HAL
open science

3D radiative transfer modeling over complex vegetation canopies and forest reconstruction from LIDAR measurements

Jianbo Qi

To cite this version:

Jianbo Qi. 3D radiative transfer modeling over complex vegetation canopies and forest reconstruction from LIDAR measurements. Continental interfaces, environment. Université Paul Sabatier - Toulouse III, 2019. English. NNT : 2019TOU30048 . tel-02498603

HAL Id: tel-02498603

https://theses.hal.science/tel-02498603v1

Submitted on 4 Mar 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 3 - Paul Sabatier

Cotutelle internationale : Faculty of Geographical Science, Beijing Normal University

Présentée et soutenue par

Jianbo QI

Le 6 juin 2019

Modélisation 3D du transfert radiatif dans les couverts végétaux et reconstruction de forêts à partir de mesures LiDAR

Ecole doctorale : **SDU2E - Sciences de l'Univers, de l'Environnement et de l'Espace**

Spécialité : **Surfaces et interfaces continentales, Hydrologie**

Unité de recherche :
CESBIO - Centre d'Etudes Spatiales de la Biosphère

Thèse dirigée par
JEAN PHILIPPE GASTELLU ETCHEGORRY et Guangjian Yan

Jury
M. Stephane JACQUEMOUD, Rapporteur
M. Huaguo HUANG, Rapporteur
M. Qing XIAO, Examinateur
M. Liangyun LIU, Examinateur
M. Jean philippe GASTELLU ETCHEGORRY, Directeur de thèse
M. Guangjian YAN, Co-directeur de thèse
M. Tiangang YIN, Co-directeur de thèse
Mme Donghui XIE, Co-directrice de thèse

Dédié à ma mère et mon père

Dedicated to my mother and father

献给我的母亲与父亲

Acknowledgement

When I write down the word “acknowledgement”, I know that something is coming—the end of my PhD career. This is a long journey from bachelor to master, and finally to PhD. During this journey, I have encountered many difficulties, however, fortunately, there are always somebody there helping and supporting me, to whom I want express my great appreciation.

First of all, I want to give my deepest gratitude to my supervisor, Prof. Jean-Philippe Gastellu-Etchegorry. When I stay in France, he gave the greatest help to me both in scientific research and my daily life. I still remember the days when he derived radiative transfer equations step by step for me, which is a lifelong and unforgettable souvenir. I am also deeply touched by his continuous hard work and enthusiasm to improve the DART model since 1992. Life can be easier, but he chooses the other way. His attitude towards scientific research has greatly influenced me, and he is the objective in my future career.

I want to thank my co-supervisor Prof. Guangjian Yan. I have been working with him more than 5 years from master to PhD. He is a gentle man, but has very strict scientific attitude. He can always give me valuable and key advice when I hit a bottleneck in my research. Besides, he also gives me a lot of advice for attitude towards life. I am also very thankful for supporting my study in France, since this experience is a unique and unforgettable in my whole life.

I would like to thank my other supervisor Prof. Donghui Xie. She is my supervisor for master degree and also for PhD degree. She gave me the initial idea to develop a 3D radiative transfer model for my research. We have discussed a lot, in both research and life, during the past few years. She gave me her maximum support when I needed it. She financially supports me to attend a lot of international conferences, which greatly broadens my horizon. She is my supervisor but also a good friend who is worth to cherish in my whole life.

I would like to thank my co-supervisor Dr. Tiangang Yin. He is a very conscientious person in scientific research. During the past few years, he helped me a lot in my research and correction of my manuscripts. We have been working together for improving the DART model, and satisfying results have been achieved. Without his help, many research work that I have done would not have been possible. I wish him having a bright future.

I would also like to thank the whole DART team: Dr. Nicolas Lauret, Eric Chavanon, Jordan Guilleux. During my stay in France, they kindly helped me. Dr. Nicolas Lauret supervises all DART computer science development and maintenance. I communicated a lot with him. He took plenty of time to explain me the details of the DART code, which made me familiar with DART much faster and easier. Eric is a very experienced engineer, who almost dealt with all the compiling problems that I have encountered during my modification of DART code. Jordan is developing a new GUI for DART in order to make it more robust in terms of computer science and also more user friendly. I wish them success and happiness in work and personal life.

Many thanks to all the persons who I have met in France, they helped me a lot in my life: Catherine Stasiulis, Ying Dai, Ying Pei, Yingjie Wang, Tianqi Cang, Biao Cao, Jie Shao, Christine Fontas, Christian Mourgues, Patrizia Tavormina, etc. Best wishes to all of you.

I would like to thank all the colleagues in Beijing Normal University, they are my best friends and give me a lot of supports. Thanks to Linyuan Li and Qing Chu, who are the first users of my LESS model; Thanks to Yue Xu, who helps me a lot during my PhD defense; Thanks to Kai Yan, who give me a lot of advice during my application for a job; Thanks to Hailan Jiang, we have many discussions about LiDAR . Many thanks to Jing Zhao, Zhonghu Jiao, Yiting Wang, Yiming Chen, Xiangyu Wang, Yan Wang, Ronghai Hu, Yahui Li, Xiaoyan Wang, Gaiyan Ruan, Dashuai Guo, Jinghui Luo, Yingji Zhou, Zhu Long, Yiyi Tong, FangLiang Jiang.

Here, I want also thank China Scholarship Council (CSC). It gave me the financial support for my work in France. Without this support, I would not have had the chance to meet the excellent DART team, which greatly impressed and influenced me by their professionalism, hard work and enthusiasm for pursuing scientific truth.

The sincerest thanks to my parents, their greatest understanding of me gives me the space to improve myself and pursuit scientific truth. I am so sorry that I did not accompany them for many years since I went to university. Your support is the strongest shield behind me. “I love you”.

List of publications

Articles

1. **Qi, J.**, Xie, D., Yin, T., Yan, G., Gastellu-Etchegorry, J.-P., Li, L., Zhang, W., Mu, X., Norford, L.K., 2019. LESS: LargE-Scale remote sensing data and image simulation framework over heterogeneous 3D scenes. Remote Sensing of Environment. 221, 695–706.
2. **Qi, J.**, Xie, D., Guo, D., Yan, G., 2017. A Large-Scale Emulation System for Realistic Three-Dimensional (3-D) Forest Simulation. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 10, 4834–4843.
3. **Qi, J.**, Xie, D., Li, L., Zhang, W., Mu, X., Yan, G., 2019. Estimating Leaf Angle Distribution from Smartphone Photos. IEEE Geoscience and Remote Sensing Letters. [In press, DOI: 10.1109/LGRS.2019.2895321]
4. Zhang, W., **Qi, J.***, Wan, P., Wang, H., Xie, D., Wang, X., Yan, G., 2016. An easy-to-use airborne LiDAR data filtering method based on cloth simulation. Remote Sensing. 8, 501.
5. Xie, D., Wang, X., **Qi, J.**, Chen, Y., Mu, X., Zhang, W., Yan, G., 2018. Reconstruction of Single Tree with Leaves Based on Terrestrial LiDAR Point Cloud Data. Remote Sensing. 10.
6. Wan, P., Zhang, W., Skidmore, A.K., **Qi, J.**, Jin, X., Yan, G., Wang, T., 2018. A simple terrain relief index for tuning slope-related parameters of LiDAR ground filtering algorithms. ISPRS journal of photogrammetry and remote sensing.
7. Gastellu-Etchegorry J.P., Lauret N., Yin T., Landier L., Kallel A., Malenovský Z., Al Bitar A., Aval J., Benhmida S., **Qi J.**, Medjdoub G., Guilleux J., Chavanon E., Cook B., Morton D., Chrysoulakis N., Mitraka Z. 2017. DART: recent advances in remote sensing data modeling with atmosphere, polarization, and chlorophyll fluorescence IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10: 6, pp 2640-2649.
8. **Qi, J.**, Yin, T., Xie, D., Chavanon, E., Laret, N., Guilleux, J., Gastellu-Etchegorry, J., Hybrid

Scene Structuring for Accelerating 3D Radiative Transfer Simulations. Submitted to Computers & Geosciences. [under review]

9. Yin, T.*, **Qi, J.***, Cook, B., Morton, D., Gastellu-Etchegorry, J.-P., Wei, S., Physical modeling and interpretation of leaf area index measures derived from LiDAR point clouds. Submitted to Remote Sensing of Environment. [under review]

Conference

1. **Qi, J.**, Gastellu-Etchegorry, J., Yin, T., 2018. Reconstruction of 3D Forest Mock-Ups from Airborne LiDAR Data for Multispectral Image Simulation Using DART Model, in: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 3975–3978.
2. **Qi, J.**, Xie, D., Yan, G., 2016a. Realistic 3D-simulation of large-scale forest scene based on individual tree detection, in: Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International. IEEE, pp. 728–731.
3. **Qi, J.**, Xie, D., Zhang, W., 2016b. Reconstruction of individual trees based on LiDAR and in situ data, in: 2nd ISPRS International Conference on Computer Vision in Remote Sensing (CVRS 2015). International Society for Optics and Photonics, p. 99011E.
4. Yin, T., **Qi, J.**, Gastellu-Etchegorry, J.-P., Wei, S., Cook, B.D., Morton, D.C., 2018. Gaussian Decomposition of LiDAR Waveform Data Simulated by Dart, in: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 4300–4303.
5. Chen, Y., Zhang, W., Hu, R., **Qi, J.**, Shao, J., Li, D., Wan, P., Qiao, C., Shen, A., Yan, G., 2018. Estimation of forest leaf area index using terrestrial laser scanning data and path length distribution model in open-canopy forests. Agricultural and Forest Meteorology 263, 323–333.
6. Cai, S., Zhang, W., **Qi, J.**, Wan, P., Shao, J., Shen, A., 2018. Applicability analysis of cloth simulation filtering algorithm for mobile lidar point cloud. International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences 42.

Abstract

Remote sensing is needed for better managing vegetation covers. Hence, three-dimensional (3D) radiative transfer (RT) modeling is essential for understanding remote sensing signals of complex 3D vegetation covers. Due to the complexity of 3D models, one-dimensional (1D) RT models are commonly used to retrieve vegetation parameters, *e.g.*, leaf area index (LAI), from remote sensing data. However, 1D models are not adapted to actual vegetation covers because they abstract them as schematic 1D layers, which is not realistic. Much effort is devoted to the conception of 3D RT models that can consider the 3D architecture of vegetation covers. However, developing an efficient 3D RT model that works on large and realistic scenes is still a challenging task. Major difficulties are the intensive computational costs of 3D RT simulation and the acquisition of detailed 3D canopy structures. Therefore, 3D RT models usually only work on abstracted scenes or small realistic scenes. Scene abstraction may cause uncertainties, and the small-scale approach is not compatible with most satellite observations (*e.g.*, MODIS). The computer graphics community provides the most accurate and efficient models (i.e., renderers). However, the initial renderer models were not designed for accurate RT modeling, which explains the difficulty to use them for remote sensing applications.

Recently emerged advanced techniques in computer graphics and light detection and ranging area (LiDAR) make it more possible to solve the above problems. 3D RT can be greatly accelerated due to the increasing computer power and improvement of rendering algorithms (e.g., ray-tracing acceleration and computational optimization). Also, 3D high-resolution information from LiDARs and photogrammetry become more accessible to reconstruct realistic 3D scenes. This approach requires new processing methods to combine 3D information and 3D RT models, which is of great importance for better remote sensing survey of vegetation.

This thesis is focused on 1) Development of a 3D RT model based on recent ray-tracing techniques and 2) Retrieval of 3D leaf volume density (LVD) for constructing 3D forest scenes.

This first chapter presents the development of an efficient 3D RT model, named LESS (**L**arg**E**-**S**cale remote sensing data and image **S**imulation framework). LESS makes full use of ray-tracing algorithms. Specifically, it simulates multispectral BRF and scene radiative budget with a weighted forward photon tracing method, and sensor images (e.g., fisheye images) or large-scale (e.g. 1 km^2) spectral images are simulated with a backward path tracing method. In the forward mode, a “virtual

photon” algorithm is used to simulate accurate BRF with few photons. The backward mode is used to simulate thermal infrared images and also atmosphere RT. LESS efficiency and accuracy were demonstrated with a model intercomparison and field measurements. In addition, LESS has an easy-to-use graphic user interface (GUI) to input parameters, construct and visualize 3D scenes.

3D forest reconstruction is done with a simulated LiDAR dataset to assess approaches that retrieve LVD from airborne LiDAR data. The dataset is simulated with the discrete anisotropic radiative transfer model (DART). First, a hybrid scene structuring scheme was designed to accelerate DART, and consequently to improve its potential for sensitivity studies. Then, an intensity-based method was designed for retrieving LVD. It only uses LiDAR ground returns with no assumption about canopy structures. In this thesis, the 3D forest scene is reconstructed through a voxel-based representation of canopies, which can be input into LESS. The comparison of LESS simulated and airborne hyperspectral images showed a good consistency.

LESS model can be used to validate other physical models, to develop parameterized models and to train neural networks. This thesis also provides a solution to fill the gap between 3D RT model and 3D canopy structures.

Résumé

La télédétection est un outil majeur pour mieux gérer les couverts végétaux. De ce fait, la modélisation du transfert radiatif (TR) tridimensionnel (3D) est essentielle pour comprendre les mesures de télédétection des couverts végétaux. Les modèles de TR unidimensionnels (1D) sont souvent utilisés pour inverser les données de télédétection en termes de paramètres de la végétation. Cependant, ils ne sont pas adaptés à la complexité des couverts végétaux, car ils les simulent comme des couches homogènes, ce qui est irréaliste. Beaucoup de travaux sont donc consacrés à la conception de modèles de TR 3D adaptés à l'architecture 3D des couverts végétaux. Cependant, développer un modèle de TR 3D efficace pour de grandes scènes réalistes est un défi en termes de modélisation du TR et d'obtention de description réaliste du couvert végétal. Ainsi, les modèles de TR 3D actuels n'opèrent en général que sur des scènes très simplifiées ou des scènes réalistes de petite dimension inadaptées à la résolution de la plupart des capteurs satellites actuels (*e.g.*, MODIS). Les modèles de la communauté "informatique graphique" (i.e., modèles de "rendu") sont les plus précis et les plus efficaces, mais ils n'ont pas été conçus pour une modélisation précise du TR. Ils sont donc peu employés pour les applications de télédétection.

Les progrès en infographie et puissance informatique permettent de beaucoup accélérer les modèles 3D de TR. De plus, les mesures 3D à haute résolution spatiale des LiDARs et caméras photogrammétriques deviennent plus accessibles pour reconstruire des couverts végétaux réalistes. Il est donc essentiel de développer des modèles de TR qui utilisent ces informations 3D.

Cette thèse est axée sur 1) le développement d'un modèle de TR 3D basé sur les techniques récentes de suivi de rayons et 2) la récupération de la densité volumique 3D des feuilles (LVD: leaf volume density) pour construire des scènes forestières. Elle présente le développement du modèle TR 3D LESS (**L**arg**E-S**cale remote sensing data and image **S**imulation framework), basé sur les algorithmes de lancer de rayons. Ainsi, LESS simule des données BRF multi-spectrales avec une méthode de lancer de photons virtuels (*i.e.*, pondérés) en mode "direct" (*i.e.*, photons lancés depuis la source) et des images de capteur à petite échelle (*e.g.*, images "fisheye") et grande échelle (*e.g.*, 1 km^2) avec une méthode de lancer de photons en mode "inverse". Le mode inverse est aussi utilisé pour simuler des images infrarouges thermiques et le TR atmosphérique. Une comparaison entre modèles et une comparaison avec des mesures "terrain" ont démontré l'efficacité et la précision du

modèle LESS. De plus, une interface utilisateur graphique conviviale permet de saisir les paramètres d'entrée, et aussi de construire et de visualiser les scènes 3D.
La reconstruction de scènes 3D de forêt est réalisée avec un jeu de données LiDAR simulées avec le modèle de TR anisotrope discret (DART), afin d'évaluer diverses approches d'obtention du LVD à partir de données LiDAR aéroportées. Dans un premier temps, un schéma de structuration de scène hybride a été conçu pour accélérer DART dans a été développé pour permettre la réalisation d'études de sensibilité des paramètres nécessitant un grand nombre de simulations. Une méthode basée sur l'intensité LiDAR permet de calculer le LVD. Elle utilise uniquement les retours au sol des impulsions LiDAR sans hypothèse concernant l'architecture de la canopée, ce qui améliore considérablement l'adaptabilité de cette méthode. La scène forestière 3D est reconstruite via une représentation sous forme de voxels, adaptée au modèle LESS, ce qui a permis de vérifier que les images LESS correspondent bien aux images hyperspectrales disponibles.

LESS peut être utilisé pour valider d'autres modèles physiques et pour entrainer des réseaux neuronaux. Cette thèse apporte une solution pour lier la modélisation 3D du TR et l'architecture 3D des paysages.

Table of Contents

Acknowledgement ..... **V**
List of publications ..... **VII**
Abstract ..... **IX**
Résumé ..... **XI**
Table of Contents ..... **XIII**
List of Figures ..... **XVII**
List of Tables ..... **XX**
List of Acronyms ..... **XXI**
Chapter 1 Introduction **1**
1.1 Motivation 2
1.2 Scope and objectives 4
1.3 Outline of the thesis 5
Chapter 2 LESS: Ray-tracing based 3D radiative transfer model **7**
2.1 Research context 8
2.2 General framework of LESS 10
2.3 Fundamentals of ray-tracing 11
2.4 3D scene description of LESS 13
2.4.1 Geometrical description of 3D scene 13
2.4.2 Ray intersection with 3D scene 14
2.5 Forward photon tracing 16
2.5.1 Real photon tracing algorithm 16
2.5.2 Virtual photon tracing algorithm 18
2.6 Backward path tracing 19
2.6.1 First-order scattering 20
2.6.2 Multiple scattering 21
2.7 Thermal infrared image simulation 22
2.8 Downward solar radiation simulation in rugged terrain 23

2.9 Atmosphere simulation with backward path tracing 24
2.9.1 Plane-parallel atmosphere model 24
2.9.2 Radiative transfer in participating media 25
2.9.3 Radiative transfer in plane-parallel atmosphere 27
2.10 Implementation and extension of LESS 29
2.11 Concluding remarks 31
Chapter 3 Accuracy evaluation of LESS 33
3.1 BRF validation 34
3.1.1 Model intercomparison 34
3.1.2 Model validation with field measurements 36
3.2 Image simulation of realistic forest stand 38
3.3 Thermal infrared simulation 41
3.4 FPAR simulation 42
3.5 Downward solar radiation in complex rugged terrain 44
3.6 Atmosphere simulation 45
3.7 Concluding remarks 46
Chapter 4 Hybrid scene structuring for accelerating 3D radiative transfer 47
4.1 Research context 49
4.2 Uniform grid in DART 50
4.3 Hybrid scene structuring 52
4.3.1 Voxel-level ray tracking 52
4.3.2 Within-voxel ray tracking 54
4.3.3 Implementation 54
4.4 Radiation tracking 56
4.5 Experiments 59
4.5.1 BASEL city scene 59
4.5.2 RAMI forest scene 60
4.6 Results and Discussion 61
4.7 Concluding remarks 63
Chapter 5 Physical interpretation of leaf area index from LiDAR data 65
5.1 Research context 67
5.2 Theoretical background 69
5.2.1 LiDAR pulse 69
5.2.2 Point cloud from Gaussian Decomposition 70

5.2.3 DART workflow of point cloud modeling 72
5.3 Review of the methods in LPI estimation 73
5.3.1 Point number based (PNB) methods 74
5.3.2 Intensity based (IB) methods 75
5.4 Comparative study of LPI/LAI estimation approaches using DART 76
5.4.1 Homogeneous scene 77
5.4.2 Heterogeneous scene 78
5.5 Results and analyses 79
5.5.1 Homogeneous scene 79
5.5.2 Heterogeneous scene 84
5.6 Influences of varied leaf dimensions 87
5.7 Influences of instrumental detection threshold 89
5.8 Further discussion and perspectives 91
5.9 Concluding remarks 94
Chapter 6 Voxel-based reconstruction and simulation of 3D forest scene 95
6.1 Research context 96
6.2 Study area and material 98
6.2.1 Study area 98
6.2.2 ALS and TLS dataset 98
6.2.3 Optical dataset 99
6.3 Voxel-based 3D PVD inversion 99
6.3.1 Single voxel PVD 99
6.3.2 Incident energy at each return 100
6.3.3 Ray traversal in voxels 102
6.3.4 Accuracy Validation 104
6.3.5 Discussion 109
6.4 3D forest scene reconstruction and simulation 112
6.4.1 3D landscape reconstruction 112
6.4.2 Spectral image simulation and comparison 113
6.5 Concluding remarks 116
Chapter 7 Conclusions and perspectives 117
7.1 Major conclusions 118
7.2 Problems and perspectives 119
Bibliography 121

Appendix 131

List of Figures

Figure 1.1: Modeling and inversion scheme in remote sensing. 3
Figure 2.1: Framework architecture of LESS. 11
Figure 2.2: ray-tracing process. 12
Figure 2.3: Forest described with triangle mesh in LESS. 13
Figure 2.4: Storage of triangle mesh in LESS. 14
Figure 2.5: Acceleration structure for the ray-object intersection. 15
Figure 2.6: Complex forest with many single trees. 15
Figure 2.7: Forward photon tracing. 17
Figure 2.8: Unit hemisphere partition. The hemisphere is projected onto the horizontal plane as a disk using equal area projection. 18
Figure 2.9: Virtual photon approach to calculate BRF. 19
Figure 2.10: Scattering calculation in backward path tracing. 21
Figure 2.11: Backward path tracing and multiple scattering calculation. 22
Figure 2.12: Thermal infrared image simulation using backward path tracing. 23
Figure 2.13: Solar radiation in rugged terrain. 24
Figure 2.14: Structures of plane parallel atmosphere. 25
Figure 2.15: Radiative transfer in participant media. 26
Figure 2.16: Simulating radiative transfer in atmosphere with backward path tracing. 28
Figure 2.17: Free path sampling in plane parallel atmosphere. 29
Figure 2.18: Implementation of a circular fisheye camera. (a) The projection diagram of a circular fisheye camera; (b) An example of simulated fisheye image using LESS. 30
Figure 3.1: Comparison of LESS and RAMI BRFs for a discrete floating sphere scene. 35
Figure 3.2: Different simulation resolution of the hybrid scene. 36
Figure 3.3: Comparison of LESS and RAMI BRFs for a hybrid spherical and cylindrical scene. 36
Figure 3.4: Comparison with field measurements and RAPID model. (a) red, principal plane; (b) red, cross-principal plane; (c) NIR, principal plane; (d) NIR, cross-principal plane.

.....38
Figure 3.5: Scene description of HET09_JBS_SUM. 39
Figure 3.6: Pixel-wise comparison between LESS and DART. 40
Figure 3.7: Comparison of simulated directional BTs over homogeneous scenes. 41
Figure 3.8: Optical properties of crops and ground: (a) Winter wheat； (b) Corn. 43
Figure 3.9: 3D display of the winter wheat and corn: (a) Winter wheat; (b) Corn. 43
Figure 3.10: Simulated FPAR and field-measured FPAR profiles. 43
Figure 3.11: The study area for downward solar radiation validation. 44
Figure 3.12: Downward solar radiation validation with field measurements. 45
Figure 3.13: Downward direct solar radiation and atmosphere radiation. 46
Figure 4.1: Radiative transfer modeling in the "Earth-Atmosphere" scene of DART. The scene of DART is divided into two parts: Earth and Atmosphere. 51
Figure 4.2: Space subdivision and data structure. 52
Figure 4.3: The proposed hybrid scene structuring scheme. 53
Figure 4.4: Simulation time varies with the number of triangles in one voxel. 54
Figure 4.5: Radiation tracking in non-empty voxels. 58
Figure 4.6: The BASEL scene under different voxel resolution. 60
Figure 4.7: The tree positions and 3D structures of the RAMI forest scene. 61
Figure 4.8: Accuracy comparison of the hybrid and uniform approaches when simulating RS images of Basel at three spatial resolutions: 1 m (a), 0.5 m (b) and 0.25 m (c). 61
Figure 4.9: Accuracy comparison of the hybrid and uniform approaches when simulating RS images of RAMI forest scene at three spatial resolutions: 0.5m (a), 0.25m (b) and 0.1 (c). 62
Figure 5.1: Workflow of DART outputs: discrete points in text format or discrete points with associated waveforms in LAS format. 72
Figure 5.2: Illustration of 3D Homogeneous test scene 78
Figure 5.3: The heterogeneous scene. It is made of randomly distributed trees with ellipsoidal and conical crowns. 79
Figure 5.4: LPI estimated by different methods with various LAI and footprint diameter. .82
Figure 5.5: Average discrete returns for all pulses. 82
Figure 5.6: The relationship between LiDAR derived LPI and Reference LPI. 83
Figure 5.7: Reference LAI plotted against $\ln LPI^{-1}$. 84
Figure 5.8: Reference LAI plotted against $\ln LPI^{-1}$ for a scene with discrete tree crowns. ...87
Figure 5.9: The accuracy of LiDAR-derived LPI against footprint size and leaf dimension.

.....89
Figure 5.10: LPI estimated by different methods with filtered point cloud 90
Figure 5.11: Average discrete returns after filtering for the homogeneous scene. 90
Figure 5.12: Reference LAI plotted against lnLPI^{-1} for heterogeneous scene after filtering. 91
Figure 6.1: Voxel PVD inversion using pulse transmittance.....100
Figure 6.2: Incident energy calculation for all echoes in a pulse. 102
Figure 6.3: Ray traversal in voxels.....104
Figure 6.4: Comparison between estimated PVD and reference PVD.....105
Figure 6.5: Voxel-based 3D distribution of PVD. 106
Figure 6.6: 2D profile of the PAD distribution and its corresponding LiDAR measurements. 107
Figure 6.7: Comparison of PAD map with established method. 107
Figure 6.8: An illustration of the alignment between ALS and TLS data 108
Figure 6.9: 3D voxel of plot L9: (a) ALS; (b) TLS; (c) Vertical profile of the PVD 109
Figure 6.10: Top of view of the horizontal PAI distribution for bottom (0-10 m), middle (10–20 m) and top (20–30 m) layers. The PAI is integrated vertically in each layer for each column. 109
Figure 6.11: Number of pulses traverse the voxel. 110
Figure 6.12: Distance between the vegetation-ground pulse to the nearest pure-ground pulse (a) and intensity distribution of pure-ground pulses (b).....111
Figure 6.13: Standard deviation of intensity in each pixel with different resolutions.....111
Figure 6.14: Base voxels with different LVD.....112
Figure 6.15: Reconstruction of a 100 m×100 m subplot scene from LiDAR point cloud. 113
Figure 6.16: Comparison between LESS simulated image and the field-measured (AISA) hyperspectral image.....115
Figure 6.17: Differences between LESS simulated image and AISA image in a resolution of 25 m. 115

List of Tables

Table 3.1: Structural and optical properties of the OA forest site. 37
Table 3.2: Optical properties of landscape elements 40
Table 3.3: Computational resources usage (for the resolution of 0.2 m)..... 41
Table 3.4: structural properties of winter wheat and corn 43
Table 3.5: Parameters for atmosphere simulation..... 46
Table 4.1: Computational resources used by simulating BASEL scene..... 62
Table 4.2: Computational resources used by simulating RAMI forest scene..... 63
Table 5.1: LiDAR parameters used in DART simulations. 77
Table 5.2: Fitted parameter c of Eq. (5-23) and R-squared value..... 86
Table 6.1: LiDAR parameters for DART simulations 104

List of Acronyms

ALS	Airborne Laser Scanning
BOA	Bottom of Atmosphere
BPT	Backward Path Tracing
BRDF	Bidirectional Reflectance Distribution Function
BRF	Bidirectional Reflectance Factor
BSDF	Bidirectional Scattering Distribution Function
BT	Brightness Temperature
BTDF	Bidirectional Transmittance Distribution Function
BVH	Bounding Volume Hierarchy
CDF	Cumulative Distribution Function
CSF	Cloth Simulation Filter
DART	Discrete Anisotropic Tadiative Transfer
DSR	Downward Solar Radiation
FPAR	Photosynthetically Active Radiation
FPT	Forward Photon Tracing
GD	Gaussian Decomposition
GUI	Graphic User Interface
ICP	Iterative Closest Point
IN	Intensity Based
LAD	Leaf Angle Distribution
LAI	Leaf Area Index
LiDAR	Light Detection and Ranging
LOD	Leaf Orientation Distribution
LPI	Laser Penetration Index
LUT	Look Up Table
LVD	Leaf Volume Density
PAD	Plant Aarea Density
PAI	Plant Area Density
PNB	Point Number Based
PVD	Plant Volume Density
RAMI	Radiation Transfer Model Intercomparison
RT	Radiative Transfer
SE	Surface Element

TLS	Terrestrial Laser Scanning
TOC	Top of Canopy
UEB	Urban Energy Budget
USR	Upward Solar Radiation

Chapter 1

Introduction

Summary

1.1 Motivation 2
1.2 Scope and objectives 4
1.3 Outline of the thesis 5

1.1 Motivation

Forests cover approximately 31% of the land surface across the globe and play a prominent role in the global carbon cycle (Mitchard, 2018; Schlamadinger and Marland, 1996; Winjum et al., 1992; Xie et al., 2008). Forests are highly complex and dynamic ecosystems that comprise various species and a large number of individual trees(Arnold et al., 2011; Bass et al., 2001). Besides, forests are an important natural resource, which is widely managed for biodiversity protection, wildlife habitat conservation, forest products and recreation (Twery and Weiskittel, 2013). Therefore, monitoring the status of the forests across the globe has great importance for understanding, utilizing and protecting the forests.

However, human activities (e.g., deforestation) have put great pressure on the environment, which changes the land cover significantly, and then influences the regional and global climatic system. These changes and the resulting effects need to be accessed regionally, as well as globally (Govaerts, 1996).

Currently, satellite remote sensing is the only technology that allows one to monitor large land surface areas with long-term observations. Through the interpretation of satellite observations, we can assess the structure, distribution and functionalities of vegetation canopy either qualitatively or quantitatively. However, optical sensors aboard satellite platforms can only acquire data from a limited number of viewing directions and spectral bands although their signals depend on many factors such as the optical properties of the observed Earth surfaces, the atmosphere, the sensor spectral bands and viewing directions (Zhang et al., 2017), etc. Understanding radiation interaction with the observed Earth surfaces and the atmosphere is essential to retrieve Earth surface information from satellite observation data. For vegetation covers, this understanding is essential for inferring canopy status such as photosynthetically active radiation (PAR).

Through the interpretation of remotely sensed electromagnetic signals (reflected or emitted) to infer the surface status is the “inversion problem” in remote sensing (**Figure 1**). In general, an inversion procedure is used to determine the input parameters of a model so that model outputs match the available measurements. Ideally, the model is a mathematical relationship. However, due to their complexity, surface structures are usually difficult to be described with simple statistical parameters. It explains why structures are usually simplified in remote sensing models. For example, a one-dimensional (1D) radiative transfer (RT) model simulates a vegetation canopy as horizontal layers with randomly distributed leaves. Based on this simplification, many 1D models, partly derived from atmospheric RT theories, have been established (Kuusk, 2018) to

formulate relationships between bidirectional reflectance factor (BRF) and vegetation parameters (e.g., leaf area index (LAI), leaf angle distribution (LAD), leaf optical properties). Up to now, 1D RT models (e.g., SAIL, SCOPE) have been widely used in remote sensing for parameter retrieval. However, due to their high degree of abstraction of the Earth surfaces, 1D models are usually too inaccurate. For example, they cannot consider the large gaps between crowns and trees with different heights. It explains that three-dimensional (3D) RT models, which can present complex heterogeneous landscapes, are needed for accurate inversion of remote sensing data.

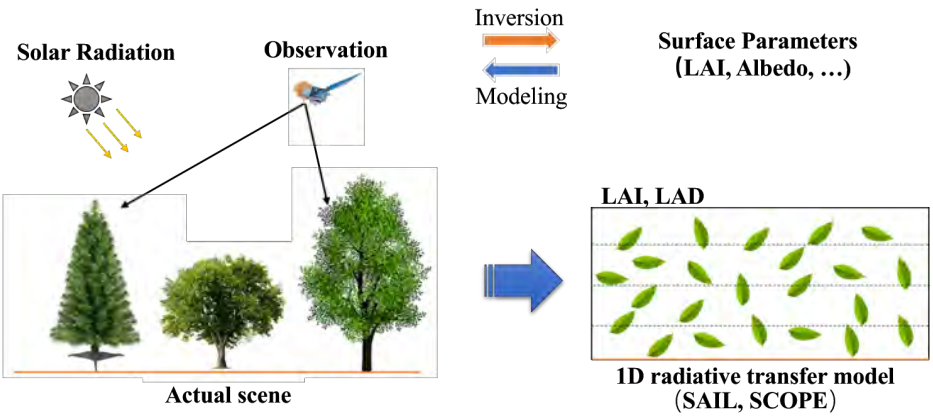


Figure 1.1: Modeling and inversion scheme in remote sensing.

3D RT models use schematic geometric objects (e.g., ellipsoid), triangle mesh and / or voxels with turbid medium to describe 3D scenes. Ray-tracing or radiosity methods are commonly used to solve the RT equation (Disney et al., 2000; Gastellu-Etchegorry et al., 2004; Huang et al., 2013; Qi et al., 2019). They can simulate remote sensing data under arbitrary conditions, which is essential for linking remote sensing signals and realistic surface structures. It allows one to maximally use the information from multi-source remote sensing data and ease the “ill-posed” problem that has confronted remote sensing inversion for many years (Liang et al., 2016).. Nowadays, the potential of 3D RT models increases with the increasing availability of 3D data from various new sensors (e.g., LiDAR: light detection and ranging), provided that 3D RT models can use these 3D data.

However, developing an efficient 3D radiative transfer model that can represent complex 3D landscapes is not an easy task due to the high heterogeneity of the Earth’s surface. To improve efficiency, current models usually work on small realistic scenes or simplify structures with simpler geometries. Although the computer graphics community provides the most accurate and efficient models (known as renderers), they were not designed specifically for performing scientific radiative transfer simulations. Thus, **an efficient and adaptable 3D RT model that**

works on large-scale and heterogeneous landscapes is important for RT modeling and is highly sought after by the remote sensing community.

Input data of 3D RT models are not always easy to obtain. Usually, realistic canopy structures, instead of simple statistical parameters, are needed. **Obtaining detailed vegetation structures over large areas is also an important aspect for successful canopy modeling works, since 3D structures are the key parameter for 3D radiative transfer model.** Due to the high complexity of vegetation canopies, traditional measurements, such as measuring LAI using LAI-2000, can give very coarse and two-dimensional (2D) parameters only, which is not compatible with 3D RT models. In recent years, LiDAR techniques, which can provide directly canopy 3D information, have been widely used to retrieve forest parameters and reconstruct 3D virtual scenes (Ackermann, 1999; Bailey and Mahaffee, 2017; Bailey and Ochoa, 2018; Bremer et al., 2017; Calders et al., 2018; Côté et al., 2009; Hancock et al., 2017; Müller-Linow et al., 2015; Qi et al., 2016). This is also one of the major topics of this thesis.

1.2 Scope and objectives

A number of problems associated to canopy RT modeling must be solved. “How to build an efficient 3D RT model that can handle large-scale forest landscapes? How to design the RT model to make it easily adaptable to various kinds of remote sensing data? How to validate the RT model? How can complex forest canopy be parameterized and input into the 3D RT model? What are the most effective parameters of the model?” etc. This thesis brings answers to these questions. For that, the objective has been 1) to develop a ray-tracing based 3D RT model (LESS: (**L**arg**E**-**S**cale remote sensing data and image **S**imulation framework) that takes full advantage of the most advanced light transport algorithms of the computer graphics community and 2) to propose a voxel-based canopy parameterization scheme by using airborne LiDAR data.

For the development of 3D RT model, this thesis makes full use of the forward and backward ray-tracing modes to simulate different remote sensing data. Specifically, a forward photon tracing (FPT) is proposed to simulate bidirectional reflectance factor (BRF) and energy-balanced related data, e.g., downward and upward solar radiation (DSR/USR) in rugged terrain and incident fraction of photosynthetically active radiation (FPAR). In this mode, a virtual photon approach is used to accelerate BRF simulation with fewer photons. The backward path tracing (BPT) is used for simulating images; it only simulates the energy that exactly goes into the sensor, which saves

a lot of computation time. The BPT is also extended to simulate thermal infrared images with given temperature distribution and to simulate the RT of a plane-parallel atmosphere.

For the 3D canopy parameterization, we first studied the estimation of laser penetration index (LPI)/leaf area index (LAI) by using simulated airborne LiDAR data from the DART model (Discrete Anisotropic Radiative Transfer). In a first step, DART was accelerated with a specifically designed hybrid scene structure scheme, in order to speed up the simulation of multiple pulse LiDAR over large areas with many vegetation elements. The designed acceleration approach makes it possible to do sensitivity analysis with many DART simulations. It relies on two complementary methods: (1) removal of empty voxels from DART's regular grids, using a bounding volume hierarchy (BVH), and (2) acceleration of ray-triangle intersection. Using the DART simulated dataset, we quantitatively analyzed several LPI/LAI estimation methods and found that the intensity-based method is the most appropriate one, which is then used to estimate 3D leaf volume density (LVD). Finally, the estimated LVD is used to construct a forest scene and is input into the proposed 3D RT model (LESS) to simulate spectral images.

