

Load Balancing Unstructured Meshes for Massively Parallel Transport Sweeps

Tarek Ghaddar

Chair: Dr. Jean Ragusa

Committee: Dr. Jim Morel, Dr. Bojan Popov

Texas A&M University

Ghaddar (TAMU) Load Balancing March 8, 2016 1/17

- Overview and Introduction
- The Transport Equation
- Parallel Transport Sweeps
- 4 Method
- **5** Load Balancing Results
- Solution Verification
- Conclusions



2 / 17

Ghaddar (TAMU) Load Balancing March 8, 2016

Motivation

- When running any massively parallel code, load balancing is a priority in order to achieve the best possible parallel efficiency.
- A load balanced problem has an equal number of degrees of freedom per processor.
- Load balancing a logically Cartesian mesh is not difficult, as the user specifies the number of cells being used.
- In an unstructured mesh, the user cannot always specify the number of cells they want per processor, and obtaining a load balanced problem is more difficult.

Ghaddar (TAMU) Load Balancing March 8, 2016 3 / 17

PDT

Ghaddar (TAMU)

- All work presented in this thesis was implemented in Texas A&M's massively parallel deterministic transport code, PDT.
- It is capable of multi-group simulations and employs discrete ordinates for angular discretization.
- Features steady-state, time-dependent, criticality, and depletion simulations. It solves the transport equation for neutron, thermal, gamma, coupled neutron-gamma, electron, and coupled electron-photon radiation.
- PDT has been shown to scale on logically Cartesian grids out to 750.000 cores.

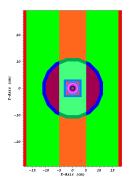
Load Balancing

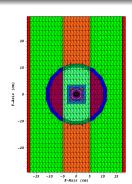
March 8, 2016

4 / 17

The Triangle Mesh Generator

 Unstructured meshes in PDT are generated using the Triangle Mesh Generator.





4 D > 4 D > 4 E > 4 E > E 9 9 P

Ghaddar (TAMU) Load Balancing March 8, 2016 5 / 17

The Transport Equation

$$\vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, E, \vec{\Omega}) + \Sigma_t(\vec{r}, E) \psi(\vec{r}, E, \vec{\Omega}) = \int_0^\infty dE' \int_{A_T} d\Omega' \Sigma_s(\vec{r}, E' \to E, \Omega' \to \Omega) \psi(\vec{r}, E', \vec{\Omega}') + S_{\text{ext}}(\vec{r}, E, \vec{\Omega})$$

$$\begin{split} \vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, E, \vec{\Omega}) + \Sigma_t(\vec{r}, E) \psi(\vec{r}, E, \vec{\Omega}) &= \\ \frac{1}{4\pi} \int_0^\infty dE' \Sigma_s(\vec{r}, E' \to E) \int_{4\pi} d\Omega' \psi(\vec{r}, E', \vec{\Omega}') + S_{\text{ext}}(\vec{r}, E, \vec{\Omega}) \\ &= \frac{1}{4\pi} \int_0^\infty dE' \Sigma_s(\vec{r}, E' \to E) \phi(\vec{r}, E') + S_{\text{ext}}(\vec{r}, E, \vec{\Omega}) \end{split}$$

Ghaddar (TAMU) Load Balancing March 8, 2016 6 / 17

The Transport Equation

$$\phi(\vec{r}, E') = \int_{A_{\tau}} d\Omega' \psi(\vec{r}, E', \vec{\Omega}')$$

$$\begin{split} \vec{\Omega} \cdot \vec{\nabla} \psi_{g}(\vec{r}, \vec{\Omega}) + \Sigma_{t,g}(\vec{r}) \psi_{g}(\vec{r}, \vec{\Omega}) &= \frac{1}{4\pi} \sum_{g'} \Sigma_{s,g' \to g}(\vec{r}) \phi_{g'}(\vec{r}) + S_{\text{ext},g}(\vec{r}, \vec{\Omega}), \\ & \text{for } 1 \leq g \leq G \end{split}$$

