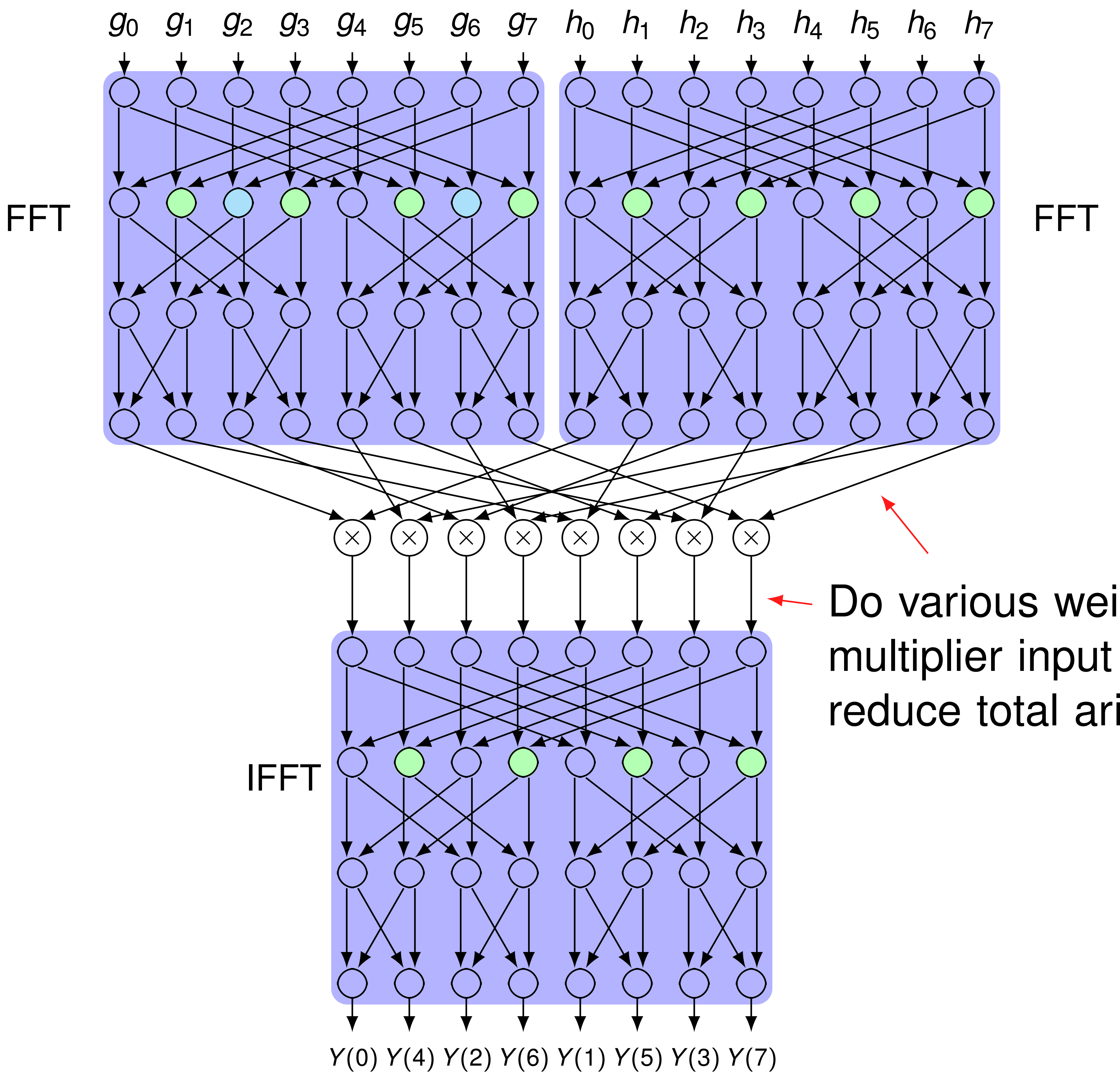
Fast Convolution Flowgraph with Problem Statement