1.3 Outline of the thesis

Based on the objectives of this thesis, the chapters are organized as follows:

The first chapter introduces the fundamentals of the ray-tracing based 3D RT LESS model. The forward photon tracing and backward path tracing are presented. These modes were extended for simulating DSR/USR, FPAR and thermal infrared radiation. The use of the backward path tracing is also used for simulating RT in a plane-parallel atmosphere.

Chapter 2 is focused on the accuracy of LESS products. Because field BRF data are difficult to obtain, the major validation scheme is cross-validation with other RT models for the case of a few heterogeneous canopies from the RAMI model inter-comparison experiment (http://rami-benchmark.jrc.ec.europa.eu/HTML/). The RAMI web site stores a few vegetation scenes and reference BRF values of these scenes. Because the RAMI web site does not store BRF images, the validation of LESS images is done with DART images of a realistic and complex forest scene. LESS FPAR and DSR are validated with field measurements. For atmosphere, LESS is compared to DART and MODTRAN (http://modtran.spectral.com/).

Chapter 3 presents a hybrid scene structuring scheme that was designed in order to accelerate DART. For that, the uniform grid approach of DART is replaced by an efficient data structure. The improvement is shown for a city and a forest scene, both simulated with many small triangles.

Chapter 4 assesses several LPI/LAI inversion approaches, using the optimized DART model with a homogeneous and a heterogeneous canopy. The modeling approach and the implementation that convert full waveform data to discrete points are detailed.

Chapter 5 introduces a voxel-based 3D forest scene reconstruction approach. It utilizes the inverted 3D LVD (cf. chapter 4) and the LESS model (cf. chapter 1). The comparison between LESS and airborne hyperspectral images of 1km forest scene are presented.

Chapter 6 concludes the thesis by summarizing the major conclusions and pointing out perspectives and issues for further researches.

The relationships between the above chapters are visualized in **Figure 2**.

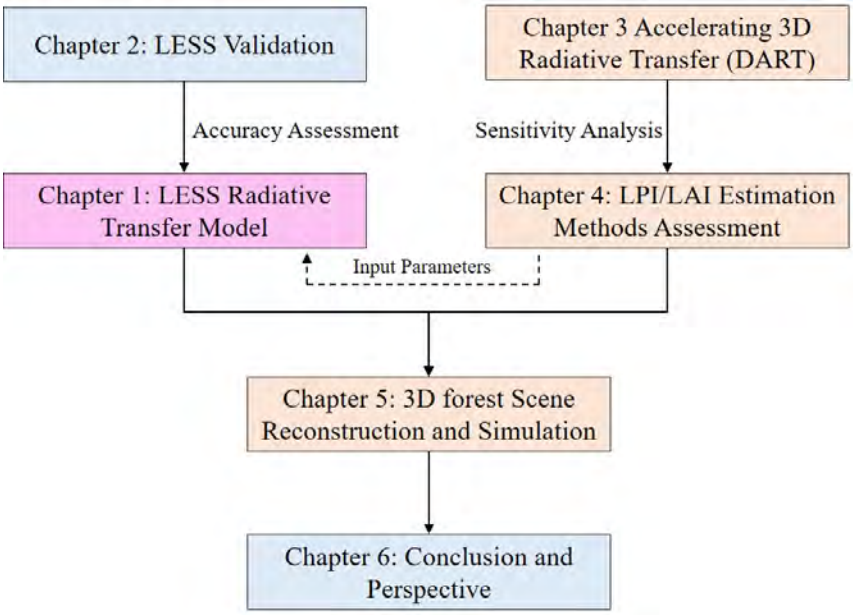


Figure 2: Relationships between different chapters

Chapter 2

LESS: Ray-tracing based 3D radiative transfer model

Summary

2.1 Research context 8
2.2 General framework of LESS 10
2.3 Fundamentals of ray-tracing 11
2.4 3D scene description of LESS 13
2.4.1 Geometrical description of 3D scene 13
2.4.2 Ray intersection with 3D scene 14
2.5 Forward photon tracing 16
2.5.1 Real photon tracing algorithm 16
2.5.2 Virtual photon tracing algorithm 18
2.6 Backward path tracing 19
2.6.1 First-order scattering 20
2.6.2 Multiple scattering 21
2.7 Thermal infrared image simulation 22
2.8 Downward solar radiation simulation in rugged terrain 23
2.9 Atmosphere simulation with backward path tracing 24
2.9.1 Plane-parallel atmosphere model 24
2.9.2 Radiative transfer in participating media 25
2.9.3 Radiative transfer in plane-parallel atmosphere 27
2.10 Implementation and extension of LESS 29
2.11 Concluding remarks 31

This chapter describes the ray-tracing based 3D RT model LESS (LargE-Scale remote sensing data and image Simulation framework) that I developed. This model employs 2 methods. (1) A forward photon tracing method to simulate multispectral bidirectional reflectance factor (BRF) or radiative budget (e.g., downward solar radiation). Photons are weighted simulate more accurate BRF with fewer photons. (2) A backward path tracing method to generate sensor images (e.g., fisheye images) or large-scale (e.g. 1 km^2) spectral images. It has been extended to simulate thermal infrared radiation by using an on-the-fly computation of the sunlit and shaded scene components. By extracting information from the photon trajectory, several kinds of remote sensing data, such as the fraction of photosynthetically active radiation (FPAR) and upward/downward radiation over rugged terrain, are simulated. Besides, the backward path tracing has been adapted to simulate atmosphere RT, which enables LESS to simulate a broad range of remote sensing datasets that can be used as benchmarks for various remote sensing applications (forestry, photogrammetry, etc.).

The chapter is presented in the paper:

"**Qi, J.**, Xie, D., Yin, T., Yan, G., Gastellu-Etchegorry, J.-P., Li, L., Zhang, W., Mu, X., Norford, L.K., 2019. LESS: LargE-Scale remote sensing data and image simulation framework over heterogeneous 3D scenes. Remote Sensing of Environment 221, 695–706."

2.1 Research context

Several 3D RT models that work with rather realistic landscapes were designed during the past decades (Kuusk, 2018). Although they differ from each other significantly, their algorithms are usually classified into two approaches for solving RT equations: (i) Radiosity; (ii) Ray tracing.

Radiosity methods, e.g., DIANA (Goel et al., 1991), RGM (Qin and Gerstl, 2000) and RAPID (Huang et al., 2013), are adapted from thermal engineering. Surfaces are usually assumed to be lambertian and the outgoing radiation of each surface is equal to the sum of reflected, transmitted and emitted radiation. This equilibrium can be represented with an equation set, the solution of which gives the radiation distribution of the 3D scene. Borel et al. (1994) is one of the major contributors who applies the radiosity method to vegetation canopy modeling. The core of the radiosity method is to compute a "view factor" between any two scattering surfaces and to store it into a matrix that reaches an unmanageable dimension if the number of surfaces grows very large. To solve this problem, Huang et al. (2013) proposed the RAPID model, which uses porous objects

to represent tree crowns. This model enables RAPID to simulate large-scale landscapes. Although the view factor matrix computation is not very efficient, a major advantage of radiosity models is that BRF computation is very fast once the view factor matrix has been computed.

Ray tracing methods are more commonly used in 3D RT models, since they are more adaptive and scalable to complex scenes with many elements (Disney et al., 2000). Depending on objectives, ray tracing methods are implemented in “forward” or “backward” mode. In the forward mode, photons are traced from illumination sources to viewing directions, while backward mode traces rays from the sensor along the viewing direction to determine the 1st, 2nd, etc. scattering order points that contribute to the sensor. Forward mode is more suitable for calculating multi-angle BRFs simultaneously and radiative budget. For example, the DART model is entirely implemented in the forward mode (Gastellu-Etchegorry et al., 2015). Radiation is tracked along discretized directions (i.e, discrete ordinate method) and is collected if it leaves the simulated scene, which enables DART to simulate many products, such as BRF, radiative budget and photosynthetically active radiation (PAR), fluorescence, etc. Forward ray tracing is also used by Raytran (Govaerts and Verstraete, 1998), Rayspread (Widlowski et al., 2006), FLiES (Kobayashi and Iwabuchi, 2008) and FLIGHT (North, 1996). Raytran is a pure Monte-Carlo based model, which traces monochromic rays from light sources. In Raytran, virtual detectors collect rays above the scene and the BRF is estimated by the number of collected rays. Because it does not use any weighting mechanism (i.e., a ray is totally scattered or absorbed), the implementation of this model is relatively straightforward. However, it also makes Raytran less efficient, because many rays are usually necessary to produce a convergent result. When calculating BRF, the rays absorbed within the scene do not contribute to the detector, which is a waste of computation time. Besides, absorption and reflectance are wavelength dependent. Therefore, simulating multiband BRF is generally computationally intensive as new rays must be sent separately per band. The Rayspread model is an extension of Raytran as it introduces a *secondary ray* mechanism, which traces a series of rays towards the detector at each intersection point of the main photon trajectory. This approach saves a lot of time when calculating BRF because it uses a much smaller number of photons. To simulate multispectral data more efficiently, the librat model (Lewis, 1999) uses a “ray bundle” concept to simulate multiband BRF in a single ray path by updating the weight for each band according to the reflectance/transmittance at each intersected point.

However, despite its advantage in simulating multiple directional BRFs in a single simulation, forward tracing is usually less efficient in simulating a particular sensor image, mostly due to the tracking of energy that ultimately does not contribute to the simulated image. The efficiency is even lower when simulating very large scenes, of which only a small part is captured by the sensor.

The weakness of the forward mode, however, is the strength of the backward mode, since backward ray tracing traces only the rays that enter the sensor. It allows one to simulate sensor images of very large scenes with many landscape elements. DIRSIG (Goodenough and Brown, 2012) is a typical representative of this kind of model, using backward path tracing to estimate surface-leaving radiance. This makes DIRSIG very efficient for performing hardware design and evaluating sensor configurations. Therefore, the ability to simulate remotely sensed signals in both forward and backward modes is an important feature for a modern and adaptive RT model.

The algorithms described above (i.e., radiosity and ray tracing) are classically used in computer graphics, especially for image rendering. In the computer graphics community, open-source renderers such as PBRT (Pharr et al., 2016) and POV-Ray (Plachetka, 1998) have been developed for handling 3D scenes and tracing rays to find intersections, using optimized techniques such as the advanced acceleration data structures (e.g., bounding volume hierarchy (Mller and Fellner, 2000)) and the CPU-level optimizations (Purcell and Hanrahan, 2004) (e.g., Streaming SIMD Extensions 2 [SSE2]). However, these renderers are not universally applicable, because they are mostly focused on image rendering for human perception rather than on radiometric accuracy for scientific applications. For example, they usually work only with three broad spectral bands (RGB) or a few fixed spectral bands in the visible region, which is not adapted to hyperspectral data simulations. Developing novel remote sensing-specific models based on these renderers or reimplementation is a pragmatic solution. For instance, Auer et al. (2016) developed a 3D SAR simulator based on POV-Ray, which can simulate radar reflection effects of 3D objects. Recently, DIRSIG has been re-designed and re-implemented, to adopt the latest advances of light transport algorithms emerged in the computer graphics community (Goodenough and Brown, 2017). DART model has integrated the ray tracing engine Embree (Wald et al., 2014), which is a collection of high-performance ray tracing kernels, to accelerate RT simulations.

2.2 General framework of LESS

The design of an operational 3D RT model requires numerous developments such as 3D scene description, illumination conditions, sensor configurations and RT modeling (Disney et al., 2000). A framework that considers all these aspects is an ideal tool to give accessibility to 3D RT simulation for most users. The LESS modeling framework has 6 major modules (**Figure 2.1**):

- *Input Data Management* module. It manages all the input parameters, including 3D landscape elements (e.g., trees), optical properties, observation geometries, etc. It offers a GUI and a set

of Python scripts for efficient input of parameters. Landscape elements in LESS are represented with geometric primitives (e.g., ellipsoid and cylinders) or triangle meshes. They are usually created by third-party software, such as Onyx Tree.

- *3D Landscape Construction* module. It includes several sub-processors to convert different types of data into a format that the *RT Model* module can use. For example, the sub-module *Terrain Processor* converts the DEM (digital elevation model) image into a triangle mesh or a height field. Usually, a height field is more efficient in terms of computation time and memory than a triangle mesh (Tevs et al., 2008). The sub-module *Optical Data Processor* calculates the reflectance and transmittance of different landscape elements by interpolating spectra of the spectral database (provided by LESS or imported by users), according to the user-defined bands.
- *Visualization* module. It is intended for 2D and 3D displays of the simulated landscape. It helps users to interactively explore the created scene and allows them to verify the correctness of the constructed 3D landscape before the actual computation.
- *RT Model* module. It is the key module of the LESS framework. It uses a parallelized ray tracing method to simulate the interaction between the solar radiation and landscape elements based on the previously constructed 3D scene and the sensor configuration. It contains several RT models, e.g., forward photon tracing and backward path tracing, to meet different simulation purposes.
- *Parallel Computing* module. It offers the ability to run LESS on a *Local Server* or a *Cluster, which* enables LESS to simulate large-scale areas.
- *Products Processing* module. It contains tools to post-process LESS outputs. For example, radiance images, combined with sun / sky irradiance, can be converted into BRF images.

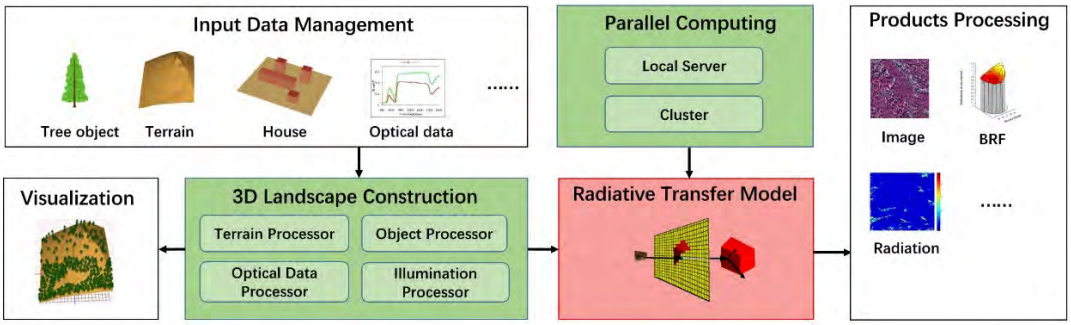


Figure 2.1: Framework architecture of LESS.

2.3 Fundamentals of ray-tracing

In computer graphics, ray-tracing is a rendering technique that generates 2D images from 3D scenes by sending rays and simulating their physical interactions with virtual objects. The radiance

leaving a point q in direction ω_o can be expressed with a rendering equation (Kajiya, 1986):

$$L_o(q,\omega_o) = L_e(q,\omega_o) + \int_{4\pi} f(q,\omega_i,\omega_o)\, L_i(q,\omega_i)|\cos\theta_i| d\omega_i \tag{2-1}$$

where $L_o(q,\omega_o)$ is the outgoing radiance from point q along direction ω_o; $f(q,\omega_i,\omega_o)$ is the Bidirectional Scattering Distribution Function (BSDF) of the intersected surface, which determines the outgoing radiance along direction ω_o at point q induced by incoming radiance along incident direction ω_i; $L_i(q,\omega_i)$ is the incoming radiance; θ_i is the angle between ω_i and the surface normal and $L_e(q,\omega_o)$ is an emission term (e.g., thermal emission). In short, the outgoing radiance is the sum of emitted and scattered radiance (**Figure 2.2**).

A complete ray-tracing process usually includes the following steps:

- **Geometrical description of the 3D scene**. The structures of the 3D scene should be explicitly expressed. Landscape elements are usually represented with geometric objects, triangle meshes and voxels with turbid media.
- **Generation of rays**. Rays are generated from light sources and propagate into the virtual scene. The sun and atmosphere are typical light sources. Some ray-tracing techniques send rays from sensors instead of light sources, which is more commonly used for simulating images only.
- **Intersection test between ray and scene**. Intersection test is the core of ray-tracing and also the most time-consuming process, since a scene may contain millions of elements and testing each ray for each element is impractical. To improve efficiency, some acceleration techniques are used, such as the bound volume hierarchy (BVH).
- **Radiance calculation**. By recording the energy change at each intersection point, the radiance recorded by sensors can be known explicitly. Besides, the ray trajectory also provides the possibilities to query other information, e.g., FPAR.

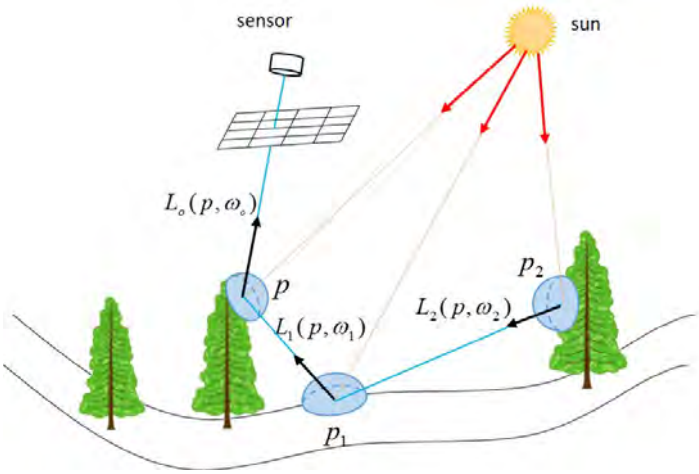


Figure 2.2: ray-tracing process.

2.4 3D scene description of LESS

2.4.1 Geometrical description of 3D scene

An accurate description of the 3D scene is the fundamental step for conducting reliable radiative transfer simulations. Generally, geometric objects are expressed with math equations, which could ease the ray-object intersection by solving analytic equations. However, it is usually used to represent some simple elements only, such as tree crowns. FLIGHT model (North, 1996) is a typical 3D model that uses geometric objects filled with turbid media to simulate 3D scenes. The concept "turbid medium" is from atmospheric science, but the difference is that turbid medium in remote sensing of vegetation is described with a group of parameters that are related to canopies, e.g., leaf volume density (LVD), LAD, leaf reflectance and transmittance.

Compared to geometric objects and turbid medium, a triangle mesh is more appropriate to accurately describe complex and heterogeneous vegetation canopies, since realistic canopies cannot be simply represented by statistical parameters. Triangle mesh uses millions of small triangles to represent the object's surfaces. Usually, the more the number of triangles is bigger and the better the representation of the original object, but it needs more computational resources.

LESS uses triangle mesh as its major scene description scheme for obtaining very accurate simulations. **Figure 2.3** shows a complex forest with more than 10 million triangles. It appears that each single leaf is composed of several small triangles in order to simulate complex leaf shapes.

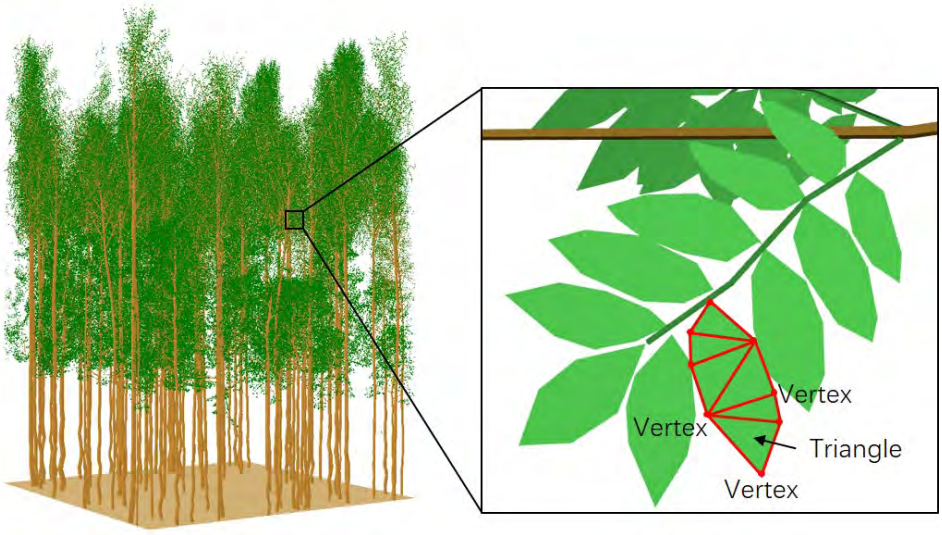


Figure 2.3: Forest described with triangle mesh in LESS.

A triangle has three vertices, thus, a natural storage approach is to store the three vertices per

triangle, which means that 3N vertices need to be stored if N triangles exist. It results in redundant information, since one vertex may belong to several triangles. A more efficient way is to store all vertices in an array, and to store the vertex indices of each triangle in another array. Hence, vertices XYZ are stored in a vertex array and vertex indices are stored in an index array (**Figure 2.4**).

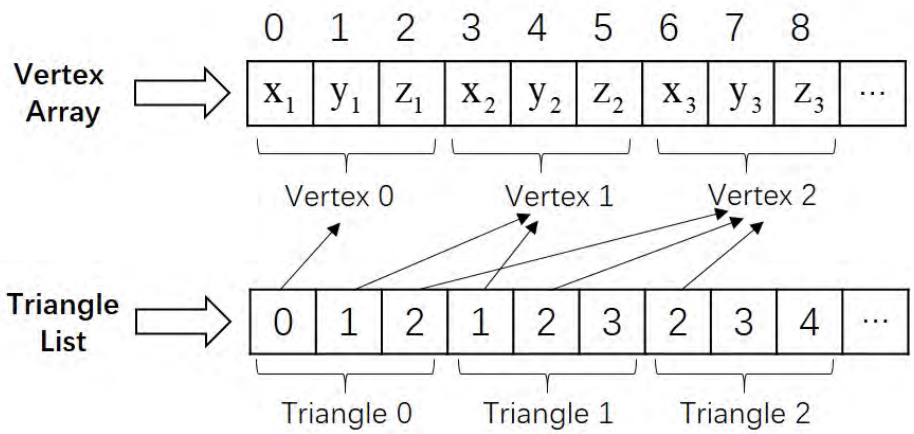


Figure 2.4: Storage of triangle mesh in LESS.

2.4.2 Ray intersection with 3D scene

A 3D scene may contain hundreds and thousands of objects, and each object may be composed of millions of small triangles. To find the intersection point of an incident ray, the intersected small triangle should first be found. The naive implementation is to test all the triangles in the scene one by one, which gives a time complexity of $O(n)$.

However, an incident ray may be intersected with objects that are close to its trajectory only. Triangles far from the raypath can be safely removed from the list of possible intersecting triangles. This selection / removal approach can be achieved with the so-called object bounding box. A bounding box is simply a cube that tightly surrounds the object (**Figure 2.5**). If a ray does not intersect with the bounding box, it does not intersect the object itself, and there is no need to test a possible intersection. Since the ray-box intersection is usually much efficient than ray-object intersection, this method can efficiently find the intersected object.

Despite the high efficiency of ray-box intersection, we still do not want to test all object bounding boxes with a ray one by one, especially for scenes with a large number of objects. Usually, an acceleration structure is used here to skip some boxes, which has been widely used in computer graphics, such as KD-Tree and BVH. For each intersected object, if it contains too many triangles, a second level of acceleration structure is usually used. Now, the basic unit is a single triangle.

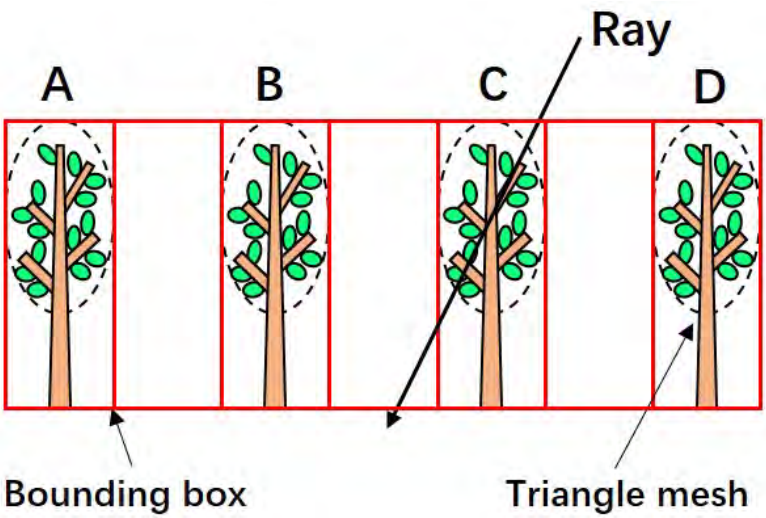


Figure 2.5: Acceleration structure for the ray-object intersection.

Let us consider a complex forest with many single trees, each one having many triangles. In **Figure 2.6**, the forest scene has more than 100,000 single trees, and each tree has more than 1 million triangles. If the triangle vertices are represented with float number, the memory used by this scene may exceed 1000 GB, which is impossible for most common computers. To handle this kind of situation, an “instancing” technique is commonly used. This method only stores several single trees in memory, and then “clones” them to different places by storing the transform matrix only for each instanced tree. Since the single tree is not actually copied, this method saves a lot of computer memory and time for copying data. Unfortunately, this approach is not adapted for simulating 3D radiative budget, at least per triangle.

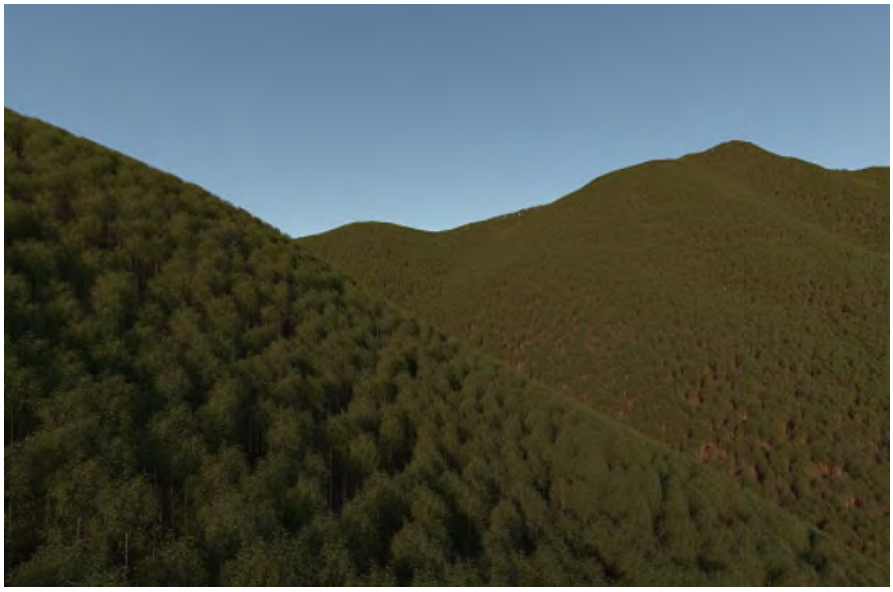


Figure 2.6: Complex forest with many single trees.

2.5 Forward photon tracing

2.5.1 Real photon tracing algorithm

Forward photon tracing (FPT) traces photon packets with the power $P(\lambda)$ into the scene from light sources. The initial power $P^0(\lambda)$ of each packet is determined by the power of light sources and the number N of generated packets. When generating photon packet in a scene with multiple light sources, a light source is randomly chosen according to the importance weight w_k, which is proportional to the power of each light source, i.e., $w_k = \frac{L_k(\lambda)}{\sum_{k=1}^{K} L_k(\lambda)}$ with $L_k(\lambda)$ being the power of light source k and K being the number of light sources. This mechanism guarantees that a light source with larger power has more sampled photon packets. The initial power of each packet, in terms of watt (W), is given as

$$P^0(\lambda) = \frac{\sum_{k=1}^{K} L_k(\lambda)}{N} \tag{2-2}$$

When a photon packet enters the scene along a path defined by its origin and direction of propagation, the occurrence of an intersection with landscape elements is tested. If an intersection occurs, the power of this packet is scaled according to the optical properties of the intersected surface, i.e., the reflectance or transmittance. For a packet with Q times of scattering before it escapes from the scene, the power becomes:

$$P^Q(\lambda) = P^0(\lambda) \cdot \prod_{q=1}^{Q} \left[\pi f(q, \omega_{i,} \omega_{o,} \lambda)/p_q\right] \tag{2-3}$$

where $f(q, \omega_{i,} \omega_{o,} \lambda)$ is the bidirectional scattering distribution function (BSDF) at the qth intersection point during its trajectory. ω_i and ω_o are the incident and outgoing directions of a photon packet, respectively. p_q is the probability that the photon is reflected or transmitted (e.g., 0.5). Since in LESS, surfaces are Lambertian, the BSDF is the bidirectional reflectance distribution function (BRDF) or bidirectional transmittance distribution function (BTDF), depending on relative configurations of $\omega_{i,}$, surface normal ω_n and ω_o.

$$f(q, \omega_i, \omega_o, \lambda) = \frac{1}{\pi} \begin{cases} \rho_{\perp,\lambda} \cdot \text{sgn}(\omega_n \cdot \omega_i) + \tau_\lambda \cdot \text{sgn}(-\omega_n \cdot \omega_i) \text{ if } \omega_o \cdot \omega_n \geq 0 \\ \rho_{\top,\lambda} \cdot \text{sgn}(-\omega_n \cdot \omega_i) + \tau_\lambda \cdot \text{sgn}(\omega_n \cdot \omega_i) \text{ if } \omega_o \cdot \omega_n < 0 \end{cases} \tag{2-4}$$

where $\text{sgn}(x) = \begin{cases} 1, & \text{x} \geq 0 \\ 0, & \text{x} < 0 \end{cases}$; $\frac{\rho_{\perp,\lambda}}{\pi}$ and $\frac{\rho_{\top,\lambda}}{\pi}$ are the upper and bottom surface BRDF, respectively; $\frac{\tau_\lambda}{\pi}$ is the BTDF. In LESS, the transmittances of the surface from both the upper and the bottom

side are assumed to be identical. The outgoing direction of a photon packet after scattering is determined by randomly sampling the BSDF function. For Lambertian surfaces, the model chooses a random direction in the outgoing hemisphere. Since a single photon trajectory can be used to simulate BRF for any wavelength by updating the power according to the spectral reflectance/transmittance, for simplicity, the symbol λ is omitted below.

A photon packet is collected by the sensor if it exits the scene through the scene top boundary. Lateral boundary effects are considered to simulate horizontally infinite scenes with a repetitive pattern. As shown in **Figure 2.7**, the photon packet which exits from the lateral boundaries will re-enter the scene from the opposite side with the same photon direction until it escapes through the top boundary of the scene. When the scattering order of a packet exceeds an user-defined threshold (e.g., 5), the propagation of the packet is randomly stopped according to the "Russian roulette" mechanism (Kobayashi and Iwabuchi, 2008), which terminates the trajectory of a packet with a probability p (e.g., 5%). If the packet survives, its power will be multiplied by $\frac{1}{1-p}$.

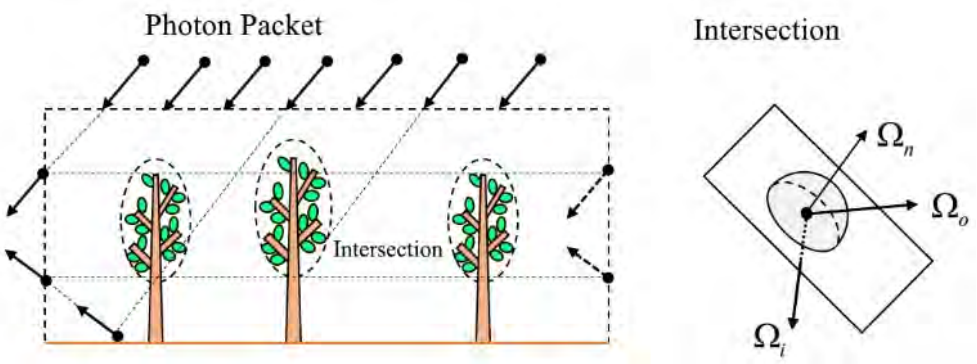


Figure 2.7: Forward photon tracing.

The collection of the escaped photon packets is achieved by placing a virtual hemisphere above the scene (Govaerts and Verstraete, 1998). The hemisphere (**Figure 2.8**) is partitioned into N_P surface elements (SE) with equal area $\Delta S = \frac{2\pi}{N_P}$, using the partition scheme of a disk via the equal area projection (i.e., $\Delta\Omega = \Delta S$). Zenith angles are defined (Beckers and Beckers, 2012) with:

$$\theta_i = \theta_{i-1} - \frac{2}{a_{aspect}} \sin\frac{\theta_{i-1}}{2} \sqrt{\frac{\pi}{k_{i-1}}}, k_i = k_{i-1}\left(\frac{r_i}{r_{i-1}}\right)^2 \tag{2-5}$$

where (θ_i, θ_{i-1}) defines a zenith range on the hemisphere with $\theta_0 = \frac{\pi}{2}$; k_i is the total number of SEs for a zenith angle θ_i with $k_0 = N_P$; r_i is the radius corresponding to θ_i with $r_i = 2\sin\frac{\theta_i}{2}$ for a unit sphere due to the equal area projection; a_{aspect} is the aspect ratio of each SE, which is

approximately enforced to 1. For each zenith range, it has $k_{i-1} - k_i$ SEs with equal area.

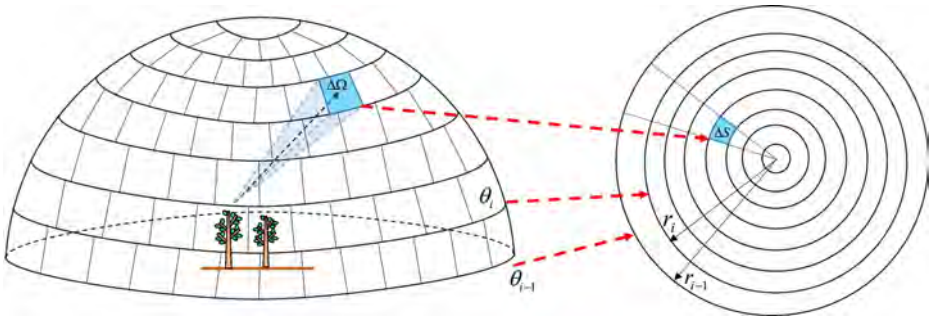


Figure 2.8: Unit hemisphere partition. The hemisphere is projected onto the horizontal plane as a disk using equal area projection.

When a photon packet exits the scene, the outgoing SE (solid angle) is determined by the photon direction only, i.e., the hemisphere is placed at an infinite position. The BRF in this SE can be estimated with (Govaerts and Verstraete, 1998):

$$f_{BRF_i} = \frac{\pi P_i^A}{\Delta\Omega_i \cdot \cos\theta_i^c \cdot P_{scene}} \tag{2-6}$$

where P_i^A is the power (watt) of all the captured photons in SE i, i.e., $P_i^A = \sum_{P^Q \in \Delta\Omega_i} P^Q$; $\Delta\Omega_i = \frac{2\pi}{N_P}$ is the solid angle of each SE; θ_i^c is the central zenith angle of solid angle $\Delta\Omega_i$; P_{scene} is the power of all the direct incident photon packets on a reference plane at the top of the scene, i.e., the incident radiation at the top of the scene. Once the power in each SE is determined, the scene albedo is computed as:

$$\omega_{albedo} = \frac{\sum_{i=1}^{N_P} P_i^A}{P_{scene}} \tag{2-7}$$

2.5.2 Virtual photon tracing algorithm

The real photon approach estimates BRF by using small SEs on the sphere. More photon packets are needed to reduce the variance when smaller SEs are used. To solve this problem, a virtual photon approach, similar to the virtual direction in DART model (Yin et al., 2015), secondary ray in Rayspread model (Widlowski et al., 2006) or some “local estimates” methods (Antyufeev and Marshak, 1990; Marchuk et al., 1980), is introduced. If a packet is intercepted by an object (e.g., a tree) in the scene without complete absorption, the packet will be scattered in a direction which is randomly sampled by the BSDF function, and a virtual photon packet will be

sent to each of the defined virtual directions. The possible scattered energy, in terms of intensity ($\mathrm{W \cdot sr^{-1}}$), is calculated as

$$I = V \cdot P^{q-1} \cdot f(q, \omega_i, \omega_v) \cdot \cos < \omega_v, \omega_n > \tag{2-8}$$

where P^{q-1} is the power of the incident photon packet at the qth intersection point along its trajectory; ω_v is a virtual direction; V is a visibility factor that is equal to zero if a landscape element occludes the virtual photon packet, and equal to 1 otherwise. When sending the occlusion testing rays, the lateral boundary effect is also considered (**Figure 2.9**). The final BRF is then:

$$f_{BRF_v} = \frac{\pi I_v^A}{\cos\theta_v \cdot P_{scene}} \tag{2-9}$$

where I_v^A is the power per unit solid angle ($\mathrm{W \cdot sr^{-1}}$) in virtual direction v and θ_v is the zenith angle of the virtual direction. An advantage of calculating a directional BRF using the virtual photon approach is that the BRF is estimated for an infinitely small solid angle (Thompson and Goel, 1998), which is the real directional BRF of a scene.

