Supplementary File of the TPDS Manuscript: High-performance Computing Implementations of Agent-based Economic Models for Realizing 1:1 Scale Simulations of Large Economies

Amit Gill, Madegedara Lalith, Sebastian Poledna, Muneo Hori, Kohei Fujita, and Tsuyoshi Ichimura

Abstract—This document provides the details of the algorithms to improve the completeness of the TPDS article – "High-performance Computing Implementations of Agent-based Economic Models for Realizing 1:1 Scale Simulations of Large Economies".

✦

- *Amit Gill is with the Department of Civil Engineering, The University of Tokyo, 7-3-1 Hongo, Bunkyo City, Tokyo, 113-8654, Japan.*
  E-mail: gill@eri.u-tokyo.ac.jp
- *Madegedara Lalith, Kohei Fujita, and Tusyoshi Ichimura are with the Department of Civil Engineering and the Earthquake Research Institute, The University of Tokyo, 7-3-1 Hongo, Bunkyo City, Tokyo, 113-8654, Japan.*
  E-mail: lalith, fujita, ichimura@eri.u-tokyo.ac.jp
- *Sebastian Poledna is with the International Institute for Applied Systems Analysis, A-2361, Laxenburg, Austria.*
  E-mail: poledna@iiasa.ac.at
- *Muneo Hori is with the Research Institute for Value-Added-Information Generation, Japan Agency for Marine-Earth Science and Technology, 3173-25, Showa-machi, Kanazawa-ku, Yokohama-city, Kanagawa, 236-0001, Japan.*
  E-mail: horimune@jamstec.go.jp

Algorithm 1: Pseudocode of the ***naive_update_distr*** version of `Update_cumulative_probability_distribution()`

Input : Vector, $\mathbf{o}^{r,s}$, of the active-sellers,
       Index, j, of the sold-out seller, and
       Vector, $\mathbf{P}^{r,s}$, of the cumulative probability distribution

Output: N_a and updated vectors $\mathbf{o}^{r,s}$ and $\mathbf{P}^{r,s}$

1 $\mathbf{o}^{r,s}.erase(\mathbf{o}^{r,s}.begin() + j)$;

2 $N_a \leftarrow \mathbf{o}^{r,s}.size()$;

3 Re-evaluate the probability $p_i^{r,s}$ for the remaining elements of $\mathbf{o}^{r,s}$

4 Generate a cumulative probability distribution vector $\mathbf{P}^{r,s} = \{P_i^{r,s} | P_i^{r,s} = \sum_{j=1}^{i} p_j^{r,s}\}$;

Algorithm 2: Pseudocode of the ***improved_update_distr*** version of `Update_cumulative_probability_distribution()`

Input : Vector, $\mathbf{o}^{r,s}$, of the active-sellers,
       Index, j, of the sold-out seller,
       Number of active-sellers, N_a, and
       Vector, $\mathbf{P}^{r,s}$, of the cumulative probability distribution

Output: N_a and updated vector $\mathbf{P}^{r,s}$

1 $N_a = N_a - 1$;

   for $i = j; i < \mathbf{o}^{r,s}.size(); i+ = 1$ **do**

2 $\lfloor$ $P_i^{r,s} = P_i^{r,s} - p_j^{r,s}$;

Algorithm 3: Pseudocode of the ***advanced_update_distr*** version of `Update_cumulative_probability_distribution()`

Input : Vector, $\mathbf{o}^{r,s}$, of the active-sellers,
       Index, j, of the sold-out seller,
       Number of active-sellers, N_a, and
       Vector, $\mathbf{P}^{r,s}$, of the cumulative probability distribution

Output: N_a and updated vector $\mathbf{P}^{r,s}$

1 $N_a = N_a - 1$;

   if $j < {}^{1}/_{2} \times \mathbf{o}^{r,s}.size()$ **then**

     for $i = 0; i < j; i+ = 1$ **do**

2 $\lfloor$ $P_i^{r,s} = P_i^{r,s} + p_j^{r,s}$;

   else

     for $i = j; i < \mathbf{o}^{r,s}.size(); i+ = 1$ **do**

3 $\lfloor$ $P_i^{r,s} = P_i^{r,s} - p_j^{r,s}$;

