The Crossbar

- The crossbar has horizontal and vertical wires
- Application of a positive (negative) voltage between a sets R of rows and C of columns writes 1s (0s) at the intersection of the rows and columns.
- If 1s (0s) are written, R × C defines an activation (deactivation) matrix.
- A set (reset) operation writes 1s (0s) in a subarray.
- Goal: Given an n × m array W, find a minimal sequence of stores and restores to program W from an all-zeros matrix.

Programming with Stores is NP-hard

ARRAY PROGRAMMING (AP)

Instance: (W,k) where W is an n × m array over {0,1} and k is an integer.

Answer: “Yes” if there exists a set of at most k activation matrices covering W.

- Later we show that AP is NP-hard when only stores are used. (A much harder proof shows that it is also NP-hard when both stores and restores are used.)
- In some circumstances it will be difficult to address individual rows and columns. That is, we will not be able to choose sets arbitrary R and C to form activation or deactivation matrices R × C.
- We introduce h-hot addressing.

h-Hot Addressing

The h-hot addressing scheme on b address wires, h < b, assigns a unique weight-h binary b-tuple a to each integer in the set \( N = \{1, ..., n\} \), \( n \leq \binom{b}{h} \).

Represent a by address h-tuple \((a_1, ..., a_h)\), \( a_j \in A \), \( A = \{1, ..., b\} \), where \( a_j \) is the index of the jth 1 in a.

Let \((a_1^{(i)}, ..., a_h^{(i)})\) be address h-tuple for integer i.

Address wire \( aw_j \) is associated with address j in A. Nanowire \( NW_i \) is associated with integer i in \( N \).

Address (nanowires) wires are either active or inactive (addressed or not addressed).

\( NW_i \) is addressed if for each \( j \in \{a_1^{(i)}, ..., a_h^{(i)}\}, aw_j \) is activated. Note: multiple NWs can be addressed simultaneously by activating \( \geq h \) address wires.

Stores and restores occur at the intersections of addressed nanowires.
**h-Hot Programming**

- **Goal:** Given an $n \times m$ array $W$, find a minimal sequence of stores and restores to program $W$ from an all-zeroes matrix using $h$-hot addressing.

**$h$-hot ARRAY PROGRAMMING (AP)**

*Instance:* $(W, k)$ where $W$ is an $n \times m$ array over $\{0,1\}$ and $k$ is an integer.

*Answer:* “Yes” if there exists a set of at most $k$ activation matrices covering $W$.

---

**Simple Array Programming Problems**

- **Diagonal** array: $[a_{i,j}]$, $a_{i,j} = 1$ only when $i = j$.

- **Half-full** array: $[a_{i,j}]$, $a_{i,j} = 1$ only when $i \leq j$.

- **$\rho$-lower full** array: $[a_{i,j}]$, $a_{i,j} = 1$ only when $i \leq j - \rho$, where $-(n-1) \leq \rho \leq (n-1)$.

- **$\rho$-upper full** array: $[a_{i,j}]$, $a_{i,j} = 1$ only when $i \geq j - \rho$, where $-(n-1) \leq \rho \leq (n-1)$.

- **Bandwidth-$\beta$** array: $[a_{i,j}]$, $a_{i,j} = 1$ only when $|i - j| \leq \beta$.

- **$s$-sparse** array: $[a_{i,j}]$, $a_{i,j} = 1$ for at most $s$ elements in each row and column.

---

**Most AP Problems are Hard**

**Theorem** For $0 \leq \varepsilon \leq (1-2/n)$, at least a fraction of $1 - 2^{-\varepsilon n^2}$ of the $2^n n \times n$ matrices require at least $(1 - \varepsilon)n^2/2$ steps when using either only stores or both stores and restores.

**Proof** The first step is a store. Subsequent steps can be stores or restores. There are $2^{2n}$ possible types of such steps and two choices at each step except the first. Thus, there are at most $2^{kn} + k - 1$ types of program that contain $k$ steps. If $2^{kn} + k - 1 \leq 2(1 - \varepsilon)n^2$, that is, $2nk + k - 1 \leq (1 - \varepsilon)n^2$, then a fraction of at least $1 - 2^{-\varepsilon n^2}$ of the arrays requires $k + 1$ or more steps. If $\varepsilon \in (1 - 2/n)$, $k \leq (1 - \varepsilon)n/2$. Since the number of types of step is smaller when only stores are used, the bounds also apply in this case.

---

**Lower Bounds for Array Programming**

**Theorem** When only stores are allowed, the diagonal and half-full arrays require at least $n$ steps. The $\rho$-lower (upper) full arrays and bandwidth-$\beta$ arrays require at least $n - |\rho|$ and $n - \beta$ steps, respectively. $s$-sparse arrays require at least $n/s$ steps.

**Proof** For the diagonal array, no activation matrix covers more than one element without also covering off-diagonal 0s. Thus, at least $n$ elements are necessary. For half-full arrays, the same argument applies. The $\rho$-lower (upper) full arrays and bandwidth-$\beta$ arrays have diagonals of size $n - |\rho|$ and $n - \beta$, respectively.