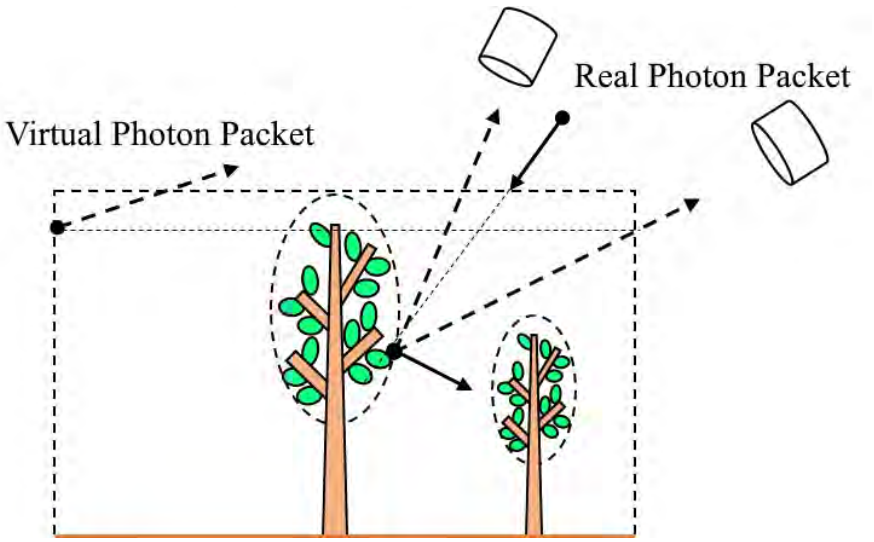


Figure 2.9: Virtual photon approach to calculate BRF.

2.6 Backward path tracing

Instead of tracing photon packets from light sources, backward path tracing sends rays from sensors into the scene. The ray directions are controlled by sensor configurations (field of view, position, orientation, etc.). The main task of this ray-tracing algorithm is to establish a connection, which is called “path”, between light sources and sensors and to determine the radiance incident onto the sensor.

2.6.1 First-order scattering

When a ray leaves the sensor and finds its first intersection point q with the virtual scene, the scattered radiation from direct incident radiation of light sources should be calculated. This is called first-order scattering. Since a scene may have both sun and atmosphere emitters, rays are randomly sent from them. When a ray reaches point q, a light source is randomly chosen between sun and atmosphere (**Figure 2.10**). Their probabilities are p_{dir} and p_{diff}, respectively. Usually, we can set $p_{dir} = p_{diff} = 0.5$ or set the value according to their emitting power, i.e., $w_k = \frac{L_k(\lambda)}{\sum_{k=1}^{K} L_k(\lambda)}$. According to the chosen light source, the scattering is calculated differently:

- **If sun has been chosen**. A ray is sent from q along negative sun direction and finds a point q'. If the ray bas been blocked between qq', then the sun does not directly contribute to point q. Otherwise, the incident irradiance from the sun can be calculated as $E_{dir}/p_{direction}^{sun}$, where E_{dir} is sun irradiance of a surface perpendicular to the sun direction and $p_{direction}^{sun}$ is the probability that the sun generates a ray in this direction. Since the sun direction is fixed, we have: $p_{direction}^{sun} = p_{dir} \cdot 1$.
- **If atmosphere has been chosen**. A random direction in the upper hemisphere is created and a ray is sent from q along this generated direction to find the intersection point q' with atmosphere, which is represented as a hemisphere in LESS. If this ray is not blocked, the atmosphere radiance is $I_{diff}/p_{direction}^{ats}$, where $p_{direction}^{ats}$ is the probability that the randomly generated ray is in this direction, i.e., $p_{direction}^{ats} = p_{diff}\frac{cos\theta_{ats}}{\pi}$ with θ_{ats} being the angle between surface normal and the generated ray.

The incident radiation will be scattered into direction ω_o, no matter which light source has been chosen. Since surfaces in LESS are Lambertian, thus, the possibility to generate a ray with specific direction θ can be expressed as $\frac{cos\theta}{\pi}$, i.e., $p_{bsdf}^{ats} = \frac{cos\theta_{ats}}{\pi}$ and $p_{bsdf}^{sun} = \frac{cos\theta_{sun}}{\pi}$. This means that we have sampled both the light sources and the BSDF functions, which is usually called multiple importance sampling. The combined probability is (Veach, 1997):

$$p_{weight}^{sun} = \frac{{p_{direction}^{sun}}^2}{{p_{direction}^{sun}}^2 + {p_{bsdf}^{sun}}^2}, \quad p_{weight}^{ats} = \frac{{p_{direction}^{ats}}^2}{{p_{direction}^{ats}}^2 + {p_{bsdf}^{ats}}^2} \tag{2-10}$$

The final contribution from the sun can be expressed as:

$$L_{sun} = \frac{E_{dir}}{p_{direction}^{sun}} \cdot p_{weight}^{sun} \cdot f(q, \omega_i, \omega_o) \cdot \cos\theta_{sun} = \frac{E_{dir}\cos\theta_{sun}}{p_{dir}} \cdot p_{weight}^{sun} \cdot f(q, \omega_i, \omega_o) \tag{2-11}$$

and the contribution from the atmosphere can be calculated as:

$$L_{ats} = \frac{I_{diff}}{p_{direction}^{ats}} \cdot p_{weight}^{ats} \cdot f(q, \omega_i, \omega_o) \cdot \cos\theta_{ats} = \frac{\pi I_{diff}}{p_{diff}} \cdot p_{weight}^{ats} \cdot f(q, \omega_i, \omega_o) \tag{2-12}$$

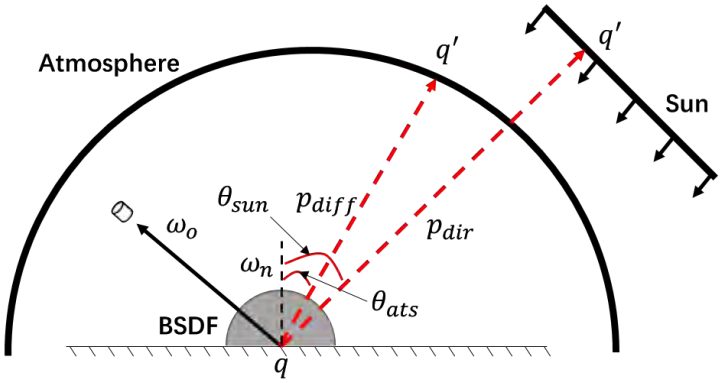


Figure 2.10: Scattering calculation in backward path tracing.

2.6.2 Multiple scattering

To calculate multiple scattering (**Figure 2.11**), a new ray is launched from point q with a direction that randomly samples the BSDF. If this ray intersects the scene at a point q_1, the same procedure for calculating radiation from light sources at point q is applied to q_1. The outgoing radiance at point q_1 along the randomly selected direction is the incoming radiance of q along direction ω_1. The multiple scattering procedure is performed recursively until reaching the user specified maximum scattering order (e.g., 5). To prevent energy loss due to the stop of scattering, the random cut-off technique ("Russian roulette") used in FPT is also applied to the sensor ray.

When simulating a horizontally infinite scene, the lateral boundary effect is considered for both the sensor ray and the illumination ray. At each intersected point (q_i), an illumination ray, which is built by randomly sampling a point q_e on the emitter, is sent towards the emitter. If this ray traverses the lateral boundary of the scene, it is also reintroduced into the scene to test whether the intersected point is occluded by other landscape elements or not.

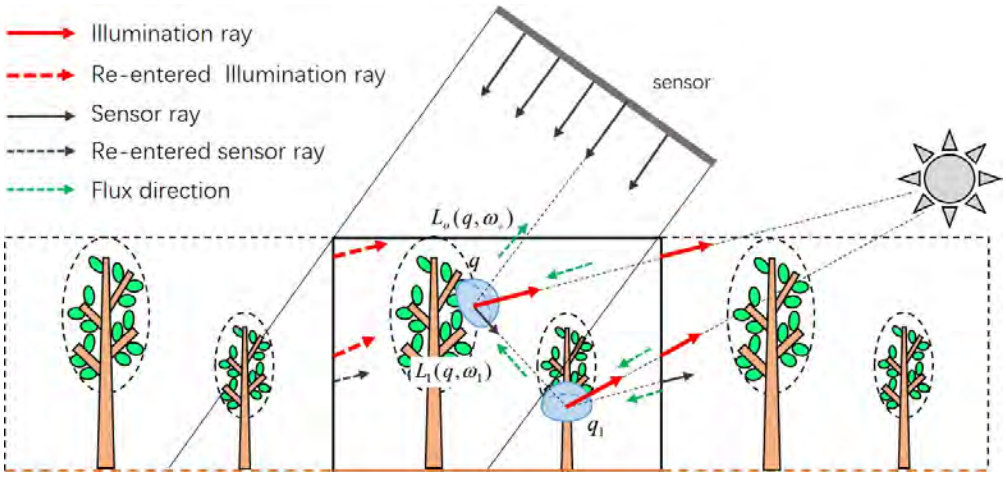


Figure 2.11: Backward path tracing and multiple scattering calculation.

2.7 Thermal infrared image simulation

When simulating thermal infrared radiation, the object itself, instead of the sun, becomes an “emitter”, which emits thermal radiation according to Planck's law and its emissivity. Thermal emission from the sun is neglected, which is usually the case for applications in the thermal infrared domain. However, the modeling is not adapted to wavelengths around 3-7µm where sun irradiance and Earth thermal exitance have the same order of magnitude. However, the presence of sun radiation will greatly influence the temperature distribution of objects due to the shadows cast between them. Scene elements are classified into four components with specific temperatures, i.e., sunlit soil, shaded soil, sunlit leaves and shaded leaves (**Figure 2.12**). The determination of these four components is computed on the fly instead of a precomputing step. This on-the-fly approach avoids the storage of emission points, which can greatly reduce the memory usage, especially for scenes with a large number of leaves. As a comparison DART pre-computes the temperature, which requires some extra computation, but allows to consider a continuous range of temperatures instead of 4 temperature values only.

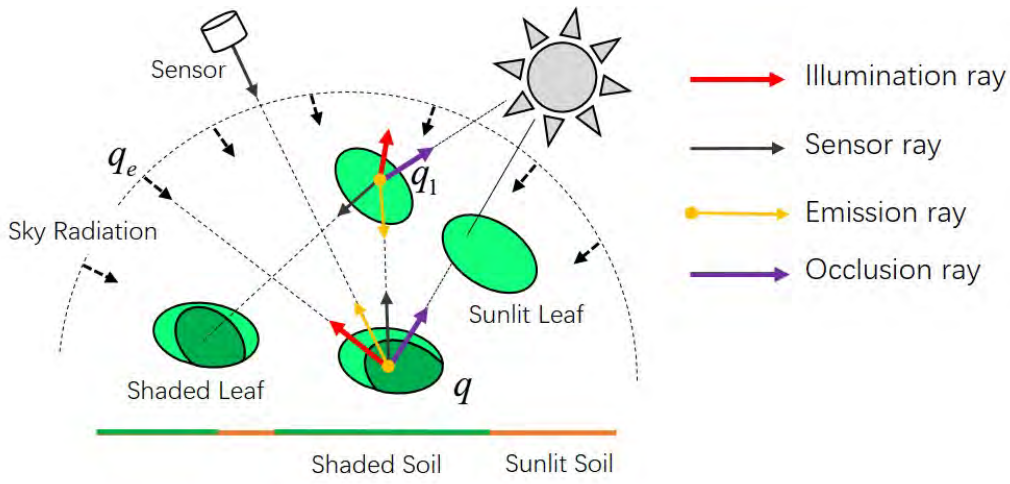


Figure 2.12: Thermal infrared image simulation using backward path tracing.

If a sensor ray is intersected in the scene (i.e., q in the scene), an emission term at this point is added. To determine the emission power, an occlusion ray is traced towards the sun. If this point is directly illuminated, i.e., the occlusion ray intersects nothing, the temperature of the sunlit component is used. Otherwise it uses the temperature of the shaded component. The emitted radiation is calculated by using Planck's law with local emissivity. In addition to the emitted power, a reflected power goes into the sensor. This is due to the reflexion of thermal emission by other objects in the scene or sky radiation. In order to consider this part of the power, a random point on a randomly selected emitter (including the sky) is sampled (e.g., q_e in **Figure 2.12**). When this point is not on the sky, the emission power is determined by sending an occlusion ray towards the sun. The contribution of power from this point is calculated by using the BSDF defined at point q if this point is not occluded by other objects. This procedure can be recursively repeated, which is the same as the procedure described in section 1.6 except for the emission term. Finally, a thermal infrared image, which records the radiance value, can be simulated.

2.8 Downward solar radiation simulation in rugged terrain

The terrain impacts local downward solar radiation (Yan et al., 2016), due to the change of incident angle between solar direction and local surface normal. Besides, surrounding terrains scatter radiation and also block some radiation from sun and sky (Wen et al., 2018). To quantitatively analyze these impacts, several models that simulate rugged terrains have been designed (Dozier and Frew, 1990; Li et al., 1999; Wang et al., 2005). Monte-Carlo ray-tracing models are usually the most accurate and are well adapted to simulate any complex terrains,

especially in presence of vegetation. LESS uses forward photon tracing to simulate the upward / downward solar radiation in complex terrains (**Figure 2.13**). The scene is first divided into regular grids, each of them representing a pixel, and solar irradiance is calculated per pixel during the simulation. If a photon that exits P_1 intersects P_2, the upward/downward solar radiation is calculated according to the spatial relationship between P_1 and P_2:

- If P_1 and P_2 belong to different pixels (here: Pixels 4 and 2), one records the upward solar radiation at exiting pixel (Pixel 4) and downward solar radiation at entering pixel (Pixel 2);
- If P_1 and P_2 belong to the same pixel. No radiation is needed to be recorded, since this is internal scattering in a pixel;
- If photon leaves from P_1 but without further intersection, we only record the upward solar radiation for the exiting pixel.

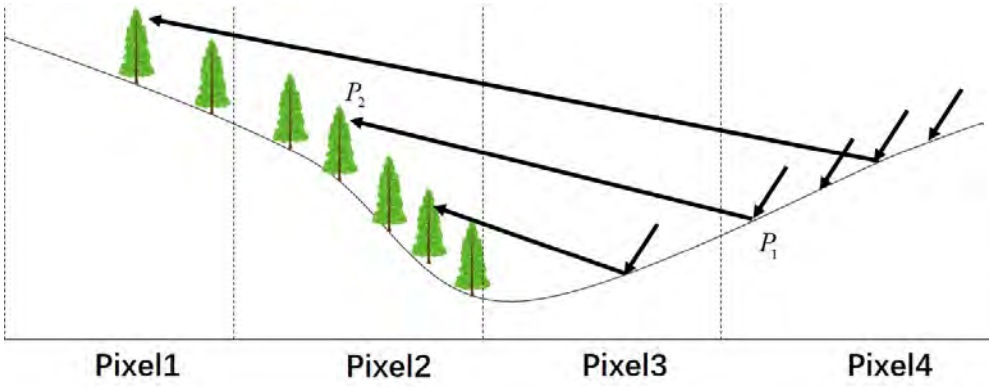


Figure 2.13: Solar radiation in rugged terrain.

2.9 Atmosphere simulation with backward path tracing

2.9.1 Plane-parallel atmosphere model

Approximately, atmosphere can be abstracted as being one-dimensional, but with different optical properties for each horizontal layer (**Figure 2.14**). Apart clouds, the atmosphere is made of gasses and aerosols with densities that decrease with altitude. For each layer, the optical properties of which can be described with 6 parameters: extinction coefficient of gases β_{gas}^{ext}, single-scattering albedo of gases ω_{gas}, scattering phase function of gases P_{gas}, extinction coefficient of aerosols $\beta_{aerosol}^{ext}$, single-scattering albedo of aerosols $\omega_{aerosol}$, scattering phase function of aerosols $P_{aerosol}$. During the simulation, these parameters are combined together in order to

compute weighted optical properties per layer:

$$\beta_{ats}^{ext} = \beta_{gas}^{ext} + \beta_{aerosol}^{ext} \tag{2-13}$$

$$\omega_{ats} = \frac{\beta_{gas}^{ext} \cdot \omega_{gas} + \beta_{aerosol}^{ext} \cdot \omega_{aerosol}}{\beta_{gas}^{ext} + \beta_{aerosol}^{ext}} \tag{2-14}$$

$$P_{ats}^{ext} = \frac{\beta_{gas}^{ext} \cdot \omega_{gas} \cdot P_{gas} + \beta_{aerosol}^{ext} \cdot \omega_{aerosol} \cdot P_{aerosol}}{\beta_{gas}^{ext} \cdot \omega_{gas} + \beta_{aerosol}^{ext} \cdot \omega_{aerosol}} \tag{2-15}$$

Gas scattering is simulated with the Rayleigh scattering phase function, while aerosol scattering is simulated with Henyey-Greenstein phase function (Henyey and Greenstein, 1941):

$$P_{gas}(\cos\theta) = \frac{3}{16\pi}(1 + \cos^2\theta) \tag{2-16}$$

$$P_{aerosol}(\cos\theta) = \frac{1}{4\pi}\frac{1 - g^2}{(1 + g^2 + 2g\cos\theta)^{3/2}} \tag{2-17}$$

where θ is the phase angle, which indicates the angle between incident ray and scattering ray; g is a anisotropy factor that ranges from -1 to 1. When g > 0, the dominant scattering is forward scattering, while for g < 0, backward scattering is dominant.

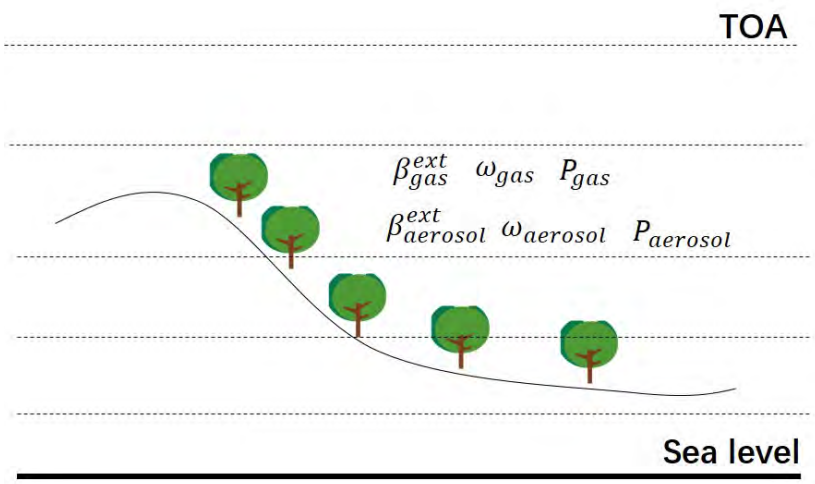


Figure 2.14: Structures of plane parallel atmosphere.

2.9.2 Radiative transfer in participating media

2.9.2.1 Fundamentals

During its propagation in participating media, light is absorbed or scattered by the particles in the media, which modifies light intensity. As illustrated in **Figure 2.15**, an incident radiation with radiance $L(0, \Omega)$ that enters the media at positon $t = 0$, becomes $L(d, \Omega)$ when it reaches

position $t = d$. The final radiance $L(d, \Omega)$ has three components: directly transmitted radiance from $L(0, \Omega)$, self-emitted radiance due to media along the ray path from $t = 0$ to $t = d$, and in-scattering radiance due to scattering of radiation from media outside the ray path:

- **Directly transmitted radiance**. It depends on direct transmittance $e^{\int_0^d -\beta(t)dt}$ over [0, d]:

$$L_{dir}(d, \Omega) = L(0, \Omega)e^{\int_0^d -\beta(t)dt} \tag{2-18}$$

  where $\beta(\mathrm{t})$ is the extinction coefficient at position t of the media. In the atmosphere, $\beta(\mathrm{t}) = \beta_{ats}^{ext}(t)$.

- **Self-emitted radiance**. At any position t, the media emits thermal radiance that is absorbed and scattered during its propagation to d.

$$L_e(d, \Omega) = \int_0^d \beta(\mathrm{t})[1 - \omega(\mathrm{t})].L_e(t, \Omega).e^{\int_t^d -\beta(t')dt'}dt \tag{2-19}$$

  where $\omega(\mathrm{t})$ is the single-scattering albedo at position t. $L_e(\mathrm{t}, \Omega)$ is the emitted radiance at positon t. In the atmosphere, $\omega(\mathrm{t}) = \omega_{ats}$.

- **In-scattering radiance**. It is the out-scattering energy of surrounding media that has been scattered in direction Ω. It is related to the phase function and single scattering albedo, i.e.,

$$L_s(d, \Omega) = \int_0^d \beta(\mathrm{t})\omega(\mathrm{t}) \int_{4\pi} P(t, \Omega, \Omega')L(t, \Omega')d\Omega' \, e^{\int_t^d -\beta(t')dt'}dt \tag{2-20}$$

  where $P(t, \Omega, \Omega')$ is the phase function at position t; $L(\mathrm{t}, \Omega')$ is the incident radiation in direction Ω' by surrounding media.

In short, the outgoing radiance is:

$$L(d, \Omega) = L(0, \Omega)T(d) + \int_0^d T(d - \mathrm{t})[\sigma_a(t)L_e(t, \Omega) + \sigma_s(t)L_{in}(t, \Omega)]dt \tag{2-21}$$

where $\sigma_a(t) = \beta(\mathrm{t})[1 - \omega(\mathrm{t})]$ is an emission coefficient; $\sigma_s(t) = \beta(\mathrm{t})\omega(\mathrm{t})$ is a scattering coefficient; $T(t) = e^{\int_0^t -\beta(t')dt'}$is the transmittance along [0,t]. For homogeneous media, which is not the case of atmosphere, the extinction coefficient $\beta(\mathrm{t})$ is a constant. $L_{in}(t, \Omega) = \int_{4\pi} P(t, \Omega, \Omega')L(t, \Omega')d\Omega'$ Then: $T(t) = e^{-\beta t}$.

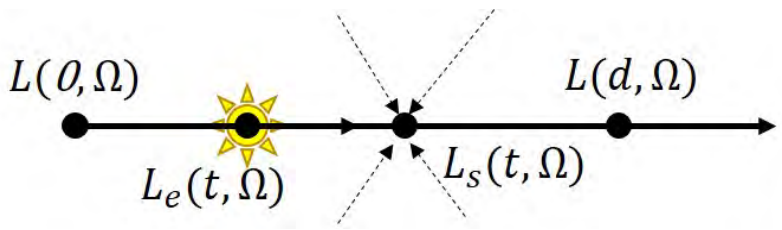


Figure 2.15: Radiative transfer in participant media.

2.9.2.2 General solution of radiative transfer equation in homogeneous media

To solve the radiative transfer equation (2-21), the integral must be computed. According to importance sampling theory (Appendix A), a density function which is approximately proportional to $T(d-\mathrm{t})[\sigma_a(t)L_e(t,\omega)+\sigma_s(t)L_{in}(t,\omega)]$ should be chosen, in order to get lower variance. In homogeneous media $T(t)$ can be analytically expressed as $e^{-\beta t}$. Thus, the density function can be $p(t)=c.T(t)$ with c being a constant. It leads to:

$$L(d,\Omega)=L(0,\Omega)T(d)+\int_0^d \frac{T(d-\mathrm{t})}{p(t)}[\sigma_a(t)L_e(t,\Omega)+\sigma_s(t)L_{in}(t,\Omega)]p(t)dt \tag{2-22}$$

The equation $\int_0^{+\infty} p(t)dt=1$ allows one to get the constant c:

$$\mathrm{c}=\frac{1}{\int_0^{+\infty} T(t)dt}=\frac{1}{\int_0^{+\infty} e^{-\beta t}dt}=\beta \tag{2-23}$$

The density function $p(t)=\beta e^{-\beta t}$ also represents the possibility that a scattering (or absorption) event occurs at position t, i.e., $p_{suc}=\beta e^{-\beta t}$. If no scattering (or absorption) event occurs during [0, d], the probability is:

$$p_{fail}=1-\int_0^d \beta e^{-\beta t}dt=e^{-\beta d} \tag{2-24}$$

To sample $p(t)$, an inverse transform sampling method can be used (Morlet, 1983). This method first computes the cumulative distribution function (CDF):

$$CDF(t)=\int_0^t \beta e^{-\beta t'}dt'=1-e^{-\beta t} \tag{2-25}$$

It results that the randomly sampled t is

$$t=-\frac{\ln(1-\xi)}{\beta} \tag{2-26}$$

where ξ is a uniformly distributed random variable with $\xi\in[0,1]$.

2.9.3 Radiative transfer in plane-parallel atmosphere

Since backward path tracing is more efficient in simulating images than forward photon tracing, the atmosphere RT is simulated in backward mode. It allows one to simulate images with atmospheric effects, and also diffuse radiation with fish-eye sensors.

Figure 2.16 illustrates how backward path tracing algorithm is used to simulate atmosphere RT. When a ray enters the atmosphere, a scattering event may occur. The position of this event is determined by the free path that is randomly sampled from the density function. Since a 3D scene

may contain other types of landscape elements, such as terrain, the radiance recorded by the sensor is computed differently depending if the ray is blocked before the atmosphere scattering event.

- **Ray not blocked during the free path**. Radiance is determined by the phase function and probability that atmosphere scattering occurs successfully. Suppose a ray starts from A and is scattered at B (**Figure 2.16**), the radiance induced by the sun and scattered into A is:

$$L_{B\to A} = T_{AB} \cdot \beta_B \omega_B \cdot P_{ats}^{ext} \cdot E_{dir} T_{B+\infty} / p_{suc}^{AB} \tag{2-27}$$

  where T_{AB} is the transmittance between [A, B]; E_{dir} is solar irradiance; $T_{B+\infty}$ is the total transmittance from point B to the top of the atmosphere along negative sun direction; p_{suc}^{AB} is the probability that the ray starts from A and is scattered at B.

- **Ray is blocked during the free path**. The radiance from point C to B (**Figure 2.16**) is:

$$L_{C\to B} = T_{BC} \cdot f(C, \Omega_i, \Omega_o) \cdot E_{dir} T_{B+\infty} / p_{fail}^{BC} \tag{2-28}$$

  where $f(C, \Omega_i, \Omega_o)$ is the BSDF at point C; p_{fail}^{BC} is the probability that a ray starts from B, but fails to induce a atmosphere scattering event.

The above process can be repeated to calculate multiple scattering, with the procedure described in section 2.6.

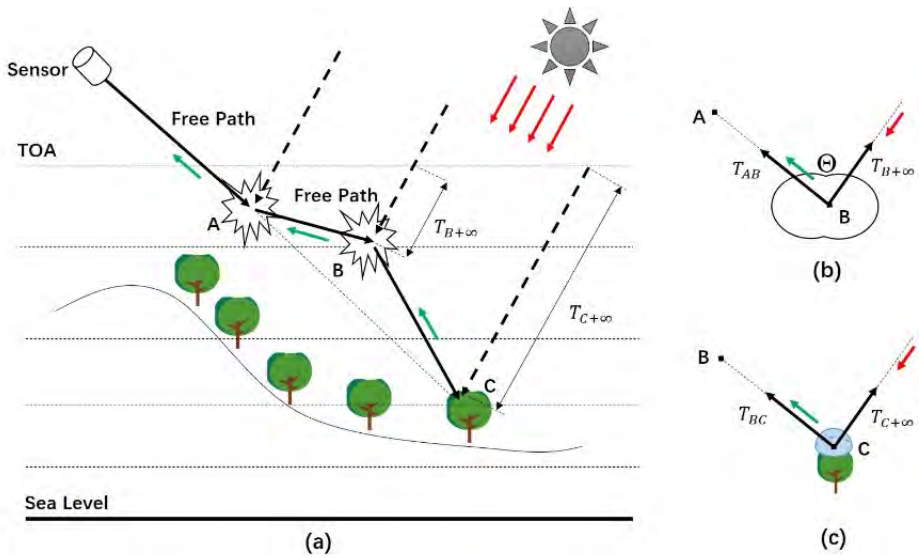


Figure 2.16: Simulating radiative transfer in atmosphere with backward path tracing.
(a) Radiative transfer simulation in atmosphere; (b) Volume scattering; (c) Surface scattering.

Equations (2-27) and (2-28) show that free paths p_{suc} and p_{fail} are important terms that need to be calculated during the simulation. For atmosphere, although each layer can be treated as homogeneous media, different layers usually have different optical properties. Thus, the method

described in homogeneous media cannot be directly applied, due to the inhomogeneous optical properties. To get the free path, we sample the optical depth instead of sampling free path directly, i.e., $\tau_x = -\ln(1-\xi)$. Further, we find a point A_n (**Figure 2.17**) in the propagation direction of the ray that starts from A_1, ensuring that the cumulated optical depth is τ_x. Since the atmosphere scattering event only occurs at point A_n, but fails for other layers, thus, the probability is:

$$p_{suc}^{A_1A_n} = p_{suc}^{A_{n-1}A_n} \cdot \prod_{i=2}^{n-1} p_{fail}^{A_{i-1}A_i} = \beta_{n-1} \cdot e^{-\sum_{i=1}^{n-1}\beta_i d_i} \tag{2-29}$$

$$p_{fail}^{A_1A_n} = \prod_{i=2}^{n} p_{fail}^{A_{i-1}A_i} = e^{-\sum_{i=1}^{n-1}\beta_i d_i} \tag{2-30}$$

where β_i is the extinction coefficient for layer i; d_i is the path length in ith layer.

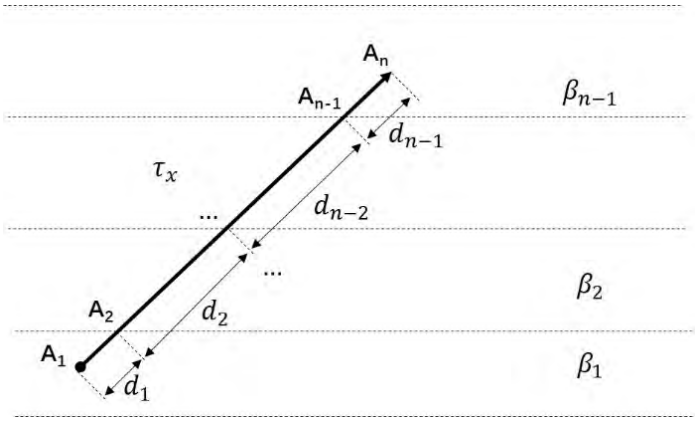


Figure 2.17: Free path sampling in plane parallel atmosphere.

2.10 Implementation and extension of LESS

The ability to complete different simulation tasks is an important aspect of a 3D model (Disney et al., 2000). The implementation of LESS is based on an open source ray-tracing code named Mitsuba (Jakob, 2010), which provides a flexible plugin architecture. It abstracts most of the components associated in the *Radiative Transfer Model* module into different groups of plugins: samplers, cameras, light sources, BRDFs, shapes, etc. This architecture makes it possible for developers to extend LESS with new functionalities without knowing and recompiling the whole system. For example, a circular fisheye camera that captures energy from a 180º field of view is implemented in LESS (**Figure 2.18**). To implement this camera, the developer must simply project the random samples from the image plane, which is provided automatically by LESS, into

spherical coordinates (Camera Ray). The radiance that follows the ray can be automatically computed by functions provided by LESS.

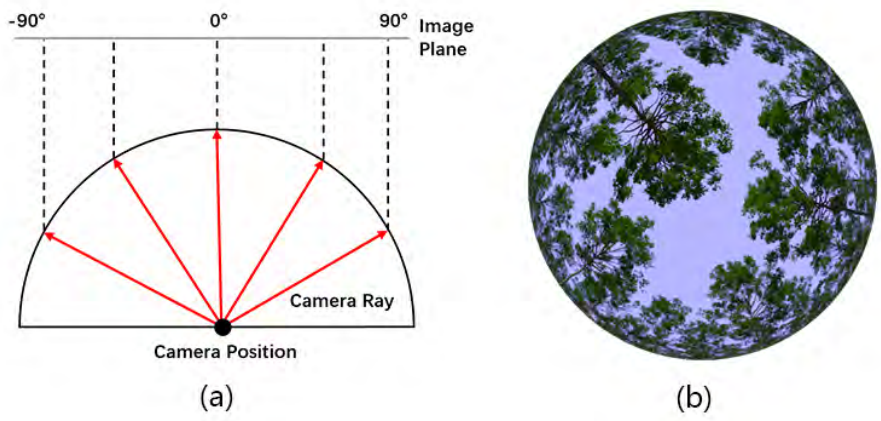


Figure 2.18: Implementation of a circular fisheye camera. (a) The projection diagram of a circular fisheye camera; (b) An example of simulated fisheye image using LESS.

Currently, both the forward (FPT) and backward (BPT) modes have been implemented in LESS. The BPT approach only considers the power that directly enters the sensor, avoiding several redundant computations. Thus, it has advantages over the FBT approach, which launches rays from the light source when only images in a few observation angles are needed. This advantage also benefits the simulation of various sensors (e.g. spherical camera and LAI-2000) because it can consider only the landscape within its field of view if a very large scene is provided. However, BPT is not suitable for simulating the energy distribution over the entire scene. In this case, FPT is needed. An advantage of LESS is that it offers a flexible system architecture that allows developers to implement new functionalities. Since most of the components in RT Core are plugins, for example, integrator, which is one of the main plugins and controls how rays are tracked and how radiances are calculated, is easy to be implemented without changing other components of the system. This allows for the development of a new integrator plugin that can work in forward mode, making energy distribution simulation possible.

A limitation of the current LESS version is that the simulation of thermal radiation is not available in forward mode, since it is highly computationally demanding and is also confronted with significant sampling errors (Bailey et al., 2016), especially for the scenes with a large number of elements. Because BPT only simulates images, thus, studying radiative budget with thermal emission is difficult for LESS. This means that more sophisticated approaches must be devised to deal with this problem, for example, using reverse ray-tracing while sensing rays from the leaf itself instead of the sensors (Bailey, 2018). Another possible future extension of LESS is the

utilization of graphics processing units (GPUs), which has been greatly improved in recent years. It has also been successfully applied to the 3D radiative transfer modeling to improve the computational efficiency (Bailey et al., 2014; Malik et al., 2017).

2.11 Concluding remarks

A large-scale remote sensing data and image simulation framework (LESS) was designed and implemented. It provides (1) a set of tools (including a GUI) to ease the management of input parameters and the creation of 3D mock-ups, and (2) a flexible 3D RT model that can simulate BRF, BT and also large-scale spectral images. The number of spectral bands is not limited. Through an on-the-fly computation of the sunlit and shaded scene components, LESS can simulate thermal infrared images with high efficiency and low memory usage. Although we mainly focus on vegetation scenes in this paper, LESS can also be applied to simulating remote sensing data over other kinds of real scenes, e.g., complex terrains, roads and buildings. LESS can also simulate remote sensing images with different spatial resolutions and multi or hyperspectral bands.

Due to the flexible system architecture, future developments of LESS will focus on the addition of new simulation capabilities for LiDAR and the atmosphere. The long-term goal of LESS is to provide a more user-friendly 3D RT simulation tool. The software can be downloaded from http://lessrt.org/. It can run on Windows and Linux operating systems and has been released as open-source (GNU GPLv3).

Chapter 3

Accuracy evaluation of LESS

Summary

3.1 BRF validation 34

3.1.1 Model intercomparison 34

3.1.2 Model validation with field measurements 36

3.2 Image simulation of realistic forest stand 38

3.3 Thermal infrared simulation 41

3.4 FPAR simulation 42

3.5 Downward solar radiation in complex rugged terrain 44

3.6 Atmosphere simulation 45

3.7 Concluding remarks 46

3.1 BRF validation

3.1.1 Model intercomparison

To evaluate the accuracy of LESS, we compared it with other models over several different homogeneous and heterogeneous canopies from the RAMI website. The LESS BRFs in principal and cross-principal planes are calculated with two approaches: forward photon tracing (FPT) and backward path tracing (BPT). For BPT, BRF image is generated per direction and the mean value is the BRF. The number of rays in each pixel for BPT is set to 128, which has been proven to be substantial for most of the heterogeneous scenes (J. Qi et al., 2017). For FPT, virtual photon is used to estimate the directional BRF, while the total number of photons is determined by the scene area and illumination photon density, which is set to 0.02 m here. Three types of landscapes are considered: discrete floating sphere canopies, random spherical and cylindrical canopies and realistic forest stand. LESS simulation of brightness temperature was also compared to simulations of the 4SAIL model (Verhoef et al., 2007).