Algorithm 4: Pseudo code of the *first version* for T periods of simulation of the ABEM. Note that `Iallgatherw()` and `Iscatterw()` are derived from `MPI_Ialltoallw()`

---

for $t = 0$; $t < T$; $t = t + \Delta t$ **do**

    if $t \neg 0$ **then**

1   Increase population and open new bank accounts;

2   Central government, at the master rank (*rank*-0), decides its consumption budget and posts `MPI_Ibcast()` to broadcast the budget amount to the local governments;

    if $t \neg 0$ **then**

3   Firms decide their production quantity, price, and labor requirements;

4   Foreign sellers decide their quantity and price;

5   *SalesOutlets* clear their previous period's data.

6   Firms decide the quantities of investment goods and input goods to buy in the current period;

7   Firms apply for loans to cover their financing gap;

8   Firms revise their budget for investment goods based on the loan received;

9   Firms hire or fire workers based on their labor requirements;

10   Households decide their consumption and investment budgets;

11   Finalize the non-blocking communication initialized in event **2**;

12   *SalesOutlets* post `MPI_Iallreduce()` to collect total demand of each sector;

13   Firms produce;

14   Share industry-wise total production and price of goods among all ranks using `MPI_Allreduce()`, and each rank calculates industry specific economic coefficients;

15   Central bank calculates the economy wide indices and posts `MPI_Ibcast()` to share the indices;

16   Post `Iallgatherw()` to share the products of each firm with their *SalesOutlets* in each rank;

17   Firms pay wages to their workers;

18   Finalize `MPI_Ibcast()` posted in the event **15**;

19   Finalize `MPI_Iallreduce()` posted in the event **12**;

20   Finalize `Iallgatherw()` posted in the event **16**, and each sales-outlet calculates and keeps its portion of products to sell;

21   All agents buy. Post `MPI_Ireduce()` to collect the quantity bought by the local governments at the main rank. Our strategies eliminate any MPI communication in this `Buy()` function which consumes about 90% of the execution time);

22   Post `MPI_Ireduce()` to sum the sales records of all the *SalesOutlets* to the master rank;

23   Local governments distribute social benefits to each household in their respective ranks;

24   Finalize `MPI_Ireduce()` posted in event **22**;

25   Post `Iscatterw()` to convey the sales records of all the *SalesOutlets* of all the firms, which are already gathered by the master rank by events **22** and 24, to the *SalesOutlets* located in the firms' home ranks;

26   Firms pay loan installments to the local banks in their respective ranks;

27   Finalize `Iscatterw()` posted in the event **25**;

28   *SalesOutlets* deliver the sales record, received in the event **27**, to their parent firms;

29   Do the end of tth period's accounting(see Algorithm 5 for details);

30   Finalize `MPI_Ireduce()` posted in the event **7** of the event **29**;

31   Agents reset their variables to prepare for the next period;

32   Finalize `MPI_Ibcast()` posted in the event **19** of the event **29**;

Algorithm 5: End of tth period's accounting for the ***first version*** of the code

---

1 Firms do period-end accounting;

2 Firms pay dividends to their investors;

3 Firms pay taxes to the local governments;

4 Firms settle their account with the local banks, and deposit extra money;

5 Foreign buyers pay taxes to the local governments;

6 Post `MPI_Ireduce()` to reduce imports, exports, and GVA at the master rank for the central bank's accounting;

7 Post `MPI_Ireduce()` to collect industry summary (i.e., GDP, production, sales, demands, employment, etc.) at the master rank;