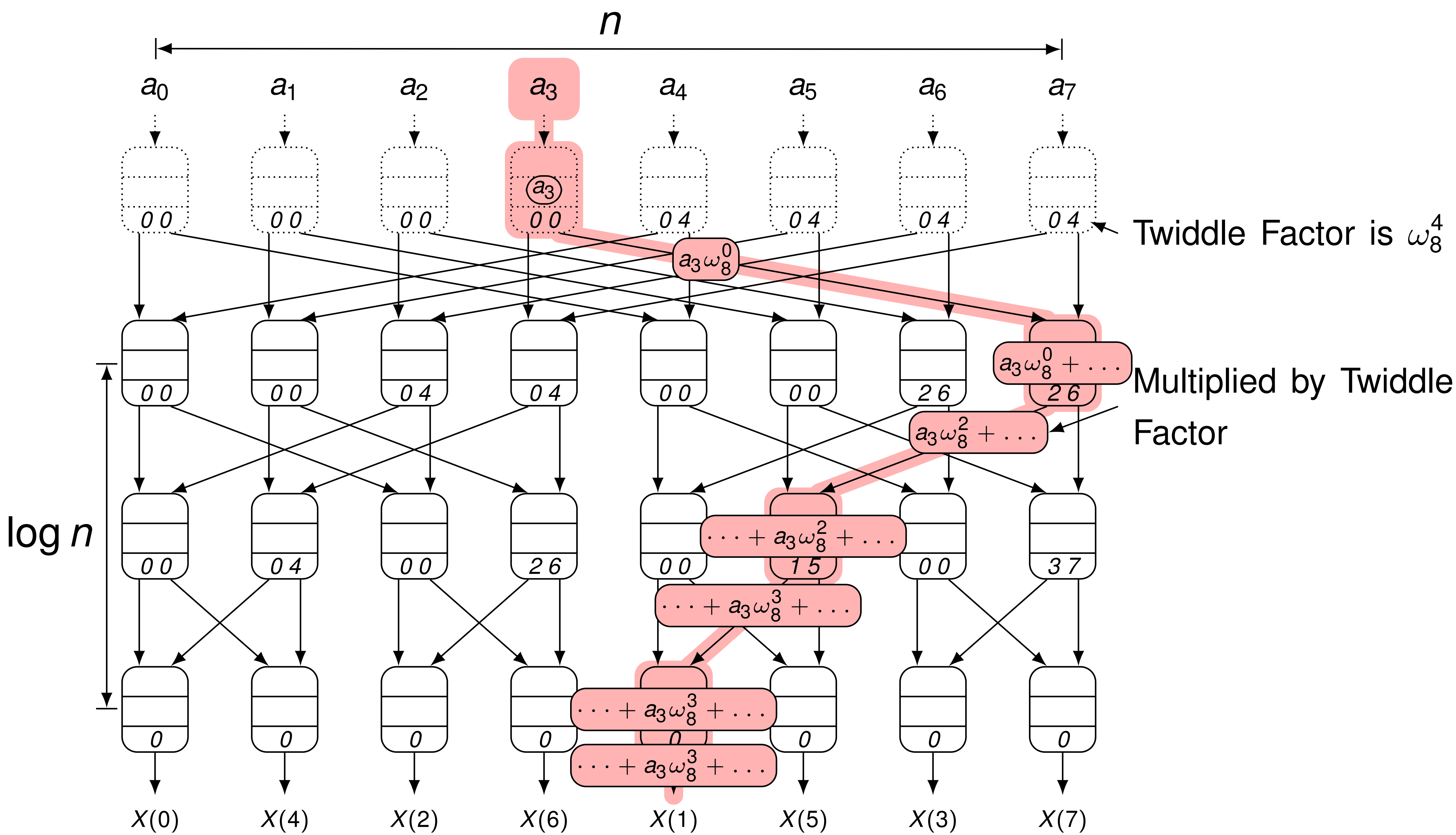
Do various weights of $e^{-i\frac{2\pi}{n}}$ on the multiplier input and output operands reduce total arithmetic complexity?

Contributions

- Novel techniques to enumerate and search families of FFT-based algorithms
- Proved minimum FLOP count of fast convolution algorithms when all FFT and IFFT twiddle factors are n^{th} roots of unity and flowgraph structure is fixed
- Found and posted new fast convolution algorithms with minimum FLOP count given formulation constraints