3.1.1.1 Discrete floating sphere canopies

Discrete floating sphere scene is a 100 × 100 m square heterogeneous scene from RAMI 3[1] made of 15 spheres (radius = 10 m). Each sphere is composed of 49999 uniformly distributed disc-shaped leaves. The LAI of each sphere is 5. LESS BRFs in red and near-infrared (NIR) bands are compared with reference BRFs (ROMCREF) provided by the ROMC. The reference BRFs correspond to averaged results of other models, such as RAYTRAN (Govaerts and Verstraete, 1998), Rayspread (Widlowski et al., 2006) and FLIGHT (North, 1996). To simulate BRFs using LESS, the 3D scene, in the format of RAMI was first converted to Wavefront OBJ File Format. Specifically, the disc-shaped leaves are converted into squares with the same leaf area. The sun zenith angle (SZA) used in the simulations is set to 20° and 50°.

Figure 3.1 shows RAMI and LESS BRFs in the principal plane. LESS BRFs are simulated with the FPT and BTP modes with differences smaller than 0.0005 and 0.001 for red and NIR bands, respectively. Compared to other models, LESS BRFs are very close, with a root mean square error (RMSE) equal to 0.0002 and 0.003 for red and NIR bands, respectively. LESS slightly

[1] http://rami-benchmark.jrc.ec.europa.eu/HTML/RAMI3/EXPERIMENTS3/HETEROGENEOUS/FLOATING_SPHERES/SOLAR_DOMAIN/DISCRETE/DISCRETE.php

overestimates the hotspot in the NIR, with a maximum difference of 0.01.

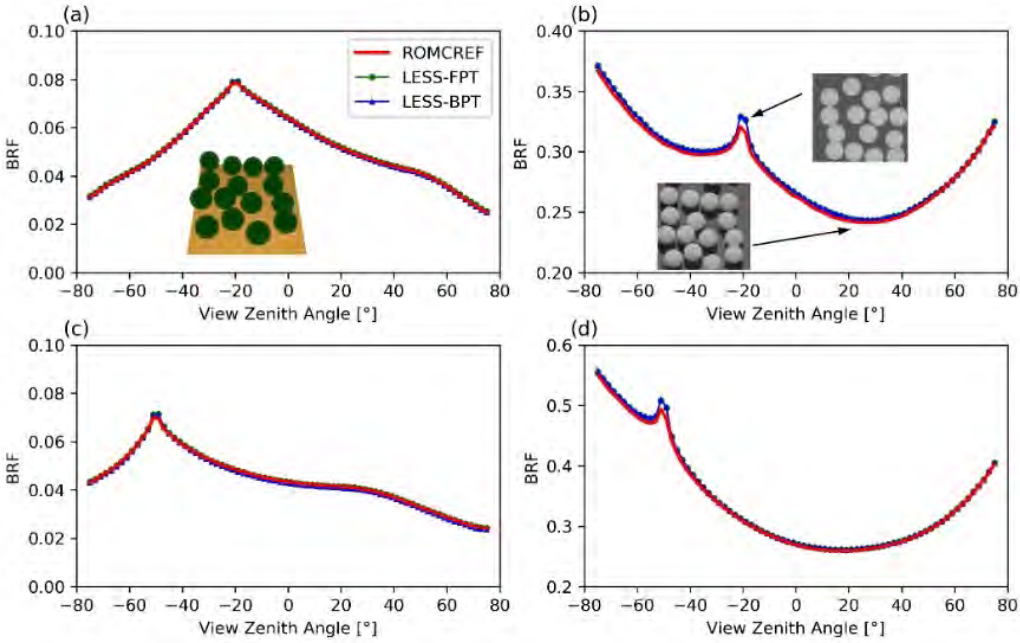


Figure 3.1: Comparison of LESS and RAMI BRFs for a discrete floating sphere scene. (a) Principal plane, red, SZA = 20º; (b) Principal plane, NIR, SZA = 20º; (c) Principal plane, red, SZA = 50º; (d) Principal plane, NIR, SZA = 50º.

3.1.1.2 Hybrid spherical and cylindrical canopies

This hybrid scene[2] is a 270 × 270 m square scene composed of randomly distributed spherical and cylindrical crowns. They consist of 31999 and 17999 uniformly distributed leaves, respectively. The LAI for the spherical and cylindrical canopy is 5, and the SZA is 20°. In order to test the ability of LESS to simulate lateral effects, the simulation was carried out for a 30 × 30 m subplot, a 90 × 90 m subplot and a 270 × 270 m subplot (**Figure 3.2**).

Results (**Figure 3.3**) show that LESS BRFs match very well the RAMI BRFs for any viewing zenith angle (VZA), for all the three subplots: the RMSE is less than 0.001.

[2] http://rami-benchmark.jrc.ec.europa.eu/HTML/RAMI3/EXPERIMENTS3/HETEROGENEOUS/REAL_ZOOM-IN/REAL_ZOOM-IN.php

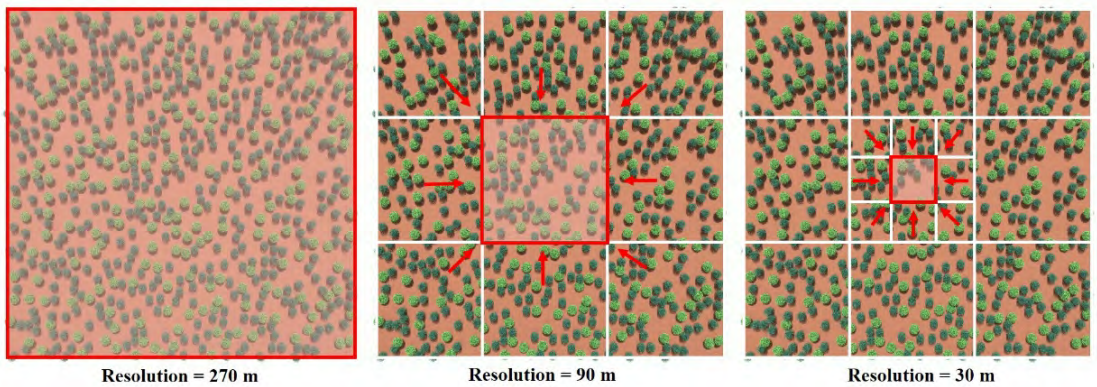


Figure 3.2: Different simulation resolution of the hybrid scene.

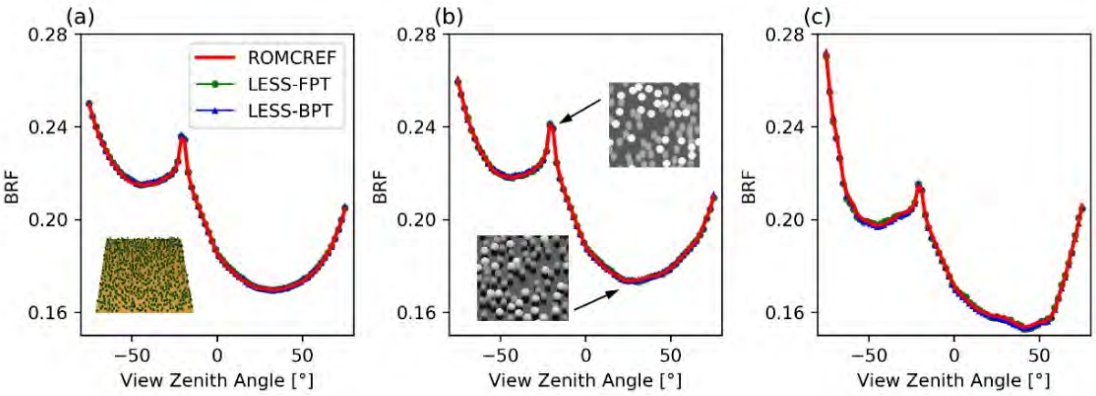


Figure 3.3: Comparison of LESS and RAMI BRFs for a hybrid spherical and cylindrical scene. (a) Principal plane, NIR, SZA = 20º, Resolution = 270 m; (b) Principal plane, NIR, SZA = 20º, Resolution = 90 m; (c) Principal plane, NIR, SZA = 20º, Resolution = 30 m.

3.1.2 Model validation with field measurements

To validate LESS with field measurements, a published dataset from an old aspen (OA) forest in Prince Albert National Park, Canada, was used (J. M. Chen et al., 1997; Leblanc et al., 1999). This dataset has also been used to validate other 3D models, such as RAPID. Thus, we also include RAPID model in this comparison (Huang et al., 2013). The OA forest site is mainly composed of old aspen trees with a height around 21.5 m and a 2 m high hazelnut understory. The LAI of the aspen tree and hazelnut understory is 1.5 and 0.5, respectively. Major structural and optical properties of this site are summarized in **Table 3.1**. Since the detailed structures of the trees are unknown, the OA crown was modeled as an ellipsoid with semi-major axis equal to 3.5 m and semi-minor axis equal to 1.5 m, and filled with randomly distributed leaves. The OA trees inside the plot are assumed to follow a Poisson distribution. The number of trees is controlled by the stem density (850 stems/ha), which is then used to derive the number of leaves inside a crown from the

LAI value and leaf size (10 × 10 cm square). The Hazelnut tree crown is more homogeneous (clumping index = 0.98) than the OA tree crown. It is modeled as a sphere with radius equal to 1 m. The in-situ BRF was measured by the POLDER (POLarization and Directionality of the Earth's Reflectances) instrument, which was mounted on a C-130 airplane on May 26, 1994. Because of the low LAI value in this site, the anisotropic effect of the soil was also considered by integrating the Soilspect model (Jacquemoud et al., 1992) in LESS. The parameters of the soil model are adopted from Huang et al. (2013).

Table 3.1: Structural and optical properties of the OA forest site.

Components	LAI	Tree height	Crown Height	Density (stems/ha)	Optical property (red/NIR)	
					Reflectance	Transmittance
Old Aspen	1.5	21.5m	7m	850	0.08/0.5	0.07/0.4
Hazelnut	0.5	2m	2m	6000	0.08/0.5	0.07/0.4
Ground (Soilspect)	$\mathrm{h} = 0.1245; \mathrm{b} = 1.2999; \mathrm{c} = 0.3087; b' = 0.1673; c' = -0.0013$ $\omega_{red} = 0.24;\ \omega_{NIR} = 0.38$					

Figure 3.4 shows the LESS, RAPID and POLDER BRFs. Overall, LESS shows similarly simulated BRF shapes in both the principal and cross-principal planes. The R^2 in the principal plane between LESS and POLDER is 0.9 and 0.91 for the red and the NIR band, respectively. The corresponding RMSEs equal to 0.0019 and 0.024, respectively. In the cross-principal plane, the correlation coefficients between LESS and POLDER are 0.91 and 0.002 for red and NIR band, respectively. Despite the low correlation in the NIR band, the RMSE is still small (0.0029 and 0.013). This low correlation is also found in the comparison campaign in Huang et al. (2013). An underestimation in the backscattering direction for both LESS and RAPID models(**Figure 3.4**c). It can be due to several factors, and in particular the unknown structure of the canopy. The tree crown is modeled as homogeneous crown in the LESS simulation, while the distribution of individual trees is assumed to be Poisson distribution. Also, the parameters used to describe the ground anisotropic scattering are those of a nearby site where no POLDER data was collected.

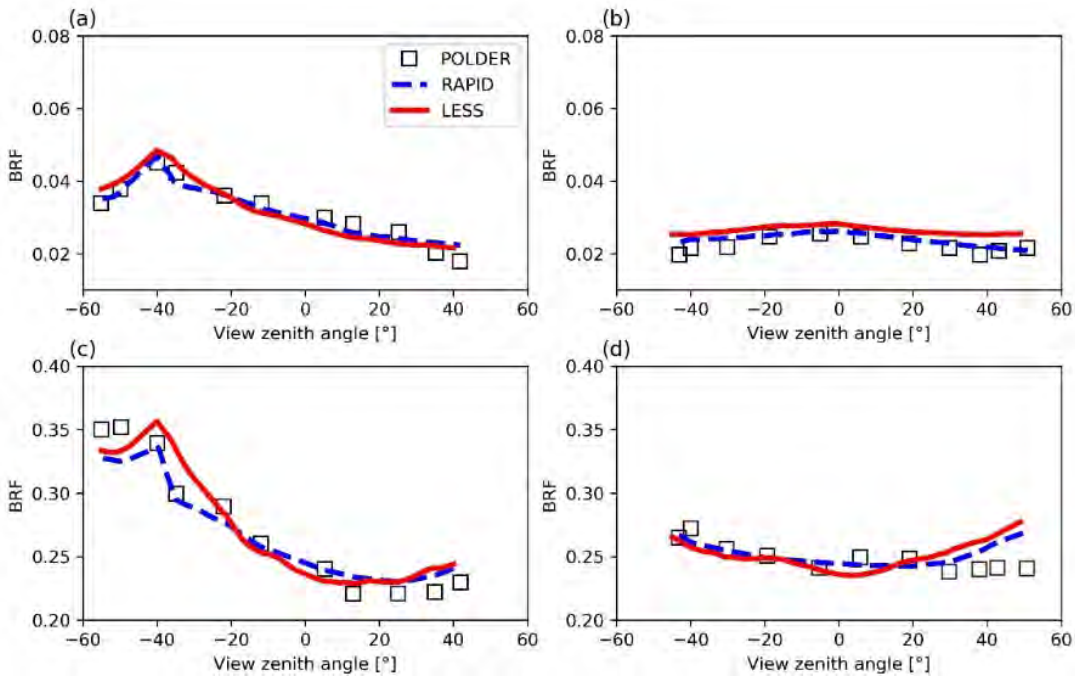


Figure 3.4: Comparison with field measurements and RAPID model. (a) red, principal plane; (b) red, cross-principal plane; (c) NIR, principal plane; (d) NIR, cross-principal plane.

3.2 Image simulation of realistic forest stand

LESS was also assessed with a realistic forest scene, HET09_JBS_SUM from RAMI website[3], through a pixel-wise comparison with a DART simulated image. This scene is a complex forest scene that contains seven species of 18 individual tree objects distributed in a 100 ×100 m plot. Each tree has detailed structure information. **Figure 3.5** and **Table 3.2** show the scene structural and optical properties. The nadir BRF image in the NIR band with a spatial resolution of 20 cm was simulated both by DART and LESS (BPT). We carried out the vegetation simulation in both triangle and turbid medium modes in DART. In the turbid mode, the leaves in each voxel were automatically converted into turbid medium according to local leaf area and leaf angle distribution. Since only one image is simulated, the number of discrete directions in DART was set to 10. **Figure 3.6**a ~ c illustrate the simulated BRF images of DART and LESS, respectively. It can be observed that these three images keep almost the same spatial pattern, even at the fine scale of small branches. To compare them more quantitatively, the pixel-wise scatter plots presented in **Figure 3.6**d and **Figure 3.6**f indicate that LESS is consistent with DART and show

[3] http://rami-benchmark.jrc.ec.europa.eu/HTML/RAMI-IV/EXPERIMENTS4/ACTUAL_CANOPIES/JARVSELJA_SUMMER_BIRCHSTAND/JARVSELJA_SUMMER_BIRCHSTAND.php

R-squared value of 0.92 for both the triangle mode and turbid mode. We also compared the results under the resolution of 1 m (**Figure 3.6**e and **Figure 3.6**g), which shows that the R-squared value of BRF increases to 0.99 and 0.97, respectively. This increase is explained by the fact that the number of rays per pixel increases if the resolution decreases, which leads to more accurate BRFs.

A significant difference between LESS and DART is the usage of computational resources (**Table 3.3**). LESS uses an "instance" technique, which only keeps one copy of each object in memory, and stores the geometric transforms of all individual trees, while DART copies each object to every position in the current version ("instance" functionality will be available in new DART version). This indicates that DART takes more memory: 236 GB and 93GB for triangle and turbid mode, compared to 5 GB of LESS. As a forward-tracking model, DART computes the scene, radiative budget and images for any number of viewing directions within a single simulation. Here, only five images, corresponding to 5 upward directions, are simulated. On the other hand, LESS (BPT) only computes the radiance which goes in one specific direction by using reverse tracking (viewing direction). Here, LESS (BPT) computes only one image and no radiative budget. The computation time is much smaller: 0.6 h for LESS, 3 h for DART turbid mode and 14.1 h for DART triangle mode. Even if DART computation time has been recently divided by a factor 3, LESS's smaller requirement of computational resources makes it appropriate for simulating large-scale (e.g. > 1 km) high-resolution images with explicitly described trees.

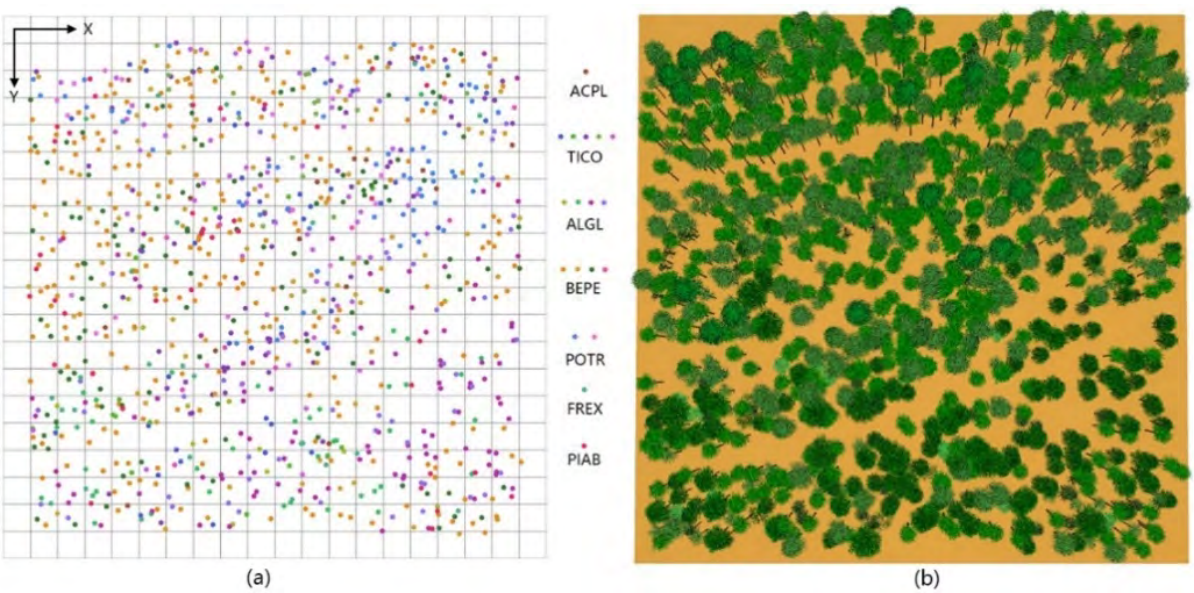


Figure 3.5: Scene description of HET09_JBS_SUM.
(a) 2D positions of each tree; (b) 3D view of HET09_JBS_SUM.

Table 3.2: Optical properties of landscape elements

Element	ρ_{leaf}	τ_{leaf}	ρ_{branch}	ρ_{trunck}	Number
ACPL	0.426	0.478	0.378	0.378	16
TICO	0.439	0.462	0.378	0.378	205
ALGL	0.500	0.383	0.378	0.378	226
BEPE	0.446	0.458	0.378	0.547	465
POTR	0.411	0.493	0.378	0.447	78
FREX	0.513	0.378	0.378	0.378	30
PIAB	0.445	0.370	0.378	0.378	39
ground	$\rho_{ground} = 0.339$				

ACPL: *Acer platanoides*; TICO: *Tilia cordata*; ALGL: *Alnus glutinosa*; BEPE : *Betula pendula*; POTR : *Populus tremuloides*; FREX : *Fraxinus excelsior*; PIAB : *Picea abies*.

ρ_{leaf}: leaf reflectance ρ_{branch}: branch reflectance

τ_{leaf}: leaf transmittance ρ_{trunck}: bark reflectance

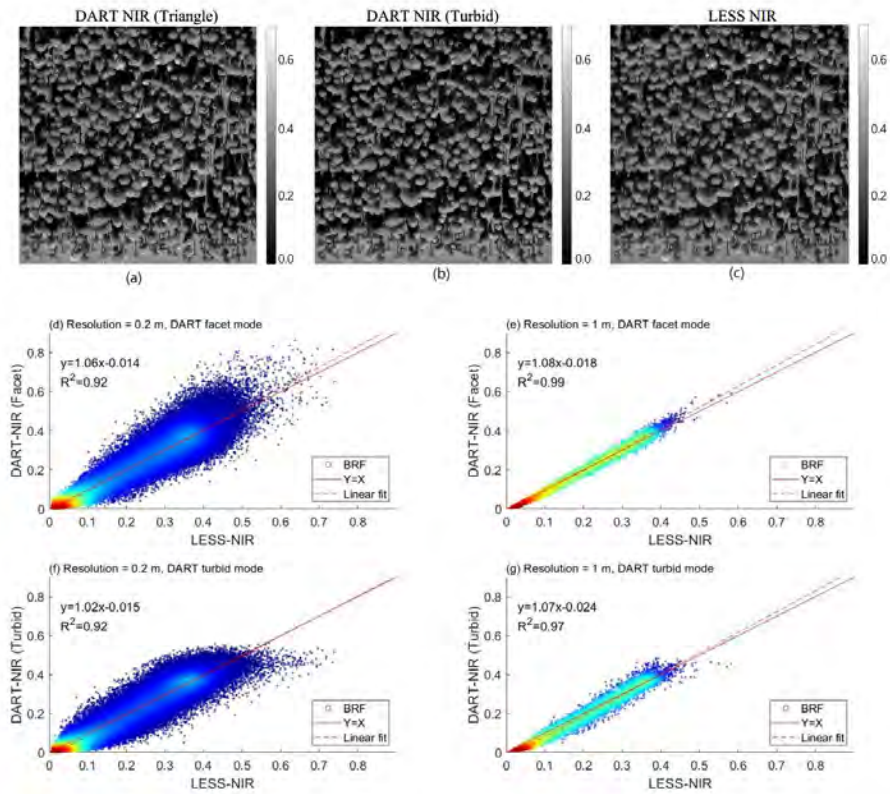


Figure 3.6: Pixel-wise comparison between LESS and DART.

(a) NIR BRF image of DART in triangle mode; (b) NIR BRF image of DART in turbid mode; (c) NIR BRF image of LESS; (d) Scatter plot between DART (triangle mode) and LESS with 0.2 m spatial resolution; (e) Scatter plot between DART (triangle mode) and LESS with 1 m spatial resolution; (f) Scatter plot between DART (turbid mode) and LESS with 0.2 m spatial resolution; (g) Scatter plot between DART (turbid mode) and LESS with 1 m spatial resolution.

Table 3.3: Computational resources usage (for the resolution of 0.2 m)

Resources	DART (Triangle)	DART (Turbid)	LESS
Memory usage	236 GB	93 GB	5 GB
Time	14.1 hours	3.0 hours	0.6 hours

Computing Platform: Intel Xeon E5-2687W, 40 cores, 500 GB memory

3.3 Thermal infrared simulation

Two homogeneous scenes with LAI = 1 and LAI = 3 are built for the comparison of simulated directional BTs with 4SAIL model. The canopy height and leaf size are set to 1 m and 0.03 m. **Figure 3.7** illustrates the simulated BTs in principal and cross-principal planes and shows that LESS simulated BTs are well matched with 4SAIL simulated BTs (R^2=0.99). The RMSE for LAI=1 and LAI=3 are 0.087 K and 0.075 K, respectively.

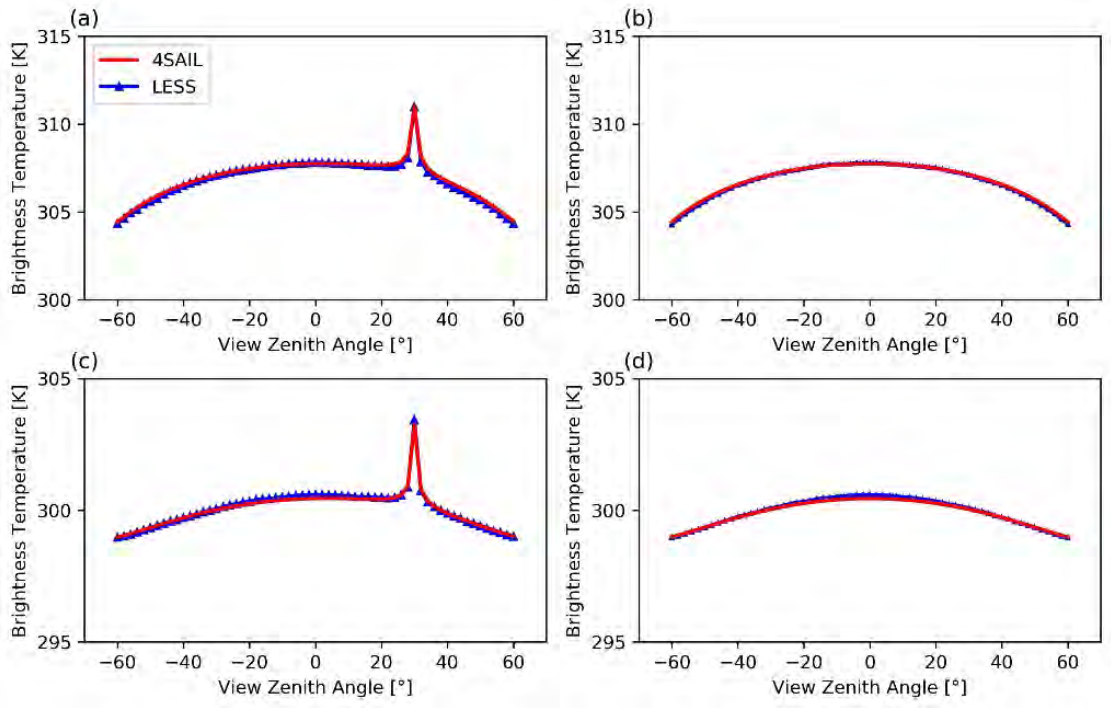


Figure 3.7: Comparison of simulated directional BTs over homogeneous scenes.
(a) LAI = 1, principal plane; (b) LAI = 1, cross-principal plane; (c) LAI = 3, principal plane; (d) LAI = 3, cross-principal plane. The temperatures of sunlit soil, shaded soil, sunlit leaves and shaded leaves are 320.65 K, 306.65 K, 299.65 K and 296.65 K, respectively. The emissivity values of soil and leaves are 0.97 and 0.99, respectively. The leaf angle distribution is spherical.

3.4 FPAR simulation

A field-measured FPAR dataset of three crops (winter wheat, JD-I corn and JD-II corn) was used to validate the accuracy of LESS for simulating vertical profile of FPAR (Xie et al., 2010). The study is located at Xiaotangshan experiment site, Beijing, China. This dataset was measured at in April 27, 2008 and August 29, 2010 for winter wheat and corn, respectively. Measurements took place in a winter wheat plot and two corn plots with different growing stages (Corn-I and Corn-II). Leaf and ground spectra were measured using a spectroradiometer (ASD FieldSpec3, Analytical Spectral Devices, USA). The structural parameters, such as LAI, crop height and row spacing, were also measured, as illustrated in **Table 3.4**. The optical properties of the three crops and the ground are illustrated in **Figure 3.8**

The FPAR was measured with a SUNSCAN canopy analysis system. For winter wheat (corn), at each 16 cm (20 cm) position, an upward and downward PAR were measured. Then the FPAR at a height H is calculated as:

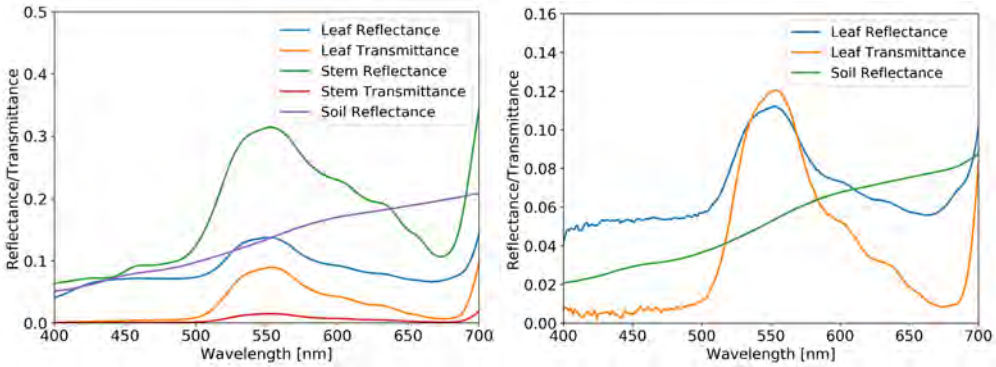
$$FPAR(H) = \frac{PAR^{\downarrow}(H_0) - PAR^{\downarrow}(H) + PAR^{\uparrow}(H) - PAR^{\uparrow}(H_0)}{PAR^{\downarrow}(H_0)} \tag{3-1}$$

where $PAR^{\downarrow}(H_0)$ is the downward PAR at top of the canopy; $PAR^{\downarrow}(H)$ is the downward PAR at height H; $PAR^{\uparrow}(H)$ is the upward PAR at height H; $PAR^{\uparrow}(H_0)$ is the upward PAR at top of the canopy. Three measurement sites were chosen in the winter wheat plot, and two kinds of measurement scheme were performed at each height position, i.e., along row (PZ8-S) and cross row (PZ8-C). For the two corn plots, only one measurement site was chosen for each one. Similarly, an along-row and a cross-row measurement were performed. Thus, there were 6 FPAR profiles for winter wheat, and 2 profiles for each of the two corns.

According to the measured parameters, the structures of the winter wheat and corns were built, which are shown in **Figure 3.9**. **Figure 3.8** shows the simulated FPAR and field-measured FPAR profiles. It can be depicted that the FPAR profiles of winter wheat have similar trends along with the change of the height. At the same height level, there are some differences among different measurements, especially at the top of the canopy, which may be caused by the different structures at different sampled sites in the field. Similarly, the measurements at the same position are also diverse for corns. However, the simulated FPAR profiles for both winter wheat and corns are consistent with field measurements, and the simulated FPAR values are between the minimum and maximum values of the measurements, which indicates that LESS successfully parameterized the crop canopy and gives accurate simulated FPARs.

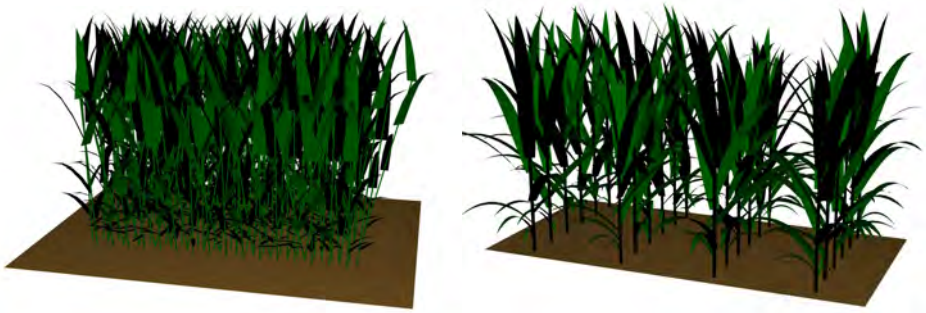
Table 3.4: structural properties of winter wheat and corn

Crop	LAI	Row spacing	Crop height
Winter wheat	3.62	18 cm	48 cm
Corn-I	3.82	60 cm	134 cm
Corn-II	3.24	60 cm	106 cm



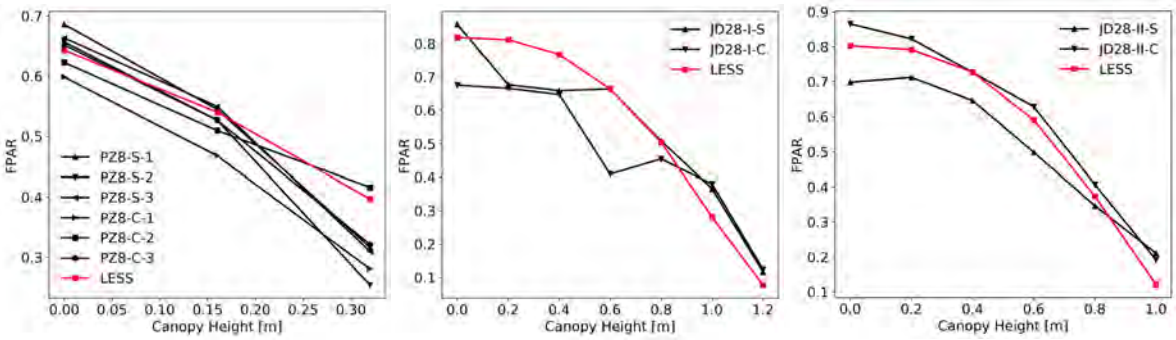
(a) (b)

Figure 3.8: Optical properties of crops and ground: (a) Winter wheat; (b) Corn.



(a) (b)

Figure 3.9: 3D display of the winter wheat and corn: (a) Winter wheat; (b) Corn.



(a) (b) (c)

Figure 3.10: Simulated FPAR and field-measured FPAR profiles.

(a) FPAR of winter wheat; (b) FPAR of Corn-I; (c) FPAR of Corn-II.

3.5 Downward solar radiation in complex rugged terrain

A field-measured dataset of downward solar radiation (DSR) in rugged terrain is used in this study to evaluate the accuracy of LESS for simulating DSR. This dataset is measured at Yuelaingshan experiment site in Chengde (42.05778N, 117.0656E), China. As shown in **Figure 3.11**, four measurement sites were chosen at different positions: hilltop (Station1), East slope (Station2), South slope (Station3) and West slope (Station4). For each measurement site, a shortwave radiometer (CNR4) was used to measure the DSR. For Station1, another radiometer (RMR100) that can measure the direct solar radiation and diffuse solar radiation is used. All the radiometer is mounted approximately parallel with the local slope. Since measurements of Station1 are seldom influenced by terrain, they were used to specify Bottom Of Atmosphere (BOA) irradiance in LESS. The terrain used in the simulation was extracted from the stereo images of ZY-3, which gives a resolution of 12.5 m.

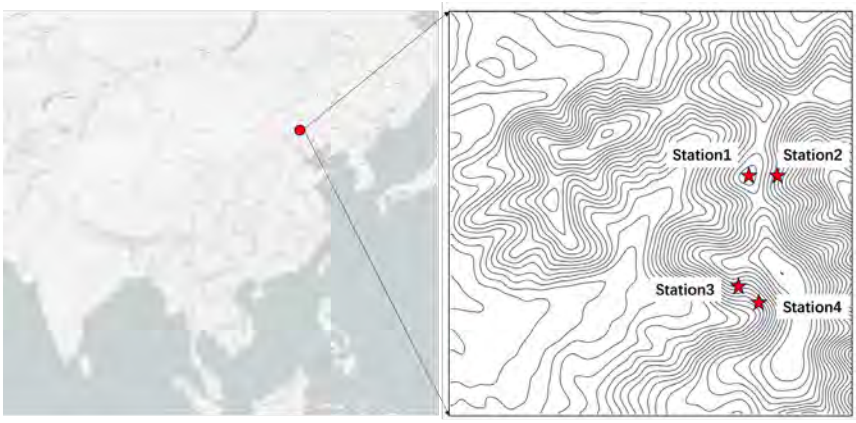


Figure 3.11: The study area for downward solar radiation validation.

Figure 3.12 illustrates the simulation results and field measurements during a whole day from 5:18 to 6:48. It can be seen that the simulation results are highly consistent with field measurements. The RMSE value of Station2, Station3 and Station4 are 33.7 $W \cdot m^{-2} \cdot \mu m^{-1}$、14.6 $W \cdot m^{-2} \cdot \mu m^{-1}$ and 22.9 $W \cdot m^{-2} \cdot \mu m^{-1}$, respectively. For Station2 (East slope), simulated DSR is a bit higher than field measurements after 12:00, while Station3 (South slope) and Station4 (West slope) give overestimation around 12:00. This inconsistency may be caused by the uncertainties of field measurements and terrain data. For example, the radiometer may not be parallel with the slope, and the extracted DEM may be different from local terrain structures due to the coarse resolution DEM data. In addition, Station2 (East slope) first reaches maximum value around 10:30, while Station3 (South slope) and Station4 (West slope) reaches maximum value around 12:00 and 14:00,

respectively. This is consistent with the change of solar azimuth angle during a day. Besides, there are abrupt changes of DSR for Station2 (East slope) around 18:00, Station3 (South slope) around 6:30 and Station4 (West slope) around 7:30, due to the occlusion of terrains. In conclusion, LESS successfully captures the influences of terrains on DSR, and the simulated DSR is consistent with field measurements.

