Rock Paper Scissors Model Using MPI

Jinzi Mac Huang

05/17/2015

Rock Paper Scissors (RPS) Model describes the competition between three reactants A, B and C with the reaction and diffusion:

\[ A + B \rightarrow AA \text{ with rate } \mu_1, \]
\[ B + C \rightarrow BB \text{ with rate } \mu_2, \]
\[ C + A \rightarrow CC \text{ with rate } \mu_3, \]
\[ A + \Box \rightarrow AA \text{ with rate } d_1, \]
\[ B + \Box \rightarrow BB \text{ with rate } d_2, \]
\[ C + \Box \rightarrow CC \text{ with rate } d_3. \]

Here \( \mu_i \) is the reaction rate and \( d_i \) is the diffusion rate. The role of A, B and C is just like the role of rock, paper and scissors in the RPS game. This system is a classical reaction diffusion system, which is related to the study of pattern formation and biodiversity. In this final project, 2D and 3D Monte Carlo styled lattice models are made to study the spatial pattern formation and temporal oscillation in the RPS model.

In the lattice simulation, the evolution in time is achieved by letting each reactant check its neighbours and react with them with a probability equals to the reaction rate. If there is any empty neighbour, the reactant can decide whether or not to occupy the neighbour with a probability equals to its diffusion rate. Since the conflict might happen if two reactants want to take the same position, a random variable is adopted to control the acceptance from different reactants with equal chance.

The result of 2D simulation is shown in the the movie[1]. In the simulation we use \( d_1 = 0.8, d_2 = 0.7, d_3 = 0.8, \mu_1 = 0.3, \mu_2 = 0.8, \mu_3 = 0.8 \) with periodic boundary condition. Each reactant is initially placed at a single point on the 1000 \times 1000 lattice. From the simulation, we see the growing reactant density quickly breaks into small structures, the population (density) of reactant goes to an equilibrium distribution as those structures occupy the whole domain. We should note that this system is far from equilibrium, as we’ll show later that the population is oscillatory in time.

We can extract the pattern formed by one reactant as the system breaks into small structures. Figure 1 shows such patterns, from the inset we can see that the pattern has a length scale of several lattice units. To study the pattern, we can use the box counting method to determine the fractal dimension. As shown in Figure 2(a), this pattern has fractal dimension 1.9, which is very close to 2D pattern. Next we can study the typical length scale of the system, one way to do this is to study the correlation length of the pattern. If the spatial correlation $C(|\mathbf{x} - \mathbf{x}'|)$ decays exponentially in distance, we can define the correlation length $l$ if $C(|\mathbf{x} - \mathbf{x}'|) \sim \exp(-|\mathbf{x} - \mathbf{x}'|/l)$. Figure 2(b) shows the log-linear plot of the spatial correlation, we see the correlation decays exponentially within a very short distance, after line fitting we conclude that our RPS system has a correlation length of $l = 4.57$ lattice units.

Next we can extend our simulation to 3D. As the computation gets more
intensive, a paralleled version of program is adopted in our simulation. The parallelism is achieved through the Message Passing Interface (MPI) running on the supercomputer Stampede.

The main computational kernel in the RPS model is to generate random numbers and check the char value on each slot. Hence the parallelism is straightforward using MPI: each core takes charge of one computing domain and evolves it into next time step, then they communicate boundary information through MPI non-blocking communication. After that, similar conflict resolving schemes are performed as we did in the 2D program.

One major deficiency in our code, is that the cubic computing domain is divided into several slices. This causes the inter-core information grows too fast as the size of computation grows. One remedy is to cutting the domain by small blocks, which can lead to the reduction of total surface area (boundary information). However doing so brings new difficulty of parallelism since we need to order each block and let them find its neighbours, hence we did not add such a feature to the program, which could be a future implementation.

We can also study the scalability of our program. This study is carried on the Stampede with 8, 16, 32 and 64 cores, the runtime result has been shown on Figure 3. We can see the program is almost strong scalable, but as we suggested before, the situation gets worse as we increase the size of computation.

In the 3D simulation, we use \( d_1 = d_2 = d_3 = 0.8 \) and \( \mu_1 = 0.4, \mu_2 = 0.6, \mu_3 = 0.8 \) on a cubic lattice of size 200 \( \times \) 200 \( \times \) 200 with periodic boundary. Though the study of spatial correlation would be expensive, we can still use the 3D simulation to study the population dynamics of RPS model. The population dynamics is shown on Figure 4, we see the population of each reactant grows exponentially at the beginning, and then different reactants start to react with each other and finally go to a stage that has small spatial structures everywhere.

Denote the population density of A, B and C as \( a(t), b(t) \) and \( c(t) \), we can

---

3

---
conclude \( a(t) + b(t) + c(t) = 1 \) for large enough \( t \). Next we write down the evolutionary equation of \( a(t) \), \( b(t) \) and \( c(t) \) with the mean field approximation:

\[
\begin{align*}
\partial_t a(t) &= a(t)[\mu_1 b(t) - \mu_3 c(t)] \\
\partial_t b(t) &= b(t)[\mu_2 c(t) - \mu_1 a(t)] \\
\partial_t c(t) &= c(t)[\mu_3 a(t) - \mu_2 b(t)]
\end{align*}
\]

This ODE system has a fixed point at \((a^*, b^*, c^*) = (\mu_2, \mu_3, \mu_1)/(\mu_1 + \mu_2 + \mu_3)\).

Denote \( \mathbf{u} = (a-\mathbf{u}^*, b-\mathbf{u}^*)^T \), we can write down the linearized equations around this fixed point:

\[
\begin{align*}
\partial_t \mathbf{u} &= \mathbf{L}\mathbf{u}, \quad \text{where} \quad \mathbf{L} = \frac{1}{\mu_1 + \mu_2 + \mu_3}
\begin{bmatrix}
\mu_2 \mu_3 & \mu_2 (\mu_1 + \mu_3) \\
-\mu_3 (\mu_1 + \mu_2) & -\mu_2 + \mu_3
\end{bmatrix}.
\end{align*}
\]

\( \mathbf{L} \) has 2 eigenvalues \( \lambda = \pm i \sqrt{\mu_1 \mu_2 \mu_3}/(\mu_1 + \mu_2 + \mu_3) \), they correspond to the temporal oscillation frequency of:

\[
f = \sqrt{\frac{\mu_1 \mu_2 \mu_3}{(\mu_1 + \mu_2 + \mu_3)}}/2\pi
\]

In our case \( \mu_1 = 0.4, \mu_2 = 0.6, \mu_3 = 0.8 \), we have \( f = 0.052 \). So we should be able to see this frequency if we look at the spectra of population. Figure 5 shows the population (mean free) and its Fourier transform, we obtain a peak at \( k = 108 \) in the frequency space. This peak corresponds to a temporal frequency of \( f = 2\pi/k = 0.0582 \), which is close to our theoretical prediction. Oscillations instead of equilibrium supports our argument that this system is actually far from equilibrium.

In conclusion, we confirmed the existence of spatial patterns and temporal oscillations in the RPS model, and the diversity is observed to be preserved in this simple model. The future work could involve the mutation among A, B and
Figure 5: Population (mean subtracted) and its Fourier transform at large $t$.

C, which is common in the competition between bacteria. The stability of diversity could be another interesting topic, the mean field ODE system is marginally stable so the thermal fluctuation might destroy the diversity eventually.

References