Solution Strategy at 20,000 Feet

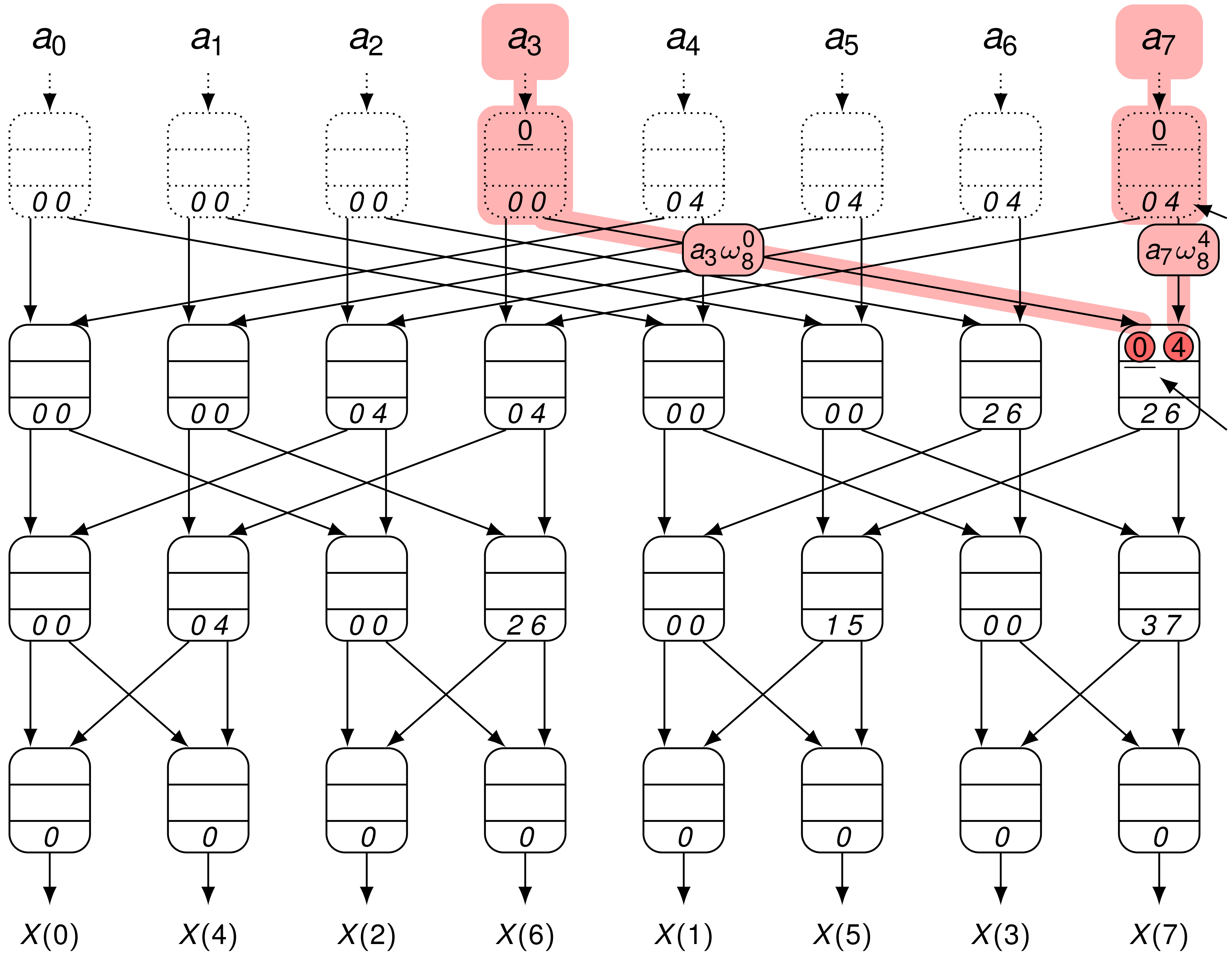
- Cast Fast Convolution's FFTs and IFFT as Satisfiability problems to find minimum FLOP count
  - Extension of techniques developed in our earlier work
  - Each FFT and IFFT is searched independently
  - State-of-the-art SAT solvers are employed
- Relax SAT formulation to allow FFTs that produce arbitrary weighted result operands
- Relax SAT formulation to allow IFFT that accepts arbitrary weighted input operands
- Cancel costs of additional weights in fast convolution's multiplication



$$X(1) = a_0\omega_8^0 + a_1\omega_8^1 + a_2\omega_8^2 + \boldsymbol{a_3\omega_8^3} + a_4\omega_8^4 + a_5\omega_8^5 + a_6\omega_8^6 + a_7\omega_8^7$$