8 Households pay taxes to the local governments;

9 Households settle their account with the local banks, and deposit extra money;

10 Finalize `MPI_Ireduce()` posted in the event **6**;

11 Central bank does period-end accounting;

12 Local banks do period-end accounting;

13 Post `MPI_Ireduce()` for collecting financial reports from the local banks at the master rank;

14 Variables of the firms and their *SalesOutlets* are reset to prepare for the next period;

15 Finalize `MPI_Ireduce()` posted in the event **13**;

16 Main bank does accounting;

17 Bank's investor receives his dividends and updates his/her bank account;

18 Main bank updates its account at the central bank;

19 Post `MPI_Ibcast()` to share the main bank's equity, loan etc. among all the local banks;

20 Main bank pays tax to the local government;

21 Post `MPI_Ireduce()` to sum tax collected and social benefits paid by the local governments at the master rank;

22 Variables of the households are reset to prepare for the next period;

23 Finalize `MPI_Ireduce()` posted in the event **21** of Algorithm **4**;

24 Finalize `MPI_Ireduce()` posted in the event **21**;

25 Central government does accounting and update its account at the central bank;

Algorithm 6: Pseudo code of the *final version* for T periods of simulation of the ABEM. `Iallgatherw()` and `Iscatterw()` are derived from `MPI_Ialltoallw()`

---

for $t = 0$; $t < T$; $t = t + \Delta t$ **do**

  if $t \neq 0$ **then**

1　　　Central bank, at the master rank (*rank*-0), estimates growth and inflation rates and posts `MPI_Ibcast()` to broadcast the estimated rates to the other ranks;

2　Central government,at the master rank, decides its consumption budget and posts `MPI_Ibcast()` to broadcast the budget amount to the local governments;

3　Firms decide their production quantity, price, and labor requirements;

4　Firms inform their labor requirements to the recruitment agencies located in their respective ranks. Recruitment agencies fire local workers and post `MPI_Isend()/Irecv()` for informing non-local fired workers to other recruitment agencies;

5　Foreign sellers decide their quantity and price;

6　Recruitment agencies finalize `MPI_Isend()/Irecv()` posted in the event **4** and hire workers of their respective ranks;

7　Master recruitment agency, at the master rank, collects number of extra labor/jobs from all recruitment agencies using `MPI_Gather()`, assigns labor from labor surplus agency to the labor deficient agency, and then inform the cross-ranks labor assignments to each recruitment agency using `MPI_Scatter()`;

8　Recruitment agencies post `MPI_Isend()/Irecv()` to initiate exchanging labor according to the cross-ranks assignments made by the master recruitment agency. It should be noted that only the numbers (representative of man-hours) are exchanged not the worker class objects;

9　Firms decide the quantities of investment goods and input goods to buy in the current period;

10　Big firms (big buyers) post `Iallgatherw()` to inform their non-local *ProcurementDivisions* about the quantities to buy;

11　Firms apply for loans to cover their financing gap;

12　Recruitment agencies finalize the `MPI_Isend()/Irecv()` posted in the event **8**;

13　Recruitment agencies assign the number of workers received in the event **12** to the available jobs;

14　Recruitment agencies post `MPI_Isend/Irecv()` to inform the job details of the remote workers to the respective remote recrutement agencies;

15　Finalize the non-blocking communication posted in the event **2**;

16　Recruitment agencies inform the firms about the workers assigned to them at the end of current period's job market ;

17　Firms produce;

18　Share industry-wise total production and price of goods among all ranks using `MPI_Allreduce()`, and each rank calculates industry specific economic coefficients;

19　Post `Iallgatherw()` to share the products of each firm with their *SalesOutlets* in each rank;

20　Firms pay the wages to the recruitment agencies, and the recruitment agencies pay to the local workers;

21　Finalize `Iallgatherw()` posted in the event **10**;

22　Finalize `MPI_Isend()/MPI_Irecv()` posted in the event **14**;

23　Recruitment agencies post `MPI_Isend()/Irecv()` to send/receive wage of workers *hired from/working at* other ranks;

24　Finalize `MPI_Isend()/Irecv()` posted in the event **23**, and the recruitment agencies pay the received wages to the workers

25　Households decide their consumption and investment budgets;

26　Big-household buyers post `Iallgatherw()` to distribute their budget to their proxies located in other ranks;

27　Finalize `Iallgatherw()` posted in the event **26**;

28　*SalesOutlets* post `MPI_Iallreduce()` to collect total demand of each sector;

