Synthesis of Multiple-Input Translinear Element Networks

Bradley A. Minch, Paul Hasler, and Chris Diorio

†Laboratory of Analogue Information Processing
School of Electrical Engineering
Cornell University
minch@ee.cornell.edu
http://www.ee.cornell.edu/~minch
A $K$-input \textit{multiple-input translinear element} (MITE) produces an output current that is exponential in a weighted sum of its $K$ input voltages.

\[ I = \lambda I_s \exp \left( \sum_{k=1}^{K} \frac{w_k V_k}{U_T} \right) \]

$I$ is the MITE’s output current
$V_k$ is the MITE’s $k$th input voltage
$w_k$ is a dimensionless positive weight that scales $V_k$ proportionally.
$I_s$ is a pre-exponential scaling current
$\lambda$ is a dimensionless factor that scales $I_s$ proportionally (e.g., a geometric factor)
$U_T$ is the thermal voltage, $\frac{kt}{q}$.

We assume that the voltage inputs draw a negligible amount of current at DC.

We assume that we have the ability to control the values of the input weights proportionally, so we can make accurate weight ratios.

If we have an integral number of inputs, each with the same weight, $w$, then we omit the $w$ associated with each symbol in the schematic for clarity.

The MITE has $K$ transconductances, each of which is \textit{linear} in the output current, $I$:

\[ g_k = \frac{\partial I}{\partial V_k} = \frac{w_k}{U_T} \lambda I_s \exp \left[ \sum_{k=1}^{K} \frac{w_k V_k}{U_T} \right] = \frac{w_k}{U_T} I \]

Using MITEs, we can construct both low-voltage translinear circuits and log-domain filters.
We can implement the weighted voltage summation either with a resistive voltage divider or with a capacitive voltage divider.

- For resistive voltage dividers, the weights are proportional to the coupling conductance.
- For capacitive voltage dividers, the weights are proportional to the coupling capacitances.

We can implement the exponential current-voltage relationship either with a bipolar transistor or with a subthreshold MOS transistor.

For each FGMOS MITE shown, the floating-gate charge provides a nonvolatile weight on the output current that we can use either to compensate for mismatch or to implement adaptive circuits.
Three Basic MITE Circuit Configurations

**Voltage-In, Current-Out**

\[ I_n \propto \exp \left[ \frac{w_{ni} V_i + w_{nk} V_k + \ldots}{U_T} \right] \]

\[ \Rightarrow I_n \propto \exp \left[ \frac{w_{ni} V_i}{U_T} \right] \exp \left[ \frac{w_{nk} V_k}{U_T} \right] \]

**Current-In, Voltage-Out**

\[ I_i \propto \exp \left[ \frac{w_{ii} V_i + \ldots}{U_T} \right] \]

**Voltage-In, Voltage-Out**

\[ I_i \propto \exp \left[ \frac{w_{ii} V_i + w_{ij} V_j + \ldots}{U_T} \right] \]

\[ \Rightarrow V_i = \frac{U_T}{w_{ii}} \log I_i - \ldots \]

\[ \Rightarrow V_i = \frac{U_T}{w_{ii}} \log I_i - \frac{w_{ij}}{w_{ii}} V_j - \ldots \]
**MITE Networks: Low-Voltage Translinear Circuits**

**Product-of-Power-Law Circuits**

\[ I_n \propto \exp \left( \frac{w_{ni} V_i}{U_T} \right) \exp \left( \frac{w_{nk} V_k}{U_T} \right) \]

\[ \Rightarrow I_n \propto \exp \left[ \frac{w_{ni}}{U_T} \left( \frac{U_T}{w_{ii}} \log I_i \right) \right] \exp \left[ \frac{w_{nk}}{U_T} \left( \frac{U_T}{w_{kk}} \log I_k \right) \right] \]

\[ \Rightarrow I_n \propto \frac{U_T}{U_T} \exp \left[ \frac{w_{ni}}{w_{ii}} \log I_i \right] \exp \left[ \frac{w_{nk}}{w_{kk}} \log I_k \right] \]

\[ \Rightarrow I_n \propto I_i^{w_{ni}/w_{ii}} I_k^{w_{nk}/w_{kk}} \]

**Quotient-of-Power-Law Circuits**

\[ I_n \propto \frac{w_{ni} V_i}{U_T} + \ldots \]

\[ \Rightarrow I_n \propto \exp \left[ \frac{w_{ni}}{U_T} \left( \frac{U_T}{w_{ii}} \log I_i - \ldots \frac{U_T}{w_{jj}} \log I_j - \ldots \right) \right] \]

\[ \Rightarrow I_n \propto \exp \left[ \frac{U_T}{U_T} \frac{w_{ni}}{w_{ii}} \log I_i \exp \left[ -\frac{U_T}{U_T} \frac{w_{ni}}{w_{jj}} \log I_j \right] \right] \]

\[ \Rightarrow I_n \propto I_i^{w_{ni}/w_{ii}} I_j^{w_{nj}/w_{jj}} \]

\[ \Rightarrow I_n \propto I_i^{w_{ni}/w_{ii}} I_j^{w_{nj}/w_{jj}} \Rightarrow I_n \propto \frac{I_i^{w_{ni}/w_{ii}}}{I_j^{w_{nj}/w_{jj}}} \]
ABCs of MITE Network Synthesis

1. Acquire a set of Translinear-Loop Equations
   - Begin with a suitable relationship to implement using MITE networks.
   - Represent the variables in terms of the ratio of positive signal currents to a unit current, $I_u$.
   - From the original relationship and the signal representations, derive a set of TL loop equations.

