1 Auroral LatticeKring Simulation Procedure

Algorithm 1: Auroral LK Steps

Result: Combine different source data to generate Auroral Map using LK

1. **Input:**
   - Data = satellite (Sate), interpolation (Inter), empirical (Emp), (and ground base Grd) data sets with same pattern: mlat, mlt, time, source
   - t = time period of interest (e.g. time interval = 10, 30 or 60 mins)
   - $ratio_{si}$ = ratio of Sate vs Inter
   - $ratio_{se}$ = ratio of Sate vs Emp
   - $ratio_{sr}$ = ratio of Sate vs Grd
2. **Output:** LK fitted auroral map data (Mean or Simulated Surface) in time period t
3. **Data Preparation:**
   - keep all the data at time t:
   - For empirical data with different scales, rescaling the data with different methods
     - **Normalization:** consider the area with satellite observations, rescaling the empirical data as $Emp_{flux} = \frac{(Emp_{flux} - min(Emp_{flux})}{range(Emp_{flux})} * range(Sate_{flux}) + min(Sate_{flux})$
     - **Regression***: consider multiple grid cells (s) at time t, average flux from satellite (points) and empirical (areas) data separately. Then fit the model:

       $$Sate_{flux}(s,t) = \beta_0(s,t) + \beta_1(s,t)Emp_{flux}(B,t) + \epsilon(s,t)$$

       Note: Since satellite data are at point level and empirical output are at grid level
   - Merge all data at time t with same scaling from different sources: All
4. **Generate Probabilistic Boundary:**
   - Create test locations: $N = nmlat * nlt * 2$ (can change)
   - Generate predict probabilities or indicator ($weight$) for each test locations using different decision boundary method (e.g., KNN method with training set: All)
5. **Downsampling Data sets: Random sampling**
   - Random sampling $N_{Sate} * ratio_{si}$ and $N_{Sate} * ratio_{se}$ for interpolation and empirical data. (e.g., $ratio_{si} = \frac{1}{6}, \frac{1}{10}$; $ratio_{se} = \frac{1}{36}, \frac{1}{60}$...)
   - Merge all satellite data at time t with downsampled inter and empirical data at t: All_{new}
6. **LatticeKrig Simulation**
   - Remove all $flux = 0$ from All_{new} and take logarithm of $flux$
   - Fit the LatticeKrig spatial model and get the mean or simulation as $exp(predict/sim.condition) * weight$ **[1]**
7. **Save and plot the fitted map in time period t**
   (Might need to adjust the flux, for example, $flux > 30 => flux = 30$)

---

2 Auroral Data

2.1 Satellite Data

- Satellite auroral Data: "*ssusi_combine_*20140219_20140221.*nc*"
- Location: Magnetic latitude (mlat): 363*363; Magnetic local time (mlt): 363*363
- Time:
  - Focus on **20140220** (24 hours)
  - time points: every 1 hour, totally (0:23)
    - For example: $\mathbf{t = 9}- > 8:30 - 9:30am$
- Hemisphere: **Northern** and Southern
- Flux and Energy

2.2 Interpolation Data

Linear interpolation from the satellite data

2.3 Empirical Output: Ovation

- Empirical Data "omni2.lst" + "premodel"
- Location: Magnetic latitude (mlat): 80; Magnetic local time (mlt): 96; Type: 3 (type==1)
- Time:
  - Focus on **20140220** (24 hours)
  - time points: every 1 hour, totally 24 (0, 1, 2,....,23)
- Hemisphere: **Northern** and Southern; Flux and Energy