$X(1)$ verified to be as defined by FFT with correct weight on a_3

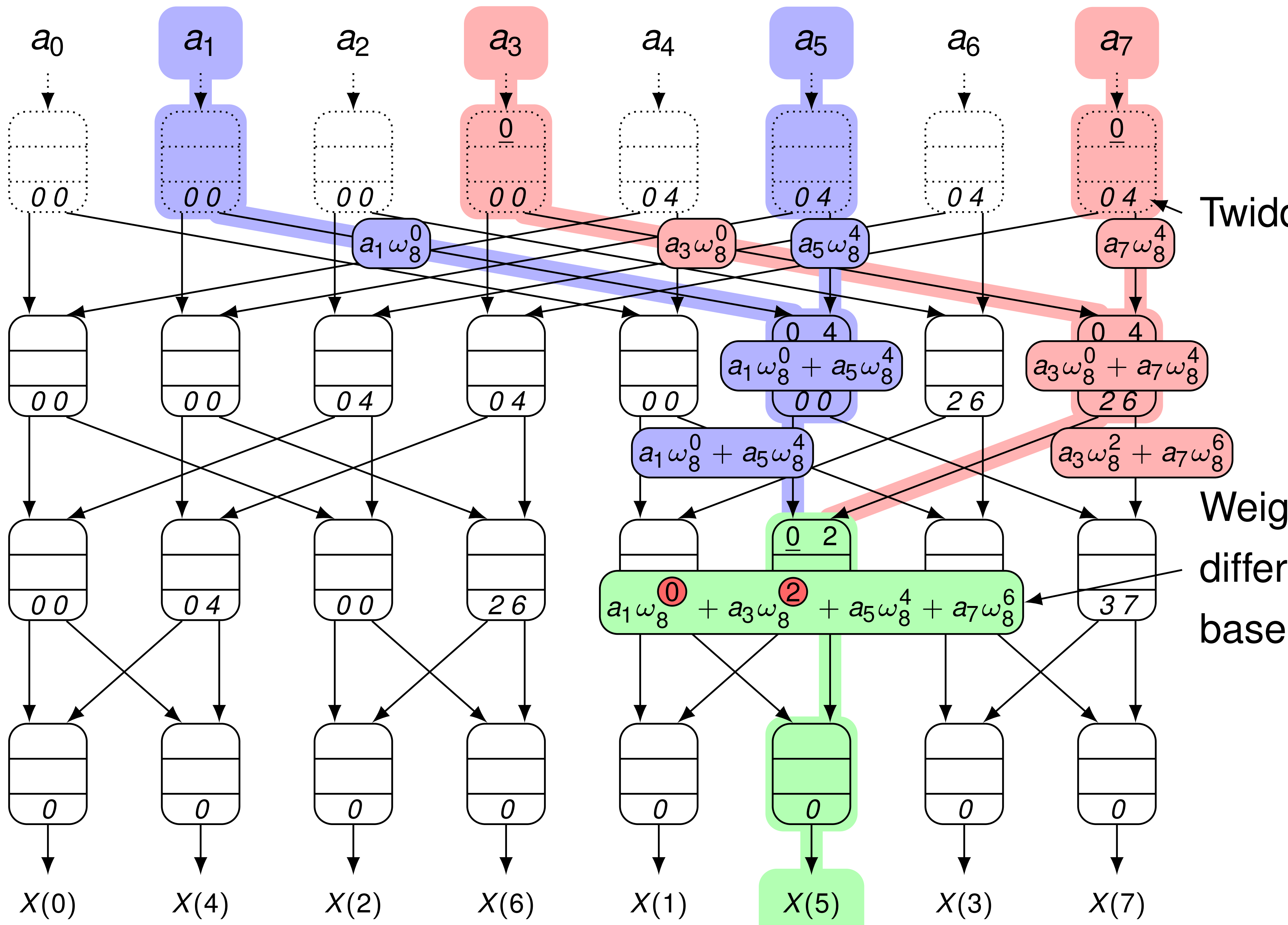
Annotate FFT Flowgraph with *Weight on Base*



