1 Definitions

Notation. We denote complex scalars by a, b,..., complex vectors by x, y,..., vectors of complex vectors by $\mathbf{x}$, $\mathbf{y}$,..., and operators by M, N,.... Component i of vector x is denoted by x_i, component i of $\mathbf{x}$ by $\mathbf{x}_i$. The direct sum of two vectors x and y glues them together:

$$x \oplus y := (x_0, \dots, x_{m-1}, y_0, \dots, y_{n-1}), \quad \mathbf{x} \oplus \mathbf{y} = (\mathbf{x}_0, \dots, \mathbf{x}_{m-1}, \mathbf{y}_0, \dots, \mathbf{y}_{n-1}).$$

We define the cross product of two vectors to produce a vector of vectors:

$$x \times y := \mathbf{x}, \quad \text{with} \quad \mathbf{x} = (x, y).$$

We define the tensor product of two vectors as usual:

$$x \otimes y := x_0 y \oplus \ldots \oplus x_{n-1} y.$$

We define the tensor product of two vectors of complex vectors componentwise:

$$\mathbf{x} \otimes \mathbf{y} := (\mathbf{x}_0 \otimes \mathbf{y}_0, \dots, \mathbf{x}_{n-1} \otimes \mathbf{y}_{n-1}).$$

We denote the canonical basis vectors of a complex vector space by

$$e_i^N \in \mathbb{C}^N.$$

Operators. We define an operator as mapping from k complex vectors to m complex vectors:

$$M : \mathbb{C}^{n_0} \times \dots \times \mathbb{C}^{n_{k-1}} \to \mathbb{C}^{N_0} \times \dots \times \mathbb{C}^{N_{m-1}}; \ \mathbf{x} \mapsto M(\mathbf{x}). \tag{1}$$

An operator $M : \mathbb{C}^m \times \mathbb{C}^n \to \mathbb{C}^k$ automatically applies the isomorphism $\mathbb{C}^m \times \mathbb{C}^n \cong \mathbb{C}^{m+n}$ when applied to only one input vector $x \in \mathbb{C}^{m+n}$.

For M defined in (1) we define the *arity* as the pair (k, m) and the *signature* as the pair of tuples $\big((n_0, \dots, n_k - 1), (N_0, \dots, N_m - 1)\big)$.

We define the *projection* operator:

$$\pi_{m_0,\dots,m_{k-1}}(\mathbf{x}) := (\mathbf{x}_{m_0}, \dots, \mathbf{x}_{k-1}). \tag{2}$$

We alternatively write

$$\pi_x(\mathbf{x}) = x \quad \text{for} \quad \mathbf{x} = (x, y),$$

if the components of the vector $\mathbf{x}$ are named.

The generalization of the identity matrix I_n is the *identity operator*

$$\mathrm{I}^{k_0 \times \dots \times k_{n-1} \to m \times \dots \times m} : \mathbb{C}^{k_0} \times \dots \times \mathbb{C}^{k_{n-1}} \to \mathbb{C}^m \times \dots \times \mathbb{C}^m \quad , \quad m = \Pi_{i=0}^{n-1} k_i$$

with

$$\mathrm{I}^{k_0 \times \dots \times k_{n-1} \to m_0 \times \dots \times m_{r-1}}(\mathbf{x}) = (\mathbf{x}_0 \otimes \dots \otimes \mathbf{x}_{n-1}, \dots, \mathbf{x}_0 \otimes \dots \otimes \mathbf{x}_{n-1})$$

We use two short-hand notations:

$$\mathrm{I}^n := \mathrm{I}^{n \to n} \quad \text{and} \quad \mathrm{I}^{m \times n} := \mathrm{I}^{m \to m} \times \mathrm{I}^{n \to n}$$

Any matrix $M \in \mathbb{C}^{m \times n}$ induces an operator

$$\mathrm{M} : \mathbb{C}^{mn} \to \mathbb{C}^{mn}; \ \mathbf{x} \mapsto \big(M \cdot \mathbf{x}_0\big)$$

with "$\cdot$" being the matrix-vector product.