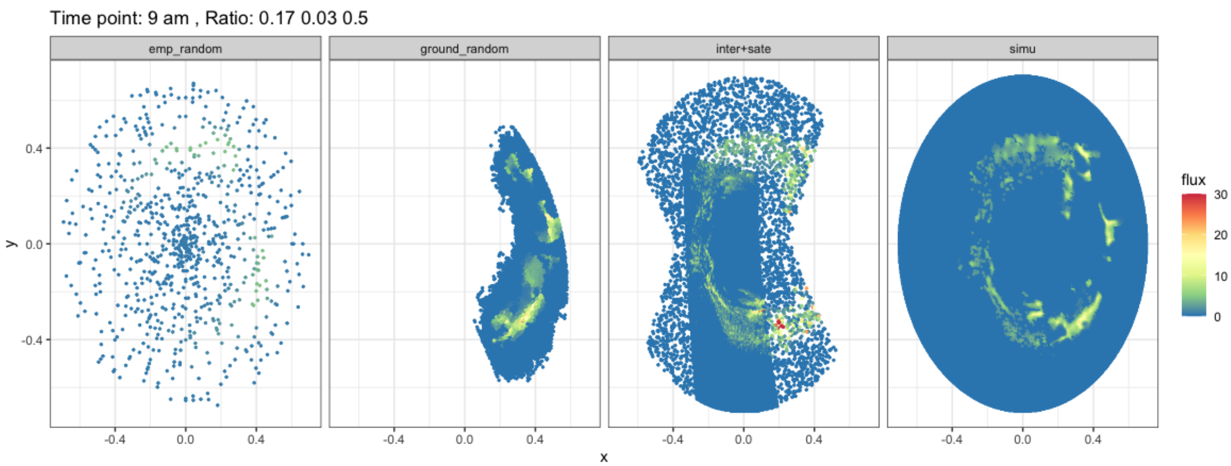
2.4 Ground Base Data

- Ground Base Data "themis2014022006_mag.nc"
- Location: Magnetic latitude (mlat): 400; Magnetic local time (mlt): 400
- Time:
  - Focus on **20140220** (24 hours)
  - time points: every 1 hour
- Hemisphere: **Northern** and Southern; Flux and Energy

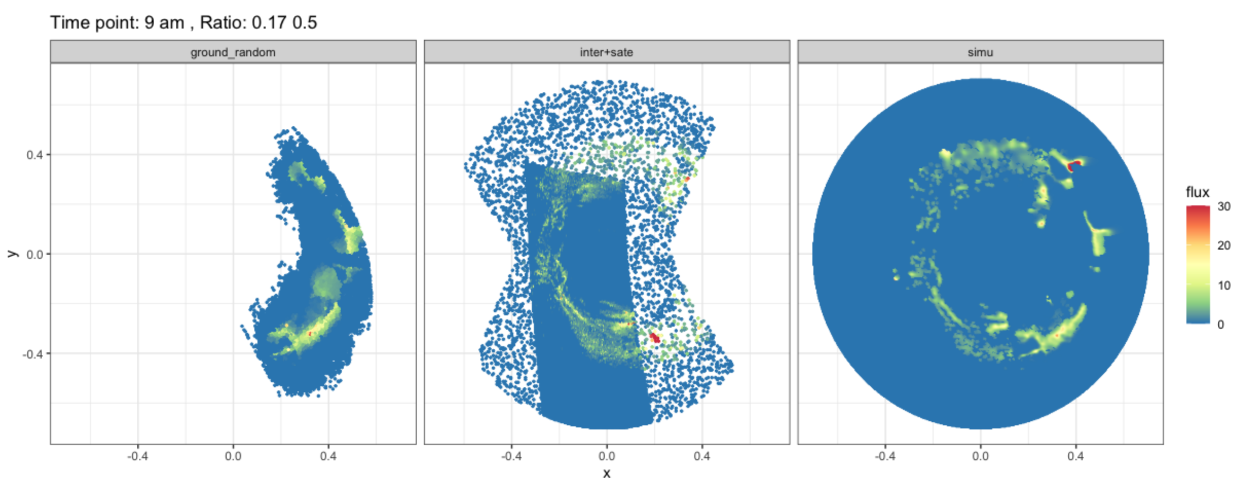
3 Simulation Results: Two Examples (Flux)

