ACA2021 - Final project
Team 5: SBM for HFT

Simulated Bifurcation Machine for High-Frequency Trading

2021/6/29 游子緒、馬健凱、陳昱行

GitHub: https://github.com/e841018/SBM4HFT

Mathematical formulation:
From exchange rates to Ising model

Derived by ourselves

Exchange rates -> QUBO -> Ising model

Inputs and outputs of SBM

- Input: `log_c`
  - A matrix of log(exchange rate)
- Output: `x`
  - A matrix of activation
  - Should contain only 1 loop

    x[0][2] (0->2): 0.04218

    x[2][1] (2->1): -0.16195

    x[1][0] (1->0): 0.12034

    sum(`log_c` * `x`): 0.00057
- Objective function to minimize:

  - sum(`log_c` * `x`) + constraints(`x`)

```
float log_c[n][n] = {
    { 0.00000, -0.12038,  0.04218},
    { 0.12034,  0.00000,  0.16208},
    {-0.04212, -0.16195,  0.00000},
};
```

```
bool x[n][n] = {
    { 0, 0, 1},
    { 1, 0, 0},
    { 0, 1, 0},
};
```

Rewrite as QUBO

Constraint 1:
For each node i,
(outflux - influx) ≈ 0

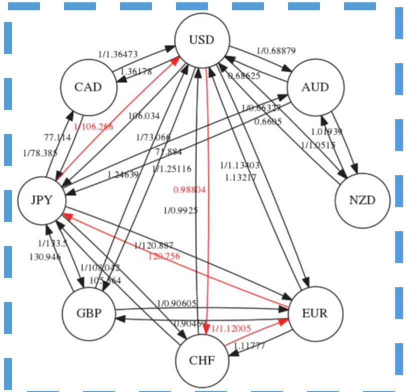
Constraint 2:
For each node i,
outflux ≈ 0 or outflux ≈ 1

$$x = \underset{x}{\operatorname{argmin}} \left\{ \sum_{i=0}^{n-1}\sum_{j=0}^{n-1}(-log_2(c_{ij}))x_{ij} + M_1 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} x_{ij} - \sum_{j=0}^{n-1} x_{ji}\right)^2 + M_2 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} x_{ij}\right)\left(\sum_{j=0}^{n-1} x_{ij} - 1\right) \right\}$$

Expand the 1st term:

$$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}(-log_2(c_{ij}))x_{ij}$$

$$= \sum_{i=0}^{n-1}\sum_{j=0}^{n-1}(-log_2(c_{ij}))x_{ij}x_{ij}$$



Expand the 2nd term:

$$M_1 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} x_{ij} - \sum_{j=0}^{n-1} x_{ji}\right)^2$$

$$= M_1 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} (x_{ij} - x_{ji})\right)^2$$

$$= M_1 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} (x_{ij} - x_{ji})\right)\left(\sum_{k=0}^{n-1} (x_{ik} - x_{ki})\right)$$

$$= M_1 \sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\sum_{k=0}^{n-1} (x_{ij} - x_{ji})(x_{ik} - x_{ki})$$

$$= M_1 \sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\sum_{k=0}^{n-1} (x_{ij}x_{ik} - x_{ij}x_{ki} - x_{ji}x_{ik} + x_{ji}x_{ki})$$

Expand the 3rd term:

$$M_2 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} x_{ij}\right)\left(\sum_{j=0}^{n-1} x_{ij} - 1\right)$$

$$= M_2 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} x_{ij}\right)\left(\sum_{k=0}^{n-1} x_{ik} - 1\right)$$

$$= M_2 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1}\sum_{k=0}^{n-1} x_{ij}x_{ik} - \sum_{j=0}^{n-1} x_{ij}\right)$$

$$= M_2 \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1}\sum_{k=0}^{n-1} x_{ij}x_{ik} - \sum_{j=0}^{n-1} x_{ij}x_{ij}\right)$$