Using this induction, the generalization of the stride permutation matrix L_m^{mn} is the *stride operator*

$$\mathrm{L}_m^{mn} : \mathbb{C}^{mn} \to \mathbb{C}^{mn};\ \mathbf{x} \mapsto \big(\mathrm{L}_m^{mn} \cdot \mathbf{x}_0\big).$$

The *vector sum* operator is defined by

$$\Sigma_n : \mathbb{C}^n \to \mathbb{C}; x \mapsto \sum_{i=0}^{n-1} x_i$$

The *constant vector* operator is defined by

$$\mathrm{C}_c : \mathbb{C}^0 \to \mathbb{C}^n; x \mapsto c$$

The *vector split* operator is defined by

$$\mathrm{S}_n : \mathbb{C}^n \to \mathbb{C}^1 \times \cdots \times \mathbb{C}^1; x \mapsto (x_0, \ldots, x_{n-1})$$

The *vector join* operator is defined by

$$\mathrm{J}_n : \mathbb{C}^1 \times \cdots \times \mathbb{C}^1 \to \mathbb{C}^n; \mathbf{x} \mapsto \mathbf{x}_0 \oplus \cdots \oplus \mathbf{x}_{n-1}$$

Operations. An n-ary operation $\diamond(\cdot, \ldots, \cdot)$ constructs a new operator M from a list of n operators M_i:

$$\big(\diamond(M_0, \ldots, M_{n-1})\big)(\mathbf{x}) := \diamond\big(M_0(\mathbf{x}), \ldots, M_{n-1}(\mathbf{x})\big). \tag{3}$$

When possible, we prefer infix notation over prefix notation:

$$A \diamond B \diamond C = \diamond(A, B, C).$$

We treat operations with multiple arities the same for any arity: "$\diamond$" is the same operator in both cases:

$$A \diamond B \quad \text{and} \quad A \diamond B \diamond C.$$

A scalar operation

$$\diamond(a, b, \ldots), \quad a, b, \ldots \in \mathbb{C}$$

naturally induces (by componentwise evaluation) the vector operation

$$\diamond(x, y, \ldots) := \big(\diamond(x_0, y_0, \ldots), \ldots, \diamond(x_{n-1}, y_{n-1}, \ldots)\big) \quad x, y, \ldots \in \mathbb{C}^n$$

and further (again by componentwise evaluation) the operation on vectors of complex vectors

$$\diamond(\mathbf{x}, \mathbf{y}, \ldots) := \big(\diamond(\mathbf{x}_0, \mathbf{y}_0, \ldots), \ldots, \diamond(\mathbf{x}_{n-1}, \mathbf{y}_{n-1}, \ldots)\big) \quad \mathbf{x}, \mathbf{y}, \ldots \in \mathbb{C}^{n_0} \times \cdots \times \mathbb{C}^{n_{n-1}}$$

which finally induces an operation on operators by (3).

We now introduce the most important operations. Compatible operators can be *composed*:

$$(M \circ N)(\mathbf{x}) := M(N(\mathbf{x})). \tag{4}$$

We denote *iterative composition* by

$$\left(\prod_{i=0}^{n-1} M_i\right)(\mathbf{x}) := \big(M_0 \circ \cdots \circ M_{n-1}\big)(\mathbf{x})). \tag{5}$$

The output of multiple operators applied to the same input is *concatenated* by the identity operation:

$$(M_0, \dots, M_n - 1)(\mathbf{x}) := (M_0(\mathbf{x}), \dots, M_{n-1}(\mathbf{x})). \tag{6}$$

The *cross product* of two operators "glues" them together:

$$(M \times N)(\mathbf{x} \oplus \mathbf{y}) := M(\mathbf{x}) \oplus N(\mathbf{y}) \tag{7}$$

The following operations are induced from scalar operations. Operators can be added (which means pointwise addition of their outputs):

$$(M + N)(\mathbf{x}) := M(\mathbf{x}) + N(\mathbf{x}). \tag{8}$$

Operators can be multiplied (which means pointwise multiplication of their outputs):

$$(M \cdot N)(\mathbf{x}) := M(\mathbf{x}) \cdot N(\mathbf{x}). \tag{9}$$

Tensor product of operators. The tensor product of two operators is only defined if both have the same arity. The resulting operator has the same arity as both factors and the signature is the pointwise multiplication of the factors' signatures. For

$$A^{m_0 \times \dots \times m_{r-1} \to M_0 \times \dots \times M_{s-1}} \quad \text{and} \quad B^{n_0 \times \dots \times n_{r-1} \to N_0 \times \dots \times N_{s-1}}$$