In the $s$-sparse case, if we maximize the number of 1s in an activation matrix, we reduce the number of such matrices. Since an activation matrix can have at most $s$ rows and columns, the result follows.
**AP is NP-hard**

**ARRAY PROGRAMMING (AP)**

*Instance*: \((W,k)\) where \(W\) is an \(n \times m\) array over \(\{0,1\}\) and \(k\) is an integer.

*Answer*: “Yes” if there exists a set of at most \(k\) activation matrices covering \(W\).

- We show that \(AP\) is \(NP\)-complete under stores.

---

**Continuous Reductions**

- Maximization (minimization) problems are defined by pairs \((I,b)\) in which instance \(I\) has value denoted by \(v(I)\).

- A pair \((I, b)\) is a “Yes” instance to a minimization (maximization) problem if \(v(I) \leq b\) (or \(v(I) \geq b\)).

**Definition** A reduction \(\rho\) between two maximization (or minimization) problems \((I,b)\) and \((I', b')\) is \((a,c)\)-bounded if \(v(I') \geq v(I)/a - c\) (or \(v(I') \leq v(I)/a - c\)).

A reduction \(\rho\) is asymptotically continuous if there is an inverse \(\gamma\) reduction such that \(\rho\) is \((a,c)\)-bounded and \(\gamma\) is \((d,e)\)-bounded where \(a,d > 0\) and \(c,e \geq 0\).

- If a reduction \(\rho\) is asymptotically continuous and \((I',b') = \rho(I,b)\), the values of \(v(I)\) and \(v(I')\) are linearly related.

---

**b-Set Basis**

**b-SET BASIS**

*Instances*: Triples \((U,S,k)\) where \(U = \{e_1, ..., e_m\}\) and \(S = \{S_1, ..., S_n\}\) a collection of subsets of \(U\), \(|S_j| \leq b\).

*Answer*: “Yes” if \((U,S,k)\) has a basis of size \(l \leq k\), that is, a collection \(B = \{B_1, ..., B_l\}\) of subsets of \(U\) such that each \(S_j\) is the union of sets in the basis.

- SET BASIS is \(b\)-SET BASIS when \(b = m\).

**Theorem** [Stockmeyer 1975] \(b\)-SET BASIS and SET BASIS are \(NP\)-complete.

---

**Covering by Complete Bipartite Graphs**

\(G = (V,E)\) is **bipartite** if \(V = X \cup Y\) and \(E \subset X \times Y\). A **biclique** is a complete bipartite graph, i.e. \(E = X \times Y\).

**COVERING BY COMPLETE BIPARTITE GRAPHS (CCB)**

*Instances*: \((G,k)\) where \(G = ((X \cup Y), E), E \subset X \times Y\).

*Answer*: “Yes” if there is a collection of at most \(k\) bicliques that covers all the edges of \(G\).

**Theorem** [Simon 1990] CCB is \(NP\)-complete.
Reduction Between AP and SET BASIS

**Theorem** An asymptotically continuous reduction exists between AP and SET BASIS.

**Proof** Given an instance \((U, S, k)\) of SET BASIS, build array \(D = [d_{i,j}]\) (instance of AP) in which \(d_{i,j} = 1\) if \(e_j\) is in \(S_i\) and 0 otherwise. If \((U, S, k)\) is a “Yes” instance, then there is a basis \(B\) of size at most \(k\) that covers \(S\). Let \(B_q\) be a basis element and let \(B_q \subseteq S_r\) for \(r\) in \(R\). The elements covered by \(B_q\) are elements in rows \(R\) of \(D\) and in columns associated with the elements in \(B_q\). Thus, these elements of \(D\) define an activation matrix. Thus, if \((U, S, k)\) is a “Yes” instance of SET BASIS, then \((D, k)\) is a “Yes” instance of AP.

Similarly, an instance \((A, k)\) of AP can be reduced to an instance \((U, S, k)\) of SET BASIS by equating rows of \(A\) with sets in the collection \(S\). An activation matrix of \(A\) corresponds to a basis set. Thus, if \((A, k)\) is a “Yes” instance, then \((U, S, k)\) is a “Yes” instance.

Reduction Between AP and CCB

**Theorem** An asymptotically continuous reduction exists between AP and CCB.

**Proof** Given an instance \((W, k)\) of AP, create an instance \(G = ((X \cup Y), X \times Y)\) in which \(X\) corresponds to the rows of \(W\) and \(Y\) corresponds to its columns. If \(w_{i,j}\) in \(W\) is 1, create an edge between the \(i\)th element of \(X\) and the \(j\)th element of \(Y\). An activation matrix now corresponds to a biclique in \(G\). It follows that \((W, k)\) is a “Yes” instance of AP (it is covered by \(k\) activation matrices) if and only if \(G\) is a “Yes” instance of CCB (its edges are covered by \(k\) bicliques).

**Theorem** AP is NP-complete.