3.1 Fitted Auroral map at 8:30 - 9:30

3.1.1 With empirical data

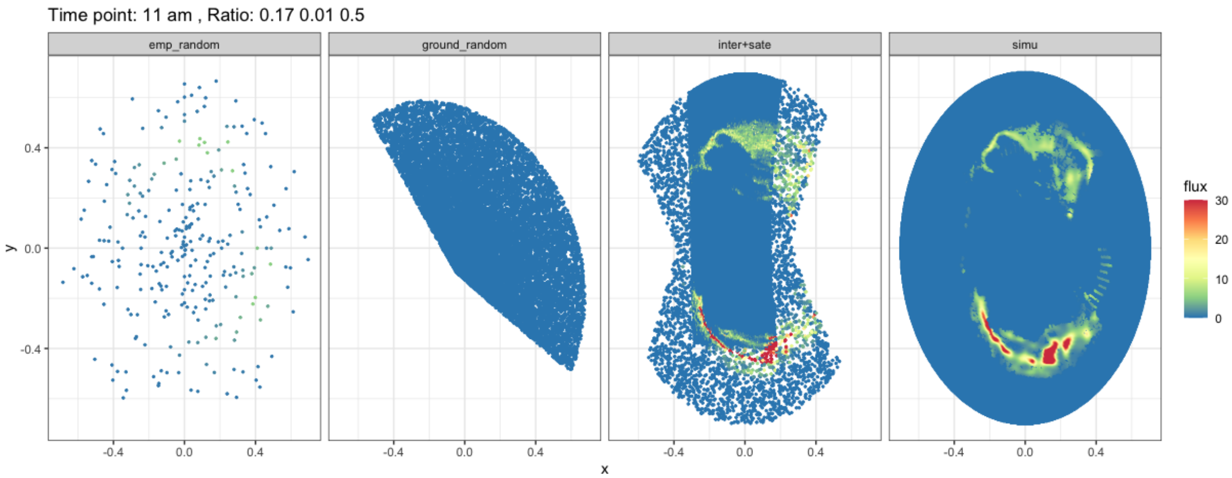


3.1.2 Without empirical data

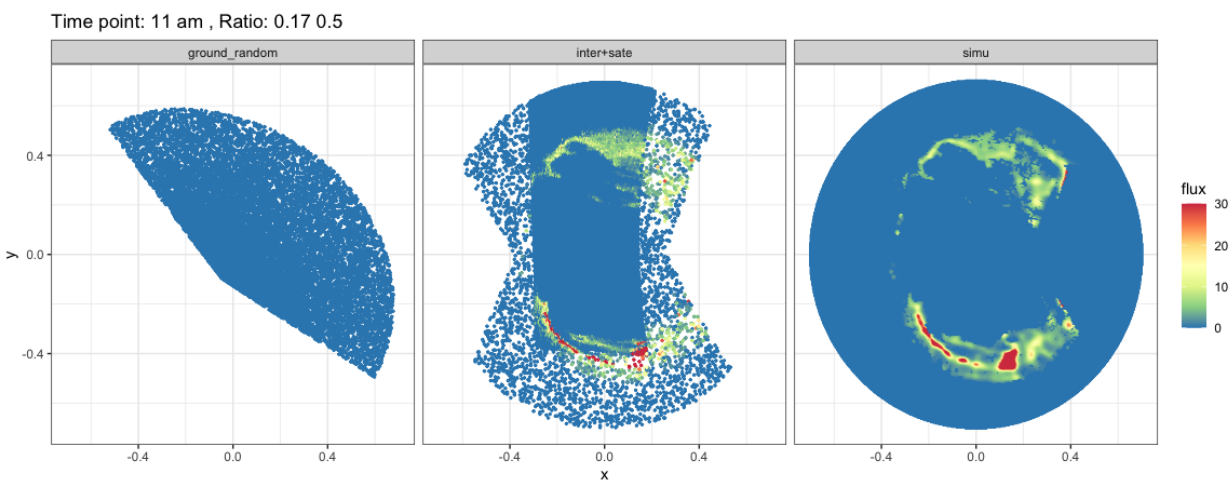


3.2 Fitted Auroral map at 10:30 - 11:30

3.2.1 With empirical data



3.2.2 Without empirical data



4 Boundary [1]

4.1 Auroral Oval: Outer Boundary (Circle Hough Transform)

- counter[:] = 0
- for (x, y) in image where flux(x,y) !=0
  for (a, b) in predefined ranges
  $r = \sqrt{(x-a)^2 + (y-b)^2}$ counter[a,b,r]++
- select the largest counter
- Transform a feature extraction problem in image space (x, y) to a counting problem in parameter space (a, b, r).

4.2 Combination of Outer Boundary and Probabilistic boundary

Consider the following weights to calculate the boundary:

- indicator weights (I_w): the indicator weights by KNN. Here we can consider the different k's (number of neighborhoods).
  - indicator weights ($I_{w,s}$): the indicator weights by KNN with a small k value.
  - indicator weights ($I_{w,l}$): the indicator weights by KNN with a large k value.
- weights (w): the probability weights by KNN
- outer boundary indicator (b): if a location is inside the outer boundary, the value is 1, otherwise, it is 0.