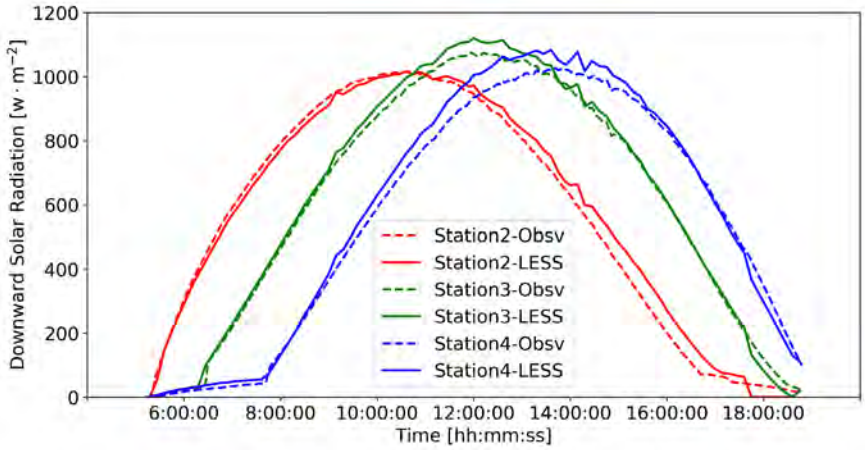


Figure 3.12: Downward solar radiation validation with field measurements.

3.6 Atmosphere simulation

To evaluate the accuracy of LESS for simulating atmosphere, we chose the US Standard atmosphere model from MODTRAN and DART model as references. The downward direct radiation and diffuse radiation are simulated for comparison. The parameters used in these simulations are indicated in **Table 3.5**.

Figure 3.13 shows the downward direct solar radiation and diffuse radiation at the bottom of atmosphere for LESS, DART and MODTRAN. It can be seen that the LESS simulated results are close to both DART and MODTRAN. For direct solar radiation, LESS is exactly the same as DART, while shows a bit difference with MODTRAN. The reason is that the atmospheric profiles that are input into LESS is extracted from the simulation results of DART. The R-squared value between LESS and MODTRAN is 0.99, the RMSE is 31.3W·m^{-2}·μm^{-1}. For diffuse radiation that is induced by atmosphere scattering, LESS, DART and MODTRAN show some differences, especially around wavelength 500 nm. However, the RMSE between LESS and MODTRAN is only around 9.6 W·m^{-2}·μm^{-1}. DART is a bit higher than MODTRAN, while LESS is between them. This comparison demonstrates that the atmosphere simulation in LESS is comparable with DART and MODTRAN.

Table 3.5: Parameters for atmosphere simulation

Parameter	Value
Solar zenith angle	30°
Altitude	0km
Aerosols	Rural VIS=23km
Julian Day	93

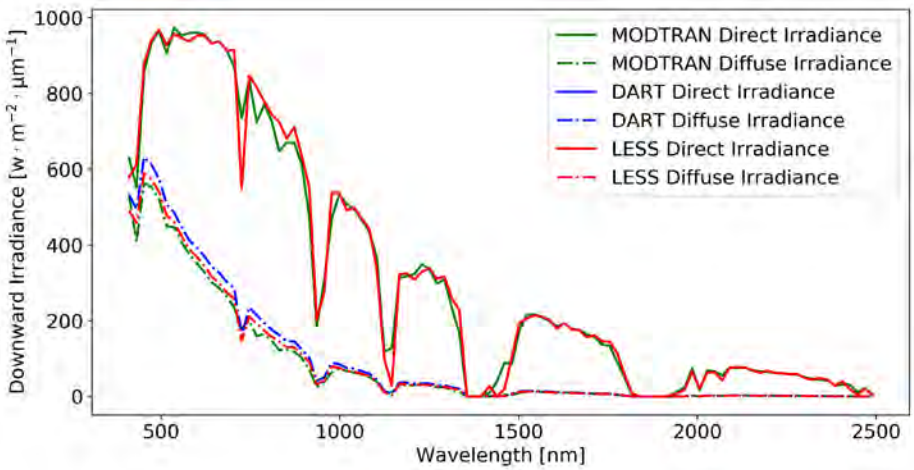


Figure 3.13: Downward direct solar radiation and atmosphere radiation.

3.7 Concluding remarks

The accuracy of LESS has been assessed against other models, using RAMI virtual scenes, LESS proved to be comparable with most 3D radiative transfer models. Besides, it was shown that the forward photon tracing and backward path tracing give nearly the same results, which stresses the self-consistency between the algorithms that are implemented in LESS. For the image simulation, LESS give results that are close to DART simulated images, but with much higher computational efficiency and lower memory usage. For other types of simulation, LESS has also been validated with other well-known models as well as field measurements. In conclusion, LESS is a multifunctional 3D radiative transfer model and gives reliable simulation results.

Chapter 4

Hybrid scene structuring for accelerating 3D radiative transfer

Summary

4.1 Research context 49
4.2 Uniform grid in DART 50
4.3 Hybrid scene structuring 52
4.3.1 Voxel-level ray tracking 52
4.3.2 Within-voxel ray tracking 54
4.3.3 Implementation 54
4.4 Radiation tracking 56
4.5 Experiments 59
4.5.1 BASEL city scene 59
4.5.2 RAMI forest scene 60
4.6 Results and Discussion 61
4.7 Concluding remarks 63

Many scientific domains (e.g., urbanism, agriculture, forestry, etc.) require accurate remote sensing (RS) products, which requires RS models that simulate satellite acquisitions with increased radiometric accuracy. Three-dimensional (3D) radiative transfer models are the most accurate RS models. However, developing such a new model from scratch up is not an easy task. A practical solution is to modify existing models. DART is one of the most comprehensive physics-based models (Gastellu-Etchegorry et al., 2004) that has been developed since 1992. It can simulate the 3D radiative transfer budget and various remote sensing acquisitions (e.g. LiDAR, spectroradiometer and solar-induced fluorescence, etc.) over the whole optical domain, including the atmosphere. Compared to other models, DART can work on scenes that are simulated with any combinations of triangles and fluids, including the so-called turbid medium that is often used for giving a statistical representation of vegetation.

However, a major problem that confronts DART is the relative low simulation efficiency. If it could be accelerated to simulate more complex and large scenes, it will be important for current DART users, since DART is a mature 3D radiative transfer model that has been widely used in the world (NASA, ESA and a lot of universities) to simulate various types of products.

In this chapter, a hybrid scene structuring approach, which is widely used in the computer graphics community, is proposed to accelerate the radiative transfer simulations while keeping the scene as realistic as possible. In a first step, a 3D description of the Earth landscape with equal-sized voxels is optimized to keep only non-empty voxels (i.e., voxels that contains triangles, fluids or turbid medium) and managed using a bounding volume hierarchy (BVH) to reduce memory usage and computation time. For any voxel that contains triangles, within-voxel BVHs are also created to accelerate the ray-triangle intersection tests. The hybrid scheme is implemented in the Discrete Anisotropic Radiative Transfer (DART) model by integrating the Embree ray-tracing kernels developed at Intel. The dual BVH implementation allows one to avoid the limitation of the Embree floating precision that is not adapted to the simulation of large Earth landscapes. In this chapter, the performance of the hybrid algorithm is compared with the original uniform grid approach implemented in DART for a 3D city scene and a forest scene.

The chapter is presented in the paper:

"**Qi, J.***, Yin, T., Xie, D., Chavanon, E., Laret, N., Guilleux, J., Gastellu-Etchegorry, J., Hybrid Scene Structuring for Accelerating 3D Radiative Transfer Simulations. Submitted to Computers & Geosciences" (under review).

4.1 Research context

The applications of radiative transfer models vary from structural (Myneni et al., 1997) and biophysical parameter retrieval (Sehgal et al., 2016) of vegetation to urban radiation flux studies (Landier et al., 2016). For vegetation monitoring, much attention has been paid to radiative transfer modeling and numerous physical models have been developed in the past decades to describe the interactions between solar radiation and complex canopies (Myneni and Ross, 2012). Compared to one-dimensional (1D) models (e.g., SAIL (Verhoef, 1984)), which treat the canopy as a horizontally homogeneous medium, three-dimensional (3D) radiative transfer (RT) models can take into account the highly heterogeneous Earth surfaces. Accurate and efficient 3D models are essential tools for deriving accurate parameters from remote sensing measurements (Gastellu-Etchegorry et al., 2004). This is notably true as 3D information is more and more available with the development of 3D acquisition technology. For example, LiDAR (Light Detection And Ranging) is more and more used in forest studies for measuring the vegetation 3D attributes (Ben-Arie et al., 2009; Lefsky et al., 2002), which are important input parameters for 3D radiative transfer models. However, a drawback of 3D RT models is that they are more difficult to parameterize, especially for some LUT (look up table)-based inversion methods of RS acquisitions. For example, hundreds of thousands of simulations and representative of different environmental and instrumental conditions may be needed to build an invertible LUT database (Banskota et al., 2015). Thus, the consideration of computational efficiency becomes crucial in this kind of applications. In particular, 3D RS models should take advantages of the techniques developed in the computer graphics domain to optimize the computation. The poor efficiency is mainly because RS models are designed to simulate RS acquisitions that are very accurate in terms of radiometry, and not simply realistic from the human point of view conversely to most computer graphics algorithms.

In 3D RT models, the complex landscape elements are described with triangle meshes, geometrical objects and turbid media. Triangle meshes can be used to simulate very precise 3D structures (e.g. position of each leaf), while turbid media can be used to simulate the canopy structure with statistical parameters (e.g., LVD: Leaf Volume Density and LOD: Leaf Orientation Distribution). To simulate large and complex scenes, the triangle and turbid elements in 3D space must be organized into efficient computer data structures to avoid unmanageable computer constraints in terms of volume memory and computation time (Disney et al., 2000), e.g., the algorithm that tests all objects one by one to determine if an intersection occurs is too

computationally expensive to simulate complex scenes (McNeill et al., 1992). Most of the RT models rely on a space subdivision technique to avoid some redundant computations. With space subdivision, objects which are located in a subspace (or voxel) which is not intersected by a ray can be directly excluded from the intersection tests, which can significantly increase the efficiency when handling scenes with a large number of elements. The space subdivision often partitions the space into hierarchical subspaces (or voxels), where the scene elements are located (Glassner, 1984). This very important technique is heavily used in some ray-tracing (Discrete-Ordinate or Monte-Carlo) based RT models for achieving computationally efficient ray-object intersections (Gastellu-Etchegorry et al., 2015; Govaerts and Verstraete, 1998; Jianbo Qi et al., 2017).

Up to now, various space subdivision approaches have been implemented, including uniform grid (Cleary and Wyvill, 1988), BSP (Binary Space Partitioning) tree (Sung and Shirley, 1992), octree (Glassner, 1984) and BVH (Bounding Volume Hierarchies) (Lauterbach et al., 2009) etc. Because the performances of these approaches usually depend on the specific scene structures, no single technique can be universally the best (Mller and Fellner, 2000). A uniform grid is well suited for homogeneous scenes, i.e., landscape elements are relatively uniformly distributed in space, while other approaches, such as BVH, are better for inhomogeneous scenes. These facts lead us to develop a hybrid scene structure which can combine the advantages of a uniform grid and BVH tree, which is also self-adaptive to different scene structures.

4.2 Uniform grid in DART

DART uses the discrete-ordinate ray-tracing method to simulate the propagation and interaction of radiation in the “Earth-Atmosphere” scene (**Figure 4.1**), which is represented as a 3D array of rectangular voxels. The tracked radiation is along N directions that discretize the 4π space. Each direction Ω_i is associated with a solid angle $\Delta\Omega_i$ with $\sum_{i=1}^{N}\Delta\Omega_i = 4\pi$ sr (Yin et al., 2013). The 3D scene contains two major parts: the atmosphere and the Earth scene. Its usefulness is illustrated here with its so-called “reflectance mode”, where sun is the only radiation source. At the beginning, the propagation of radiation in the atmosphere is simulated (Grau and Gastellu-Etchegorry, 2013), which gives the direct sun irradiance and diffuse atmospheric radiation that reach the voxels of the top of the Earth scene at the bottom of atmosphere (BOA). Then, this so-called BOA radiation illuminates the Earth scene. A typical Earth scene usually can contain landscape elements, such as trees, grass, houses and water. These elements are represented with any combination of triangles, fluids and turbid media.

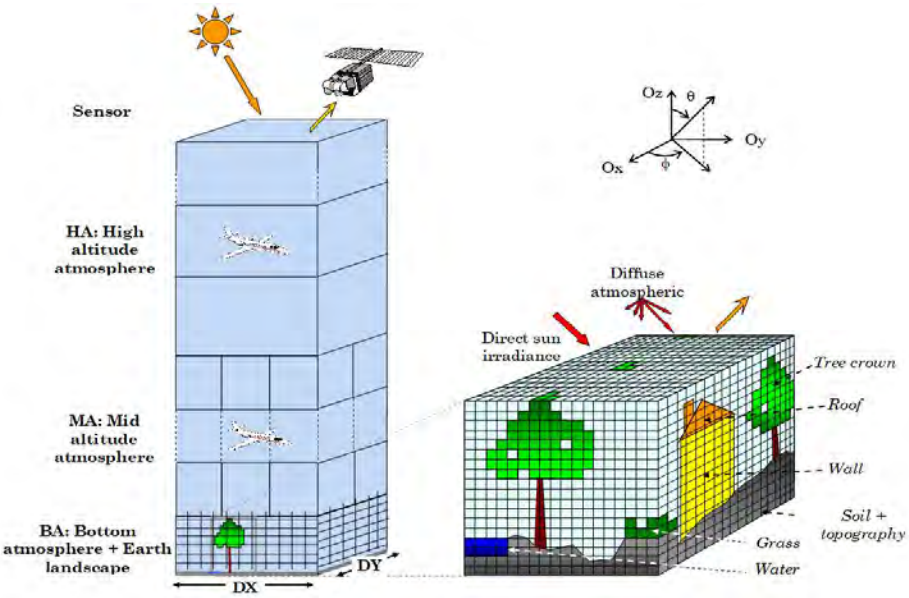


Figure 4.1: Radiative transfer modeling in the "Earth-Atmosphere" scene of DART. The scene of DART is divided into two parts: Earth and Atmosphere.

As presented in **Figure 4.1**, DART partitions the Earth scene into equal-sized voxels, which are used to organize the landscape elements (triangles, fluids and turbid media). Equal-sized voxels are mandatory for simulating the 3D radiative budget (e.g., absorbed radiation) of Earth landscapes. For each triangle, a triangle-voxel intersection test (Akenine-Möller, 2005) is performed to determine the voxel that contains it. The turbid medium is just a set of descriptive statistical parameters (e.g., LVD, LOD for vegetation), and it usually fills the whole voxel. To store this information in memory for radiation tracking, a list of voxels is first created. The length of this list is equaled to the number of voxels in the scene. For each voxel, DART stores a list of references of all the triangles which intersect this voxel and a list of turbid media that are in the voxel (**Figure 4.2**).

The radiative transfer with the Earth scene (landscape) starts by tracing rays from the BOA illumination plane to the scene. **Figure 4.2** illustrates the case of a ray that enters a voxel at the entry point A and then crosses five voxels. When a ray enters a voxel, ray-triangle tests are performed sequentially for all the triangles inside the voxel. Intersection occurs for the closest triangle that intersects the incident ray. Usually, the order of the intersections is not known. Thus, a sorting algorithm is performed, which may require large computation time if too many triangles are inside the voxel. If the ray intersects a triangle, scattering is computed using the optical properties of this triangle. If present, turbid media is taken into account, both for transmittance and scattering, including within voxel single and multiple scattering. If no triangle intersection is found, the ray crosses the voxel, and an exit point (e.g. point B in **Figure 4.2**) will be calculated by a ray-

plane intersection algorithm (Hughes and Foley, 2014). Then, it is very straightforward to find the next voxel that the ray will enter, because the exit face of the current voxel is also the entry face of the next voxel. It is a major advantage of the uniform grid partitioning technique. Only objects that belong in part or in total are taken into account. However, this step-by-step algorithm cannot skip empty spaces (e.g. voxel 4, 8 and 9 in **Figure 4.2**), which may lead to many redundant computations in highly heterogeneous scenes with large empty gaps. This uniform voxel approach may also consume a lot of memory, since all the voxels (empty or non-empty) need to be created.

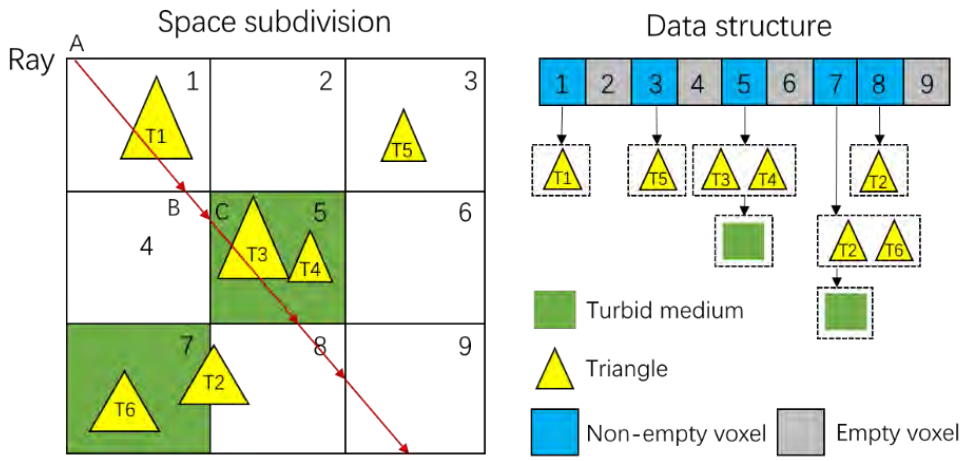


Figure 4.2: Space subdivision and data structure.

4.3 Hybrid scene structuring

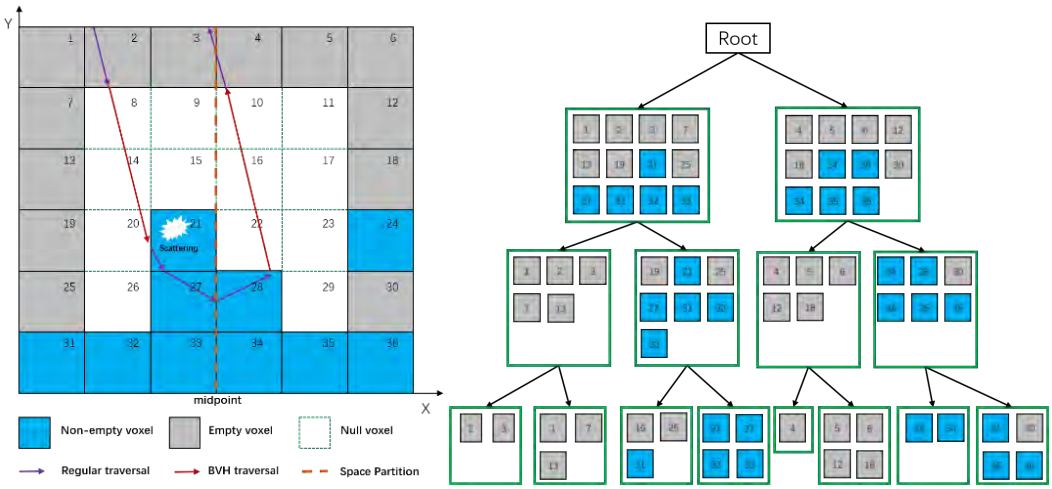
4.3.1 Voxel-level ray tracking

DART uses the voxel as the basic unit to calculate the transferred energy in the Earth scene. This approach is also adopted in our new hybrid scene approach, since voxels can also support the turbid medium, which makes it possible to simulate triangle and turbid medium simultaneously in a single scene. The voxels in the 3D space are categorized into three groups: non-empty voxel, empty voxel, and null voxel (**Figure 4.3**). The non-empty voxel is the voxel that contains triangles or fluids/turbid media. The empty voxel contains nothing and is on the side of the scene. It is created in the voxel list because it is used for setting up the source rays and storing radiation that exits the Earth scene. The null voxel only has its position in 3D space but it is not created in memory. This approach reduces the memory usage especially if the 3D scene contains many empty spaces (e.g. landscape with topography).

To skip the empty spaces (null voxels) during the ray tracing method, the empty and non-empty voxels are organized using a BVH tree, which is well adapted to non-uniform geometry

distributions. The BVH used in this study is a binary tree that structures the bounding volumes of all the voxels into a hierarchy structure. As shown in **Figure 4.3**b, it first creates a root node that represents the bounding volume of the whole scene. Then all the voxels are divided into two sides around the midpoint along the longest scene axis (e.g. the x axis). For each side, a bounding volume containing the voxels inside it is created. In the BVH tree, these two bounding volumes are the child nodes of the root. This partitioning process is recursively repeated until a level such that the "leaf" node of this level only contains a small number of voxels (e.g. 1 or 2).

To track rays in this hybrid scene, the algorithms of uniform grid and BVH are combined. This is illustrated here with a ray that enters a voxel (i.e., voxel 2) on the side of the simulated earth scene (**Figure 4.3**a). Then, the computation of the exit point gives the next possible voxel (i.e., voxel 8). If the next voxel is a null voxel, the BVH algorithm is used to determine the next non-null voxel (i.e., voxel 21). For that, the BVH algorithm tests if the ray intersects the bounding volume of the root node. If there is an intersection, the algorithm tests the occurrence of intersection with child nodes. At the leaf node, the ray will test ray-voxel intersection with all the voxels, one by one, inside the node using a ray-box intersection algorithm (Williams et al., 2005). This binary search scheme is very efficient for finding voxels that are far from the current voxel. However, it is less efficient if the next cell is very close, since the BVH top-down algorithm searches from the root node to leaf node for ray. Thus, as long as a ray meets non-null voxels, the grid tracking method is used. This mechanism makes the model to choose tracking scheme automatically, and it can adapt to different scene structures without user intervention.



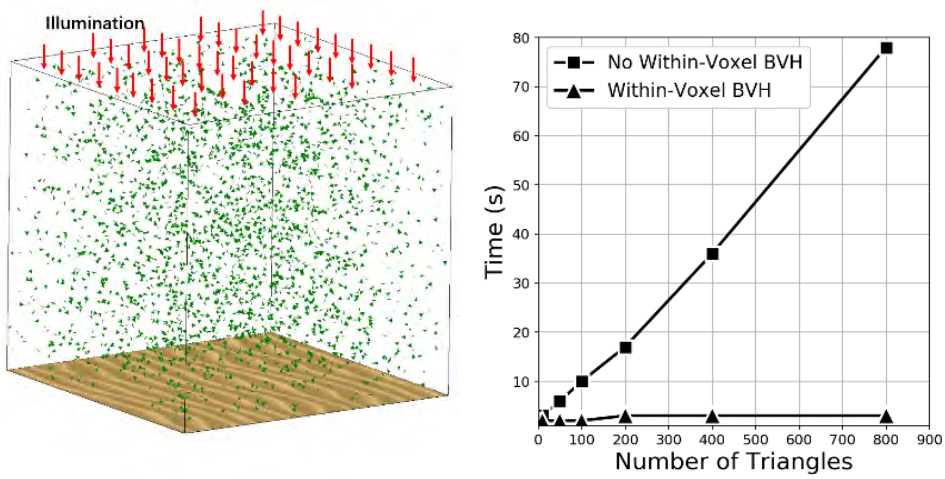
(a) (b)

Figure 4.3: The proposed hybrid scene structuring scheme.

(a) The structure of a typical scene with non-uniform geometry distribution; (b) The corresponding BVH tree which organizes all the non-null voxels.

4.3.2 Within-voxel ray tracking

In the hybrid structuring scheme, the triangles are not directly managed by BVH but organized by voxels. When a ray hits a voxel, the algorithm tests the intersection with all the triangles one by one. For some applications (e.g., large voxel size), each voxel may contain many triangles (e.g., 106 triangles). Hence, testing triangle intersection one by one may also slow down the radiative transfer process. Thus, if the number of triangles in a voxel exceeds a threshold, a BVH tree (so called within-voxel BVH) is created inside this voxel to organize all the triangles directly. In that case, during ray tracking, the test of triangle intersection if calculated with the BVH tree instead of the plain list of triangles. **Figure 4.4** shows a simple 1 m× 1m scene, which contains only one voxel with N randomly distributed triangles. The number N takes the value 1, 5, 10, 50, 100, 200, 400 and 800. The illumination density is set to 106 rays per square meter, which gives a total number of 106 rays. The total intersection time is measured, which is shown in **Figure 4.4**b. It can be seen that the total time increases linearly with the increase of N if the within-voxel BVH algorithm is not used, while it is nearly constant if the within-voxel BVH is used. When N is as low as 10, the time is nearly the same. It explains that the number 10 is used as a threshold to indicate whether a BVH tree should be created in a voxel, which avoids creating BVH trees for voxels that contain only a few triangles (i.e., less than 10). This approach is useful because the BVH tree also consumes memory.



(a) (b)

Figure 4.4: Simulation time varies with the number of triangles in one voxel.

(a) 1 m × 1 m scene with randomly distributed triangles; (b) Simulation time under a different number of triangles.

4.3.3 Implementation

The hybrid structuring scheme was implemented in DART using a ray tracing library named

Embree (Wald et al., 2014), which is a collection of high-performance ray tracing kernels and is optimized for Intel® processors with support for SSE and AVXs (AVX, AVX2, and AVX-512) instructions. A user-defined geometry that represents the voxel needs to be defined to build the voxel level BVH. The shape of this voxel geometry is specified through two callback functions: RtcVoxelBoundsFunc and RtcVoxelIntersectionFun. They are registered to Embree by rtcSetGeometryBoundsFunction and rtcSetGeometryIntersectFunction, respectively. The bounds function RtcVoxelBoundsFunc calculate the bounding box of the voxel geometry, and its code is illustrated in **Listing 1**. The geometryUserPtr, which is a user pointer set by rtcSetGeometryUserData, provides the accessibility to member functions of the geometry (e.g., getLowestCornerX() gets the X coordinate of the lowest corner). The intersection function RtcVoxelIntersectionFun is called to determine the real intersection point when a ray intersects the bounding box of the geometry. This function is implemented with a ray-box intersection algorithm (Williams et al., 2005), since voxels in DART are simply boxes with the same dimensions.

When the number of triangles within a voxel exceeds a threshold (e.g., 10), a BVH tree which manages all the triangles inside the voxel is built. In DART, a triangle may belong to different voxels. To avoid copying a triangle several times, we store all the triangles from the whole scene in a vertex array, as illustrated in **Figure 2.4** (Chapter 2). Each voxel only stores a list of references to the triangles contained. Thus, all the within-voxel BVHs can share the same vertex array in memory by calling the function rtcSetSharedGeometryBuffer provided by Embree.

Due to floating-point precision problems, a ray scattered by a triangle can be intercepted again by the same triangle. This self-intersection can significantly decrease radiation scattering if it is not well handled. Here, an intersection filtering mechanism is used to avoid this self-intersection problem. By registering a call back function RTCTriangleFilterFunc (**Listing 2**), through the function rtcSetGeometryIntersectFilterFunction provided by Embree, an intersection is first set "valid" if intersection occurs. Then, if self-intersection occurs, the tag "geomID" is set to "invalid”, which implies that the ray will ignore this intersection and continue to find other potential intersections. The geometryUserPtr here is still set by rtcSetGeometryUserData, but it represents a list of triangles. When a ray is initialized, the previously intersected triangle preItsTriangle will be provided, which is used in the RTCTriangleFilterFunc function to determine whether the ray intersects it again.

Listing 1: Sample bounds function code of the voxel geometry

```
void RtcVoxelBoundsFunc(const struct RTCBoundsFunctionArguments *args){
 const Voxel * voxel = (const Voxel *) args->geometryUserPtr;
 RTCBounds *bounds_o = args->bounds_o;
      bounds_o->lower_x = voxel->getLowestCornerX();
      bounds_o->lower_y = voxel->getLowestCornerY();
      bounds_o->lower_z = voxel->getLowestCornerZ();
      bounds_o->upper_x = voxel->getHighestCornerX();
      bounds_o->upper_y = voxel->getHighestCornerY();
      bounds_o->upper_z = voxel->getHighestCornerZ();
 }
```

Listing 2: Intersection filtering function code

```
void RTCTriangleFilterFunc(const RTCFilterFunctionNArguments * args){
        vector<Triangle*> * triList = (vector<Triangle *> *) (args->geometryUserPtr);
        RTCRay *ray = (RTCRay*) args->ray;
         int* valid = args->valid;
        RTCHit* hit = (RTCHit*) args->hit;
         IntersectContext* context = (IntersectContext*)args->context;
        Triangle * preItsTriangle = (Triangle *)context->userPtr;
        if((*preItsTriangle) == (*(triList->at(hit->primID)))){
             hit->geomID = RTC_INVALID_GEOMETRY_ID;
             ray->tfar = embree::inf;
             valid[0] = 0;
       }
 }
```

4.4 Radiation tracking

When a ray enters a non-empty voxel (i.e., a voxel contains triangles or turbid media), scattering may occur. **Figure 4.5** illustrates the scattering process inside the non-empty voxels. For a turbid voxel, suppose an incident ray with radiant power $W_{inc}(\Omega_i)$ enters the cell from direction Ω_i.Then the transmitted radiation which exits the voxel is

$$W_{trans}(\Delta l, \Omega_i) = W_{inc}(\Omega_i) \cdot T(\Delta l, \Omega_i) \tag{4-1}$$

where$T(\Delta l, \Omega_i)$ is the transmittance along segment Δl within the cell, it is calculated as:

$$T(\Delta l, \Omega_i) = e^{-G(\Omega_i)\cdot u_f \cdot \Delta l} \tag{4-2}$$

where $G(\Omega_i) = \int_{2\pi} \frac{g_f(\Omega_f)}{2\pi} |\Omega_f \cdot \Omega_i| d\Omega_f$, which is the leaf projection coefficient for direction Ω_i, Ω_f is the leaf normal orientation vector, $\frac{g_f(\Omega_f)}{2\pi}$ is the normalized leaf angle distribution function, andu_f is the foliage volume density. Thus, the total intercepted radiation within the voxel is $W_{Int}(\Omega_i) = W_{inc}(\Omega_i) \cdot [1 - T(\Delta l, \Omega_i)]$. Scattering events may happen anywhere along the interval$[0, \Delta l]$. To make it simpler, two scattering points (SP) are defined along Δl: one is for upward scattering, and the other one is for downward scattering. The angular distribution of the scattered radiation is determined by the phase function of medium defined in this cell, which is given by

$$\frac{P(\Omega_i, \Omega_s)}{4\pi} = \frac{\int_{2\pi} \frac{g_f(\Omega_f)}{2\pi} |\Omega_f \cdot \Omega_i| f(\Omega_f, \Omega_i \to \Omega_s) d\Omega_f}{\int_{2\pi} \frac{g_f(\Omega_f)}{2\pi} |\Omega_f \cdot \Omega_i| d\Omega_f} \tag{4-3}$$

where $f(\Omega_f, \Omega_i \to \Omega_s)$ denotes the leaf scattering phase function. For a Bi-Lambertian leaf facet, it is determined by the leaf diffuse reflectance (ρ_{front} and ρ_{back} for front and back reflectance, respectively) and transmittance (τ for both sides). Thus, the scattered radiation in direction Ω_s is $W_{sca}(\Omega_s) = W_{inc}(\Omega_i) \cdot [1 - T(\Delta l, \Omega_i)] \cdot \frac{P(\Omega_i, \Omega_s)}{4\pi}$. This radiation will be attenuated again during its way from the SP to the border of the cell, which gives the first-order scattering radiation outside the cell $W_{first}(\Omega_s) = W_{sca}(\Omega_s) \cdot T(\Delta l_i, \Omega_s)$. When the radiation transfers from the SP to the outside of the cell, it will be scattered again, the total intercepted radiation for each direction Ω_s is $W_{Int-first}(\Omega_s) = W_{sca}(\Omega_s) \cdot [1 - T(\Delta l_i, \Omega_s)]$. Since each scattering event will produce N new rays, the number of rays becomes unmanageable for higher order scattering events. Therefore, we approximate all higher order (≥ 2) scatterings with only one scattering point, which is defined as the center of the cell, and then a phase function $\frac{P_h(\Omega_i, \Omega_s)}{4\pi}$ which can describe the angular distribution of all the higher scattering is defined. The detailed derivation of this phase function can be found in the Appendix of (Gastellu-Etchegorry et al., 1996).

After the calculation of first-order and multi-order within cell scattering, the scattered radiation may exit the cell from any positions of the six faces, and they will enter the next non-empty voxel or exit the scene. To track the rays which exit the current voxel, each voxel is divided into n^3 sub voxels, and each voxel face is divided into n^2 sub-faces. Actually, there are $n^2.m^2$ sub-faces where m is used to improve the geometry of the scattered rays. All the J rays that exit the

same sub-face along the same direction are grouped as an unique exit point (P_E), which is calculated from all the exit points (P_1,…,P_J) weighted by their respective energy (W_j)

$$P_E = \frac{\sum_{j=1}^{J} P_j . W_j}{\sum_{j=1}^{J} W_j} \tag{4-4}$$

where W_j is the flux of the ray exit from P_j. This approach greatly reduces the number of rays. This number of rays is always smaller than the number of sub-faces, and in many cases much smaller.

For any voxel that contains triangles or parts of triangles, if a ray-triangle intersection point occurs, the scattered radiation is determined by the bidirectional reflectance distribution function (BRDF) of the triangle.

Any ray $W_{i,j}(\Omega_v)$ that exits the top scene through the voxel $V(i,j,k_{top})$ along the direction $(\Omega_v, \Delta\Omega_v)$ is stored at the top level of the top cells, in order to create a 2D power distribution (i.e., image creation) per DART upward discrete direction. If the sun is the unique source of radiation, the total power per voxel $V(i,j,k_{top})$ is transformed into radiance $L_{i,j}(\Omega_v)$ along direction Ω_v:

$$L_{i,j}(\Omega_v) = \frac{W_{i,j}(\Omega_v)}{\Delta x . \Delta y . cos\theta_v . \Delta\Omega_v} \tag{4-5}$$

where Δx and Δy are the x and y dimensions of the cell, respectively. θ_v is the zenith angle of observation. $\Delta\Omega_v$ is the associated solid angle along direction Ω_v.

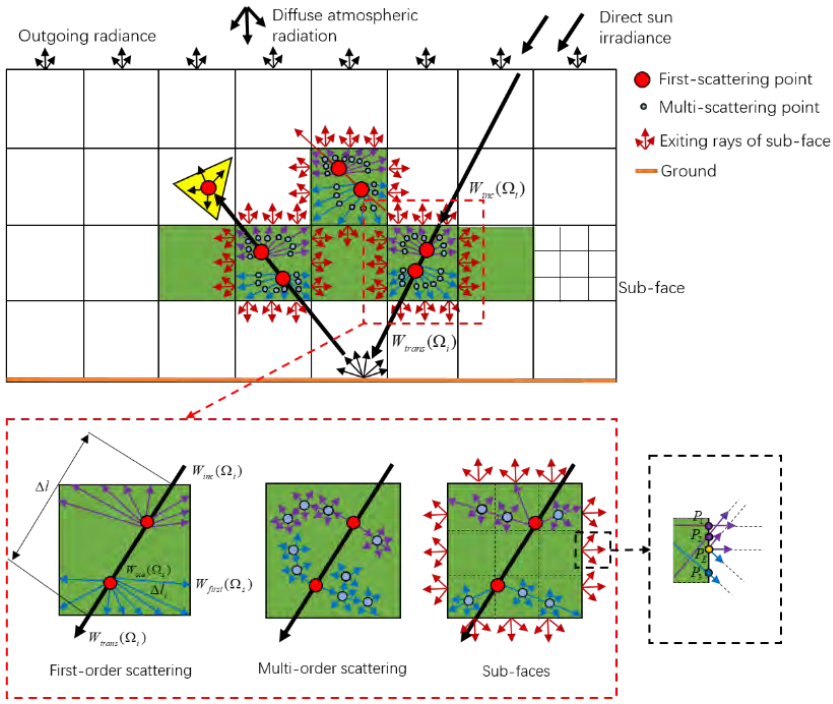


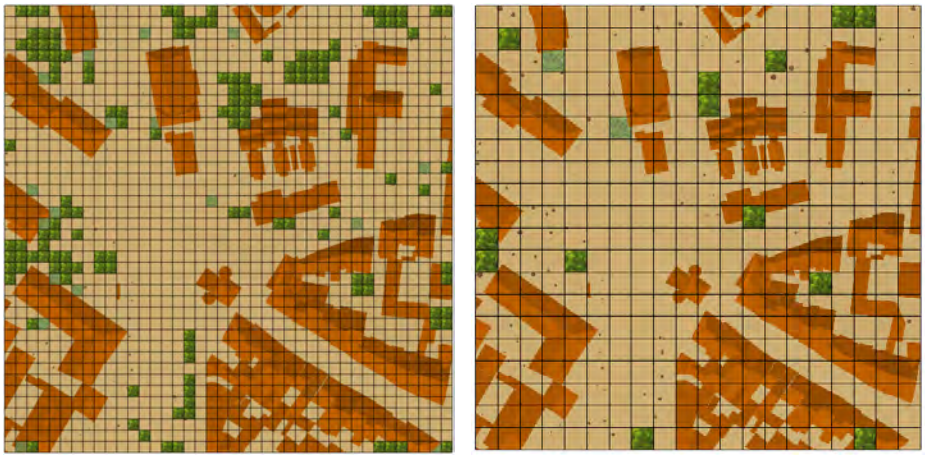
Figure 4.5: Radiation tracking in non-empty voxels.