Change subscripts and rewrite QUBO as Ising model

- Change subscripts: $x_{ab} \rightarrow x_{a\times n+b}$
  Now each expanded term is of the form $Q_{ij}x_ix_j$
- QUBO: matrix ***Q***

$$Q \in \mathbb{R}^{N\times N}, Q = Q^T, x_i \in \{0,1\}$$

$$f_Q(Q,x) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} Q_{ij}x_ix_j$$

$$x_{opt} = \underset{x}{\operatorname{argmin}}\, f_Q(Q,x)$$

- Ising model: matrix ***J*** and vector ***h***

$$J \in \mathbb{R}^{N\times N}, J = J^T, h \in \mathbb{R}^N, s_i \in \{-1,1\}$$

$$H(J,h,s) = -\frac{1}{2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} J_{ij}s_is_j + \sum_{i=0}^{N-1} h_is_i$$

$$s_{opt} = \underset{s}{\operatorname{argmin}}\, H(J,h,s)$$

$$\text{Let } s_i = 2x_i - 1$$

$$\text{Let } J_{ij} = \frac{-Q_{ij}}{2}, h_i = \left(\sum_{j=0}^{N-1}\frac{Q_{ij}}{4} + \sum_{j=0}^{N-1}\frac{Q_{ji}}{4}\right), C = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\frac{Q_{ij}}{4}$$

$$f_Q(Q,x) = -\frac{1}{2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} J_{ij}s_is_j + \sum_{i=0}^{N-1} h_is_i + C = H(J,h,s) + C$$

Example: An Ising problem of 9 spins

M1 = M2 = 0.2

J =	(0, 0)	(0, 1)	(0, 2)	(1, 0)	(1, 1)	(1, 2)	(2, 0)	(2, 1)	(2, 2)
(0, 0)	0	-0.1000	-0.1000	0	0	0	0	0	0
(0, 1)	-0.1000	-0.2602	-0.2000	0.2000	0	0.1000	0.1000	-0.1000	0
(0, 2)	-0.1000	-0.2000	-0.1789	0.1000	0	-0.1000	0.2000	0.1000	0
(1, 0)	0	0.2000	0.1000	-0.1398	-0.1000	-0.2000	-0.1000	0.1000	0
(1, 1)	0	0	0	-0.1000	0	-0.1000	0	0	0
(1, 2)	0	0.1000	-0.1000	-0.2000	-0.1000	-0.1190	0.1000	0.2000	0
(2, 0)	0	0.1000	0.2000	-0.1000	0	0.1000	-0.2211	-0.2000	-0.1000
(2, 1)	0	-0.1000	0.1000	0.1000	0	0.2000	-0.2000	-0.2810	-0.1000
(2, 2)	0	0	0	0	0	0	-0.1000	-0.1000	0

	(0, 0)	(0, 1)	(0, 2)	(1, 0)	(1, 1)	(1, 2)	(2, 0)	(2, 1)	(2, 2)
h =	0.2000	0.2602	0.1789	0.1398	0.2000	0.1190	0.2211	0.2810	0.2000

Simulated Bifurcation Machine:
An algorithm that solves QUBO by simulating Ising models

We encountered some problems and haven’t fully solved them...

Parameter tuning

- SBM involves a lot of parameters
  - The objective function involves **2 constants**:
    - M1, M2
  - SBM involves **5 constants** and **2 functions of time**:
    - N_step, N_substep, alpha0, beta0, Delta_t
    - alpha(t), eta(t)
- We can’t find a set of parameters that fits the example 9-spin problem…