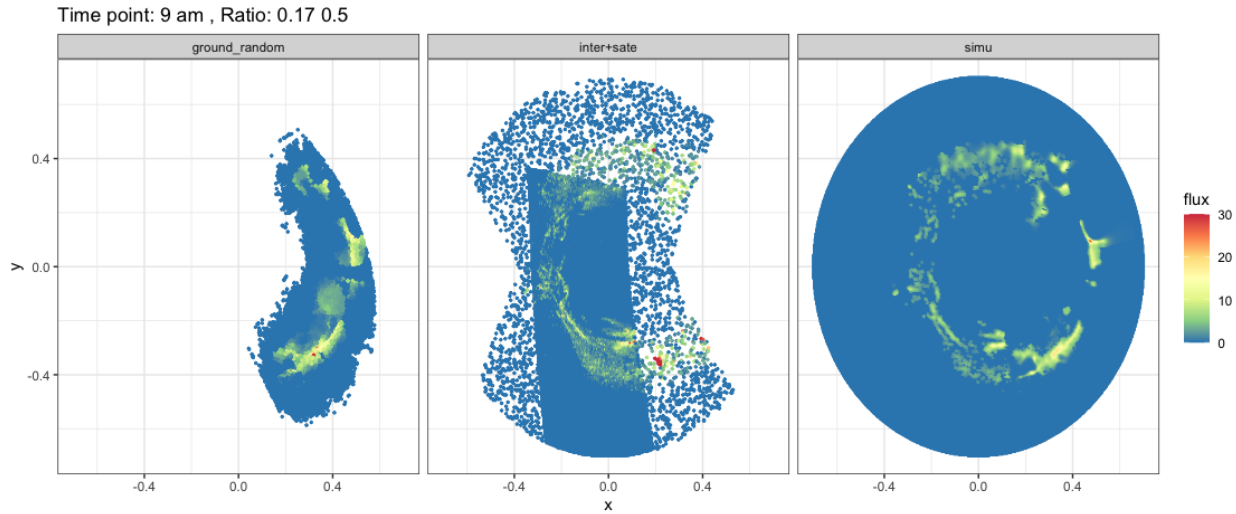
Then adjust the weights using the following methods,

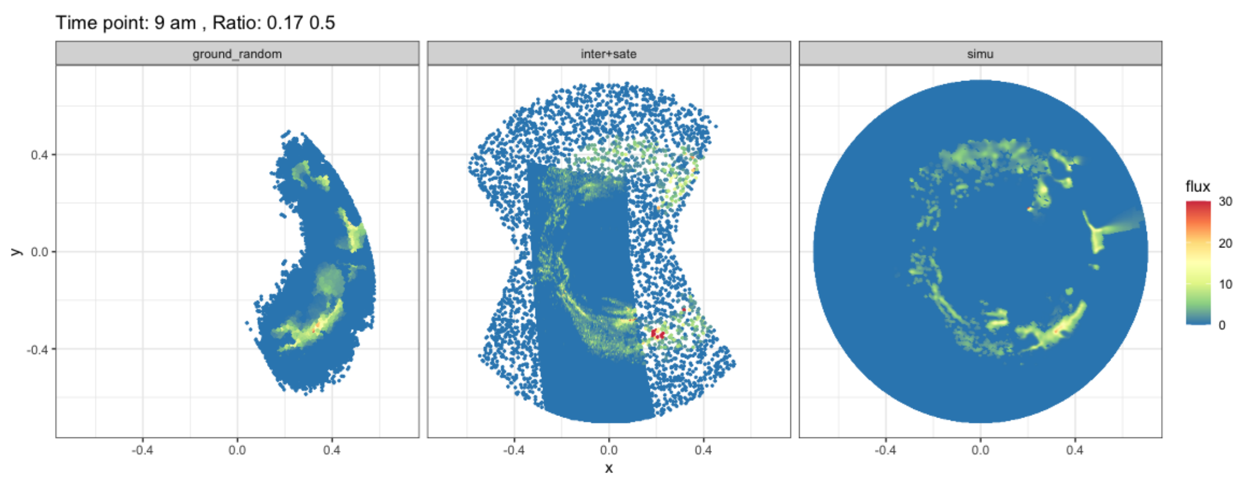
1. New Weights 1 = $I_w \times I(b = 1) + I_w \times I(b = 0)$ (I_w for inside and outside of the outer boundary are the same)
2. New Weights 2 = $I_{w,l} \times I(b = 1) + I_{w,s} \times I(b = 0)$
3. New weights 3 = $I_w \times I(b = 1) + w^v \times I(b = 1)$, where v is constant, $v \geq 1$.
4. New weights 4 = $w^u * I(b = 1) + w^v * I(b = 0)$, where $u = 0$, and $v \geq 1$. (same result as 3))