we define the *left tensor product*

$$(A \grave{\otimes} B)\big(\Sigma_{i=0}^{m_0-1} e_i^{m_0} \otimes \mathbf{x}_0^i, \dots, \Sigma_{i=0}^{m_0-1} e_i^{m_{r-1}} \otimes \mathbf{x}_{r-1}^i\big) := \sum_{i_0=0}^{m_0-1} \cdots \sum_{i_{r-1}=0}^{m_{r-1}-1} A\big(e_{i_0}^{m_0}, \dots, e_{i_{r-1}}^{m_{r-1}}\big) \otimes B\big(\mathbf{x}_0^{i_0}, \dots, \mathbf{x}_{r-1}^{i_{r-1}}\big) \tag{10}$$

and the *right tensor product*

$$(A \acute{\otimes} B)\big(\Sigma_{i=0}^{n_0-1} \mathbf{x}_0^i \otimes e_i^{n_0}, \dots, \Sigma_{i=0}^{n_0-1} \mathbf{x}_{r-1}^i \otimes e_i^{n_{r-1}}\big) := \sum_{i_0=0}^{n_0-1} \cdots \sum_{i_{r-1}=0}^{n_{r-1}-1} A\big(\mathbf{x}_0^{i_0}, \dots, \mathbf{x}_{r-1}^{i_{r-1}}\big) \otimes B\big(e_{i_0}^{n_0}, \dots, e_{i_{r-1}}^{n_{r-1}}\big). \tag{11}$$

As notation, we use the *tensor* product for the identity operator:

$$\mathrm{I} \otimes A := \mathrm{I} \grave{\otimes} A \quad \text{and} \quad A \otimes \mathrm{I} := A \acute{\otimes} \mathrm{I}\,.$$

2 Examples

2.1 Signal processing

Cooley-Tukey FFT. We now reexpress the Cooley-Tukey FFT using our new operator-notation, generalizing the well-known Kronecker product formulation. The original formulation is the matrix factorization

$$\mathrm{DFT}_{mn} = (\mathrm{DFT}_m \otimes \mathrm{I}_n)\, \mathrm{T}_m^{mn} (\mathrm{I}_m \otimes \mathrm{DFT}_n)\, \mathrm{L}_m^{mn} \tag{12}$$

which describes the actual computation

$$\mathrm{DFT}_{mn} \cdot x = (\mathrm{DFT}_m \otimes \mathrm{I}_n) \cdot (\mathrm{T}_m^{mn} \cdot ((\mathrm{I}_m \otimes \mathrm{DFT}_n) \cdot (\mathrm{L}_m^{mn} \cdot x)))$$

with "$\cdot$" being the matrix-vector product. We define the DFT-specific operators

$$\mathrm{DFT}_{mn} : \mathbb{C}^{mn} \to \mathbb{C}^{mn};\ \mathbf{x} \mapsto \mathrm{DFT} \cdot \mathbf{x} \quad \text{and} \quad \mathrm{T}_n^{mn} : \mathbb{C}^{mn} \to \mathbb{C}^{mn};\ \mathbf{x} \mapsto \mathrm{T}_n^{mn} \cdot \mathbf{x}.$$

Then (12) becomes

Rule 1 (Cooley-Tukey FFT)

$$\mathrm{DFT}_{mn} = (\mathrm{DFT}_m \otimes \mathrm{I}^{n\to n}) \circ \mathrm{T}_m^{mn} \circ (\mathrm{I}^{m\to m} \otimes \mathrm{DFT}_n) \circ \mathrm{L}_m^{mn} \tag{13}$$

and the actual computation becomes

$$\mathrm{DFT}_{mn}(\mathbf{x}) = (\mathrm{DFT}_m \otimes \mathrm{I}^{n\to n})(\mathrm{T}_m^{mn}((\mathrm{I}^{m\to m} \otimes \mathrm{DFT}_n)(\mathrm{L}_m^{mn}(\mathbf{x})))).$$

Convolution. Convolution of two signals cannot be expressed as matrix-vector product. However, it can easily be defined as operator,

$$\mathrm{Conv}_n : \mathbb{C}^n \times \mathbb{C}^n \to \mathbb{C}^n$$

and a breakdown for it written as

Rule 2 (Fast convolution)

$$\mathrm{Conv}_n = \mathrm{iDFT}_n \circ \big((\mathrm{DFT}_n \circ \pi_0) \cdot (\mathrm{DFT}_n \circ \pi_1)\big). \tag{14}$$

2.2 Matrix multiplication

In this section we use the MMM to showcase the extension of the Spiral approach to more general operators. We define the MMM as operator, investigate the meaning of expressions of the identity, stride, and MMM operator using "$\circ$", "$\otimes$", and "$\times$". We cast the well-known tiling of the MMM as three breakdown rules (one for each dimension). Next we show how we obtain a non-recursive formula for tiling of two dimensions by applying two recursive rules plus formula manipulation.