Hypothesis 1: Inappropriate objective function

We tried another objective function from [1]

```
J =          ( 0, 0) ( 0, 1) ( 0, 2) ( 1, 0) ( 1, 1) ( 1, 2) ( 2, 0) ( 2, 1) ( 2, 2)
           +------------------------------------------------------------------------
    ( 0, 0)| -0.5000 -0.2500 -0.2500 -0.2500       0       0 -0.2500       0       0
    ( 0, 1)| -0.2500 -25.0753 -0.7500  0.5000 -0.2500  0.5000  0.5000 -0.7500       0
    ( 0, 2)| -0.2500 -0.7500  7.4355  0.5000       0 -0.7500  0.5000  0.5000 -0.2500
    ( 1, 0)| -0.2500  0.5000  0.5000 23.0673 -0.2500 -0.7500 -0.7500  0.5000       0
    ( 1, 1)|       0 -0.2500       0 -0.2500 -0.5000 -0.2500       0 -0.2500       0
    ( 1, 2)|       0  0.5000 -0.7500 -0.7500 -0.2500 31.4154  0.5000  0.5000 -0.2500
    ( 2, 0)| -0.2500  0.5000  0.5000 -0.7500       0  0.5000 -9.4245 -0.7500 -0.2500
    ( 2, 1)|       0 -0.7500  0.5000  0.5000 -0.2500  0.5000 -0.7500 -33.3891 -0.2500
    ( 2, 2)|       0       0 -0.2500       0       0 -0.2500 -0.2500 -0.2500 -0.5000

             ( 0, 0) ( 0, 1) ( 0, 2) ( 1, 0) ( 1, 1) ( 1, 2) ( 2, 0) ( 2, 1) ( 2, 2)
h =           1.5000 25.5753 -6.9355 -22.5673  1.5000 -30.9154  9.9245 33.8891  1.5000
```

The result was still not good enough…

```
maximize {  obj0   +    obj1   +    obj2   +    obj3 }
          65.0055  -   2.0000  -   0.0000  -   0.0000  = 63.0055
```

```
activation =
 0 0 1
 1 0 0
 0 0 0
```

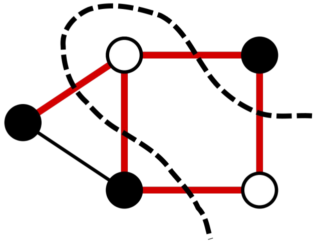
[1] “A Currency Arbitrage Machine Based on the Simulated Bifurcation Algorithm for Ultrafast Detection of Optimal Opportunity.” ISCAS2020

Hypothesis 2: The 9-spin problem is too small

K2000: *Weighted Max Cut problem for 2000-vertex complete graph*

Simulated Bifurcation Machine

- Number of steps = 4000
- Number of substeps = 2
- Delta_t = 0.7
- x_init_max = p_init_max = 0.001



https://upload.wikimedia.org/wikipedia/commons/thumb/c/cf/Max-cut.svg/1200px-Max-cut.svg.png

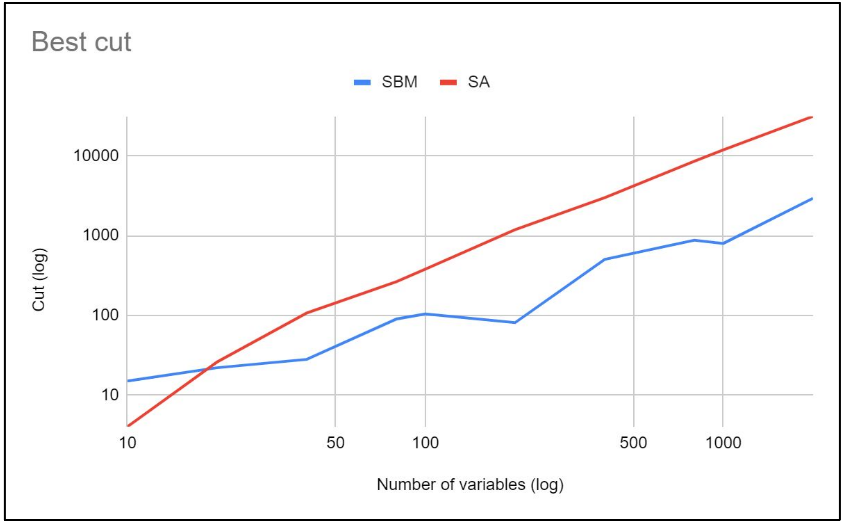
Simulated Annealing https://github.com/hariby/SA-complete-graph/releases