4 ロ ト 4 昼 ト 4 昼 ト 4 昼 ト 4 回 ト

Ghaddar (TAMU) Load Balancing March 8, 2016 7 / 17

The Transport Equation

$$\vec{\Omega}_{\textit{m}} \cdot \vec{\nabla} \psi_{\textit{g},\textit{m}}(\vec{r}) + \Sigma_{\textit{t},\textit{g}}(\vec{r}) \psi_{\textit{g},\textit{m}}(\vec{r}) = \frac{1}{4\pi} \sum_{\textit{r},\textit{r}} \Sigma_{\textit{s},\textit{g}' \rightarrow \textit{g}}(\vec{r}) \phi_{\textit{g}'}(\vec{r}) + S_{\textit{ext},\textit{g},\textit{m}}(\vec{r})$$

$$\phi_{g}(\vec{r}) \approx \sum_{m=M}^{m=M} w_{m} \psi_{g,m}(\vec{r}).$$

$$\vec{\Omega}_m \cdot \vec{\nabla} \psi_m^{(l+1)}(\vec{r}) + \Sigma_t \psi_m^{(l+1)}(\vec{r}) = q_m^{(l)}(\vec{r})$$

401491471717

Ghaddar (TAMU) Load Balancing March 8, 2016 8 / 17

The Transport Sweep

A parallel sweep algorithm is defined by three properties:

- partitioning: dividing the domain among available processors
- aggregation: grouping cells, directions, and energy groups into tasks
- scheduling: choosing which task to execute if more than one is available

<ロ > < 回 > < 回 > < 巨 > くき > しき > しき の < ♡

9 / 17

The Sweep

4	5	6	7
3	4	5	6
2	3	4	5
1	2	3	4



| Ghaddar (TAMU) | Load Balancing | March 8, 2016 | 10 / 17

990

Aggregation

- $A_x = \frac{N_x}{P_x}$, where N_x is the number of cells in x and P_x is the number of processors in x
- $A_y = \frac{N_y}{P_y}$, where N_y is the number of cells in y and P_y is the number of processors in y
- $N_g = \frac{G}{A_g}$
- $N_m = \frac{M}{A_m}$
- $N_k = \frac{N_z}{P_z A_z}$
- $\bullet \ N_k A_x A_y A_z = \frac{N_x N_y N_z}{P_x P_y P_z}$



11 / 17

Ghaddar (TAMU) Load Balancing March 8, 2016

Parallel Efficiency

$$\begin{split} \epsilon &= \frac{T_{\mathsf{task}} N_{\mathsf{tasks}}}{[N_{\mathsf{stages}}][T_{\mathsf{task}} + T_{\mathsf{comm}}]} \\ &= \frac{1}{[1 + \frac{N_{\mathsf{idle}}}{N_{\mathsf{tasks}}}][1 + \frac{T_{\mathsf{comm}}}{T_{\mathsf{task}}}]} \end{split}$$

$$T_{\text{comm}} = M_L T_{\text{latency}} + T_{\text{byte}} N_{\text{bytes}}$$

 $T_{\text{task}} = A_x A_y A_z A_m A_\sigma T_{\sigma \text{rind}}$

4□ > 4□ > 4 = > 4 = > = 99

 Ghaddar (TAMU)
 Load Balancing
 March 8, 2016
 12 / 17

Metric Definitions

$$\bullet \ f = \frac{\max_{ij}(N_{ij})}{\frac{N_{tot}}{I \cdot I}}$$

•
$$f_I = \max_i [\sum_j N_{ij}] / \frac{N_{toi}}{I}$$

•
$$f_I = \max_i \left[\sum_j N_{ij} \right] / \frac{N_{tot}}{I}$$

• $f_J = \max_i \left[\sum_i N_{ij} \right] / \frac{N_{tot}}{J}$



Ghaddar (TAMU) Load Balancing March 8, 2016 13 / 17

Algorithm

```
//I, J subsets specified by user
//Check if all subsets meet the tolerance
while (f < tol_subset)
  //Mesh all subsets
  else
    if (f_l > tol_column)
      Redistribute(X);
    if (f_J > tol_row)
      Redistribute(Y);
```

Ghaddar (TAMU) Load Balancing March 8, 2016 14 / 17

Overview and Introduction The Transport Equation Parallel Transport Sweeps Method Load Balancing Results Solution Verifica

Ghaddar (TAMU) Load Balancing March 8, 2016 15 / 17

Overview and Introduction The Transport Equation Parallel Transport Sweeps Method Load Balancing Results Solution Verifica

Ghaddar (TAMU) Load Balancing