Definition of MMM as operator. We define the matrix-matrix multiplication operator

$$\mathrm{MMM}^{N,M,K} : \mathbb{C}^{KN} \times \mathbb{C}^{KM} \to \mathbb{C}^{MN} \tag{15}$$

by

$$C = \mathrm{MMM}^{N,M,K}(A, B) :\Leftrightarrow C_{iM+j} = \sum_{k=0}^{K-1} A_{iK+k} \cdot B_{kM+j} \quad , \quad 0 \le i < N,\ 0 \le j < M.$$

MMM operator and tensor products. We now investigate the meaning of the tensor products

$$\mathrm{I}^{N/N_B \times 1 \to N/N_B} \otimes \mathrm{MMM}^{N_B,M,K} \quad \text{and} \quad \mathrm{MMM}^{N,M_B,K} \otimes \mathrm{I}^{1\times M/M_B \to M/M_B}$$

for this operator.

First we compute the operation of $\mathrm{I}^{N/N_B \times 1 \to N/N_B} \otimes \mathrm{MMM}^{N_B,M,K}$. Therefore we cut the linearized $N \times K$-matrix $A \in \mathbb{C}^{KN}$ into N/N_B horizontal $N_B \times K$-stripes $A^r \in \mathbb{C}^{KN_B}$:

$$A = \sum_{r=0}^{N/N_B-1} e_r^{N/N_B} \otimes A^r \quad \text{with} \quad A^r = \sum_{i=0}^{N_B-1} \sum_{k=0}^{K-1} \big(e_i^{N_B} \otimes e_k^K\big) A_{rN_BK+iK+k}.$$

We now evaluate $\big(\mathrm{I}^{N/N_B \times 1 \to N/N_B} \otimes \mathrm{MMM}^{N_B,M,K}\big)(A, B)$ using (10):

$$\begin{aligned}
&\big(\mathrm{I}^{N/N_B \times 1 \to N/N_B} \otimes \mathrm{MMM}^{N_B,M,K}\big)(A, B) \\
&\quad = \big(\mathrm{I}^{N/N_B \times 1 \to N/N_B} \otimes \mathrm{MMM}^{N_B,M,K}\big)\big(\Sigma_{r=0}^{N/N_B-1} e_r^{N/N_B} \otimes A^r, B\big) \\
&\quad = \sum_{r=0}^{N/N_B-1} \mathrm{I}^{N/N_B \times 1 \to N/N_B} \big(e_r^{N/N_B}, e_0^1\big) \otimes \mathrm{MMM}^{N_B,M,K} \big(A^r, B\big) \\
&\quad = \sum_{r=0}^{N/N_B-1} e_r^{N/N_B} \otimes \mathrm{MMM}^{N_B,M,K} \big(A^r, B\big).
\end{aligned}$$

This shows that the operator $\mathrm{I}^{N/N_B\times 1\to N/N_B}\otimes \mathrm{MMM}^{N_B,M,K}$ expresses blocking of $\mathrm{MMM}^{N,M,K}$ by cutting the output and the first input into horizontal stripes of width N_B.

Next we compute the operation of $\mathrm{MMM}^{N,M_B,K}\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}$. Therefore we cyclically distribute the columns of the linearized $K\times M$-matrix $B\in\mathbb{C}^{KM}$ into M/M_B vertical groups and collect them as $K\times M_B$-matrices $B^r\in\mathbb{C}^{KM_B}$:

$$B=\sum_{r=0}^{M/M_B-1} B^r\otimes e_r^{M/M_B} \quad \text{with} \quad B^r=\sum_{k=0}^{K-1}\sum_{j=0}^{M_B-1}\big(e_k^K\otimes e_j^{M_B}\big)B_{kM+jM/M_B+r}.$$

We now evaluate $\big(\mathrm{MMM}^{N,M_B,K}\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}\big)(A,B)$ using (11):

$$\begin{aligned}
&\big(\mathrm{MMM}^{N,M_B,K}\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}\big)(A,B)\\
&\quad=\big(\mathrm{MMM}^{N,M_B,K}\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}\big)\big(A,\Sigma_{r=0}^{M/M_B-1}B^r\otimes e_r^{M/M_B}\big)\\
&\quad=\sum_{r=0}^{M/M_B-1}\mathrm{MMM}^{N,M_B,K}\big(A,B^r\big)\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}\big)\big(e_0^1,e_r^{M/M_B}\big)\\
&\quad=\sum_{r=0}^{M/M_B-1}\mathrm{MMM}^{N,M_B,K}\big(A,B^r\big)\otimes e_r^{M/M_B}.
\end{aligned}$$

This shows that the operator $\mathrm{MMM}^{N,M_B,K}\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}$ expresses blocking of $\mathrm{MMM}^{N,M,K}$ by cutting the output and the second input into vertical stripes of width 1. Each smaller $\mathrm{MMM}^{N,M_B,K}$ operates on M_B of these stripes at distance M/M_B.