Twiddle Factor is ω_8^4

Record weight on a_i with lowest i for both incoming sums as weight on base

Weight Stride is an Invariant of the Canonical FFT Flowgraph



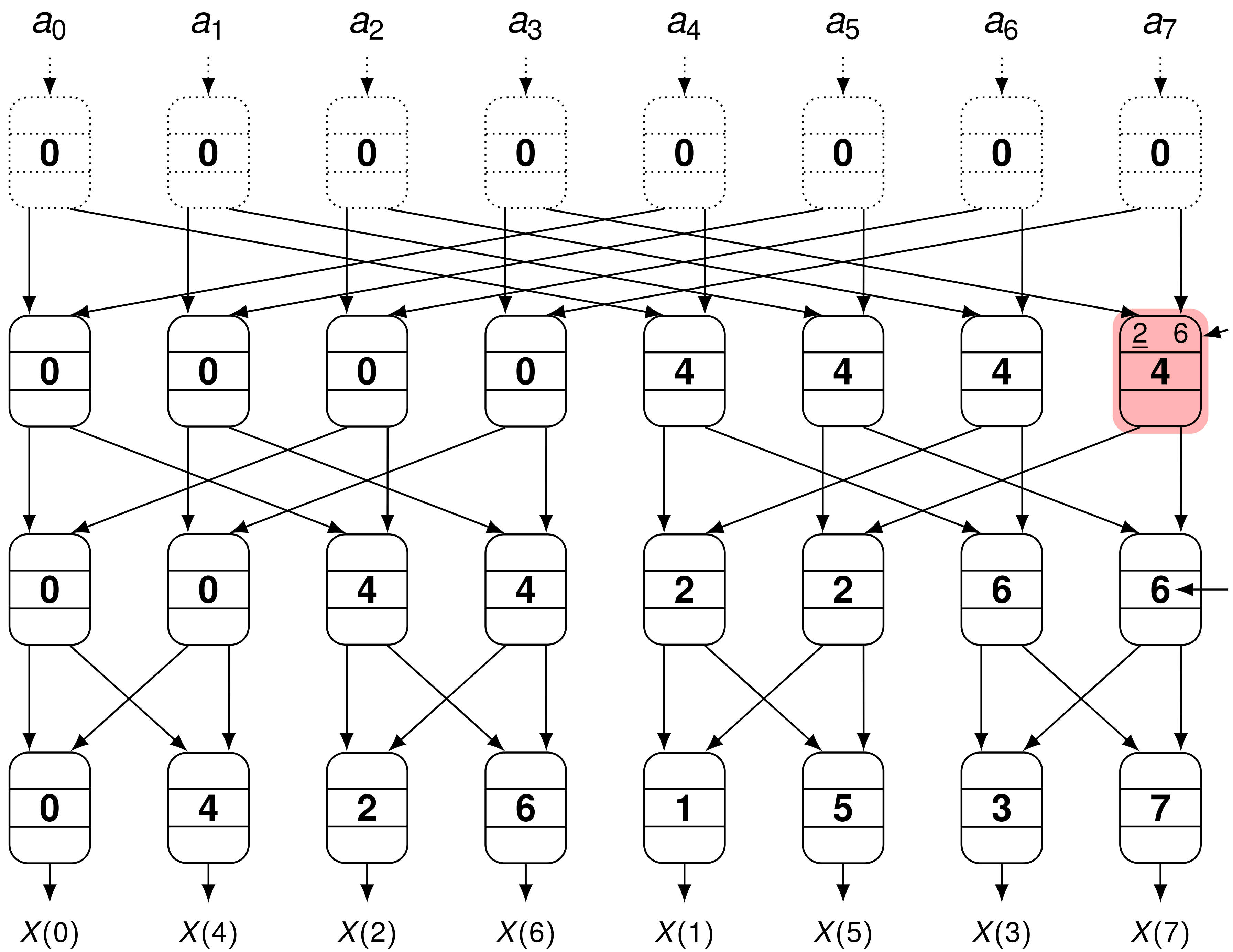
Twiddle Factor is ω_8^4

Weight Stride is difference of weight on bases, $W_{stride} = 2 - 0$

$$X(5) = a_0\omega_8^{0} + a_1\omega_8^5 + a_2\omega_8^{2} + a_3\omega_8^7 + a_4\omega_8^{4} + a_5\omega_8^1 + a_6\omega_8^{6} + a_7\omega_8^3$$

