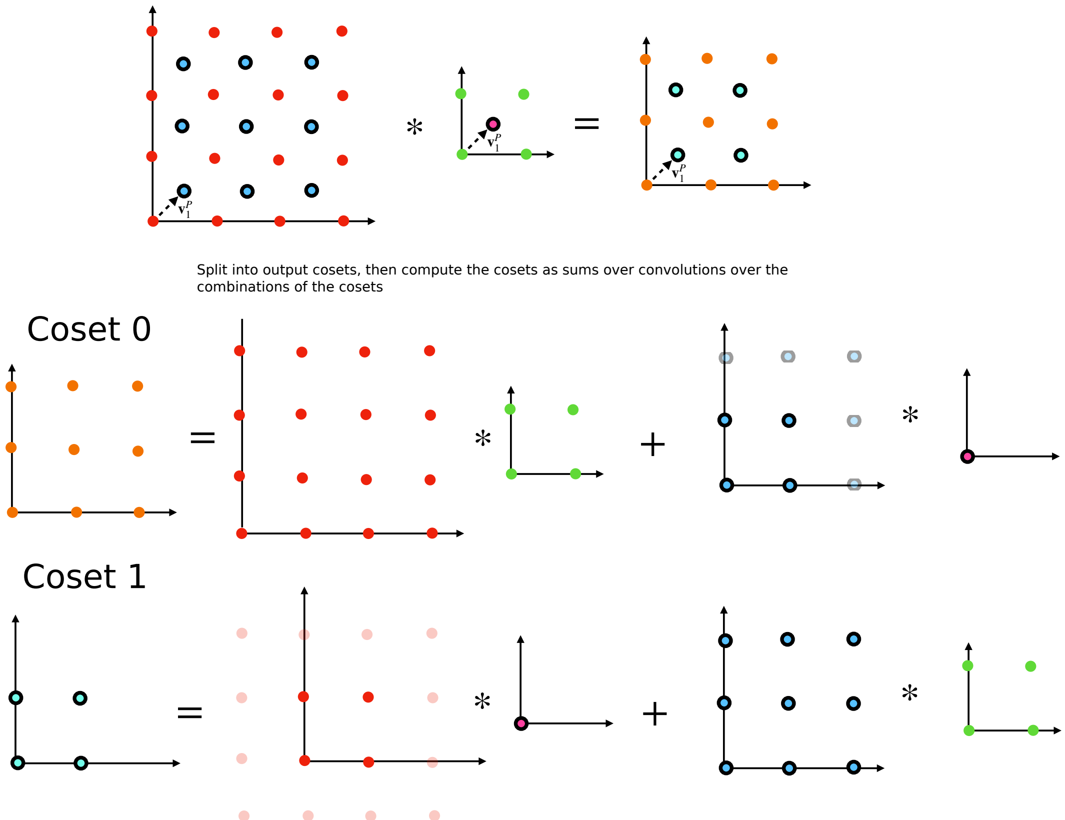

Figure 1: An illustration of the convolution decomposition. The shaded points are not considered in the convolution operations. The κ and δ terms simply generalize the intuition behind these figures.

we split this sum over the cosets of the filter, amd re-arrange the sums

$$\tilde{o}_i[n, l, \mathbf{p}] = \sum_{j=0}^{C-1} \sum_{k=0}^{c-1} \sum_{s \in \text{supp}(\tilde{f}_j)} a[n, k, \mathbf{D}\mathbf{p} + \mathbf{v}_i^{\mathcal{R}} + \mathbf{D}\mathbf{s} + \mathbf{v}_j^{\mathcal{S}}] f[l, k, \mathbf{D}\mathbf{s} + \mathbf{v}_j^{\mathcal{S}}]. \tag{4}$$

We can immediately simplify by definition

$$\tilde{o}_i[n, l, \mathbf{p}] = \sum_{j=0}^{C-1} \sum_{k=0}^{c-1} \sum_{s \in \text{supp}(\tilde{f}_j)} \tilde{a}_{\kappa_+(i,j)}[n, k, \mathbf{p} + \mathbf{s} + \delta(i, j)] \tilde{f}_j[l, k, \mathbf{s}] \tag{5}$$

which gives

$$\tilde{o}_i = \sum_{j=0}^{C-1} \tilde{a}_{\kappa_+(i,j)} \left[\cdot + \delta(i, j)\right] * \tilde{f}_j. \tag{6}$$