MMM operator and the stride operator. In order to obtain blocking B into M/M_B vertical stripes of width M_B, we need to reorder the columns of B before and after the tensor product. This is achieved by the operator

$$\mathrm{I}^{KN\to KN}\times\big(\mathrm{I}^{K\to K}\otimes \mathrm{L}_{M_B}^{M}\big)$$

on the input side and

$$\mathrm{I}^{N\to N}\otimes \mathrm{L}_{M/M_B}^{M}$$

on the output side.

Tiling of MMM as operator breakdown rules. Collecting the results from this section, we can express two blocking strategies for MMM as operator breakdown rules for the operator $\mathrm{MMM}^{N,M,K}$.

Rule 3 (Base case)

$$\mathrm{MMM}^{1,1,1}\to \pi^{(A,B)\to A}\cdot\pi(A,B)\to B \tag{16}$$

Rule 4 (Blocking of the N dimension)

$$\mathrm{MMM}^{N,M,K}\to \mathrm{I}^{N/N_B\times 1\to N/N_B}\otimes \mathrm{MMM}^{N_B,M,K} \tag{17}$$

```
// MMM by tiling the N-dimension with block size NB
// C: NxM, A:NxK, B: KxM
// Ci: NBxM, Ai:NBxK tiles
for i=0:NB:N/NB-1
   Ci = MMM(Ai, B)
```

$$\begin{bmatrix} C_0\\ \hline \vdots \\ \hline C_{N/N_B-1}\end{bmatrix}=\begin{bmatrix} A_0\\ \hline \vdots \\ \hline B_{N/N_B-1}\end{bmatrix}B$$

Rule 5 (Blocking of the M dimension)

$$\mathrm{MMM}^{N,M,K} \rightarrow \left(\mathrm{I}^{N\rightarrow N} \otimes \mathrm{L}^{M}_{M/M_B}\right) \circ \left(\mathrm{MMM}^{N,M_B,K} \otimes \mathrm{I}^{1\times M/M_B \rightarrow M/M_B}\right) \circ \left(\mathrm{I}^{KN\rightarrow KN} \times (\mathrm{I}^{K\rightarrow K} \otimes \mathrm{L}^{M}_{M_B})\right) \quad (18)$$

```
// MMM by tiling the M-dimension with block size MB
// C: NxM, A:NxK, B: KxM
// Cj: NxMB, Bi:KxMB tiles
for j=0:MB:M/MB-1
   Cj = MMM(A, Bj)
```

$$\left[\; C_0 \;\middle|\; \dots \;\middle|\; C_{M/M_B-1} \;\right] = A \left[\; B_0 \;\middle|\; \dots \;\middle|\; B_{M/M_B-1} \;\right]$$

Rule 6 (Blocking of the K dimension)

$$\mathrm{MMM}^{N,M,K} \rightarrow \left(\Sigma_{K^2/K_B^2} \otimes \mathrm{I}^{MN\rightarrow MN}\right) \circ \left(\mathrm{I}^{K/K_B \times K/K_B \rightarrow K^2/K_B^2} \otimes \mathrm{MMM}^{N,M,K_B}\right) \circ \left((\mathrm{L}^{NK/K_B}_{K/K_B} \otimes \mathrm{I}^{K_B\rightarrow K_B}) \times \mathrm{I}^{KM\rightarrow KM}\right) \quad (19)$$

```
// MMM by tiling the K-dimension with block size KB
// C: NxM, A:NxK, B: KxM
// Ak: NxKB, Bk:KBxM tiles
C = 0
for k=0:KB:K/KB-1
   C = C + MMM(Ak, Bk)
```

$$C = A_0B_0 + \cdots + A_{K/K_B-1}B_{K/K_B-1} \quad \textit{with} \quad A = \left[\; A_0 \;\middle|\; \dots \;\middle|\; A_{K/K_B-1} \;\right] \quad \textit{and} \quad B = \begin{bmatrix} B_0 \\ \hline \vdots \\ \hline B_{K/K_B-1} \end{bmatrix}$$

The algorithm space spanned by Rules 3–6 contains the algorithm space of ATLAS. (Not the implementation space, though.)