Weight stride still equal to 2 for contributed terms

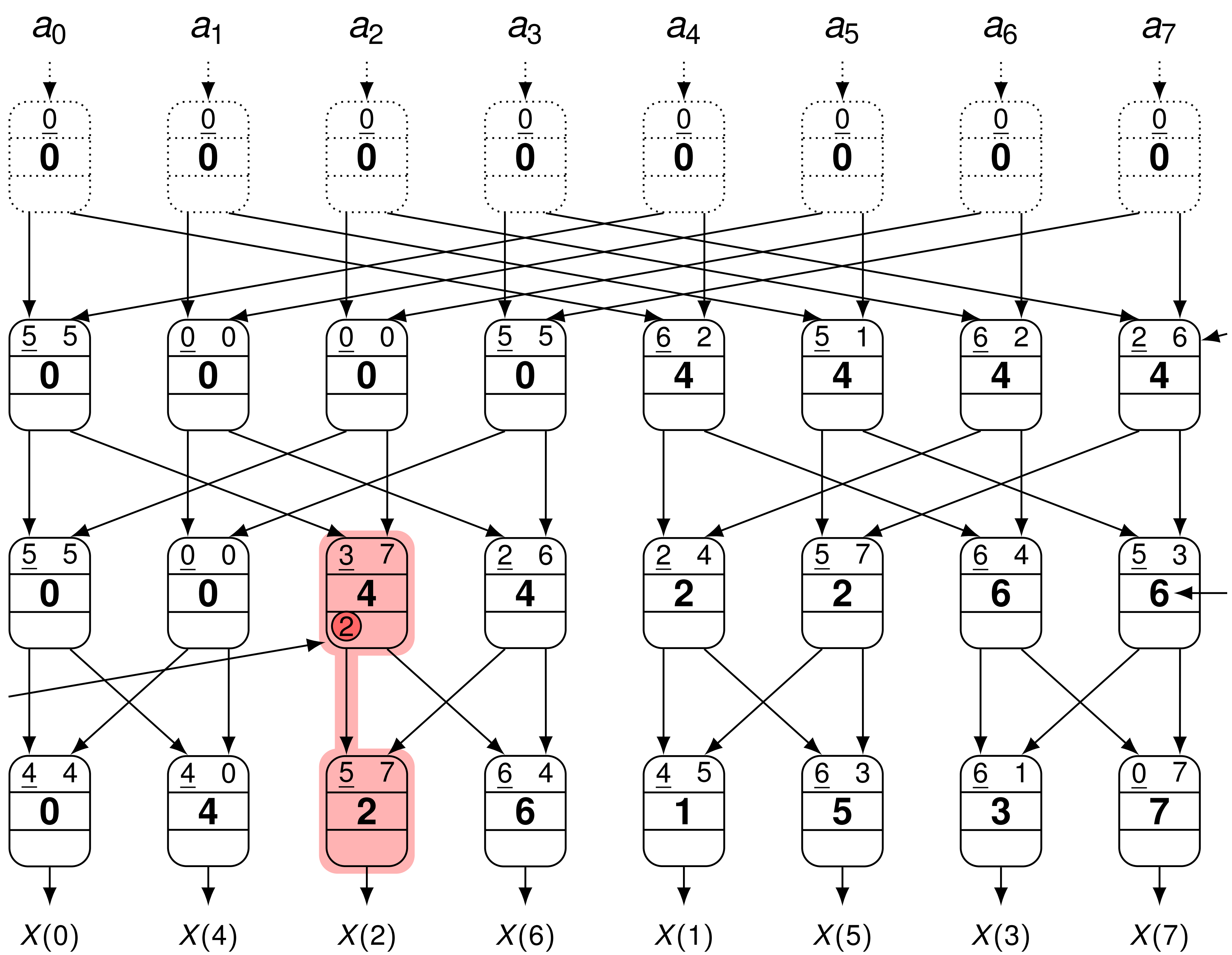
FFT Flowgraph with Weight Strides and Two Random Weights on Base



Random choice for
Left $W_{base} = 2$,
Right W_{base} is
Left $W_{base} + W_{stride} = 6$