The only thing that remains is to ensure that $\delta(i, j)$ is well defined — that is, we need to show that $\mathbf{v}_{\kappa_+(i,j)}^{\mathcal{R}} - \mathbf{v}_i^{\mathcal{R}} - \mathbf{v}_j^{\mathcal{S}} = \mathbf{v}_h^{\mathcal{R}} - (\mathbf{v}_i^{\mathcal{R}} + \mathbf{v}_j^{\mathcal{S}}) \in \mathbf{D}\mathbb{Z}^s$. But this is true, since $(\mathbf{v}_i^{\mathcal{R}} + \mathbf{v}_j^{\mathcal{S}})$ is on the same coset as $\mathbf{v}_{\kappa_+(i,j)}^{\mathcal{R}}$, finishing the proof. $\square$

We now turn to the derivative computation.

Proposition 3.3. *For a lattice tensor $a_{\mathcal{L}_\mathcal{R}}$ and filter $f_{\mathcal{L}_\mathcal{P}}$ where the region $\mathcal{P}$ is strictly positive, with convolution defined as*

$$o[n, i, \mathbf{p}] := (a_{b,c,\mathcal{L}_\mathcal{R}} \star f_{c,k,\mathcal{L}_\mathcal{P}})[n, i, \mathbf{p}] \tag{7}$$

whose output lattice tensor is in $\mathcal{L}_{\mathcal{S}}$ where $\mathcal{L}_R = \mathcal{L}_{\mathcal{S}} \oplus \mathcal{L}_{\mathcal{P}}$, the derivatives of the loss h with respect to the filter and input lattice tensor are given by

$$\frac{\partial h}{\partial a[n,i,\mathbf{k}]} = (o \star \overline{f_{c,k,\mathcal{L}_{\mathcal{P}}}})[n,i,\mathbf{k}], \tag{8}$$

$$\frac{\partial h}{\partial f[i,j,\mathbf{k}]} = \sum_{n=0}^{b-1} \left(\sum_{\mathbf{p}\in\mathcal{L}_{\mathcal{S}}} \frac{\partial h}{\partial o[n,i,\mathbf{p}]} \cdot a[n,j,\mathbf{p}+\mathbf{k}] \right), \tag{9}$$

where $\overline{f}$ mirrors the filter and swaps the channels, i.e. $\overline{f}[i,j,\mathbf{k}] := f[j,i,-\mathbf{k}]$.

Proof. The proof for this is mechanical, we start from the chain rule, then simplify

$$\begin{aligned}
\frac{\partial h}{\partial a[n,i,\mathbf{k}]} &= \sum_m \sum_j \sum_{\mathbf{p}} \frac{\partial h}{\partial o[m,j,\mathbf{p}]} \cdot \frac{\partial o[m,j,\mathbf{p}]}{\partial a[n,i,\mathbf{k}]} && (10)\\
&= \sum_m \sum_j \sum_{\mathbf{p}} \frac{\partial h}{\partial o[m,j,\mathbf{p}]} \cdot \frac{\partial}{\partial a[n,i,\mathbf{k}]} \sum_{l=0}^{c-1} \sum_{s\in\mathrm{supp}(f)} a[n,l,\mathbf{p}+\mathbf{s}] f[j,l,\mathbf{s}] && (11)\\
&= \sum_j \sum_{\mathbf{p}} \frac{\partial h}{\partial o[n,j,\mathbf{p}]} \sum_{s\in\mathrm{supp}(f)} \frac{\partial}{\partial a[n,i,\mathbf{k}]} a[n,i,\mathbf{p}+\mathbf{s}] f[j,i,\mathbf{s}] && (12)\\
&= \sum_j \sum_{\mathbf{p}} \frac{\partial h}{\partial o[n,j,\mathbf{p}]} f[j,i,\mathbf{k}-\mathbf{p}] && (13)\\
&= \sum_j \sum_{\mathbf{q}} \frac{\partial h}{\partial o[n,j,\mathbf{k}-\mathbf{q}]} f[j,i,\mathbf{q}] && (14)\\
&= \sum_j \sum_{q\in\mathrm{supp}(f)} \frac{\partial h}{\partial o[n,j,\mathbf{k}-\mathbf{q}]} f[j,i,\mathbf{q}] && (15)\\
&= \sum_j \sum_{q\in\mathrm{supp}(f)} \frac{\partial h}{\partial o[n,j,\mathbf{k}-\mathbf{q}]} f[j,i,\mathbf{q}] && (16)
\end{aligned}$$