2.3 LU factorization

Definition of LU factorization as operator. We define the LU factorization and triangular solve for multiple right-hand sides as operators:

$$\mathrm{LU}^N : \mathbb{C}^{N^2} \rightarrow \mathbb{C}^{N^2} \quad (20)$$

$$\mathrm{TS}^N_{AL^{-1}} : \mathbb{C}^{N^2} \times \mathbb{C}^{N^2} \rightarrow \mathbb{C}^{N^2} \quad (21)$$

$$\mathrm{TS}^N_{U^{-1}A} : \mathbb{C}^{N^2} \times \mathbb{C}^{N^2} \rightarrow \mathbb{C}^{N^2} \quad (22)$$

Rule 7 (Recursive LU factorization)

$$
\begin{aligned}
\mathrm{LU}^{N} \rightarrow & \left(\mathrm{L}_{N}^{4N} \otimes \mathrm{I}^{N/4 \rightarrow N/4} \right) \\
& \circ \left(\mathrm{I}^{3N^2/4 \rightarrow 3N^2/4} \oplus \left(\mathrm{LU}^{N/2} \circ (\Sigma_2 \otimes \mathrm{I}^{N^2/4 \rightarrow N^2/4}) \circ (\mathrm{MMM}^{N/2,N/2,N/2} \oplus \mathrm{I}^{N^2/4 \rightarrow N^2/4}) \right) \right) \\
& \circ \left(\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \otimes \mathrm{I}^{N/4 \rightarrow N/4} \right) \\
& \circ \left(\mathrm{I}^{N^2/4 \rightarrow N^2/4} \oplus \mathrm{TS}_{AL^{-1}}^{N/2} \oplus \mathrm{TS}_{U^{-1}A}^{N/2} \oplus \mathrm{I}^{N^2/4 \rightarrow N^2/4} \right) \\
& \circ \left(\begin{bmatrix} 1 & & & \\ 1 & & & \\ & 1 & & \\ 1 & & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \otimes \mathrm{I}^{N/4 \rightarrow N/4} \right) \circ \left(\mathrm{LU}^{N/2} \oplus \mathrm{I}^{3N^2/4 \rightarrow 3N^2/4} \right) \circ \left(\mathrm{L}_{4}^{4N} \otimes \mathrm{I}^{N/4 \rightarrow N/4} \right)
\end{aligned} \tag{23}
$$

2.4 Viterbi Decoder

A Viterbi decoder is parameterized by the following choices:

- State machine size: 2^m
- Number of code words to work on: n
- Code words: $c \in \mathbb{R}^N$
- Error metric: $e_{i,j,k} : \mathbb{R}^{nN} \times \mathbb{R} \rightarrow \mathbb{R}$
- Choice function: $f : \mathbb{R}^2 \rightarrow \mathbb{R} \times \mathbb{N}$
- Start vector: $x \in \mathbb{R}^{2^m}$

We define the computationally most expensive step within Viterbi decoding as an operator:

$$
\mathrm{Vit}_{m,n,N}^{e,f,x} : \mathbb{R}^{nN} \rightarrow \mathbb{R}^{2^m} \times \mathbb{N}^{2^m n}. \tag{24}
$$

The actual decoding is done by evaluating

$$
(y, d) = \mathrm{Vit}_{m,n,N}^{e,f,x} \left(c_0 \oplus \cdots \oplus c_{n-1} \right)
$$

with c_i being the received codewords. y is the probability vector and d the decision vector used by the traceback part of the Viterbi decoder. All computation is performed within Viterbi *butterflies*:

$$
\mathrm{V}_{i,j}^{e,f} : \mathbb{R}^2 \times \mathbb{R}^{2n} \times \mathbb{R}^{nN} \rightarrow \mathbb{R}^2 \times \mathbb{R}^{2n} \times \mathbb{R}^{nN}; \left(x, d, c \right) \mapsto \left(y, d', c \right) \quad , \quad 0 \leq i < n, \ 0 \leq j < 2^{m-1} \tag{25}
$$

with

$$
\begin{aligned}
(y_k, v_k) &= f\left(e_{i,j,2k}(c, x_0), e_{i,j,2k+1}(c, x_1) \right) \quad , \quad 0 \leq k < 2 \\
d'_r &= \begin{cases} v_{r \bmod 2} & \text{if } \lfloor r/2 \rfloor = i \\ d_r & \text{else} \end{cases} .
\end{aligned}
$$