Invariant Weight Stride,
$W_{stride} = 6$

Twiddle Factors from Weights on Base

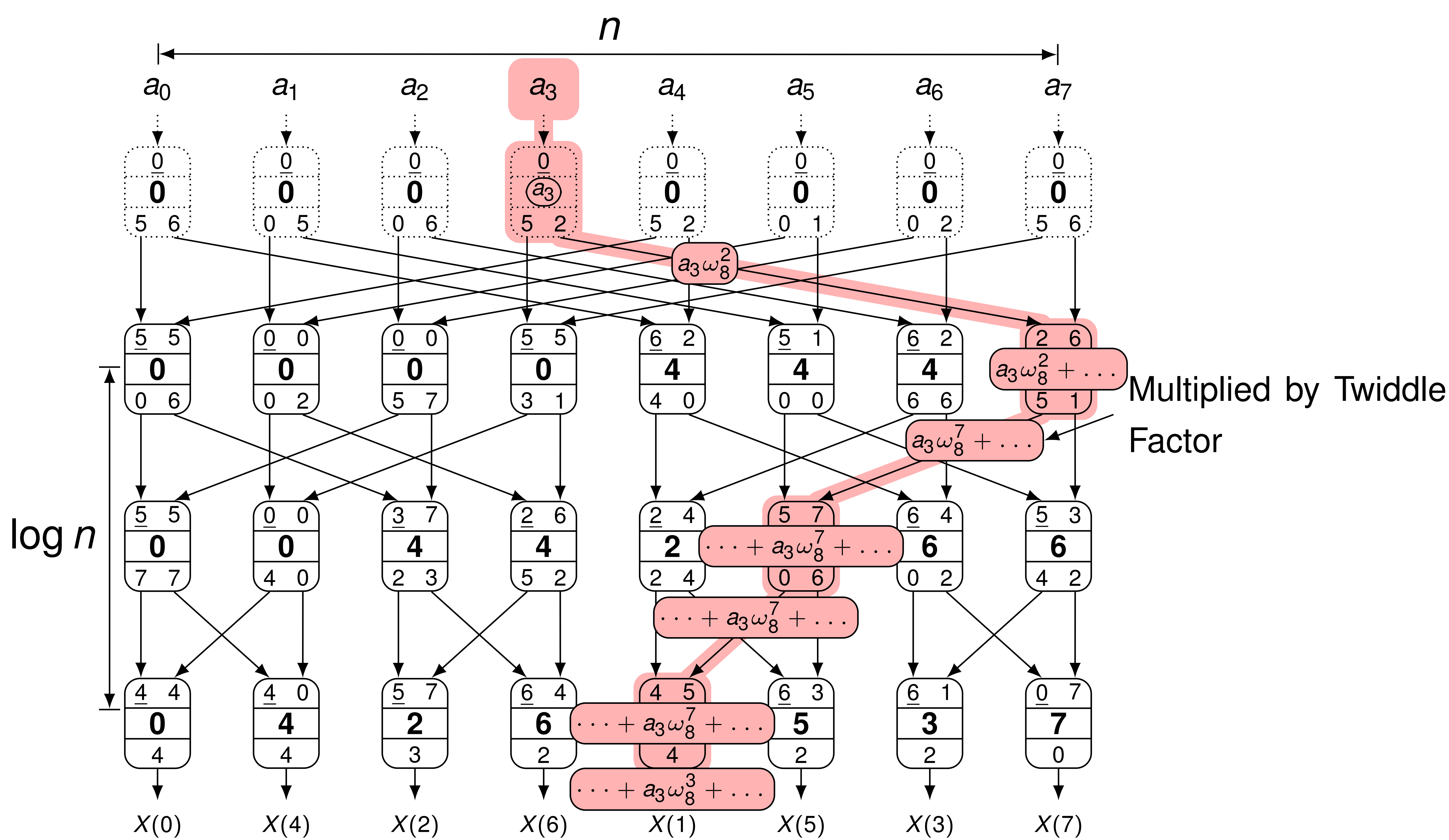


Random choice for *Left* $W_{base} = 2$, *Right* W_{base} is *Left* $W_{base} + W_{stride} = 6$

Invariant Weight Stride, $W_{stride} = 6$

New twiddle factor is difference of W_{base}, $5 - 3 = 2$

Random Member FFT with One Path Verified



$$X(1) = a_0\omega_8^0 + a_1\omega_8^1 + a_2\omega_8^2 + \boldsymbol{a_3\omega_8^3} + a_4\omega_8^4 + a_5\omega_8^5 + a_6\omega_8^6 + a_7\omega_8^7$$

$X(1)$ verified to be as defined by FFT with correct weight on a_3

How to Generate a Random Member Weighted FFT Algorithm

Input: Size-n flowgraph with labeled invariants
Output: Size-n flowgraph with twiddle factors assigned

foreach $nd \in flowgraph$ **do**
    if *nd not in top row* **then**
        $nd.W_{base} \leftarrow rand()\ (\text{mod } n)$
        $nd.rW_{base} \leftarrow nd.W_{base} + nd.W_{stride}\ (\text{ mod } n)$
    else
        $nd.W_{base} \leftarrow 0$
foreach $nd \in flowgraph$ **do**
    if *nd not in top row* **then**
        $nd.lp.tfp \leftarrow nd.W_{base} - nd.lp.W_{base}\ (\text{mod } n)$
        $nd.rp.tfp \leftarrow nd.rW_{base} - nd.rp.W_{base}\ (\text{mod } n)$
    if *nd in bottom row* **then**
        $nd.tfp \leftarrow rand()\ (\text{mod } n) - nd.W_{base}\ (\text{mod } n)$