Applying the definition of $\overline{f}$ gives the final result. We follow a similar derivation for the filter f.

$$\begin{aligned}
\frac{\partial h}{\partial f[i,j,\mathbf{k}]} &= \sum_n \sum_m \sum_{\mathbf{p}} \frac{\partial h}{\partial o[n,m,\mathbf{p}]} \cdot \frac{\partial o[n,m,\mathbf{p}]}{\partial f[i,j,\mathbf{k}]} && (17)\\
&= \sum_n \sum_m \sum_{\mathbf{p}} \frac{\partial h}{\partial o[n,m,\mathbf{p}]} \cdot \frac{\partial}{\partial f[i,j,\mathbf{k}]} \sum_{l=0}^{c-1} \sum_{s\in\mathrm{supp}(f)} a[n,l,\mathbf{p}+\mathbf{s}] f[m,l,\mathbf{s}] && (18)\\
&= \sum_n \sum_m \sum_{\mathbf{p}} \frac{\partial h}{\partial o[n,m,\mathbf{p}]} \cdot \sum_{l=0}^{c-1} \sum_{s\in\mathcal{L}} a[n,l,\mathbf{p}+\mathbf{s}] \frac{\partial f[m,l,\mathbf{s}]}{\partial f[i,j,\mathbf{k}]} && (19)\\
&= \sum_n \sum_{\mathbf{p}} \frac{\partial h}{\partial o[n,i,\mathbf{p}]} \cdot a[n,j,\mathbf{p}+\mathbf{k}] && (20)
\end{aligned}$$

which gives the final result. □

Pooling Although Figure 1 illustrates convolution, the same intuition applies; we break up the operation over the cosets of the lattice. We encapsulate this idea in the following proposition.

Proposition 3.4. *For a lattice tensor $a_{\mathcal{L}_{\mathcal{R}}}$ and stencil geometry $\mathcal{L}_{\mathcal{P}}$ where the region $\mathcal{P}$ is strictly positive, the result of the lattice pooling operation $o_{n,c,\mathcal{L}_{\mathcal{R}}}[l,m,\mathbf{n}] = \max_{\mathbf{s}\in\mathcal{L}_{\mathcal{P}}}\{a[l,m,\mathbf{n}+\mathbf{s}]\}$ can be written in terms of its output cosets as*

$$\tilde{o}_i[l,m,\mathbf{n}] = \max_{0\le j<C} \left\{ \max_{\mathbf{s}\in\mathbf{D}^{-1}(\mathcal{L}_{\mathcal{P}}-\mathbf{v}_j)\cap\mathbb{Z}^s} \left\{ \tilde{a}_{\kappa_+(i,j)}\left[\mathbf{n}+\mathbf{s}+\delta(i,j)\right] \right\} \right\} \tag{21}$$

where the maximum operation running over the set $\mathbf{D}^{-1}(\mathcal{L}_{\mathcal{P}}-\mathbf{v}_j)\cap\mathbb{Z}^s$ is the traditional max pool operator, restricted to the j^{th} coset of the stencil.

Proof. We proceed by first restricting the output of

$$o_{n,c,\mathcal{L}_{\mathcal{R}}}[l,m,\mathbf{n}] = \max_{\mathbf{s}\in\mathcal{L}_{\mathcal{P}}}\{a[l,m,\mathbf{n}+\mathbf{s}]\} \tag{22}$$

to the i^{th} coset. By definition, we have

$$\tilde{o}_i[l,m,\mathbf{n}] = \max_{\mathbf{s}\in\mathcal{L}_{\mathcal{P}}}\{a[l,m,\mathbf{D}\mathbf{n}+\mathbf{v}_i^{\mathcal{R}}+\mathbf{s}]\}. \tag{23}$$

We can take the maximum over the maximum of the cosets (the max of the max is still the max), explicitly the restriction to the set $\mathbf{D}^{-1}(\mathcal{L}_{\mathcal{P}}-\mathbf{v}_j)\cap\mathbb{Z}^s$ is the restriction to the j^{th} coset of the stencil

$$\tilde{o}_i[n,l,\mathbf{n}] = \max_{0\le j<C}\left\{\max_{\mathbf{s}\in\mathbf{D}^{-1}(\mathcal{L}_{\mathcal{P}}-\mathbf{v}_j)\cap\mathbb{Z}^s}\{a[l,m,\mathbf{D}\mathbf{n}+\mathbf{v}_i^{\mathcal{R}}+\mathbf{D}\mathbf{s}+\mathbf{v}_j^{\mathcal{P}}]\}\right\}. \tag{24}$$