We now write the well-known Viterbi algorithm as operator breakdown rule:

Rule 8 (Pease Viterbi algorithm)

$$
\begin{aligned}
\mathrm{Vit}_{m,n,N}^{e,f,x} \rightarrow & \ \pi^{(x,d,c) \rightarrow (x,d)} \\
& \circ \left(\prod_{i=0}^{n-1} \left((\mathrm{I}^{2^{m-1} \rightarrow 2^{m-1}} \times \mathrm{I}^{2^{m-1} \rightarrow 2^{m-1}} \times \mathrm{I}^{1 \rightarrow 1}) \otimes_j \mathrm{V}_{i,j}^{e,f} \right) \circ \left(\mathrm{L}_{2^{m-1}}^{2^m} \times \mathrm{I}^{2^m n \rightarrow 2^m n} \times \mathrm{I}^{nN \rightarrow nN} \right) \right) \\
& \circ \left(\mathrm{C}_x \times \mathrm{C}_{\vec{0}} \times \mathrm{I}^{nN \rightarrow nN} \right)
\end{aligned} \tag{26}
$$

$$
\begin{aligned}
\mathrm{A}^{k\times m\to n}\otimes \mathrm{I}^{1\times r\to r} &\to \mathrm{L}^{rn}_{n}\circ\big(\mathrm{I}^{1\times r\to r}\otimes \mathrm{A}^{k\times m\to n}\big)\circ\big(\mathrm{I}^{k\to k}\times \mathrm{L}^{rm}_{r}\big) && (29)\\
\mathrm{I}^{r\times s\to t}\otimes\big(\mathrm{A}^{n\to n}\circ \mathrm{B}^{k\times m\to n}\big) &\to \big(\mathrm{I}^{t\to t}\otimes \mathrm{A}^{n\to n}\big)\circ\big(\mathrm{I}^{r\times s\to t}\otimes \mathrm{B}^{k\times m\to n}\big) && (30)\\
\mathrm{I}^{r\times s\to t}\otimes\big(\mathrm{A}^{k\times m\to n}\circ \mathrm{B}^{k\times m\to k\times m}\big) &\to \big(\mathrm{I}^{r\times s\to t}\otimes \mathrm{A}^{k\times m\to n}\big)\circ\big((\mathrm{I}^{r\to r}\times \mathrm{I}^{s\to s})\otimes \mathrm{B}^{k\times m\to k\times m}\big) && (31)\\
\mathrm{I}^{r\times s\to rs}\otimes \mathrm{I}^{t\times u\to tu} &\to \mathrm{I}^{rt\times su\to rstu} && (32)\\
\big(\mathrm{I}^{km\to km}\otimes \mathrm{A}^{r\to n}\big)\circ\big(\mathrm{I}^{k\to k}\otimes \mathrm{B}^{s\to mn}\big) &\to \mathrm{I}^{k\to k}\otimes\big((\mathrm{I}^{m\to m}\otimes \mathrm{A}^{r\to n})\circ \mathrm{B}^{s\to mn}\big) && (33)\\
\big(\mathrm{I}^{k\to k}\otimes \mathrm{L}^{mn}_{n}\big)\circ \mathrm{L}^{kmn}_{km} &\to \big(\mathrm{L}^{kn}_{k}\otimes \mathrm{I}^{m\to m}\big) && (34)\\
\big(\mathrm{A}^{k\times m}\times \mathrm{B}^{r\times s}\big)\otimes\big(\mathrm{C}^{t\times u}\times \mathrm{D}^{v\times w}\big) &\to \big(\mathrm{A}^{k\times m}\otimes \mathrm{C}^{t\times u}\big)\times\big(\mathrm{B}^{r\times s}\otimes \mathrm{D}^{v\times w}\big) && (35)\\
\big(\mathrm{A}^{k\times m}\times \mathrm{B}^{r\times s}\big)\circ\big(\mathrm{C}^{t\times u}\times \mathrm{D}^{v\times w}\big) &\to \big(\mathrm{A}^{k\times m}\circ \mathrm{C}^{t\times u}\big)\times\big(\mathrm{B}^{r\times s}\circ \mathrm{D}^{v\times w}\big) && (36)\\
\mathrm{I}^{1\to 1}\otimes \mathrm{A} &\to \mathrm{A} && (37)\\
\mathrm{I}^{k\to k}\circ \mathrm{I}^{k\to k} &\to \mathrm{I}^{k\to k} && (38)\\
\mathrm{L}^{kmn}_{n}\circ\big(\mathrm{I}^{k\to k}\otimes \mathrm{L}^{mn}_{m}\big) &\to \mathrm{L}^{kn}_{n}\otimes \mathrm{I}^{m\to m} && (39)
\end{aligned}
$$

Table 1: Operator identities.