4.5 Experiments

4.5.1 BASEL city scene

BASEL is a full 3D city model of the Basel city (Switzerland), which was adapted in the frame of the European Union funded H2020 project UrbanFluxes (http://urbanfluxes.eu/). It has a size of 7 km × 6.5 km with 1593457 triangles. One of the goals of this project was to address the challenge of mapping and mitigating urban heat by using Earth observation (EO) to identify urban energy budget (UEB) spatial patterns. For that, DART was used to simulate a time series of maps of the radiative budget with the help of satellite images and urban geometric databases (Landier et al., 2016). The approach required to run thousands of DART simulations with changing parameters (e.g., sun direction), which stressed the usefulness of the hybrid algorithm to reduce computation time and memory. Since the city scene mainly consists of houses simulated with triangles, there are large empty spaces between houses and inside the houses (**Figure 4.6**). This is ideal for testing the acceleration performance of the new scene structuring scheme. The trees included in this scene are represented as a pure turbid medium, which is described with a set of statistical parameters (LAD, LOD, etc.). Thus, the dimensions of voxels will influence the shapes of the simulated trees. For example, if voxels are too large (e.g. 10 m), trees may be only represented by a single voxel or totally removed from the scene if this tree only occupies less than 20% of a voxel. To get a reasonable representation of trees, the dimensions of voxel should be relatively small (e.g., 0.5 m). However, the realistic description of scene elements (i.e., very small voxels) and computation time are to some extent contradictory. As illustrated in **Figure 4.6** (c)-(d), a 100 m × 100 m subzone, which is composed of 3236 triangles, was selected to quantify the computational performance of the hybrid algorithm under different resolutions.

(a) (b)



(c) (d)

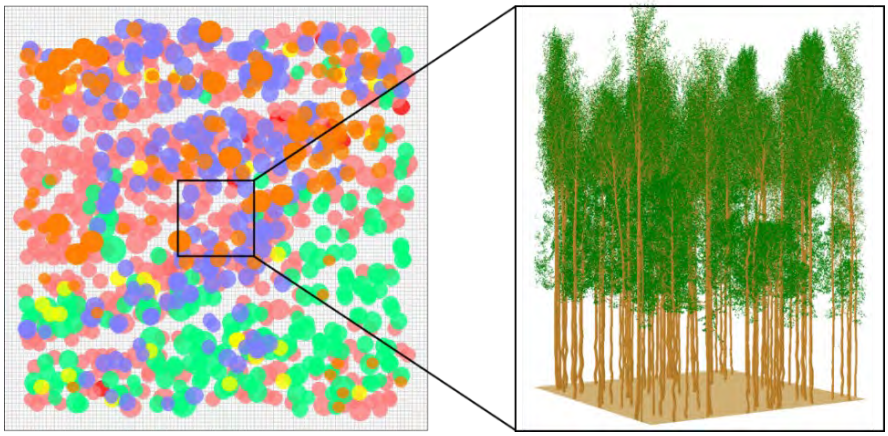
Figure 4.6: The BASEL scene under different voxel resolution.

(a) Overview of the BASEL scene; (b) 3D view of a part of the scene; (c) Voxelization of the scene in 5 m resolution; (d) Voxelization of the scene in 10 m resolution.

4.5.2 RAMI forest scene

We consider a heterogeneous forest scene, named Järvselja Birch Stand (Summer), that was used by the RAMI experiment (http://rami-benchmark.jrc.ec.europa.eu/HTML/RAMI-IV/RAMI-IV.php), in order to inter-compare radiation transfer models (Widlowski et al., 2013). This scene contains 7 species of different trees. 1059 individual trees of these species are distributed in a 100 m × 100 m plot. Because these trees are extremely well simulated and due to the limitation of computational resources, a 20 m × 20 m subplot was chosen for testing.

Figure 4.7 shows the 2D and 3D view of this subplot. Since all the elements in this scene (branches and leaves) are converted into triangle mesh, this subplot has more than 51,000,000 triangles. The bidirectional reflectance factor (BRF) image at nadir was simulated under different resolutions (0.1 m, 0.25 m and 0.5 m). The corresponding computation time and memory (RAM) were also recorded for hybrid and uniform structuring schemes.

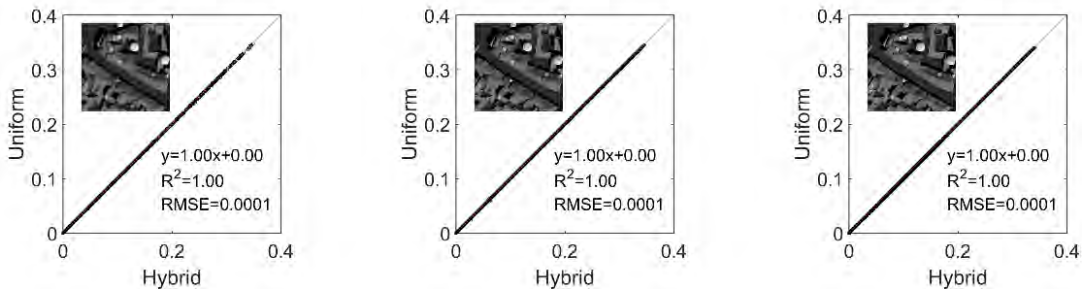


(a) (b)

Figure 4.7: The tree positions and 3D structures of the RAMI forest scene.
(a) 2D view of the 20 m × 20m subplot; (b) 3D view of this subplot.

4.6 Results and Discussion

The simulations were carried out under near-infrared bands (800 nm). Simulation accuracy, computation time and memory usage of the hybrid scene structuring approach and the original uniform approach are compared for the city of BASEL (**Figure 4.8**) and the RAMI forest scene (**Figure 4.9**). In both cases, there is no bias in accuracy at all spatial resolutions. The averaged RMSE is 0.0001 and 0.0056 for Basel city scene and the forest scene, respectively. Conversely to the uniform approach, the accuracy of the hybrid approach is nearly independent of the voxel resolution. For Basel city, simulations are accelerated by 1.4×, 1.7× and 3.7× for 1 m, 0.5 m and 0.25 m voxel resolutions, respectively (**Table 4.1**). This decrease in computer time is mainly due to the removal of the null voxels, which are more numerous at small resolutions (e.g., 0.25 m). It also improves memory usage because the null voxels are not created in the memory.



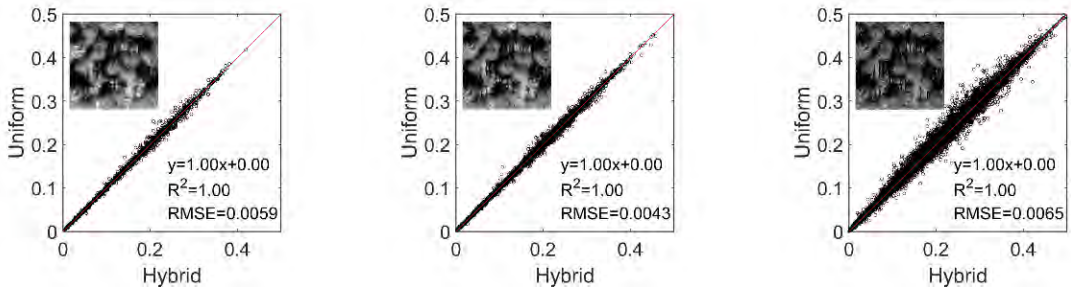
(a) (b) (c)

Figure 4.8: Accuracy comparison of the hybrid and uniform approaches when simulating RS images of Basel at three spatial resolutions: 1 m (a), 0.5 m (b) and 0.25 m (c).

Table 4.1: Computational resources used by simulating BASEL scene

BASEL scene	Null voxel	Time (s)			Memory (GB)		
Resolution	(%)	Uniform	Hybrid	Speedup	Uniform	Hybrid	Decrease
1 m	85.6%	21	15	1.4×	0.18	0.17	6%
0.5 m	91.3%	161	94	1.7×	0.86	0.56	35%
0.25 m	95.2%	2000	540	3.7×	5.4	2.7	50%

For RAMI (**Table 4.2**), the hybrid approach reduces the computation time from 67 hours to 15 minutes (258.5× speedup) when the resolution is set to 0.5 m. The main factor that influences the computation time in this situation is the ray-triangle intersection within a voxel. Under coarser resolution (e.g., 0.5 m), there are many more triangles in a voxel. The uniform approach tests the intersection for all triangles within a voxel for each incident ray, and then sorts the order according to the distance to the origin of the ray. In hybrid mode, a ray can find the closest intersected triangle directly without sorting, which can significantly accelerate the simulation when there are many triangles. With better spatial resolutions, the average number of triangles in each voxel decreases, the time used by a uniform approach also decreases. This can be confirmed by the results in **Table 4.2**. When voxel resolution changes from 0.5 m to 0.1 m, the speedup deceases from 258.5× to 4.9×. As for the memory, the reduction due to the removal of null voxels does not overcome the memory increase caused by within BVH data structures. Actually, 1181824, 193580 and 36081 within-voxel BVHs are built for resolution 0.1 m, 0.25 m and 0.5 m, respectively. It can be seen that using less within BVHs is better, in terms of memory usage, than using more BVHs to manage the same number of triangles. It stresses that for triangle-based scenes, best improvements are obtained for coarser voxel resolutions.



(a) (b) (c)

Figure 4.9: Accuracy comparison of the hybrid and uniform approaches when simulating RS images of RAMI forest scene at three spatial resolutions: 0.5m (a), 0.25m (b) and 0.1 (c).

Table 4.2: Computational resources used by simulating RAMI forest scene.

Forest Scene	Null voxel	Time (s)			Memory (GB)		
Resolution	(%)	Uniform	Hybrid	Speedup	Uniform	Hybrid	Increase
0.5 m	57.0	**244311**	**945**	**258.5×**	23.5	28.1	16%
0.25 m	69.7	31622	1221	25.9×	25.6	32.4	21%
0.1 m	84.9	14277	2911	4.9×	41.2	57.4	28%

4.7 Concluding remarks

In this chapter, a hybrid scene structuring scheme has been proposed to accelerate the radiative transfer process in complex landscapes. This approach utilizes the BVH to skip large empty areas between voxels and also to accelerate ray-triangle intersection inside a voxel. Compared to the uniform approach, the hybrid approach usually can reduce the computation time by 2× ~3× in city scenes, and more than 100 in forest scenes. In terms of accuracy, it is worth noting, that a hybrid algorithm allows one to avoid an important limitation of the Embree ray-tracing kernels library: all geometry is managed with float coding which implies important limitations in terms of accuracy for remote sensing applications over large landscapes. The hybrid algorithm escapes the limitation by using a double BVH algorithm for voxels and for triangles within the voxels.

The hybrid algorithm opens up new opportunities for allowing DART to derive very accurate remote sensing products for many scientific domains such as forestry, agriculture, urbanism, etc. For example, it allows one to use DART to simulate a time series of maps for urban radiative budget. The inversion of satellite images is another major domain of application, because it usually requires the simulation of hundred thousands of simulations.

Finally, it is worth noting that a recent DART improvement allows one to avoid the problem of trees simulated with turbid medium as mentioned above. Indeed, instead of being filled with turbid medium, the tree crowns can be filled with triangles that have the same LAD and LOD as the actual trees. This approach is expected to combine the advantages that were noted for the urban scene and the forest scene.

Chapter 5

Physical interpretation of leaf area index from LiDAR data

Summary

5.1 Research context 67
5.2 Theoretical background 69
5.2.1 LiDAR pulse 69
5.2.2 Point cloud from Gaussian Decomposition 70
5.2.3 DART workflow of point cloud modeling 72
5.3 Review of the methods in LPI estimation 73
5.3.1 Point number based (PNB) methods 74
5.3.2 Intensity based (IB) methods 75
5.4 Comparative study of LPI/LAI estimation approaches using DART 76
5.4.1 Homogeneous scene 77
5.4.2 Heterogeneous scene 78
5.5 Results and analyses 79
5.5.1 Homogeneous scene 79
5.5.2 Heterogeneous scene 84
5.6 Influences of varied leaf dimensions 87
5.7 Influences of instrumental detection threshold 89
5.8 Further discussion and perspectives 91
5.9 Concluding remarks 94

Canopy radiative transfer modeling is a complex process that combines the solution of the radiative transfer equations, as well as the canopy structure parameterization. The latter is important for understanding the interaction between solar radiation and vegetation canopies. Airborne LiDAR is one of the ideal tools to achieve this goal. Compared with terrestrial LiDAR, airborne LiDAR can acquire three-dimensional (3D) information of forest over large areas in a relatively short time. Airborne LiDAR data, nowadays, has been widely used to retrieve forest structure parameters, such as leaf area index (LAI).

However, using airborne LiDAR data to retrieve forest parameters requires a full understanding of the physical interactions between the emitted laser pulse and targets (e.g., leaves, branches and ground). Current methods to estimate the laser penetration index (LPI) and the LAI from airborne laser scanner (ALS) systems rely on point number-based (PNB) or intensity-based (IB) empirical approaches, developed using exponential or linear correlations with ground measurements of gaps, LAI, and clumping. These empirical approaches are not universal, and therefore necessitate site-specific parameterizations, as opposed to more physical retrieval methods, based on radiative transfer theory, which take into account specific attributes of the LiDAR instrument and the environmental conditions.

In this chapter, we develop an approach for a radiometric interpretation of LiDAR point cloud data generated from Gaussian Decomposition (GD) based on a combination of radar remote sensing theory and the radiative transfer models. We investigated a diversity of metrics for LiDAR point "intensity" definitions, including the peak, the standard deviation, the integral of Gaussian profiles, and the derived apparent reflectance. After a comparative review of the existing LPI and effective LAI estimation approaches with ALS data, we present a sensitivity study with various configurations for both homogeneous and heterogeneous forest 3-D scenes using a new GD point cloud modeling module developed for the latest version of the discrete anisotropic radiative transfer (DART) model. Initial modeling results show that the PNB methods become inappropriate to capture the within-crown gaps for LAI estimation as footprint size increases, and the methods using point intensity defined using either the integral of Gaussian profile or apparent reflectance can potentially be used to accurately estimate the LAI with a broad range of configurations and without any ground-based measurements.

The chapter is presented in the paper:

"Yin, T.*, **Qi, J*.**, Cook, B., Morton, D., Gastellu-Etchegorry, J.-P., Wei, S., Physical modeling and interpretation of leaf area index measures derived from LiDAR point clouds. Submitted to Remote Sensing of Environment." (under review).

5.1 Research context

LiDAR devices usually use the time-of-flight technique to generate precise range measurements based on the reflected signals of laser pulses. Lidar systems are typically categorized according to the platform and corresponding laser footprint diameter [i.e., spaceborne: larger than 10 m (Abdalati et al., 2010; Dubayah and Drake, 2000), airborne: 10-100 cms (Asner et al., 2012; Cook et al., 2013; Hyde et al., 2006) and terrestrial: tiny footprint down to a few mms (Côté et al., 2009)] and the detector system for recording return energy (e.g., discrete return, waveform, online-waveform processed point, and photon counting). Among all of them, waveform LiDAR stores the most comprehensive information for each pulse due to the entire radiometric-temporal energy profiles, which has been used for derivation of forest biophysical parameters (Hmida et al., 2017; Koetz et al., 2006) and for tree segmentations/classifications (Reitberger et al., 2009, 2008). A waveform can be converted into discrete points by deconvolution, thresholding, zero-crossing, or Gaussian Decomposition (Wagner et al., 2004), etc., or theoretically into photon counting point by modeling acquisition of a single photon detector (Yin et al., 2016). All these possibilities explain the increasing development of waveform LiDAR models in the past years (Gastellu-Etchegorry et al., 2016; North et al., 2010; Sun and Ranson, 2000) that accurately simulate the temporal radiometric amplitude profile under various instrumental (beam width, divergence, acquisition rate…), environmental (flight altitude, look angle range, terrain slope…), and vegetation (foliar reflectivity, leaves distribution…) configurations.

Despite the comprehensive information retrieved from the waveform, efficient storage of LiDAR data is critical for a tremendous number of pulses, especially for ALS and TLS (terrestrial laser scan) acquisitions with pulse repetition frequencies up to hundreds of kHz. Compared to the waveform, discrete points are easier to be managed and processed, but they cause uncertainties (Anderson et al., 2016). A point is usually stored with its position associated with an “intensity” value, which should be correlated with return power. However, precise radiometric information can be difficult to extract, especially for proprietary onboard processing systems that tend to be a black box for LiDAR user.

Converting waveform into points by using self-developed algorithms is a solution that helps to understand the background of point generation and to avoid misuse of different definitions, such as “intensity”. Among these algorithms, Gaussian Decomposition (GD) techniques were developed to approximately describe the waveform as a combination of Gaussian profiles defined by time shift centroid (t_0), peak amplitude ($\hat{P}$) and temporal standard deviation (s_p): $P(t) = \hat{P} \cdot$

$e^{-\frac{(t-t_0)^2}{s_p^2}}$ (Wagner et al., 2006). Depending on the specific LiDAR device and on the user's choice, the "intensity" could be a quantity correlated with $\hat{P}$, s_p, $\hat{P} \cdot s_p$ (Milenković et al., 2017), or the target's reflectivity [i.e., a product from Riegl's V-line series (Ullrich and Pfennigbauer, 2011)]. $\hat{P} \cdot s_p$ (proportional to the integral of the profile) is correlated with the scattering cross-section parameter in the radar equation, which depends on the size, reflectivity, and orientation of an ideal target. Although these idealized concepts and the waveform radiometric calibration (Wagner, 2010) have been discussed previously, the physical interpretation of each quantity under realistic conditions has not been completely addressed (e.g. oblique target, influences of footprint size, target size and background terrain, as well as the merged Gaussian profile by a cluster of leaf targets, etc.). For actual devices (e.g., Riegl), the radiometric validity of the output reflectivity still requires reference-target calibration at various distances (Pfennigbauer and Ullrich, 2010).

LiDAR point cloud data are regularly used to estimate the laser penetration index (LPI) and Plant / Leaf Area Index (PAI/LAI) over vegetation. Given the influence of ambiguous coefficients and residual radiometric issues, there is considerable controversy over point cloud inversion methods. For example, the accuracy of TLS inversions is affected by partial hits that depend on laser beam size and leaf size, and the estimation precision depends on the pulse density (Grau et al., 2017). For ALS, the point density within an area or a volume is usually used to estimate the LAI (Korhonen et al., 2011; Morsdorf et al., 2006; Solberg, 2010; Solberg et al., 2009, 2006) or the Leaf Volume Density (LVD) (Schneider et al., 2014). More recently, the ambiguous "intensity" information from LiDAR points also began to be employed (She-Zhou et al., 2013). However, most studies are empirical, and therefore lack a theoretical underpinning that would enable a universal application of the inversion method under various instrumental and environmental conditions. The direct computation of LPI and $\ln(LPI^{-1})$ (canopy interceptions described by Beer Lambert's law) from these approaches are divergent, and the validations of these approaches were built on regressions with ground measurements using TLS, LAI2200 or hemispherical cameras (exponent correlation of estimated LPI with reference LPI and linear correlation with reference LAI). The linear LAI correlations showed promising accurate fittings (high R^2 and low RMSE) but the fitting slopes either overestimated or underestimated results without an explicit radiometric linkage to the leaf size, angle distribution, occlusion, and clumping. Furthermore, each of the past empirical studies focused only on a single test site (limited environmental configuration) with a specific LiDAR device (fixed instrumental configuration) and very few airborne flights (limited experimental configurations). A general lack of rigorous sensitivity studies could raise questions regarding the accuracy and limitations of each approach: e.g., can current methods be

applied universally?

To physically understand the LiDAR point data and the inversion methods, a robust radiometric linkage should be built for various quantities that are involved in the point data processing, and a model should be developed to efficiently generate LiDAR point data and evaluate the inversion approaches and the sensitivity. The radiative transfer model (RTM) based on ray tracing is a suitable tool due to the least approximation in both the realistic scene representation and the temporal-spatial radiometric interactions. During recent years, many waveform RTMs have been developed based on the extension of existing credible models, including FLIGHT (North et al., 2010), DIRSIG (Goodenough and Brown, 2017), RAYTRAN (Govaerts and Verstraete, 1998), LIBRAT (Disney et al., 2009), FLiES (Kobayashi and Iwabuchi, 2008) and DART (Gastellu-Etchegorry et al., 2016). These RTMs have shown their capabilities in modeling large footprint LiDAR waveform of a single pulse (Widlowski et al., 2013). Among them, the DART LiDAR module has recently been extended to adapt multi-pulse mode (Yin et al., 2016), which can efficiently simulate ALS and TLS waveform data.

5.2 Theoretical background

The radiometric theories and the modeling in DART are presented in this section. The nomenclature of in radar remote sensing are adapted for LiDAR signal representation (Wagner, 2010; Wagner et al., 2006). For the other parts, we use the classical radiative transfer nomenclature. Detailed scientific and technical aspects of the DART waveform LiDAR module were described in (Gastellu-Etchegorry et al., 2015, 2016, Yin et al., 2016).

5.2.1 LiDAR pulse

DART has improved the modeling of LiDAR realistic acquisition geometry and energy distribution. For a LiDAR device, two Gaussian profiles are usually defined:

1. The 1-D temporal convolution of the transmitted pulse and the receiver response function:

$$S_t = \hat{S} e^{-\frac{r^2}{2s_s^2}} \tag{5-1}$$

2. The 2-D energy profile $P_l(\beta)$ within the footprint cone such that the ratio to the central maximum ($\hat{P}_{l,\beta}$) is 0.5 (i.e., FWHM), $1/\mathrm{e}^2$ or $1/\mathrm{e}$. The energy distribution within the circular boundary can be expressed as:

$$P_l(\beta) = \hat{P}_{l,\beta} e^{-\frac{\beta^2}{2s_\beta^2}} \tag{5-2}$$

where β represents the angular offset from the pulse direction and s_β is the standard deviation of the angular divergence within footprint. The energy distribution is preserved along the cone path while the footprint area increases. The half divergence of the cone is represented by β_t. The reception is defined with telescope area $A_t = \frac{\pi D_r^2}{4}$.

This configuration guarantees that the energy spread within the footprint area (radius $r_{fp} = R \cdot tan\,\beta_t + \frac{d_l}{2}$) is precisely defined, where R is the sensor-to-target range and d_l is the beam cross-section diameter at the "exit gate" of the laser generator. d_l is negligible for the airborne and spaceborne platform, but it has a critical influence on TLS footprint dimension.

5.2.2 Point cloud from Gaussian Decomposition

DART uses Gaussian Decomposition (GD) (Wagner et al., 2006) to extract points from simulated waveform data. If there are N echoes, then the temporal energy profile recorded by the receiver $P_r(t)$ can be approximated as:

$$P_r(t) = \sum_{i=1}^{N} \hat{P}_i e^{-\frac{(t-t_i)^2}{2s_{p,i}^2}} \tag{5-3}$$

For each Gaussian profile, t_i is the temporal centroid to derive the point position; $\hat{P}_i$ is the time-gated peak amplitude expressed as:

$$\hat{P}_i = \frac{\sigma_i}{C_{cal} R_i^4 s_{p,i}} \tag{5-4}$$

where C_{cal} is a calibration constant with pre-defined instrumental and experimental configurations; σ_i the radar back-scattering differential cross-section described as:

$$\sigma_i = \frac{4\pi}{\hat{\Omega}} \rho_{a,i} A_{s,i} \tag{5-5}$$

where $\hat{\Omega}$ represent an effective solid angle in which all reflected powers are assumed to be uniformly distributed; $\rho_{a,i}$ the target apparent reflectance and $A_{s,i}$ the footprint area at a distance R_i along the pulse direction $A_{s,i} = \frac{\pi R_i^2 \beta_t^2}{4}$ (neglecting d_l).

We define the apparent reflectance $\rho_{a,i}$ as the ratio of reflected radiant flux from a surface over a perpendicular Lambertian surface with reflectance = 1, which can be considered similar to the biconical reflectance factor of passive sensors with the same incident and view direction (Schaepman-Strub et al., 2006), but it is associated with every return from a pulse. Following this definition, $\hat{\Omega} = \int_{2\pi} cos\theta d\Omega = \pi$; and

$$\sigma_i = \pi \rho_{a,i} R_i^2 \beta_t^2 \tag{5-6}$$

In Eq. (5-3), $s_{p,i}$ can be expressed as:

$$s_{p,i} = \sqrt{s_s^2 + s_i^2} \tag{5-7}$$

where s_i represents the additional broadening from an oblique surface or a cluster of leaves that cannot be distinguished in a single return. s_s is an instrument-specific constant unless the temporal profiles of every transmitted pulse are known (Mallet and Bretar, 2009). $s_{p,i} \approx s_s$ if $s_s \gg s_i$ (negligible footprint size compared to the corresponding distance of the pulse duration). The ability of GD to distinguish different targets depends on several variables including target size, surface angle, gap size, and acquisition frequency, etc.

For each point output, the subjected results are categorized into five types of "intensity" values (DART implementation described in Appendix A):

1. Peak Amplitude by fast detection $\hat{P}_i$ (without GD): The corresponding peak amplitude determined by zero-crossing.
2. Fitted Peak Amplitude $\hat{P}_i$ (with GD): Non-linear least-square minimization and curve-fitting (Newville et al., 2016) is used to determine t_i, $\hat{P}_i$ and $s_{p,i}$.
3. Standard Deviation $s_{p,i}$ (GD) (unit: number of bins).
4. Integral I_i (GD) described by Milenković et al. (2017):

$$I_i = \int \hat{P}_i e^{-\frac{(t-t_i)^2}{2s_{p,i}^2}} dt = \sqrt{2\pi} \cdot \hat{P}_i \cdot s_{p,i} \tag{5-8}$$

   I_i corresponds to the total return power of each Gaussian profile, which could be linked to σ_i by

$$\sigma_i = \frac{I_i C_{cal} R_i^4}{\sqrt{2\pi}} \tag{5-9}$$

5. Apparent reflectance $\rho_{a,i}$ (GD) : By substituting Eq.(5-9) into Eq. (5-6), we get $\rho_{a,i}$ as a function of I_i:

$$\rho_{a,i} = \frac{I_i C_{cal} R_i^2}{\pi\sqrt{2\pi} \cdot \beta_t^2} \tag{5-10}$$

   $\rho_{a,i}$ is physically equivalent to the LiDAR backscattering coefficient provided in Wagner (2010) and Milenković et al. (2017). The additional $\pi \cdot \beta_t^2$ in the denominator converts it to a more intuitive reflectance value. Comparing with the cross-section, the backscattering coefficient was suggested for waveform radiometric calibration due to the normalization relative to the footprint area.

5.2.3 DART workflow of point cloud modeling

Figure 5.1 shows the workflow for simulating LiDAR points. Initially, DART reads the 3-D scene (i.e., abstract description or field of 3-D objects) with user-defined experimental configurations for LiDAR acquisition (e.g., altitude, platform trajectory, pulse divergence, scan density, control points, etc.). The stages enclosed in the dashed box are DART internal processes. DART uses a quasi-Monte Carlo Ray Tracing approach to model the waveform for each pulse (Gastellu-Etchegorry et al., 2016). DART-simulated waveforms can be directly exported or processed with GD before being exported:

- Output exportation to LAS

The DART-simulated waveforms are stored as a binary file. In the next step, this binary file can be converted into standard LiDAR data formats to adapt the existing processing software. Previous work documented the output conversion into the sorted pulse data (SPD) format (Gastellu-Etchegorry et al., 2016). Here, we implement an approach that processes the binary file into LAS 1.3 point cloud through GD based on laspy library. The points are stored in the LAS file, with the possibility to export the waveforms to another WDP file that is linked to each decomposed point (Header description in (ASPRS, 2009), point type 1 or 4). The file is supported by existing LAS visualization or processing software, e.g., CloudCompare (Girardeau-Montaut, 2011).

- Internal waveform processing

Waveforms are decomposed into points in DART code that are saved as matrices in a text file. Each line represents a point. Columns store 3D position and "intensity" information including peak amplitude $(\hat{P})$, temporal standard deviation ($s_{p,i}$), integral ($\propto \hat{P} \cdot s_{p,i}$), return index, number of returns per pulse, etc. This text file can be imported into CloudCompare for further processing.

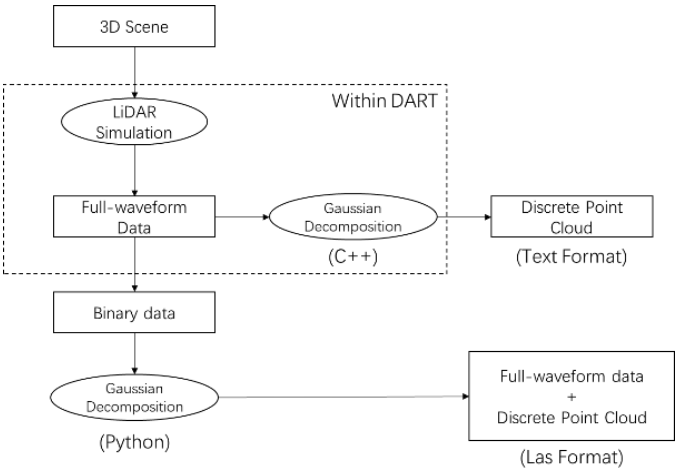


Figure 5.1: Workflow of DART outputs: discrete points in text format or discrete points with associated waveforms in LAS format.

The dual output option provides great flexibilities in different applications. For example, Option 1 supports the validation of algorithms that work with ALS waveform data in LAS format. Option 2 may be preferable for numerous points in TLS simulations instead of waveforms. In that case, storing waveforms in a binary file would require tremendous unnecessary computer memory and hard disk space. Also, users may want to investigate the sensitivity of various input instrumental variables (d_l, β_{fp},...) or various output intensity selections ($\hat{P}, \sigma$, or $\hat{P} \cdot \sigma$...) for different applications (e.g., classification, LAD inversion, etc.).

5.3 Review of the methods in LPI estimation

The LPI is usually defined as the percentage of the power that passes through a vegetation volume:

$$LPI = e^{-G \cdot \omega \cdot LAI / \cos\theta} \tag{5-11}$$

where G is the unit leaf area projection along the pulse direction; ω the clumping index to convert from effective LAI into actual LAI (Jing M Chen et al., 1997); θ is the zenith angle of the incident power that passes the vegetation.

The DART simulated point cloud is applied for sensitivity studies in LPI and Plant Area Index / Leaf Area Index (LAI) estimation using ALS. The current approaches equally divide the scene into grids and aggregate information (point number or intensity) within each grid area and convert results into a raster map. If all the LiDAR pulse direction are assumed to be nadir, and only the vegetation and ground returns are classified for the 2D mapping, then $\theta = 0°$. This is also the assumption for all existing empirical approaches to generate LPI/LAI map. The effective LAI is: $LAI_{eff} = \omega \cdot LAI$. Explicitly, the LPI can be sampled by either the area fraction or power fraction of ground returns:

$$LPI = \frac{P_{t,g}}{P_{t,g} + P_{t,v}} = \frac{A_g}{A_g + A_v} = \frac{\frac{I_g \cdot R_g^2}{\rho_{a,g}}}{\frac{I_g \cdot R_g^2}{\rho_{a,g}} + \frac{I_v \cdot R_v^2}{\rho_{a,v}}} \tag{5-12}$$

where $P_{t,g}$ and $P_{t,v}$ are the total incident LiDAR power onto the ground and leaves, respectively; A_g and A_v are the cover area of ground and leaves, respectively; I, R and ρ_a are aggregated values within the footprint. $P_{t,g}$ and $P_{t,v}$ cannot be directly estimated from LiDAR data. Depending on the expression behind the two right-side equal signs, the ALS-based inversion of LPI is categorized into either point-number-based (PNB) or intensity-based (IB).

5.3.1 Point number based (PNB) methods

Assume the pulses are equally or randomly distributed within the area, then the number of returns from leaves (N_v) and ground (N_g) serve as statistical samplings of A_v and A_g. Given a small footprint size such that only one return is retrieved from each pulse, and a large pulse density to get a convergent value with a low variance, the LPI is expressed as:

$$LPI_{all} = \frac{N_g}{N_g + N_v} \tag{5-13}$$

where N_g and N_v are the total point number of ground and vegetation, respectively. Monitoring larger regions with ALS requires higher altitude which makes footprint size larger and increases the chance of partial hits. Multi-return LiDAR can capture several returns while a pulse penetrates through a canopy. LPI_{all} describes the sampling of A_v and A_g for only single-return LiDAR, but it has also been studied for multi-return LiDARs (Hu et al., 2018; She-Zhou et al., 2013). To balance the weight of each return from a pulse, Fleck et al. (2012) and Schneider et al. (2014) proposed the weighted LPI expression below:

$$LPI_{weighted} = \frac{N_{g,1} + \frac{1}{2}N_{g,2} + \frac{1}{3}N_{g,3} + \frac{1}{4}N_{g,4} \dots}{N_{t,1} + \frac{1}{2}N_{t,2} + \frac{1}{3}N_{t,3} + \frac{1}{4}N_{t,4} \dots} \tag{5-14}$$

where $\mathrm{N_{t,i}} = \mathrm{N_{g,i}} + \mathrm{N_{v,i}}$ is the total number of returns per pulse with i=1, 2, 3… returns ($N_{t,i} = i$), and $N_{g,i}$ is the respective number of ground returns. For $LPI_{weighted}$, equal weights are assigned to each pulse and a weight of $1/i$ is assigned to each point. In old systems which record either 2 returns or 4 returns in maximum (Lovell et al., 2003; Solberg, 2010; Solberg et al., 2006), LPIs were also estimated by using either the first or the last returns:

$$LPI_{first} = \frac{N_{g,only} + N_{g,first}}{N_{only} + N_{first}} \tag{5-15}$$

$$LPI_{last} = \frac{N_{g,only} + N_{g,last}}{N_{only} + N_{last}} \tag{5-16}$$

where N_{first} and N_{last} are the numbers of first and last returns, respectively; N_{only} is the number of pulses with only one return generated from a single target (e.g., $N_{g,only}$ is the number of pulses that directly hit the ground) or from merged multiple points with indistinguishable targets (e.g., $\mathrm{N_{v,only}}$ is the number of pulses that hit a dense vegetation cluster). LPI_{first} tends to inform on between-crown gaps, whereas LPI_{last} informs on both the between-crown gaps and the within-crown gaps. Usually, LPI_{first} underestimates and LPI_{last} overestimates the penetration.

To balance that, LPI_{both} based on the average number of first and last returns (Solberg et al., 2009) was stated as being more appropriate:

$$LPI_{both} = \frac{N_{g,only} + 0.5(N_{g,first} + N_{g,last})}{N_{only} + 0.5(N_{first} + N_{last})} \tag{5-17}$$

It is critical to note that the numbers N in the denominators in Eq. (5-13)-(5-17) correspond to laser pulses that produce at least one return. Pulses that do not give returns are filtered.

5.3.2 Intensity based (IB) methods

5.3.2.1 With prior knowledge of vegetation and ground

Theoretically, the intensity-based LPIs are more sensitive to LAI and LAD. However, the IB methods were much less addressed due to the uncertain definition ($\hat{P}$, I, $s_{p,i}$, or ρ_a) and measurements. Here, I is assumed to be the intensity output. If we define the distance-weighted integral of Gaussian profile $I' = I \cdot R^2$, Eq. (5-12) becomes:

$$LPI_{IB,\gamma} = \frac{\sum I_g{}'}{\sum I_g{}' + \gamma \cdot \sum I_v{}'} \tag{5-18}$$

where all I' are cumulated within the square area; $\gamma = \frac{\rho_{a,g}}{\rho_{a,v}}$ is the apparent reflectance ratio of ground and leaf that are statistically aggregated.

$\rho_{a,g}$ and $\rho_{a,v}$ can be derived from the natural reflectivity measurements of the ground ($\rho_{n,g}$) and the leaf ($\rho_{n,v}$). If the variation in the elevation slopes within the area is negligible and the surface is Lambertian, $\rho_{a,g} = \rho_{n,g} \cdot cos\theta_g$, where θ_g is averaged over the area. The estimation of θ_g relies the terrain model reconstruction using the ground returns (Zhang et al., 2016). $\rho_{a,v}$ is linked to the back-scattering transfer function $T(\Omega_i)$ and the reflectivity of the leaf ($\rho_{n,v}$) by

$$\rho_{a,v} = \rho_{n,v} \cdot T(\Omega_i) = \rho_{n,v} \cdot \frac{\int_{2\pi} \frac{g(\Omega_f)}{2\pi} cos^2\,\theta_{f,i}\, d\Omega_f}{\int_{2\pi} \frac{g(\Omega_f)}{2\pi} cos\,\theta_{f,i}\, d\Omega_f} \tag{5-19}$$

where $\frac{g(\Omega_f)}{2\pi}$ is the leaf angle distribution (LAD) of a leaf cluster; $\theta_{f,i}$ is the angle between leaf normal and Ω_i. Thus, we get

$$\gamma = \frac{\rho_{n,g} \cdot cos\theta_g}{\rho_{n,v} \cdot T(\Omega_i)} \tag{5-20}$$

γ was usually considered as a constant for simplicity. For example, Lefsky et al. (1999) proposed $\gamma = 0.5$ for LiDAR devices with 1064 nm wavelength. The same value was also used at 1550nm (She-Zhou et al., 2013). For the ideal configuration {horizontal terrain, vertical pulse incidence, spherical leaf angle distribution}, we get $cos\theta = 1$ and $T(\Omega_i) = 2/3$, which gives:

$$\gamma_{\text{ideal}} = \frac{3 \cdot \rho_{\text{n,g}}}{2 \cdot \rho_{\text{n,v}}} \tag{5-21}$$

where $\rho_{n,g}$ and $\rho_{n,v}$ can be retrieved from ground measurements as prior values. Eq. (5-21) is a general expression for any wavelength with measured $\rho_{\text{n,g}}/\rho_{\text{n,v}}$.