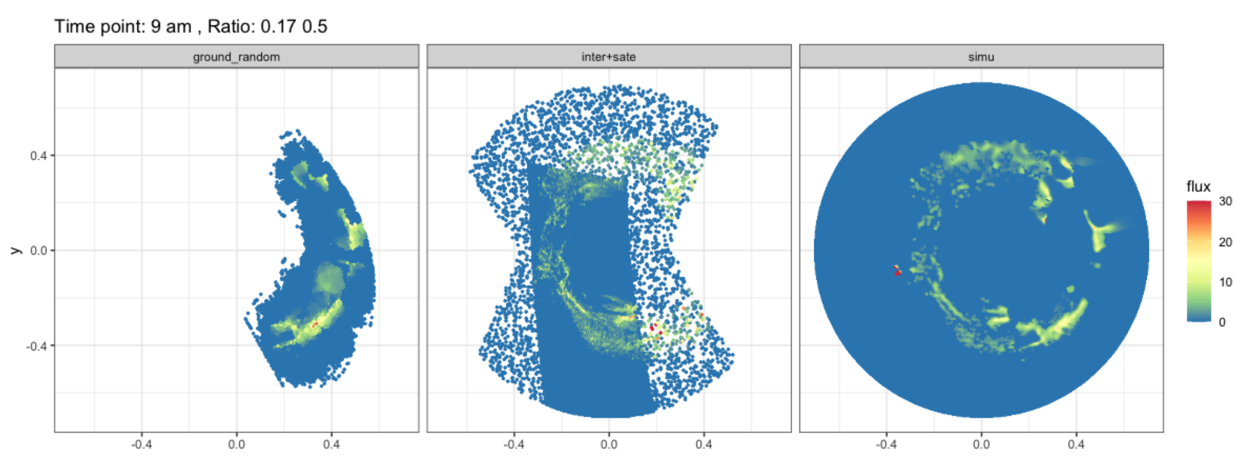
The plots on the following pages are: **Top** the fitted Auroral map using **New Weights 1**; **Middle** the fitted Auroral map using **New Weights 2**; **Bottom** the fitted Auroral map using **New Weight 3/4**.