2.5 Numerical Integration

The numeric integral of a function

$$f : \mathbb{R} \to \mathbb{R}$$

using the Gauss quadrature formula for n points is obtained by

$$Q_n(f) = \sum_{i=1}^{n} \lambda_{i,n} f(x_{i,n}) \tag{27}$$

with $x_{i,n}$ being the abscissas and λ_{rn} being the Cotes numbers.

We define the *integration operator* as constant operator $\mathrm{Q}_{f,n}$, parameterized by a function f:

$$\mathrm{Q}_{f,n} : \mathbb{R}^0 \to \mathbb{R}, \quad \text{with} \quad f : \mathbb{R} \to \mathbb{R} \tag{28}$$

We write (27) as breakdown rule for (28):

Rule 9 (Gauss integration)

$$\mathrm{Q}_{f,n} \to \Sigma_n \circ \Big(\big((\mathrm{I}^{n\to n}\otimes f)\circ \mathrm{C}_{x_{i,n}}\big)\cdot \mathrm{C}_{\lambda i,n} \Big)$$

2.6 All pairs shortest path

3 Operator manipulation

Generalizing tensor product identities.

Example: manipulating MMM formulas. We now show that formula manipulation generalizes to the operators. We start with

$$\mathrm{MMM}^{N,M,K} \to \big(\mathrm{I}^{N\to N}\otimes \mathrm{L}^{M}_{M/M_B}\big)\circ\big(\mathrm{MMM}^{N,M_B,K}\otimes \mathrm{I}^{1\times M/M_B\to M/M_B}\big) \circ \big(\mathrm{I}^{KN\to KN}\times(\mathrm{I}^{K\to K}\otimes \mathrm{L}^{M}_{M_B})\big) \tag{40}$$

and apply (29):

$$\mathrm{MMM}^{N,M_B,K} \otimes \mathrm{I}^{1\times M/M_B \to M/M_B} \to \\ \mathrm{L}^{MN}_{M_B N} \circ \big(\mathrm{I}^{1\times M/M_B \to M/M_B} \otimes \mathrm{MMM}^{N,M_B,K} \big) \circ \big(\mathrm{I}^{KN\to KN} \times \mathrm{L}^{KM}_{M/M_B} \big). \tag{41}$$

Using (34) we can simplify the first line of (44):

$$\big(\mathrm{I}^{N\to N} \otimes \mathrm{L}^{M}_{M/M_B} \big) \circ \mathrm{L}^{MN}_{M_B N} \to \mathrm{L}^{NM/M_B}_{N} \otimes \mathrm{I}^{M_B \to M_B} \tag{42}$$

Using (36), (38), and (39) we simplify

$$\begin{aligned} &\big(\mathrm{I}^{KN\to KN} \times \mathrm{L}^{KM}_{M/M_B} \big) \circ \big(\mathrm{I}^{KN\to KN} \times (\mathrm{I}^{K\to K} \otimes \mathrm{L}^{M}_{M_B}) \big) \\ &\qquad \to \mathrm{I}^{KN\to KN} \times \big(\mathrm{L}^{KM}_{M/M_B} \circ (\mathrm{I}^{K\to K} \otimes \mathrm{L}^{M}_{M_B}) \big) \\ &\qquad \to \mathrm{I}^{KN\to KN} \times \big(\mathrm{L}^{KM/M_B}_{M/M_B} \otimes \mathrm{I}^{M_B\to M_B} \big). \end{aligned} \tag{43}$$

Substituting (42) and (43) into (44) yields

$$\mathrm{MMM}^{N,M,K} \to \big(\mathrm{L}^{NM/M_B}_{N} \otimes \mathrm{I}^{M_B\to M_B} \big) \\ \circ \big(\mathrm{I}^{1\times M/M_B \to M/M_B} \otimes \mathrm{MMM}^{N,M_B,K} \big) \circ \big(\mathrm{I}^{KN\to KN} \times (\mathrm{L}^{KM/M_B}_{M/M_B} \otimes \mathrm{I}^{M_B\to M_B}) \big) \tag{44}$$

for Rule 5.

Parallelization through formula manipulation.