then by definition, we have

$$\tilde{o}_i = \max_{0\le j<C}\left\{\max_{\mathbf{s}\in\mathbf{D}^{-1}(\mathcal{L}_{\mathcal{P}}-\mathbf{v}_j)\cap\mathbb{Z}^s}\left\{\tilde{a}_{\kappa_+(i,j)}\left[\mathbf{n}+\mathbf{s}+\delta(i,j)\right]\right\}\right\} \tag{25}$$

as desired. □

We again turn to derivative computation.

Proposition 3.5. *For a lattice tensor $a_{\mathcal{L}_{\mathcal{R}}}$ and filter geometry $\mathcal{L}_{\mathcal{P}}$ where the region $\mathcal{P}$ is strictly positive, lattice pooling*

$$o_{b,c,\mathcal{L}_R}[n,i,\mathbf{k}] = \max_{\mathbf{s}\in\mathcal{L}_P}\{a[n,i,\mathbf{k}+\mathbf{s}]\} \tag{26}$$

whose output lattice tensor is in $\mathcal{L}_{\mathcal{S}}$, has the gradients given by

$$\frac{\partial h}{\partial a[n,i,\mathbf{k}]} = \sum_{\mathbf{p}\in\mathcal{L}_{\mathcal{P}}}\frac{\partial h}{\partial o[n,i,\mathbf{k}-\mathbf{p}]}\cdot\mu[n,i,\mathbf{k}-\mathbf{p},\mathbf{p}] \tag{27}$$

where

$$\mu[n,i,\mathbf{p},\mathbf{q}] := \begin{cases}1 & \textit{if } \max_{\mathbf{s}\in\mathcal{L}_P}\{a[n,i,\mathbf{p}+\mathbf{s}]\} = a[n,i,\mathbf{p}+\mathbf{q}]\\ 0 & \textit{otherwise}.\end{cases} \tag{28}$$

Proof. The proof of this is, again, mechanical. We start by noting that

$$\begin{aligned} o_{b,c,\mathcal{L}_R}[n,i,\mathbf{k}] &= \max_{\mathbf{s}\in\mathcal{L}_P}\{a[n,i,\mathbf{k}+\mathbf{s}]\} && (29)\\ &= \sum_{\mathbf{s}\in\mathcal{L}_{\mathcal{P}}}\mu[n,i,\mathbf{k},\mathbf{s}]\cdot a[n,i,\mathbf{k}+\mathbf{s}] && (30)\end{aligned}$$

We proceed by noting the chain rule

$$\begin{aligned} \frac{\partial h}{\partial a[n,i,\mathbf{k}]} &= \sum_m\sum_j\sum_{\mathbf{p}}\frac{\partial h}{\partial o[m,j,\mathbf{p}]}\cdot\frac{\partial o[m,j,\mathbf{p}]}{\partial a[n,i,\mathbf{k}]} && (31)\\ &= \sum_m\sum_j\sum_{\mathbf{p}}\frac{\partial h}{\partial o[m,j,\mathbf{p}]}\cdot\frac{\partial}{\partial a[n,i,\mathbf{k}]}\sum_{\mathbf{s}\in\mathcal{L}_{\mathcal{P}}}\mu[m,j,\mathbf{p},\mathbf{s}]a[m,j,\mathbf{p}+\mathbf{s}] && (32)\\ &= \sum_{\mathbf{p}}\frac{\partial h}{\partial o[n,i,\mathbf{p}]}\cdot\frac{\partial}{\partial a[n,i,\mathbf{k}]}\sum_{\mathbf{s}\in\mathcal{L}}\mu[n,i,\mathbf{p},\mathbf{s}]\cdot a[n,i,\mathbf{p}+\mathbf{s}] && (33)\end{aligned}$$

Where this last simplification, simply expand the range of the last sum (since we may sum over additional points of the lattice, μ will nullify these points). Finally, we may treat μ as a constant, as it remains constant in a small neighborhood of a. This gives

$$\begin{aligned} \frac{\partial h}{\partial a[n,i,\mathbf{k}]} &= \sum_{\mathbf{p}}\frac{\partial h}{\partial o[n,i,\mathbf{p}]}\cdot\mu[n,i,\mathbf{p},\mathbf{k}-\mathbf{p}] && (34)\\ &= \sum_{\mathbf{p}}\frac{\partial h}{\partial o[n,i,\mathbf{k}-\mathbf{p}]}\cdot\mu[n,i,\mathbf{k}-\mathbf{p},\mathbf{p}] && (35)\end{aligned}$$

where the last line follows from a change of variables, restricting the sum to the filters stencil gives the desired result. □

2 Expanding on Padding and Down/Upsampling

Padding Figure 2 illustrates the differences between left and right padding. Right padding is practically trivial. However, when padding to the left, we must ensure that the coset vectors remain consistent.