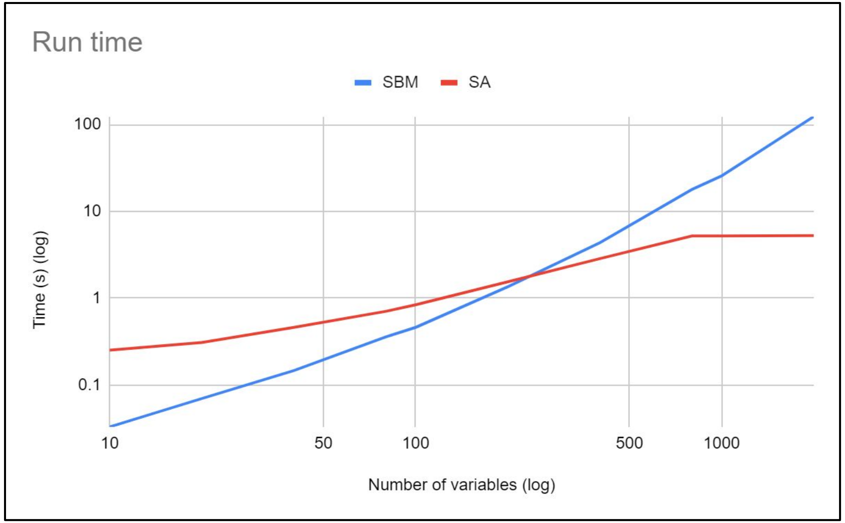
- thread = 1
- sync steps = 4000

Experimental result Experiments

SBM v.s. SA in different graph sizes

Best cut & **Run time** in different graph sizes (10, 20, …, 1000, 2000)
Our SBM can handle K2000 case, but suffers from poor solution results.