29　Central bank calculates the economy wide indices and posts `MPI_Ibcast()` to share the indices;

30　Finalize `MPI_Iallreduce()` posted in the event **28**;

31　Finalize `Iallgatherw()` posted in the event **19**, and each sales-outlet calculates and keeps its portion of products to sell;

32　Firms buy (i.e., Firms' `buy()`);

33　Post `MPI_Ireduce()` to sum the quantities bought by *ProcurementDivisions* to the master rank (see event **10**);

34　Consumers, except the firms, buy (i.e., Final consumers' `buy()`);

35	Finalize `MPI_Ibcast()` posted in the event **29**;
36	Finalize `MPI_Ireduce()` posted in the event **33**, and post `Iscatterw()` to convey the quantities thus gathered by the master rank to the *ProcurementDivisions* of the big firms located in their home ranks;
37	Post `MPI_Ireduce()` to sum the quantities bought by the proxies of big-household buyers to the master rank;
38	Post `MPI_Ireduce()` to collect the quantities of goods bought by the bank's investor;
39	Post `MPI_Ireduce()` to sum the sales records of all the *SalesOutlets* to the master rank;
40	Local governments distribute social benefits to each household in their respective ranks;
41	Finalize `MPI_Ireduce()` posted in the event **39**, and post `Iscatterw()` to convey the sales records thus gathered by the master rank to the *SalesOutlets* located in the firms' home ranks;
42	Firms pay loan installments to the local banks in their respective ranks;
43	Finalize `Iscatterw()` posted in the event **41**. *SalesOutlets* deliver the sales record, thus received, to their parent firms;
44	Finalize `MPI_Ireduce()` posted in the event **37**, and post `Iscatterw()` to convey the quantities thus gathered by the master rank to the non-local buyers of the big household buyers located in the their home ranks;
45	Finalize `Iscatterw()` posted in the event **36**. The *ProcurementDivisions* deliver the total quantity, thus received, to their parent firms;
46	Finalize `Iscatterw()` posted in the event **44**. The non-local buyers deliver the total quantity, thus received, to their parent households;
47	Do the end of tth period's accounting (see Algorithm 7 for details);
48	Finalize `MPI_Ireduce()`s posted in the events **7** and **10** of the event **47**;
49	Agents reset their variables to prepare for the next period;
50	Finalize `MPI_Ibcast()` posted in the event **20** of the event **47**;

Algorithm 7: End of tth period's accounting for the ***final version*** of the code

1 Firms do period-end accounting;

2 Firms pay dividends to their investors;

3 Firms pay taxes to the local governments;

4 Firms settle their account with the local banks, and deposit extra money;

5 Foreign buyers pay taxes to the local governments;

6 Post `MPI_Ireduce()` to reduce imports, exports, and GVA to the master rank for the central bank's accounting;

7 Post two `MPI_Ireduce()`s to collect industry summary (i.e., GDP, production, sales, demands, employment, etc.) and foreign buyers summaries (i.e., consumption budget, quantity of goods bought) at the master rank;

8 Households pay taxes to the local governments;

9 Households settle their account with the local banks, and deposit extra money;

10 Post `MPI_Ireduce()` to sum household summary (i.e., consumption and investment budget, quantities of consumption and investment goods bought, wages received, etc.) to the master rank;

11 Finalize `MPI_Ireduce()` posted in the event **6**;

12 Central bank does period-end accounting;

13 Local banks do period-end accounting;

14 Post `MPI_Ireduce()` for collecting financial reports from the local banks at the master rank;

15 Variables of the firms and their *SalesOutlets* are reset to prepare for the next period;

16 Finalize `MPI_Ireduce()` posted in the event **14**;

17 Main bank does accounting;

18 Bank's investor receives his dividends and updates his/her bank account;

19 Main bank updates its account at the central bank;

20 Post `MPI_Ibcast()` to share the main bank's equity, loan etc. among all the local banks;

21 Main bank pays tax to the local government;

22 Post `MPI_Ireduce()` to sum quantity bought, tax collected, and social benefits paid by the local governments at the master rank;

23 Variables of the households are reset to prepare for the next period;

24 Finalize `MPI_Ireduce()` posted in the event **22**;

25 Central government does accounting and update its account at the central bank;