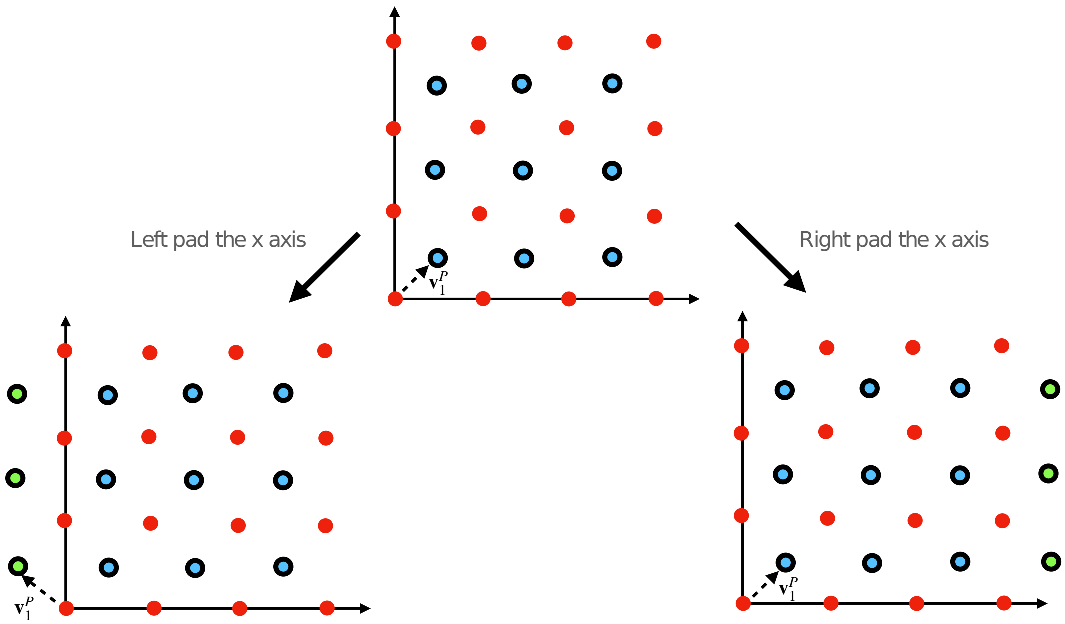

Figure 2: Padding by a unit of 1 in both left and right cases. Left padding is more involved, since it requires shifting the coset vectors.

Downsampling & Upsampling Figure 3 illustrates down and up sampling with a non-dyadic subsampling lattice. Only two lattices are involved with the Quincunx lattice, however, a similar example exists in 3D, where we subsample from the Cartesian lattice to the FCC lattice, to the BCC lattice, then back to the Cartesian lattice.

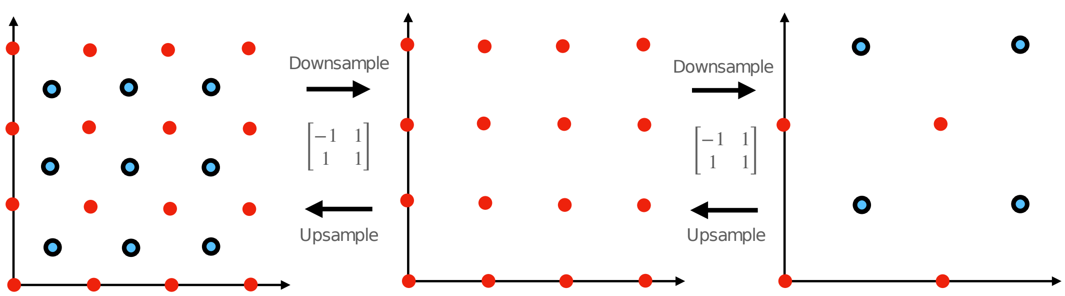

Figure 3: A non dyadic downsampling scheme. The lattice is down/upsampled by the given matrix.