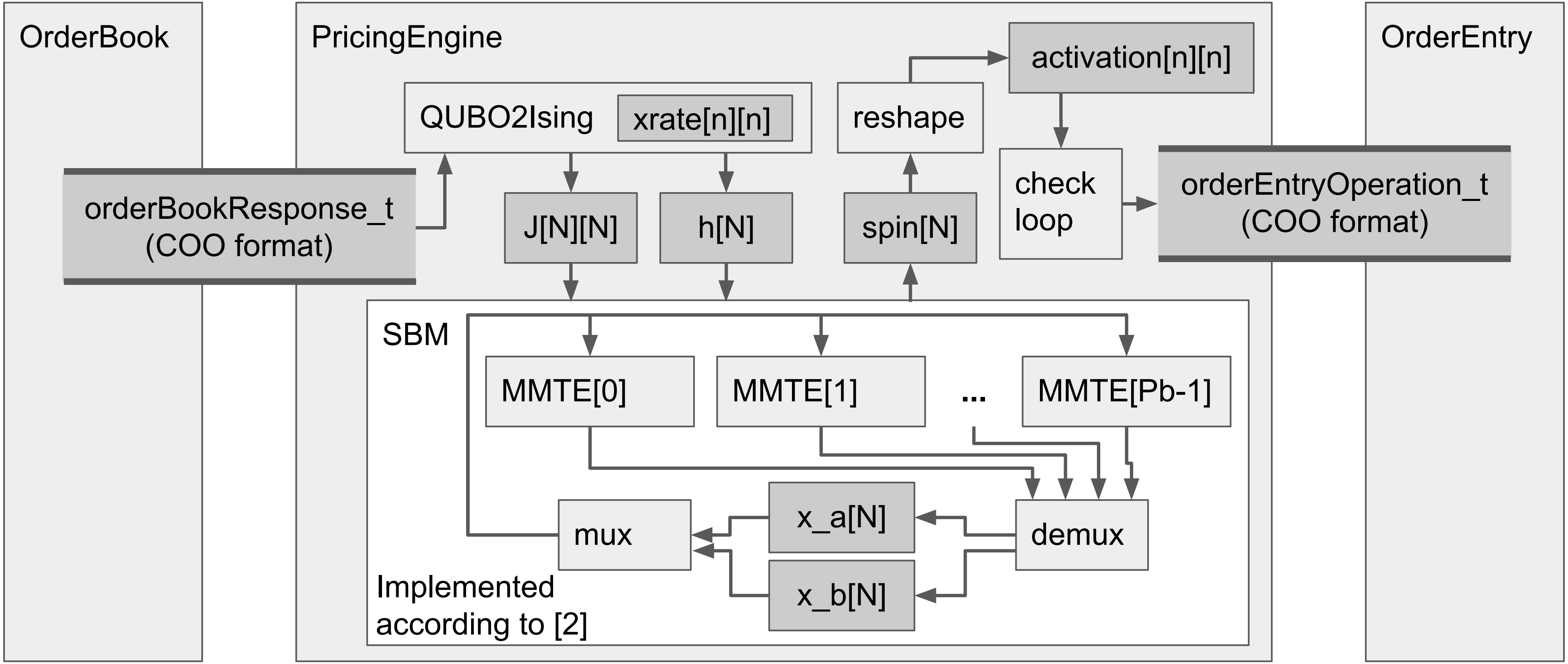
FPGA implementation of SBM

We did not find any existing open-source SBM library,
so we implemented one.

Function blocks

N = n * n

: Packed and streamed

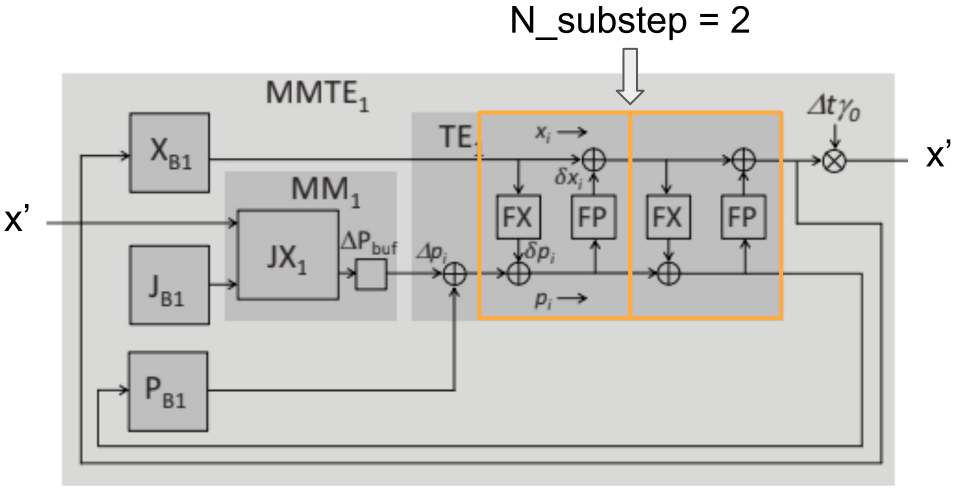


[2] “FPGA-based Simulated Bifurcation Machine,” FPL2019

Inside MMTE

- MM (Matrix-vector multiplication)
- TE (Time evolution)
  - $FX(x_i, h_i) = \delta t * (-(\alpha 0 - \alpha + \beta 0 * x_i * x_i) * x_i - \eta * h_i$
  - $FP(p_i) = \delta t * p_i$