Searching a Family of FFT Algorithms

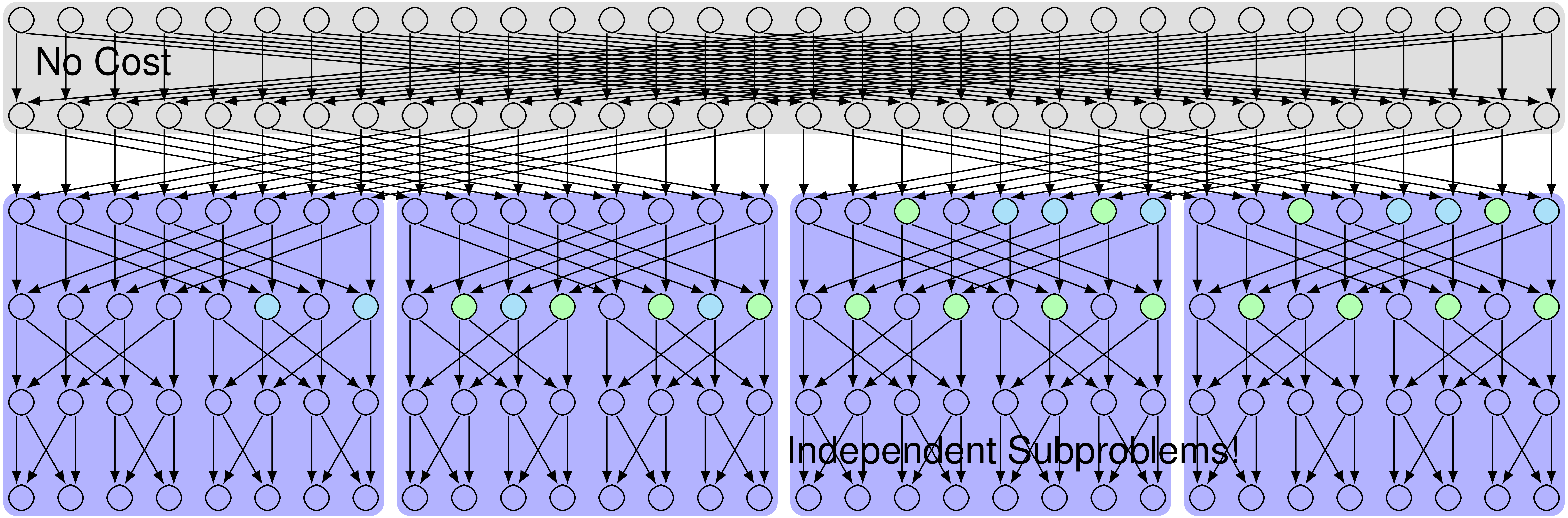
- All family members are not equally desirable
  - Some require fewer FLOPs
  - Others have “better” twiddle factor sets
  - Need a way to search and find desirable members: SAT!
- How many family members are there?
  - $2^{n \log_2 n \log_2 n}$
  - For a 256-point FFT: 2^{16384}
  - Only 1 in 2^{18432} chance of guessing correct twiddle factors
  - Estimated atoms in the universe is 2^{264}
  - Fastest supercomputer performs 2^{144} FLOPS

A SAT Formulation

- Directly cast "Random Member Algorithm" as SAT
- Must also calculate FLOP count directly in SAT model
  - This Psuedo-Boolean constraint adds complexity
- Naïve formulation only works for small size-n
  - Size-32 455 FLOP search UNSAT in 30 seconds
  - Time-out of 24 hours reached for size-64 1159 FLOP search
- Techniques required to solve larger more interesting cases
  - Exclude cost symmetries
  - Share twiddle factors
  - Partition
  - Exclude local symmetries

FFT Partitioning for Fast Convolution

Initial weights known



Terminal weights unknown for fast convolution

Terminal weights known for FFT and hence smaller partitions possible

Symmetrical IFFT partitioning when terminal weights known and initial weights unknown

Results

- Brute-force proof of lowest possible FLOP count within search constraints
  - FFT and IFFT twiddle factors are n^{th} roots of unity
  - FFT and IFFT flowgraph structure is same as generated by common power-of-two FFTs
  - SAT-based search limits problem size-n to $n = 128$
- Witness algorithms are posted

Table: FFT or IFFT FLOP Counts

FFT or IFFT	$n = 64$	$n = 128$
Split-Radix	1160	2824
Unweighted Tangent $\lvert\omega_n^*\rvert = *$	1152	2792
Weighted Tangent $\lvert\omega_n^*\rvert = *$	1120	2720
Weighted SAT Search $\lvert\omega_n^*\rvert = 1$	1136	2744

Conclusions and Future Work

- Conclusions
  - Extended work on brute-force search of FFT algorithms to fast convolution
    - Proved minimum FLOP count of fast convolution algorithms when all FFT and IFFT twiddle factors are n^{th} roots of unity and flowgraph structure is fixed
    - Found and posted new fast convolution algorithms with minimum FLOP count given formulation constraints
- Future Work
  - Describe new FFT algorithms with abstract algebra
    - Enable better reasoning
    - Break problem size constraints imposed by SAT
  - Expand solution space
    - Alter underlying graph structure
    - Allow twiddle factors that are not n^{th} roots of unity
  - Expand search objectives
    - Not just minimize FLOP count
    - Minimize cost and complexity of implementation
    - Maximize overall performance