5.3.2.2 Without prior knowledge

Without any assumption, the four unknowns in Eq. (5-20) can vary a lot across the area, so the constant assumption of γ is imprecise. Indeed, the denominator of Eq. (5-18) represents the cumulated ground intensity as if there is no vegetation, as γ is the factor to calibrate vegetation intensities into ground intensities. Milenković et al. (2017) proposed an alternative method for full waveform LiDAR, which utilizes the pure-ground pulse (pulses that generate only ground returns). All the pulses and the return intensities were classified into three types: pure-ground (intensity I'^{pure}_{g}), pure-vegetation, and vegetation-ground (with ground return intensity $\text{I}_{\text{g,v}}$). The returns of 2nd and 3rd types were associated with the nearest 1st types (defined as $I^{nearest}_{v}$ and $I^{nearest}_{g,v}$). Then Eq. (5-18) becomes:

$$LPI_{IB,nearest} = \frac{\sum I'^{pure}_{g} + \sum I'_{g,v}}{\sum I'^{pure}_{g} + \sum I'^{nearest}_{g,v} + \sum I'^{nearest}_{v}} \tag{5-22}$$

A great advantage of $LPI_{IB,nearest}$ compared to $LPI_{IB,\gamma}$ is that the prior measurement of γ is unnecessary, but $LPI_{IB,nearest}$ relies on the pure-ground pulses, which requires large gaps to increase the probability of pure-ground pulses. Another critical assumption is that $\rho_{a,g}$ for both the pure-vegetation and the vegetation-ground pulses are approximately equal to $\rho_{a,g}$ of the nearest pure-group pulses. Considering the $\rho_{a,g}$ calculation, $\rho_{n,g}$ could vary due to understory defoliations and wastes, and θ_g could be influenced by insufficient sampling, whereas more samples (including ground returns of vegetation-ground pulses) can be used for calculating $\theta_{g,i}$ (from terrain model reconstruction) in $LPI_{IB,\gamma}$ estimation.

5.4 Comparative study of LPI/LAI estimation approaches using DART

In practice, the estimated LPI is not directly used to estimate LAI. Instead, it is correlated with the ground-measured penetration index that may be equivalent to gap fraction/transmittance of a vertical direction or within a certain solid angle near nadir or the LAI or effective LAI derived from LAI-2000 or hemispherical camera images. Reference and estimated LPI and LAI are linked:

$$LPI_{ref} = LPI_{est}^{c}, \qquad LAI_{ref} = \alpha\, ln\left(\frac{1}{LPI_{est}}\right) = \alpha G\omega \cdot\ LAI_{est} \tag{5-23}$$

where c and α are the exponents and linear fitting coefficients with the reference values for the estimated LPI and LAI, respectively. In this section, the simulated point clouds of various critical parameters are studied and inter-compared with the LPI/LAI estimation approaches in Section 5.3.

The DART scene is a 3D array of cells that contain the Earth's elements (i.e., turbid medium, facets, fluids). Vegetation can be simulated either as facets or as turbid medium. If needed, for example if the number of facets is very large, facets used for simulating vegetation can be converted into turbid medium, with statistical information (LVD, LAD, leaf dimension and element shape, etc.) determined per cell. Two approaches are possible for creating scenes with vegetation (e.g., trees) that is simulated with facets. 1) DART imports 3D elements, possibly very complex, that are created elsewhere. 2) DART directly creates 3D elements with schematic shapes (e.g., tree with ellipsoidal crown) and homogeneously or randomly filled with facets with user defined LAI, LAD, etc. Here, the later approach is used to create a homogeneous scene and an heterogeneous scene and to study their surface interactions with LiDAR pulses. With this, DART can conveniently set up the ideal reference parameters and compare with the inversion results.

We use the configurations of Riegl VQ-480i (**Table 5.1**), which is a multi-returns device with online waveform processing. To simulate realistic ALS flight, we compute the position of each pulse according to flight height, moving speed, pulse repetition rate, and scanning speed, etc. The average pulse density is estimated with $\frac{PRF}{2vR\cdot tan30°}$ where the pulse repetition frequency PRF and the flight speed v are assumed to be constants. As we are interested in ideal configurations, the incident zenith angle of each pulse is manually modified to be 0° ($\theta_{g,i} = 0$) for each incident pulse. At 1550 nm, leaf and ground reflectance are set to 0.34 (general leaf) and 0.24 (brown moss), respectively.

Table 5.1: LiDAR parameters used in DART simulations.

Parameter	Value	Parameter	Value
Wavelength	1550nm	Cross-track FOV	±30°
Laser sampling interval	1 ns	Scanning Speed	100 lines/second
Laser beam divergence	0.3 mrad	Laser PRF	200 kHz

5.4.1 Homogeneous scene

The test scene consists of a square vegetation plot and a flat ground surface (**Figure 5.2**). Each

leaf is represented by a square (5 cm × 5 cm). The plot has a height of 10m with 2m space below the canopy, which facilitates the classification of ground returns and vegetation returns. The leaves within the canopy volume (22 m × 22 m × 10 m) have a spherical angle distribution. Therefore, we set $cos\theta = 1$, $G = 1/2$, $\omega = 1$, and $T(\Omega_i) = 2/3$ in Eq. (5-11) and (5-21).

The simulated surface is $22\text{m} \times 22\text{m}$ to avoid boundary effects. In this study, the sensor height R varies from 50 m to 1000 m. The corresponding footprint diameter varies from 0.015 m to 0.30 m. I_i is recorded for each point decomposed from the simulated waveforms. LPI is calculated for each $2\text{m} \times 2\text{m}$ area. The average LPI and the standard deviation over the whole region are computed and compared with the reference LPI estimated by Eq. (5-11), i.e., $LPI_{ref} = e^{-0.5LAI_{ref}}$. The footprint diameter ($\propto R$) and LAI_{ref} are the variables for this study.

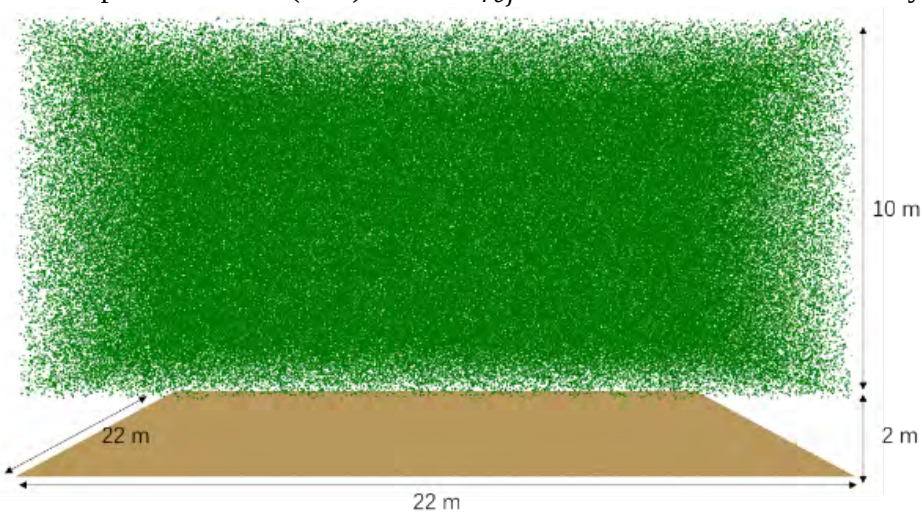


Figure 5.2: Illustration of 3D Homogeneous test scene

5.4.2 Heterogeneous scene

The heterogeneous scene (100 m × 100 m) is aimed to test the influences of large gaps between tree crowns on the LiDAR-derived PAI (**Figure 5.3**). An equal number of ellipsoidal and conical crowns (327 each) are randomly distributed. The height and diameter of the crowns are 12 m and 3.6 m, respectively. Two simulations were conducted under different LVDs, which are constantly set to 0.25 and 0.5 for each crown. This makes the LAI for the whole scene equal to 1 and 2, respectively. The LiDAR data is simulated with a flight altitude R =1000 m and a 0.30 m footprint diameter. The LPI is calculated with a 10 m × 10 m plot area. In this case, LPI_{ref} cannot be directly estimated due to the inhomogeneous distribution of leaves. Instead, the radiation received by the ground surface is estimated using a tremendous amount of nadir incident rays. DART can map the fraction map of intercepted irradiance by the ground to the total top-of-scene irradiance as LPI_{ref} for each area. The associated LAI map is also computed.

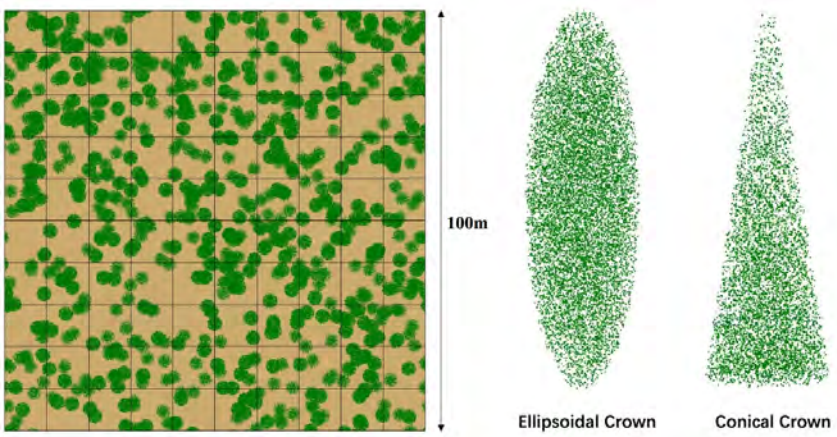


Figure 5.3: The heterogeneous scene. It is made of randomly distributed trees with ellipsoidal and conical crowns.

5.5 Results and analyses

5.5.1 Homogeneous scene

The estimations from both PNB ($LPI_{all}, LPI_{weighted}, LPI_{first}, LPI_{last}, LPI_{both}$) and IB ($LPI_{\gamma=1.06}, LPI_{nearest}$) methods are illustrated in **Figure 5.4** with varying LAI (1 - 6) and footprint diameter ($0.015\text{m} - 0.30\text{m}$). The dashed line gives the reference LPI value, and the shaded region of each line represents the variance of LPI estimation. $\gamma = 1.06$ is the ideal value computed from Eq. (5-21) for the IB methods. I'^{pure}_{g} can be precomputed precisely with known LiDAR configuration and ground reflectance. For the PNB methods, LPI_{first} always gives the minimum underestimation and LPI_{last} always gives the maximum overestimation. The estimations of the other approaches are in-between these two values with the following trend: $LPI_{last} > LPI_{both} > LPI_{weighted} \approx LPI_{all} > LPI_{first}$, which is consistent with past works. As reference LAI increases, at first, the methods tend to underestimate the reference value, and then tend to overestimate it. For a fixed LAI, it could be depicted that for infinitesimal footprint size (much smaller than the leaf dimension), all the estimations converge to the reference value because only one return from either vegetation or ground could be retrieved for each pulse ($N_{g,first} = N_{g,last} = 0; N_g = N_{g,only}$). This is also the reason why TLS can measure the gap fraction more precisely (Danson et al., 2007). The total number of returns are illustrated in **Figure 5.5**a. The average number of returns per pulse converges to 1 as footprint size tends to 0.

LPI_{last}, LPI_{first} and LPI_{both} synchronously converge to 1, 0, and 0.5 respectively with increasing footprint size. LPI_{first} depicts the large gaps relying on the number of the pure-ground

pulses, whereas LPI_{last} depicts small gaps within the canopy replying on the number of vegetation-ground pulses. From the case of infinitesimal footprint size, as footprint size becomes larger, the chance of getting vegetation-ground pulse grows in 2 ways:

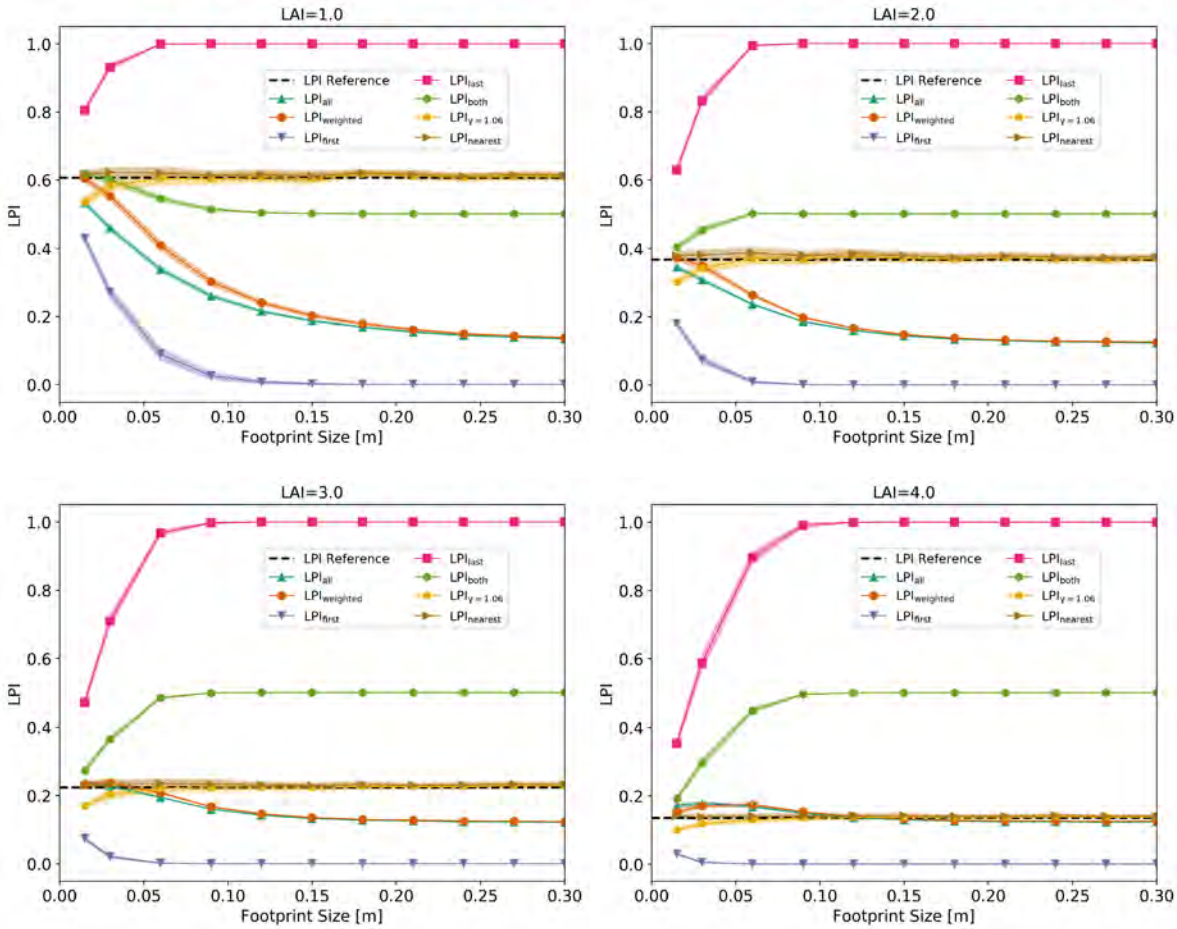
- The initial pure-ground pulse becomes vegetation-ground pulse, which reduces $N_{g,only}$, but increases $N_{g,last}$ with the same amount of points. Under this condition, the numerator of Eq. (5-15) becomes smaller (decreasing LPI_{first}), but the numerator of Eq. (5-16) keeps constant (no effect on LPI_{last}).
- The original pure-vegetation pulse becomes vegetation-ground pulse, which reduces $N_{v,only}$ but increase $N_{g,last}$ with the same amount of points. This condition does not affect LPI_{first}, but increases LPI_{last}.

It should be noticed that although LPI_{both} balances the underestimation of LPI_{first} and the overestimation of LPI_{last}, it does not support the arguments that LPI_{both} gives a more accurate estimation, and further that LPI_{both} can directly estimate LAI without ground measurements as mentioned by Hu et al. (2018). The number of ground returns per pulse is shown in **Figure 5.5**b. The trend, as expected is coherent with the convergence of LPIs estimated by these three approaches. As the footprint diameter increases, every pulse tends to produce a ground return, so the value converges to 1. As the LAI increases, the convergence becomes slower because of the reduced probability of energy penetrating through the canopy.

On the other hand, the other two PNB methods $LPI_{weighted}$ and LPI_{all} performs similarly except for the cases with smaller footprint sizes where $LPI_{weighted}$ is a bit closer to LPI_{ref} than LPI_{all}. The saturation speeds of these two approaches are much slower than for the other three approaches because all returns are counted. The average number of returns per pulse shown in **Figure 5.5**a converges together with $LPI_{weighted}$ and LPI_{all}. Given a large enough pulse density to compute the LPI with statistically low variance, the speed of LPI convergence relies on the size of within-crown gaps that is determined by two factors: the ratio of footprint diameter over the leaf dimension, and the LAI under homogeneous environment. The former parameter determines the probability of a pulse's capability in generating multiple returns in leaf interactions. The later parameter determines the number of returns that could be generated for each pulse. For actual LiDAR acquisition over a homogeneously dense forest, the footprint size should be controlled within a domain such that as LAI decreases, the LPI estimation does not converge to a constant value (e.g., 1, 0.5, 0, etc.). Otherwise, the sensitivity will vanish. The estimation of this domain requires a complete understanding of the leaf dimension, leaf density, and device configurations. Therefore, from the sensitivity study above $LPI_{weighted}$ and LPI_{all} are theoretically preferable comparing with the other three approaches for a broader acceptable range of the value of footprint

size and LAI.

The two IB methods $LPI_{\gamma=1.06}$ and $LPI_{nearest}$ are much closer to the $\mathrm{LPI_{ref}}$ than the PNB methods. They are generally consistent with the reference LPI at any footprint size, except for $LPI_{\gamma=1.06}$, which underestimates the LPI when footprint size is smaller than 0.1 m. Since $LPI_{\gamma=1.06}$ depends on the statistic parameters of vegetation (i.e. LAD), it requires that each pulse should cover a considerable number of leaves. When footprint size is too small, this method will produce biases. For $LPI_{nearest}$, it only considers the ground returns and performs well for all the footprint sizes, regardless of the structural and optical properties of vegetation. Indeed, IB methods are more adaptable for various LAIs because they well depict the within-crown gaps in the scene. In contrast, under certain cases, some PNB methods are not sensitive to LAI change. As LAI increases, the intensity of ground returns becomes weaker than vegetation returns, but it can still produce a point. Thus, the average number of returns for different LAIs converges to the same value in **Figure 5.5**.



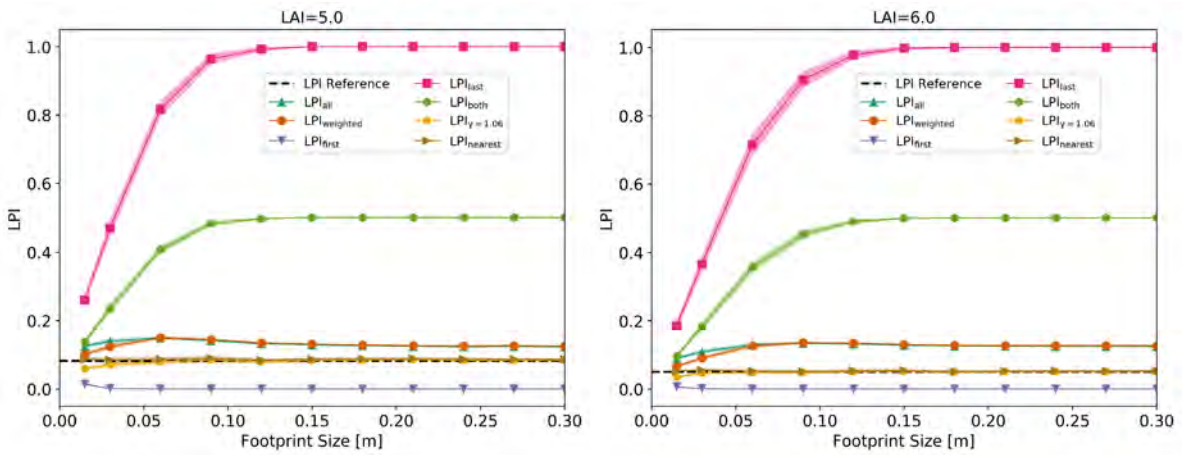
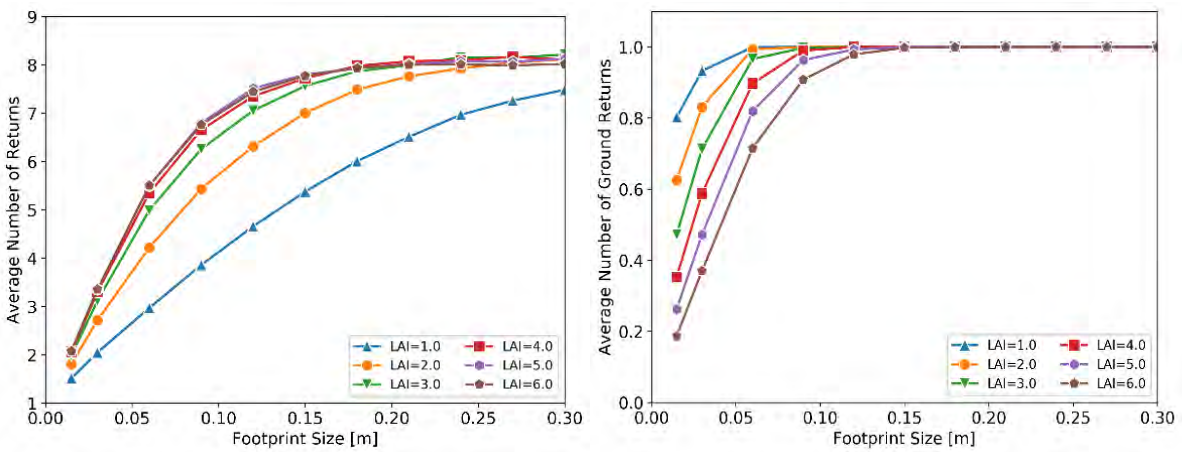


Figure 5.4: LPI estimated by different methods with various LAI and footprint diameter.



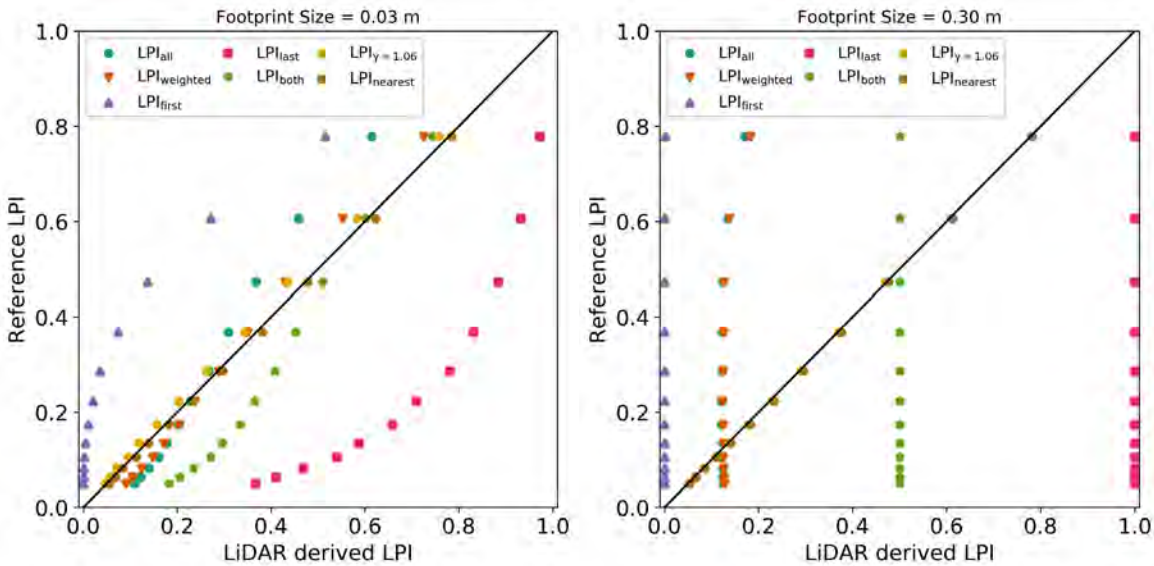
(a) (b)

Figure 5.5: Average discrete returns for all pulses.

(a) Average number of returns under different footprint size; (b)Average number of ground returns.

It should be noted that $LAI_{eff} = LAI$ in the simulated scene because $\omega = 1$. The product αG derived from Eq. (5-11) serves as the linear correlation coefficient that links the LAI_{ref} with the estimated LAI_{est}. This correlation is studied for various LPI estimation approaches. For ideal fitting, $c = 1$, $\alpha = 2$, and $\alpha \cdot G = 1$. In this experiment, three footprint diameters (0.03m, 0.15m and 0.30m) are chosen for the evaluation of linear correlation using the same dataset presented in **Figure 5.4**. From the subfigures, for 0.03 m footprint size that mimics a TLS configuration, the evaluations of all the approaches are not saturated. For 0.30 m footprint size of a regular ALS configuration, LPI_{first}, LPI_{last} and LPI_{both} become saturated, and $LPI_{weighted}$ and LPI_{all} are not saturated for small LAI, but saturated for large LAI. **Figure 5.6** (for LPIs) and **Figure 5.7** (for LAIs) illustrate the scatterplot of the references and the estimations using different methods.

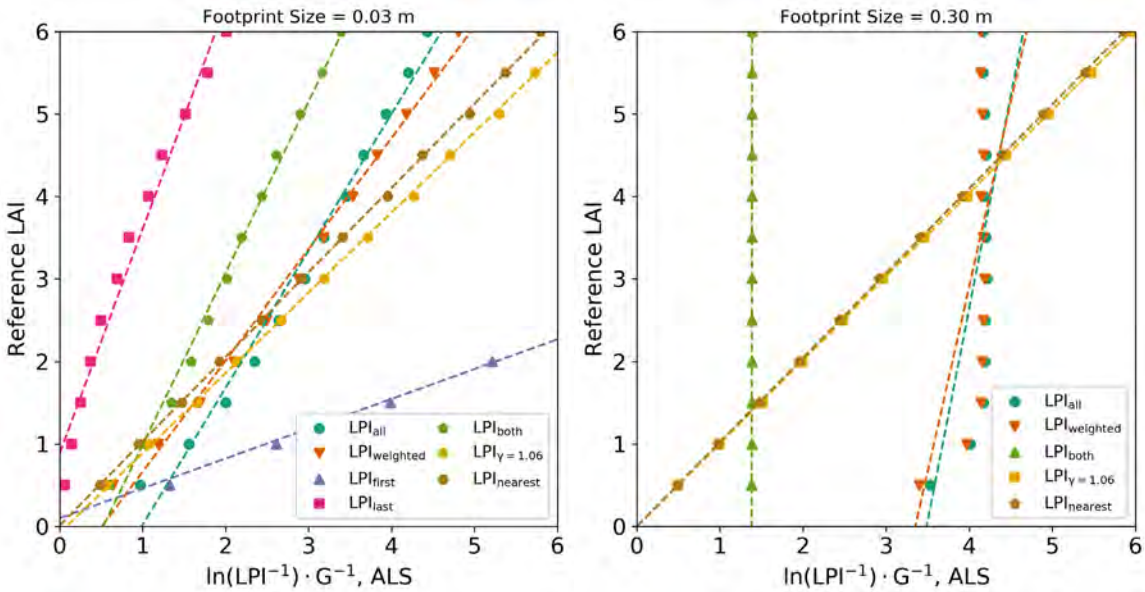
For tiny footprint size in subfigures (a), both the exponential and the linear correlations are well-established for all the approaches. Indeed, the footprint diameter at this level is close to the TLS measurements. An almost 1:1 relationship for LPI ($c_{\gamma=1.06} = 0.95, c_{nearest} = 1.02$) and LAI ($[\alpha G]_{\gamma=1.06} = 0.95$, $[\alpha G]_{nearest} = 1.02$) is observed for IB methods ($LPI_{\gamma=1.06}$ and $LPI_{nearest}$). For PNB methods, the c coefficient for $LPI_{weighted}$ and LPI_{all} are closed to 1 ($c_{weighted} = 1.15$ and $c_{all} = 1.16$) while the other PNB methods might provide either an underestimation ($c_{last} = 3.38$, $c_{both} = 1.67$) or an overestimation ($c_{first} = 0.37$). In **Figure 5.7**a, the linear regressions of PNB methods do not pass through the origin. The LAI correlations are almost linear for the PNB methods for LAI values > 1 and converge to the origin with varying c as LAI value approaching 0, which is caused by the boundary partial hit effect due to the non-negligible ratio of footprint size over leaf size with small total LAI value. It shows that the configuration of TLS can capture the correlation of within-crown gaps with tiny footprint laser.



(a) (b)

Figure 5.6: The relationship between LiDAR derived LPI and Reference LPI.

(a) Footprint size = 0.03 m; (b) Footprint size = 0.3 m.



(a) (b)

Figure 5.7: Reference LAI plotted against lnLPI^{-1}.

(a) Footprint size = 0.03 m; (b) Footprint size = 0.3 m; LPI_{first} and LPI_{last} are excluded in (b), since their $\ln(LPI^{-1})$ values are always equal to infinity and 0, respectively.

The comparison results of PNB methods infer that multi-return LiDAR can more accurately estimate the leaf penetration and leaf area than single-return LiDAR, and that he pulse-weighted approaches ($LPI_{weighted}$ and LPI_{both}) are more precise than the point-weighted approaches (LPI_{all}). The fitted coefficient of αG of LPI_{first} using modeling approaches can correct LAI measured from single-return TLS data (Béland et al., 2011).

For larger footprint size (0.15 m and 0.30m), IB methods still maintain a 1:1 correspondence. However, all the PNB methods show an ill-performed sensitivity for both LPI and LAI fittings (almost vertical lines) which become even worse as footprint size increases. LPI_{both} shows a negligible variation as LAI changes. Non-linear correlation can be observed for $LPI_{weighted}$ and LPI_{all} with LAI<3 (0.15 m footprint) and LAI<1.5 (0.30 m footprint). As LAI increases, $LPI_{weighted}$ and LPI_{all} get saturated gradually. This effect is explained by the merging of Gaussian profiles with large LAI, which makes the PNB methods inappropriate for capturing within-crown small gaps, but they can still be used for larger gaps in a heterogeneous scene as illustrated below.

5.5.2 Heterogeneous scene

A major part of the heterogamous scene is occupied by empty space between tree crowns. Conversely to the last section, the work is focused on the landscape-scale areas (10 m × 10 m),

large enough such that large crown gaps always exist. By assuming that each area consists of a homogeneous vegetation part with mean LAI and an empty part if the vertical canopy cover fraction within each area is defined as f_{vcc}, we get an approximated expression for the LPI in the heterogeneous scene:

$$LPI = f_{vcc} \cdot e^{-G \cdot \omega \cdot LAI / f_{vcc}} + (1 - f_{vcc}) \tag{5-24}$$

where LAI represents the landscape scale, and LAI/f_{vcc} is the LAI of the canopy cover. It should be noted that ω represents only the within-crown clumping index, and $\ln(LPI^{-1})$ is not linear with LAI, although the past publications tested exponent relation for the estimated LPI and linear relation for estimated $\ln(LPI^{-1})$ as described in Eq. (5-23).

Figure 5.8 shows the correlation between LiDAR-derived LPIs and the reference values (both LPI and LAI) under different LAIs, with the same larger footprint size used in the study of the homogeneous scene (0.3 m). The reference LPIs are also evaluated by using DART radiative budget of the ground. For LPI correlation (**Figure 5.8**a and **Figure 5.8**c), $LPI_{nearest}$ and $LPI_{\gamma=1.06}$ are constantly consistent with Reference LPI. LPI_{first}, as expected, shows an almost equal fit with the fraction of vertical crown cover (f_{vcc}) (red scatter in **Figure 5.8**a and **Figure 5.8**c), therefore can give an estimation of f_{vcc}. The underestimation is because of the multiple returns at the boundary of the crown, where less first returns from the ground are acquired comparing with a LiDAR with smaller footprint size. Compared to the LPI, LPI_{first} shows a larger underestimation, since LPI_{first} fails to characterize the small gaps within the crown. $LPI_{weighted}$ tends to give a closer correlation than LPI_{first}, because it can detect the weighted points at the crown boundaries to mitigate the multiple-return effect. LPI_{last} provides a very strong overestimation and rather saturated values with low sensitivity. LPI_{both} averages the inaccuracy of both LPI_{first} and LPI_{last}. Most of the estimated results are diverse but can be approximated to follow Eq. (5-23) except for LPI_{last}. As illustrated in **Table 5.2**, the fitted parameter c is 18.87 (LAI = 2) or even cannot get a value (LAI = 1), since pulse always produces a ground return in this simulation.

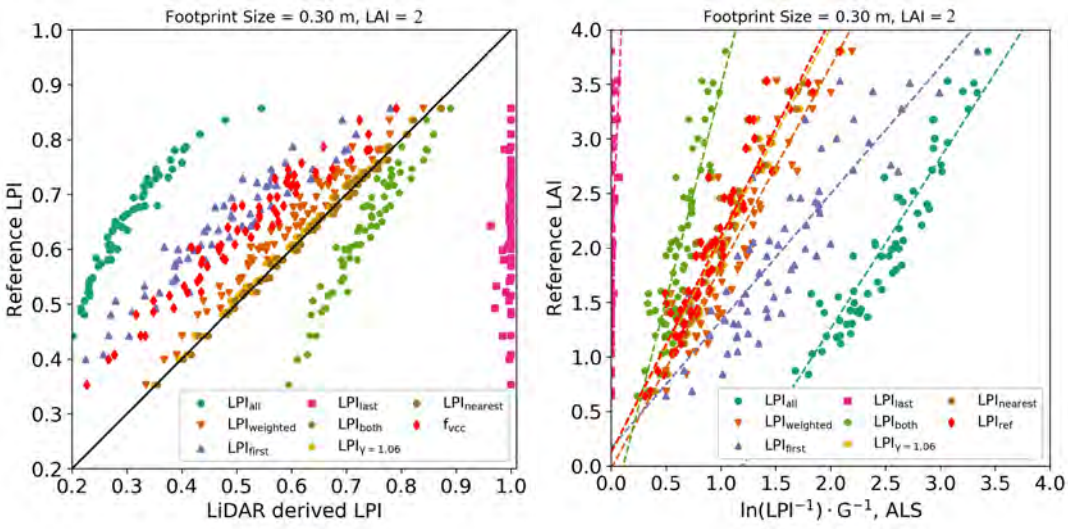
In **Figure 5.8**b and **Figure 5.8**d, all estimated $\ln(LPI^{-1})$ from various approaches can be observed to be linearly correlated with the reference LAI calculated within each grid. It can be observed that estimations using all the multiple returns (LPI_{all}, $LPI_{weighted}$, $LPI_{\gamma=1.06}$, and $LPI_{nearest}$) are higher correlated than the approaches using partially the points (LPI_{first}, LPI_{last}, and LPI_{both}). LPI_{both} has the highest correlation among the three approaches. Due to the non-linearity, there is not any explicit analytical value for the linear fitting slope in this study. However, the linear fitted scattered points of computed $\ln\left(LPI_{ref}^{-1}\right) \cdot G^{-1}$ is added in **Figure 5.8**b and

Figure 5.8d as the reference. Results show that the two IB methods give results that are very closed to the references. With this study, although violating Eq. (5-24), the linear regressions show relatively acceptable fittings in our study range, which verifies the linear fitting of ground-measured gap fraction for LAI map generation in the past publications.