4.3 Experiment Results 1: with outer boundary (Flux¿1), without empirical data

4.3.1 8:30 - 9:30

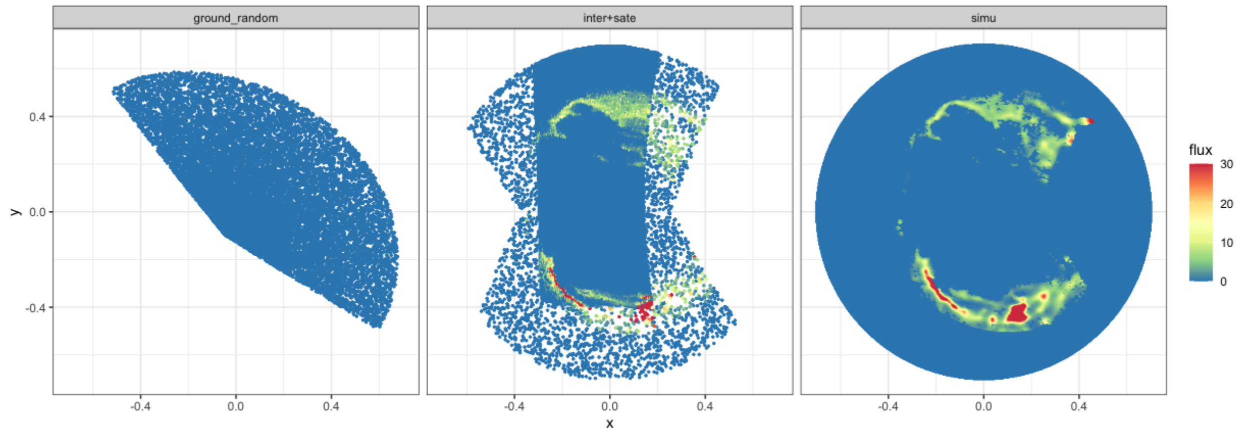




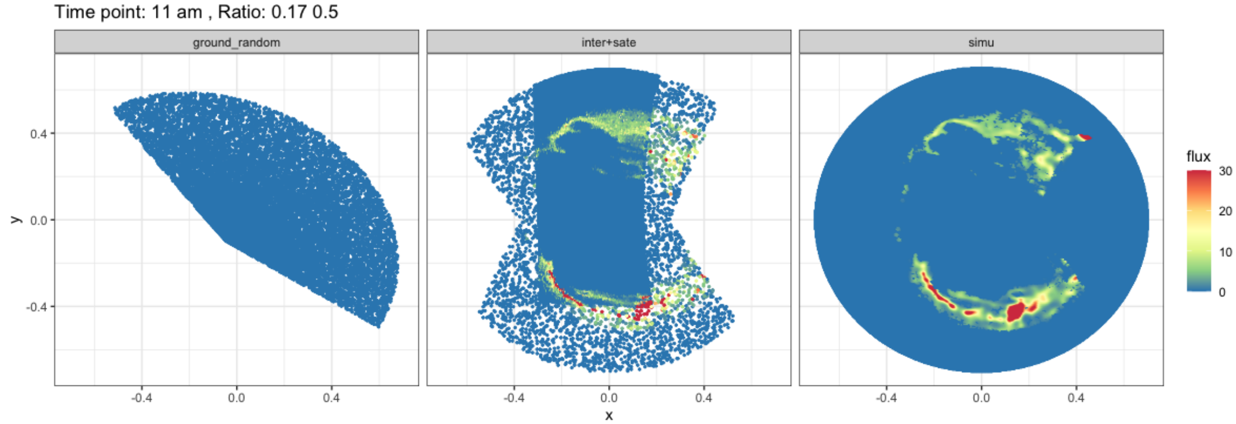


4.3.2 10:30 - 11:30

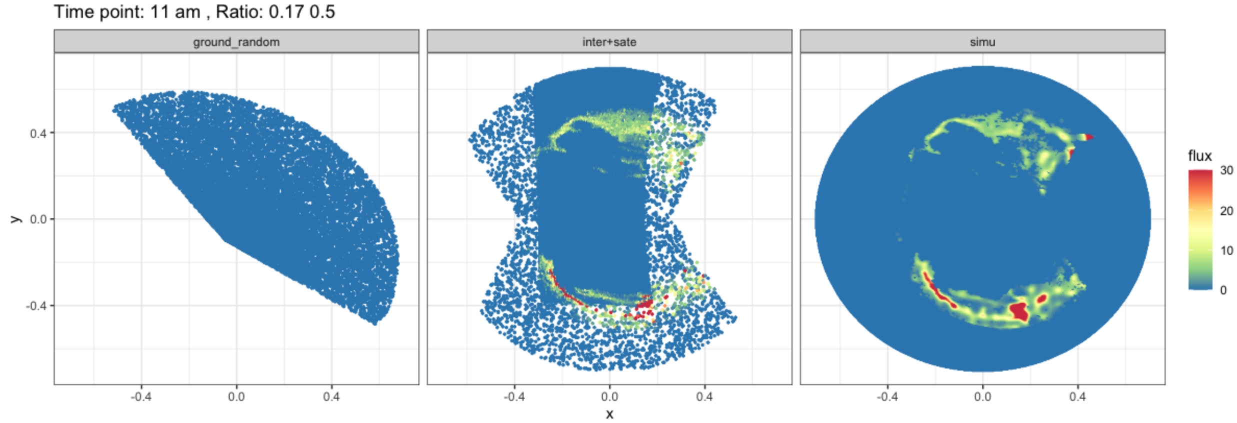
Time point: 11 am , Ratio: 0.17 0.5



Time point: 11 am , Ratio: 0.17 0.5



Time point: 11 am , Ratio: 0.17 0.5



5 Appendix

5.1 Outer Boundary Examples