2. Begin the Network
   - Begin with a TL loop equation: $\prod_{n \in \text{"CW"}} I_n^{k_n} = \prod_{n \in \text{"CCW"}} I_n^{k_n}$
   - Pick a current from each set (e.g., $I_i$ from “CW” and $I_j$ from “CCW”), make a new MITE for each, make a new node in the circuit, and couple it into MITE $Q_i$ through $k_j$ unit inputs and into MITE $Q_j$ through $k_i$ unit inputs. If $k_i$ and $k_j$ have a factor in common, they can both be divided by that factor in determining the number of unit inputs.
3. Build the Network

For each additional current (e.g., $I_k$ from “CW”), make a new MITE and connect it to an existing MITE whose current is from the opposite set (e.g., $I_j$ from “CCW”), by making a new node and coupling it into MITE $Q_k$ through $k_j$ unit inputs and into MITE $Q_j$ through $k_k$ unit inputs. If $k_j$ and $k_k$ have a factor in common, they can both be divided by that factor in determining the number of unit inputs.

4. Balance the Network

Suppose that the largest MITE fan-in is $K$. Add a sufficient number of grounded inputs to all MITEs, so they each have a fan-in of $K$. 
5. Bias the Network

Bias the MITE network by diode connecting those MITEs whose currents are inputs.

6. Complete the Network

Complete the MITE network by connecting all of the grounded inputs to the collector of one of the diode-connected MITEs, avoiding the creation of feedback loops.
Synthesis of a Two-Layer MITE Network

2. **Beginning** the network

\[ I_1 I_2 I_3^3 = I_4 I_5 I_6^2 \]

“CW”

3. **Building** the network

\[ I_1 I_2 I_3^3 = I_4 I_5 I_6^2 \]

“CCW”

2:1

1

6

2:3

6
Synthesis of a Two-Layer MITE Network

3. Building the network (con’t.)

\[ I_1I_2I_3^3 = I_4^4I_5^2I_6^2 \]

“CW”

“CCW”

3. Building the network (con’t.)

\[ I_1I_2I_3^3 = I_4^4I_5^2I_6^2 \]

“CW”

“CCW”
Synthesis of a Two-Layer MITE Network

3. Building the network (con’t.)

4. Balancing the network

\[ I_1 I_2 I_3^3 I_4^4 = I_4^4 I_5^2 I_6^2 \]

“CW” “CCW”
5. **Biasing the network**

\[
I_6 = \frac{I_1^2I_2I_3^2}{I_4^2I_5}
\]

6. **Completing the network**

\[
I_6 = \frac{I_1^3I_2^3I_3^2}{I_4^2I_5}
\]
Synthesis of a **Cascade** MITE Network

2. **Beginning** the network

3. **Building** the network (con’t.)

\[ I_1 I_2 I_3 = I_4 I_5 I_6 \]

3. **Building** the network

3. **Building** the network (con’t.)

\[ I_1 I_2 I_3 = I_4 I_5 I_6 \]
Synthesis of a **Cascade MITE Network**

3. **Building the network (con’t.)**

\[ I_2 I_4^2 I_3^4 = I_4^4 I_5^2 I_6^2 \]

4. **Balancing the network**

\[ I_2 I_4^3 I_3^4 = I_4^4 I_5^2 I_6^2 \]

5. ** Biasing the network**

\[ I_6 = \frac{I_2 I_4^2 I_3^2}{I_4 I_5} \]

6. **Completing the network**

\[ I_6 = \frac{I_2 I_4^2 I_3^2}{I_4 I_5} \]
MITE Log-Domain Filters

\[
H(s) = \frac{I_{\text{out}}(s)}{I_{\text{in}}(s)} = \frac{1}{1 + \tau_1 s + \tau_1 \tau_2 s^2} = \frac{1}{1 + \frac{\tau s}{Q} + (\tau s)^2}
\]

\[
\tau_1 = \frac{C U_T}{w L_{\tau_1}}, \quad \tau_2 = \frac{C U_T}{w L_{\tau_2}}, \quad \tau \equiv \sqrt{\tau_1 \tau_2}, \quad Q \equiv \sqrt{\frac{\tau_2}{\tau_1}}
\]