// Matrix-Vector Multiplication (MM)
for $i = 1$ **to** N **do**
    $\Delta p_i \leftarrow \mathrm{JX}(\mathbf{J}_i, \mathbf{x})$
end
// Time Evolution (TE)
for $i = 1$ **to** N **do**
    $p_i \leftarrow p_i + \Delta p_i$
    // Self Evolution
    for $m = 1$ **to** M **do**
        $p_i \leftarrow p_i + \delta p_i = p_i + \mathrm{FX}(x_i, h_i)$
        $x_i \leftarrow x_i + \delta x_i = x_i + \mathrm{FP}(p_i)$
    end
end



csynth report

- The host program for Vitis passed SW emulation but failed HW emulation
  - No csynth report generated for U50
  - **(Update: we just passed HW emulation!)**
- The testbench for Vivado HLS passed cosim
  - U50 can not be found in the device list of Vivado HLS 2019.2
  - The closest device is U280 (with HBM)

Display Name	Part	Family
Alveo U280 Data Center Accelerator Card	xcu280-fsvh2892-2L-e	virtexuplusHBM
Alveo U250 Data Center Accelerator Card	xcu250-figd2104-2L-e	virtexuplus
Alveo U200 Data Center Accelerator Card	xcu200-fsgd2104-2-e	virtexuplus

csynth report with device=U280, 16 currencies (256 spins)

We only added pragmas on interface …

256 spins: C synth spent 35 mins

9 spins: Cosim result correct

Module	Latency (absolute) min	Latency (absolute) max	Pipeline Type
SBM	**348 ms**	**348 ms**	**none**
QUBO2Ising	5.791 ms	5.791 ms	none
total	354 ms	354 ms	none

== Utilization Estimates

* Summary:

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	317	-
FIFO	-	-	-	-	-
Instance	126	40	7072	11333	0
Memory	118	-	2	4	0
Multiplexer	-	-	-	768	-
Register	-	-	685	-	-
Total	244	$\boxed{40}$	7759	12422	$\boxed{0}$

csynth report with device=U50, 3 currencies (9 spins)

We only added pragmas on interface ...

```
================================================================
== Utilization Estimates
================================================================
* Summary:
```

Name	BRAM_18K	DSP	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	2781	-
FIFO	-	-	-	-	-
Instance	10	63	12343	12029	-
Memory	0	-	98	134	-
Multiplexer	-	-	-	2713	-
Register	-	-	3890	416	-
Total	10	63	16331	18073	0
Available SLR	1344	2976	871680	435840	320
Utilization SLR (%)	~0	2	1	4	0
Available	2688	5952	1743360	871680	640
Utilization (%)	~0	1	~0	2	0

Host program

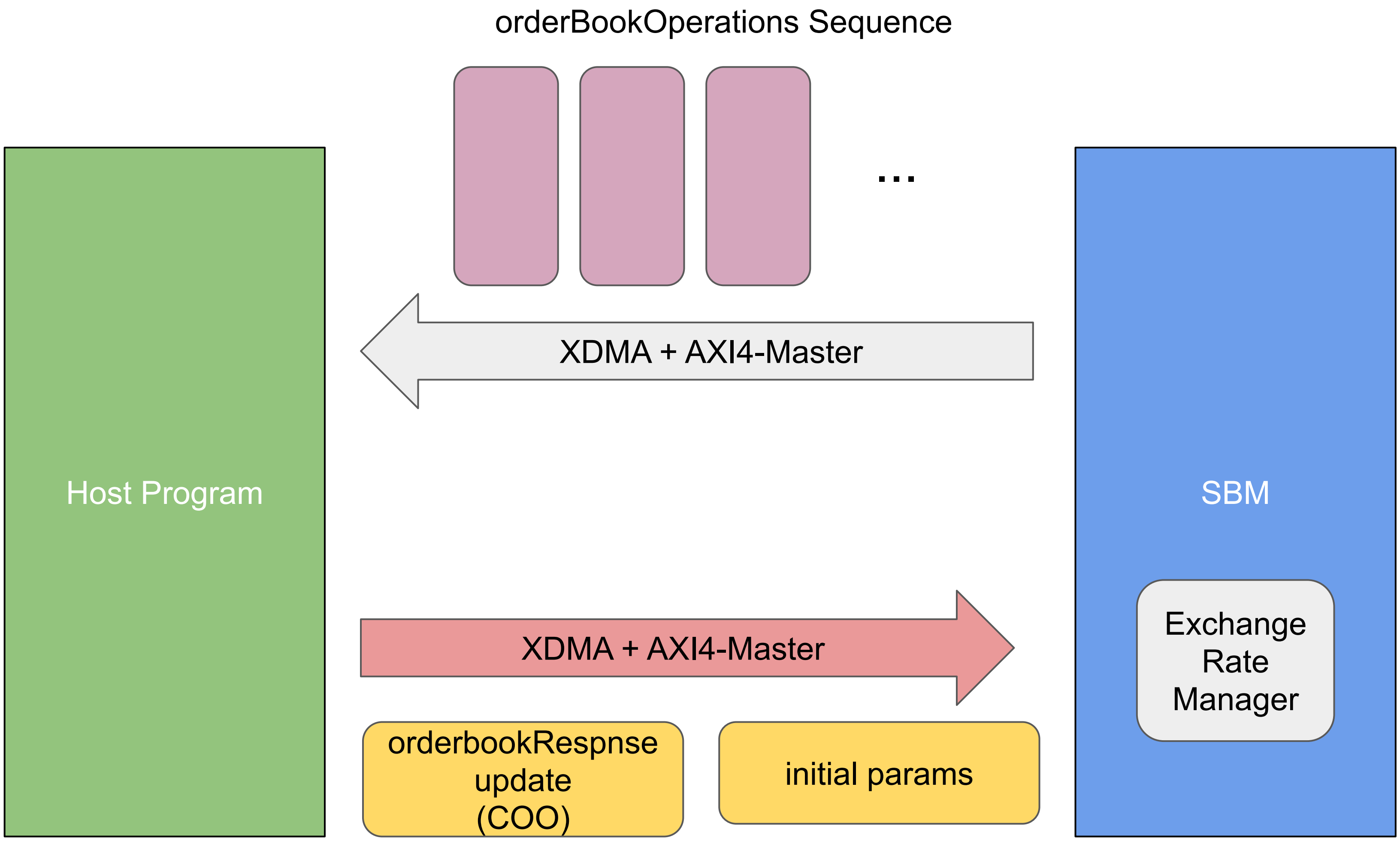
For testing kernel on U50

OpenCL/C++

- Use Coordinate format to pass log(exchange rate) data
- Use Coordinate format to receive a sequence of book order.
- Some problems encoutered:
    - AXI4-Lite don’t support custom data type directly
    - Use xdma + AXI4 Master instead, but would take 4096 bytes to aligned the memeory, though the actual data type size is only 12 bytes.

Interface reference from MM POC

- MM POC is designed for **single-asset** arbitration,
while the currency exchange application is **multi-asset**
- An asset in currency exchange is an exchange rate
  - For example, 4 currencies will need (16 - 4 = 12) assets (symbolIndex)
- Each incoming orderBookResponse_t is regarded as an exchange rate update
  - **0 or multiple orderEntryOperation_t will be generated, instead of 1**
- Changes are **not** required in OrderBook and OrderEntry
  - The interfaces to other IPs are unchanged


orderBookOperations Sequence
...
XDMA + AXI4-Master
Host Program
SBM
XDMA + AXI4-Master
Exchange Rate Manager
orderbookRespnse update (COO)
initial params

Contributions

- 游子緒
  - Mathematical derivations
  - Backbone of kernel code
- 陳昱行
  - MM PoC interface
  - Host program
- 馬健凱
  - Kernel optimization and refactoring
  - Test the kernel with K2000 and compare with SA