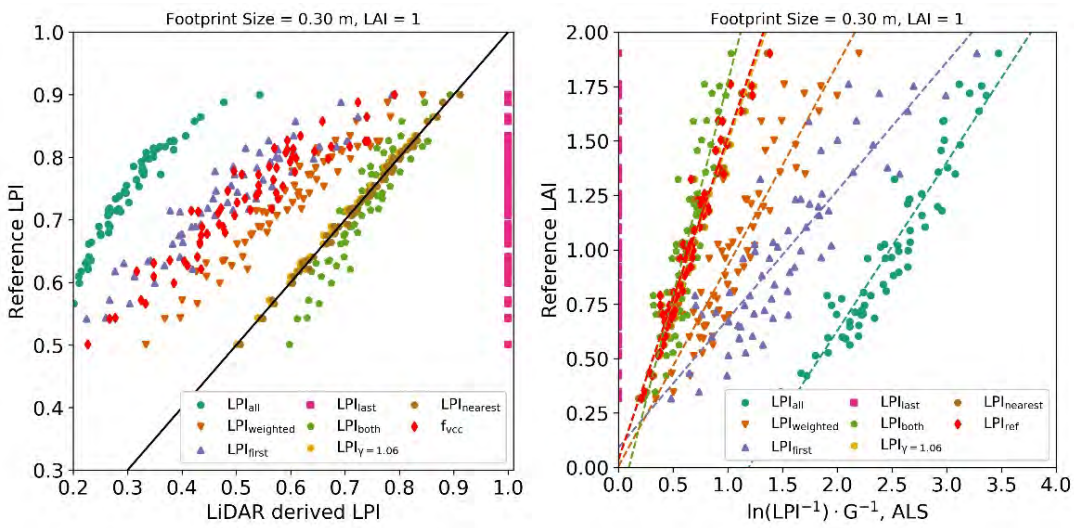
When LAI changes from 2 to 1, $LPI_{weighted}$ shows a underestimation with $c_{weighted}$ varies from 0.93 to 0.61. This is mainly caused by the change of the average number of returns from within-crown pulses. This has already been discussed in the homogeneous case described in section 5.5.1, where the conclusion is that $LPI_{weighted}$ is not sensitive to the changes of LAIs, and only gives satisfied results at some specific LAI values. It can also be noted that LPI_{first} is not sensitive to the changes of LAI, and the results are nearly the same for LAI = 2 and LAI = 1.

Table 5.2: Fitted parameter c of Eq. (5-23) and R-squared value

LAIs	Value	LPI_{all}	$LPI_{weighted}$	LPI_{first}	LPI_{last}	LPI_{both}	$LPI_{\gamma=1.06}$	$LPI_{nearest}$
LAI=3	c	0.41	0.93	0.65	18.87	1.61	0.97	1.00
	R^2	0.90	0.89	0.79	0.48	0.83	0.90	0.90
LAI=1	c	0.27	0.61	0.42	-	1.09	0.98	1.00
	R^2	0.90	0.84	0.78	-	0.77	0.95	0.95



(a) (b)



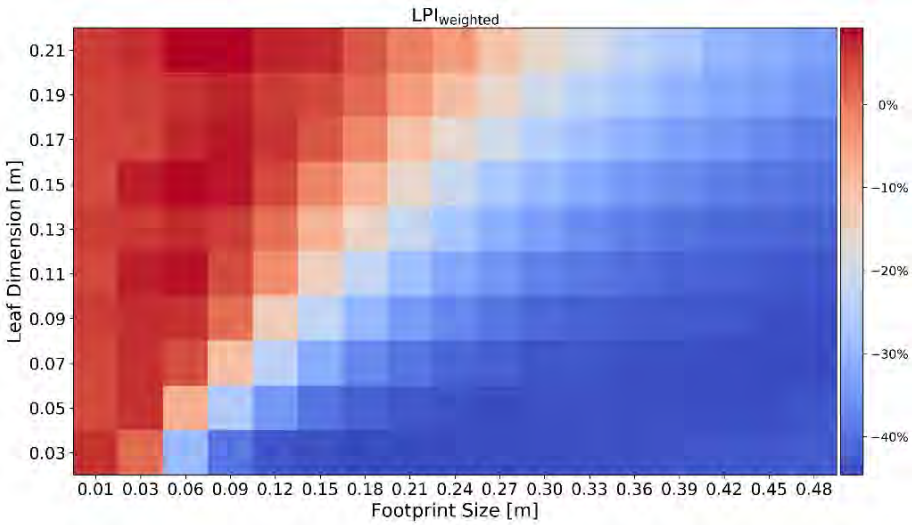
(c) (d)

Figure 5.8: Reference LAI plotted against lnLPI^{-1} for a scene with discrete tree crowns. (a) Relationship between reference LPI and LiDAR-derived LPI with LAI = 2; f_{vcc} is the fraction of vertical crown cover estimated from DART simulation. (b) The relationship between Reference LAI and $\ln(LPI^{-1}) \cdot G^{-1}$ with LAI = 2; (c) Relationship between reference LPI and LiDAR-derived LPI with LAI = 1; (d) The relationship between Reference LAI and $\ln(LPI^{-1}) \cdot G^{-1}$ with LAI = 1.

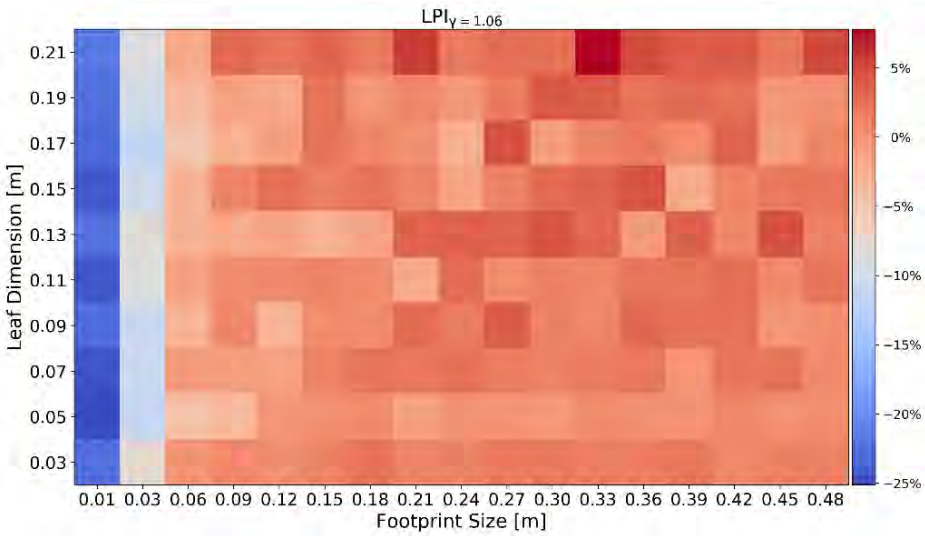
5.6 Influences of varied leaf dimensions

In a forest, individual leaf dimensions can vary a lot due to tree species and growing stages, e.g., from millimeter scale of coniferous trees to centimeter scale of broad-leaved trees. Even for the same LAI and same footprint size, different leaf dimension will influence the estimated LPI. **Figure 5.9** shows the relative error of LiDAR-derived LPIs under different footprint sizes and square leaf dimensions of the homogeneous scene. The LAI is fixed at 3. LPI estimated by PNB is sensitive to the footprint size and the leaf dimensions. Overall, relatively smaller ALS footprint size and larger leaf dimensions produce lower errors, due to the increased fractions of pure-ground and pure-vegetation pulses. Only $LPI_{weighted}$ (best-performing PNB method) and $LPI_{\gamma=1.06}$ are studied here. For IB methods, e.g., $LPI_{\gamma=1.06}$ in **Figure 5.9**b, is less sensitive to leaf dimensions. Even though smaller leaf dimension improves the accuracy, this is not significant. We define a factor r as the ratio of footprint size over the leaf size to characterize the changes of accuracy. **Figure 5.9**c and **Figure 5.9**d plot the relative error against r for $LPI_{weighted}$ and $LPI_{\gamma=1.06}$, respectively. It can be seen that $LPI_{weighted}$ gets closer to the reference if r is approaching 1.0, which can also be confirmed by **Figure 5.4** (LAI = 2). If footprint size equals

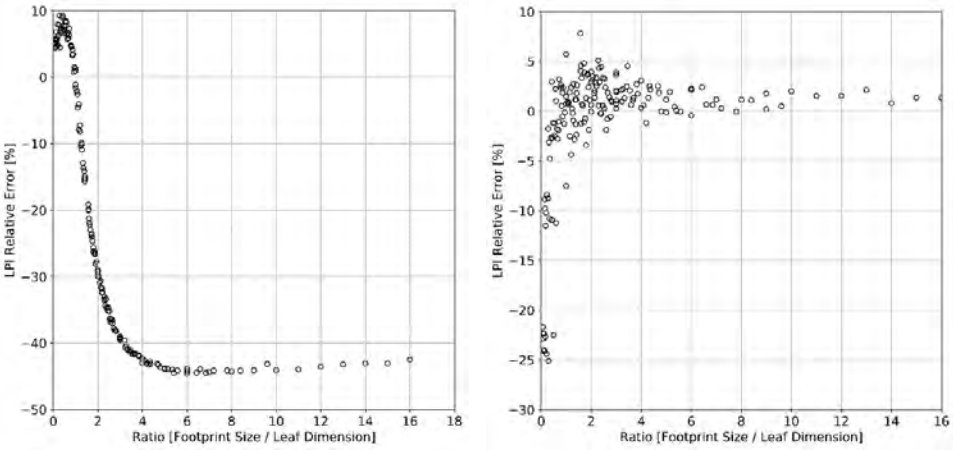
0.05, which is the leaf size, $LPI_{weighted}$ approximately equals the reference LPI. This suggests that $LPI_{weighted}$ (and other PNB methods) is sensitive to the footprint size and leaf dimension when it is used to estimate small within-crown gaps. $LPI_{\gamma=1.06}$ gets stable if r is larger than 6, which ensures that a LiDAR pulse covers at least several leaves.



(a)



(b)



(c) (d)

Figure 5.9: The accuracy of LiDAR-derived LPI against footprint size and leaf dimension. (a) $LPI_{weighted}$; (b) $LPI_{\gamma=1.06}$; (c) $LPI_{weighted}$ relative error vs. the radio of footprint size and leaf dimension; (d) $LPI_{\gamma=1.06}$ relative error vs. the radio

5.7 Influences of instrumental detection threshold

In addition to the experimental and environmental variables that can influence the LPI estimation result, the instrumental variables can also play a crucial part. For actual devices, the parameters that determine whether a return can be detected by a LiDAR device are *R* and ρ_a due to the sensor capability. In this section, the points produced by GD are filtered with threshold $\rho_a > 0.9\%$. This is an empirical value retrieved from the point cloud generated by Riegl 480i.

Figure 5.10 illustrates the LPI calculated by the filtered point cloud under different LAIs for the homogeneous scene (LAI = 1, 3, 4, 6). Compared to the unfiltered LPIs **Figure 5.4**, the filtered LPIs have a larger variance, especially for IB methods under lower LAIs. When LAI is small, the vegetation return usually has lower ρ_a which may be filtered out. This causes the LPI to be overestimated by $LPI_{nearest}$, since some energy from vegetation backscattering has been removed. For PNB methods, the estimated LPIs are slightly lower than the ones from the unfiltered point cloud, because the average number of returns has been reduced by 2 (**Figure 5.11**) approximately. **Figure 5.11** indicates also that the average number of ground returns decrease with the increase of the LAI, which means there are more pure vegetation pulses.

For the heterogeneous scene, the filtering process changes the LiDAR-derived LPI slightly, as illustrated in **Figure 5.12**. LPI_{last} and LPI_{both} become closer to reference LPI, because some ground returns from vegetation-ground pulses have been filtered out due to its low reflectance, which makes the influence of inaccuracy estimation of within crown gaps smaller for LPI_{last}.

LPI_{first} is not affected by this filtering process, since it mainly affects the pulse with multiple returns within crowns.

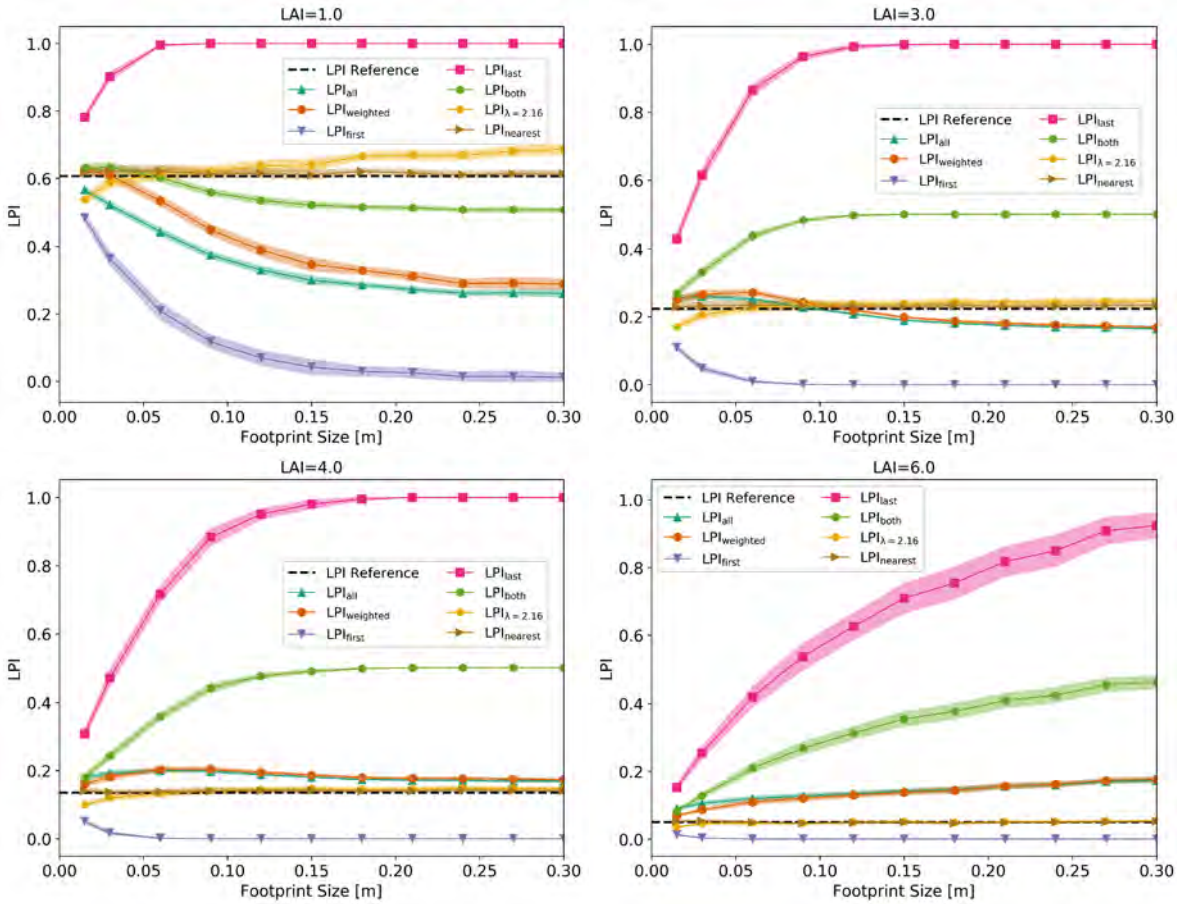
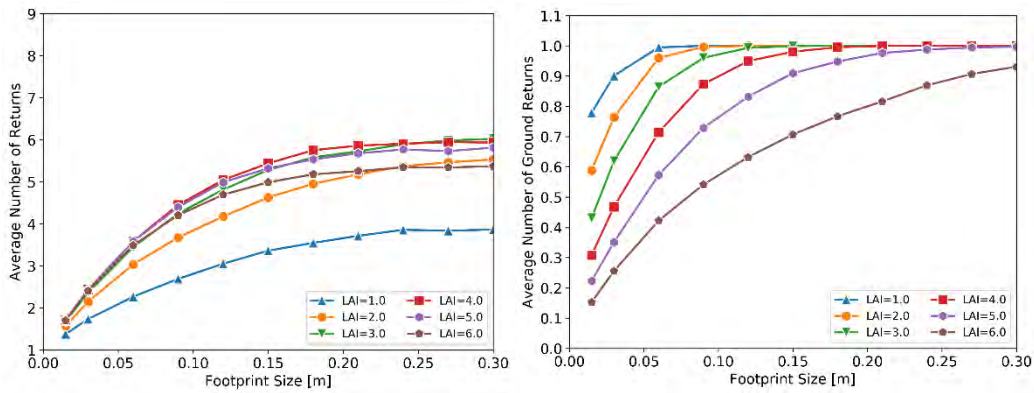


Figure 5.10: LPI estimated by different methods with filtered point cloud



(a) (b)

Figure 5.11: Average discrete returns after filtering for the homogeneous scene.

(a) Average number of returns under different footprint size; (b) average number of ground returns.

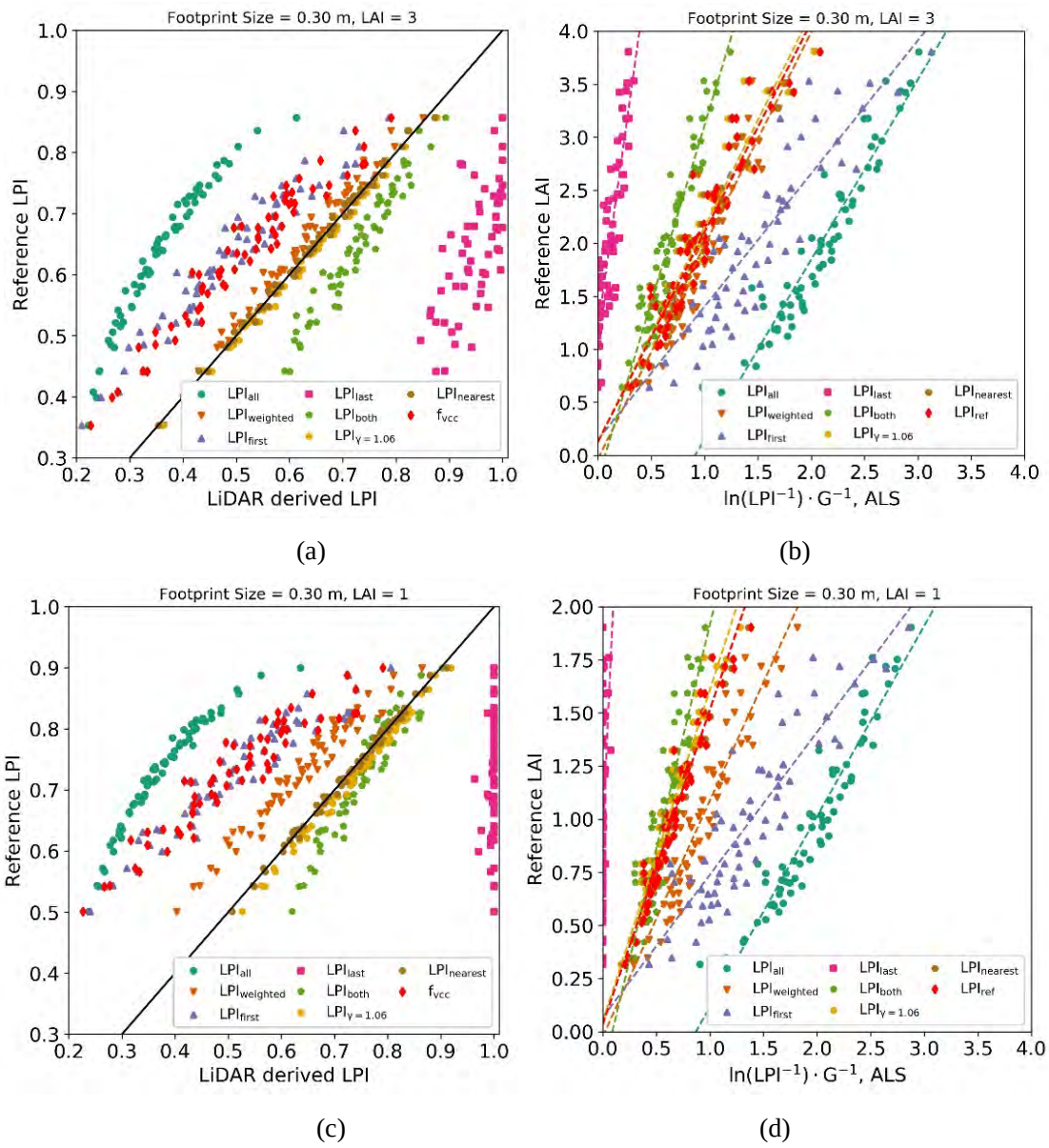


Figure 5.12: Reference LAI plotted against lnLPI^{-1} for heterogeneous scene after filtering. (a) Relationship between Reference LPI and LiDAR-derived LPI with LAI = 2; (b) Relationship between Reference LAI and $\ln(LPI^{-1}) \cdot G^{-1}$ under footprint size 0.3 m with LAI = 2; (c) Relationship between Reference LPI and LiDAR-derived LPI with LAI = 1; (d) Relationship between Reference LAI and $\ln(LPI^{-1}) \cdot G^{-1}$ under footprint size 0.3 m with LAI = 1.

5.8 Further discussion and perspectives

The analyses based on simulations in Section 5.5 show that the IB methods in estimating LPI and effective LAI are more precise and less influenced by the variations of footprint size, LAI, vegetation cover, and leaf dimensions than the PNB methods. For the homogeneous scene,

modeling results show that IB methods can estimate LAI_{eff} directly without necessarily linear correlation with ground measurements. It also has the flexibility for a wide range of instrumental, experimental and environmental configurations. This flexibility cannot be achieved by PNB methods unless the footprint size is infinitesimally small, which corresponds to TLS configuration. As footprint size grows, the estimated LPI from PNB methods becomes insensitive to the LAI variation. This is in contrary with the statement from Hu et al. (2018) that PNB methods are sensitive to the variation of path length and clumping within a tree crown.

For the heterogeneous scene of larger landscape-scale study, PNB methods are sensitive to the large gaps, the IB methods can still capture the precise LPI with existing gaps of various size. We have also shown that I_i and ρ_a are the only suitable quantities for inversion, instead of $\hat{P}_i$. For the estimation of the LAI, the analyses of the fitting coefficient α is not completely addressed although the approximated analytical expression is provided (Eq. (5-24)). The within crown leaf clumping index other than 1 and the leaf angle distribution other than random are not studied in this work. In practice, the intensity values measured by actual LiDAR device also have more unresolved uncertainties than point number. In addition to the issues regarding quantity definition and the radiometric calibration, a LAS file stores the intensity values as 8-bit or 16-bit digital numbers, and this discretization may introduce further precision issues. Furthermore, not all LiDAR device can produce an "intensity" value that is suitable for the IB methods. Traditional LiDAR has $\hat{P}$ as the intensity output. Recent Riegl devices using the online waveform processing technique provides separately $\hat{P}$, $s_{p,i}$ and "reflectivity" (could be ρ_a) as output options (Pfennigbauer and Ullrich, 2010; Ullrich and Pfennigbauer, 2011). However, it was also mentioned that the pulse broadening and system response on $s_{p,i}$ are too weak to be measured (Pfennigbauer et al., 2013). For the IB methods, the constant γ assumption in estimating LPI_λ heavily dependents on the statistically convergent LAD and consistent natural reflectivity ratio of leaf and ground. The understory vegetation and fallen leaves can influence the ground reflectivity. The $LPI_{nearest}$ is a more practical approach. The possible sources of bias are insufficient pure ground pulses and an abrupt change of the ground property or slope which influences the nearest pure-ground pulse.

For the scientific community, we also endeavor to address the following two practical questions mentioned below:

1. **Is it possible to estimate LAI only from ALS data without or with fewer ground measurements?**

Based on the results, the answer could be potentially positive. In this work, we solve the dependency of the linear fitting coefficients. For homogeneous vegetation, we have shown that

LAI can be estimated with presumed G and ω (index of only within-crown clumping). For the heterogeneous scene, according to Eq. (5-24), the computations of $f_{vcc}e^{-G\cdot LAI_{eff}}$ (influence by within-crown small gaps) and $1-\mathrm{f_{vc}}$ (influenced by between-crown large gaps) could be separated, so the between-crown clumping effect is removed. To achieve that, f_{vcc} should be estimated first, following by the estimation of $f_{vcc}e^{-G\cdot LAI_{eff}}$ using the IB methods. This approach can be generally similar to Hu et al. (2018) approach, except that IB method should be used instead of PNB method which has proved insensitive to LAI variation and inseparable from linear correlation with ground measurements. The trickiest part is indeed the footprint size configuration. From our analyses in Section 5.5.2, LPI_{first} can give an accurate estimate f_{vcc} with a proper footprint size. Therefore, we state that the ultimate setting of footprint size should be larger than the within-crown gaps but smaller than the holes and the between-crown gaps. Fortunately, because most recent ALS data satisfy this condition (e.g. Cook et al., 2013), LAI can be directly mapped. However, it is also important to note that an accurate LAI estimate relies on accurate estimates of ω and G.

2. What can we do with a specific ALS LiDAR device and the existing data?

All ALS data are diverse due to the specific experimental and environmental configurations. The acquisition methods, configurations, and the point processing algorithms from different manufacturers are also quite diverse. The most appropriate ways to understand the capability of a device or inverse of an existing data with known configurations is to go through physical modeling instead of directly applying a specific approach that was only verified on a single experiment.

In a nutshell, we present a relatively robust and complete study of LiDAR point cloud in terms of theory, comparative review and sensitivity studies. There are interconnected through DART model. We only investigated ALS points in estimating LPI and effective LAI by DART simulations. However, the applications can also be extended to the limitations and structural/sensor influence of the TLS points, e.g., the partial hit effect. The point cloud module has been implemented in the latest DART release (Version higher than 5.7.0). The recent development of DART LiDAR module now supports the conversion of DART simulated waveforms into point clouds with the storage of waveforms and point cloud data in the standard LAS LiDAR data format. These advances help to bridge the gap between simulated data and current data processing software, with the goal to improve the usage of LiDAR points for assessing the accuracy of ALS and TLS inversions.

5.9 Concluding remarks

This chapter gives a comprehensive insight to modeling discrete point cloud from waveform data and its application in estimating laser penetration index (LPI) and leaf area index (LAI). The comparative study shows that point number based method (PNB) are insensitive to LAI changes and is only valid for some LAI range, while intensity based (IB) methods are more appropriate to describe large gaps as well as small within crown gaps, especially the $LPI_{nearest}$ method that only considers the ground returns. However, the unknown “intensity” value type provided by various LiDAR instruments prevents the usage of intensity information, since not all of them are suitable for IB methods. A more practical way is to use the GD method, described in this chapter, to extract the wanted information from waveform data. In conclusion, accurate estimation of LPI/LAI is important for extract structural information from LiDAR data, which serves as a basis for a voxel-based reconstruction that is described in the following chapter.

Chapter 6

Voxel-based reconstruction and simulation of 3D forest scene

Summary

6.1 Research context 96
6.2 Study area and material 98
6.2.1 Study area 98
6.2.2 ALS and TLS dataset 98
6.2.3 Optical dataset 99
6.3 Voxel-based 3D PVD inversion 99
6.3.1 Single voxel PVD 99
6.3.2 Incident energy at each return 100
6.3.3 Ray traversal in voxels 102
6.3.4 Accuracy Validation 104
6.3.5 Discussion 109
6.4 3D forest scene reconstruction and simulation 112
6.4.1 3D landscape reconstruction 112
6.4.2 Spectral image simulation and comparison 113
6.5 Concluding remarks 116

The above chapters have proposed an efficient 3D radiative transfer model named LESS and have investigated different methods to accurately estimate laser penetration index (LPI)/ leaf area index (LAI). The LAI is an important parameter for radiative transfer modeling, however, 2D LAI is usually not compatible with 3D radiative transfer models. This chapter bridges the gaps between the canopy structure and the 3D radiative transfer model by proposing a voxel-based 3D leaf volume density (LVD) inversion method. The inversion is heavily based on the conclusions that has been drawn in Chapter 5, i.e., intensity based LPI estimation approach is usually more reliable and adaptable to different forests. Using the inverted 3D LVD matrix, a 1km forest scene is reconstructed and input into LESS to simulate spectral images that are then compared to airborne hyperspectral images.

The chapter is presented in the paper:

"**Qi, J.**, Yin, T., Xie, D., Yan, G., Pang, Y., Liu, Q., Gastellu-Etchegorry, J.-P., Voxel-based 3D Estimation of Plant Area Density from Airborne LiDAR Data using Ray-tracing Algorithm. Forests." (under review).

6.1 Research context

3D structural information is an important input parameter for 3D radiative transfer modeling, which provides the ability to upscale leaf-level observations (e.g., leaf reflectance) to canopy-level observations (Schneider et al., 2014). However, how the canopy structure is parametrized is still an open question, due to the complexity of plants. To construct a realistic forest scene, individual tree based approach is usually used (Calders et al., 2018; Côté et al., 2009; Jianbo Qi et al., 2017; Qi et al., 2016). This approach first constructs individual trees with detailed structures from terrestrial LiDAR data (Bailey and Ochoa, 2018; Calders et al., 2018; Wang et al., 2014) or from plant simulation software (e.g., Onyx TREE: www.onyxtree.com). Using these individual trees, realistic forests can be reconstructed by combining spatial information, such as tree position, crown diameter and tree height, etc. A benefit of the individual tree approach is that the constructed forest scene has very precise structures from leaf scale to canopy scale. However, it is difficult to reconstruct a scene that is identical to an actual forest. Indeed, it is nearly impossible to reconstruct every individual tree in a forest, especially for large-scale areas and because the number of already reconstructed trees is limited.

To mitigate the problem raised by the individual tree approach, voxel-based reconstruction approach is proposed, which has been proved to be more accurate to represent an existing forest

plot from airborne LiDAR data (Schneider et al., 2014). Compared with terrestrial laser scanning (TLS) data, which usually works on individual tree or plot level but with high scanning resolution, airborne laser scanning (ALS) data can cover large areas in relative short time. However, the obtained point cloud accuracy is usually not as high as TLS data, which makes difficult to distinguish branches from leaves with ALS data. Hence, the inverted LVD is also called plant volume density (PVD) (Hosoi and Omasa, 2006). The contradiction is that 3D radiative transfer models usually need high resolution (e.g., 1 m) PVD information to provide reliable simulations, which requires the derivation of high resolution 3D PVD from relatively low-density ALS point cloud.

A few attempts have been made (Lin and West, 2016; Schneider et al., 2014; Song et al., 2011) in previous studies to achieve that goal. They use the number of ground echoes (also called “return”) and vegetation echoes to characterize the transmittance, which is then related to PVD by Beer’s law (Yang et al., 1993). However, as demonstrated in Chapter 5, point number based (PNB) methods are usually not adequate to describe fine-scale gaps within leaves, especially for the low density and large footprint ALS data. Instead, intensity, in terms of energy, has advantages in characterizing small gaps because a single pulse already covers at least several leaves, which can be used to evaluate transmittance by using the backscattered energy.

When using LiDAR data intensity for estimating PAD, several factors, such as pulse target reflectance, should be carefully considered because some of them (e.g., ground reflectance) are spatially varying. To deal with this problem, some studies use an empirical ratio between ground and vegetation reflectance to derive canopy transmittance (Lefsky et al., 1999; She-Zhou et al., 2013). Recently, Milenković et al. (2017) proposed a method that only considers ground echoes, without a-priori knowledge of the vegetation structural and optical properties.

Another crucial factor that needs to be considered when dealing with ALS data is the pulse incident angle, which may influence the ALS-derived PVD and is often overlooked (Liu et al., 2018). For ALS, scanning angles can be larger than 20°, which gives an obvious oblique incident pulse when penetrating the canopy. For voxel-based PVD inversion, an individual pulse may enter through one voxel and exit from another neighboring voxel. Thus, the column-based PVD inversion may produce uncertainties. On the other hand, path lengths inside voxels depend on the scanning angle. One way to quantify the effect of incident angle is to trace rays according to the direction of each pulse using ray-tracing technique. This is a useful tool to analyze light regime (Bittner et al., 2012), radiative transfer (Disney et al., 2000) and sensor configurations (Zhao et al., 2015). Kükenbrink et al. (2017) proposed a ray-tracing based method to quantify the occluded volume in ALS acquisitions and also analyzed its dependency on pulse density, flight strip overlap,

etc. To obtain voxel-level PAD, rays should be traced into uniformly distributed voxels. This can be achieved by using a voxel traversal algorithm (Amanatides et al., 1987), which can provide all the intersected voxels along a ray and the corresponding path length passing each voxel.

In this chapter, we propose an approach to estimate voxel-level PVD from ALS intensity measurements using a ray-tracing algorithm. We first (i) establish the formula to calculate the total transmittance of a voxel, which depends on the incident energy at each echo. Then (ii) a ground echo based method is developed to calculate the incident energy, and (iii) PAD of each voxel was estimated by the ray-tracing algorithm. We also compare the ALS-derived PAD with TLS measurements at the same plot. Finally, a forest scene is constructed based on the estimated PVD and is input into LESS.

6.2 Study area and material

6.2.1 Study area

The study area is located in the Genhe Forestry Reserve (120°12′ to 122°55′E, 50°20′ to 52°30 N), Greater Khingan of Inner Mongolia, Northeastern China. This site has slight topography (80% of the slopes are less than 15°) and a mean altitude of 1000 m. Forests occupy 75% of the whole area. The dominate tree species are Dahurian Larch (*Larix gmelinii (Rupr.)* Rupr) and white birch (*Betula platyphylla* Suk).

6.2.2 ALS and TLS dataset

ALS data was collected by a RIEGL LMS-Q680i system, which is onboard on a Yun-12 aircraft. This system operates at wavelength of 1550 nm with a beam divergence of 0.5 mrad. It uses a rotation scanning system with an angle range of ±30°. This flight was flown over the study area with a 1000 m above ground level (AGL) in August, 2016. The final point density, considering flight line overlapping, is around 9 points/m^2. The data used in this study covers an area of 500 m × 500 m. The ALS data is first classified into ground and non-ground parts using the cloth simulation filter (CSF) (Zhang et al., 2016), which is used to determine the ground echo for each pulse. In this study, echoes lower than 0.5 m are assumed to be ground echoes.

A 30 m × 30 m forest plot mainly composed of larch trees was chosen in this study area. TLS data was acquired on 13th, August, 2016 by a Leica ScanStation C10 scanner that operates at 532 nm, with a 4.5 mm spot size at range of 50 m. Five scanning positions were selected: one in the plot center and four on the outside of the four edges. The scanning resolution was 0.03°. Data from

the five positions were referenced in a common coordinate system using the Cyclone software (Leica GeoSystems) and ground control points.

6.2.3 Optical dataset

Optical properties of leaves, branches and ground are also needed to construct a realistic 3D landscape. These datasets were measured by a spectroradiometer (ASD FieldSpec3, Analytical Spectral Devices, USA) equipped with an integrating sphere. The measurements were performed on August, 2016. Twenty samples of birch and larch leaves from different height and different places in the forest were collected and immediately measured. Since the leaf size of larch tree is very small, we placed them side by side and glued them together to measure them with ASD. The reflectance of barks was also measured by collecting samples from different heights of a tree. The ground reflectance was measured in 20 randomly selected sites in the forest, including bare soil and grassland.

6.3 Voxel-based 3D PVD inversion

6.3.1 Single voxel PVD

Beer's law was used to estimate the PVD per voxel because it links the total transmittance to plant area density:

$$T(\Omega) = e^{-\rho_f \cdot G(\Omega) \cdot \Delta l(\Omega)} \tag{6-1}$$

where $T(\Omega)$ is the directional transmittance along direction Ω; ρ_f is the plant area density [m^2/m^3]; $\Delta l(\Omega)$ is the path length through a voxel. $G(\Omega) = \int_{2\pi} \frac{g(\Omega_L)}{2\pi} |\Omega \cdot \Omega_L| d\Omega_L$ is leaf projection coefficient for direction Ω, with $\frac{g(\Omega_L)}{2\pi}$ the leaf angle distribution. In this study, the leaf angle distribution is assumed to be spherical ($g(\Omega_L) = 1$) which gives $G(\Omega) = 0.5$.

Let a voxel j that is crossed by several pulses (**Figure 6.1**). If pulses have an infinitely small diameter, its total transmittance can be calculated as:

$$T_j = \frac{I_j^{out}}{I_j^{in}} = \frac{I_j^{out}}{\sum_{k=1}^{P} I_{j,k}^{in}} \tag{6-2}$$

where I_j^{in} is the total energy that is incident onto voxel j, and I_j^{out} is the total outgoing energy. I_j^{in} is composed of energy from different pulses with energy equal to $I_{j,k}^{in}$ ($k = 1, 2, \ldots, P$),

where P is the total number of pulses incident on voxel j. Since the path length ($\Delta l_{j,k}$) of different pulses changes with the incident angles and cell entry points, an averaged path length is used to estimate PVD:

$$LAD_j = -\frac{2\,ln\,T_j}{\Delta \bar{l}_j} \tag{6-3}$$

where $\Delta \bar{l}_j$ is the averaged path length for all pulses that are incident on voxel j.

As shown above, the energy that enters and exist a voxel must be known for calculating PVD. The energy incident on a voxel cannot be estimated directly. Here, it is assumed that it is equal to the incident energy at the positon of the first echo for each pulse in voxel j, i.e., $I_{j,k}^{in} = I_{j,k,1}$. The outgoing energy I_j^{out} is the energy incident on other voxels. It can be calculated through the same way as calculating incident energy, which is an iterative process. Thus, the problem turns out to calculate the incident energy ($I_{j,k,1}$) at the position of the first echo of pulse k in voxel j.

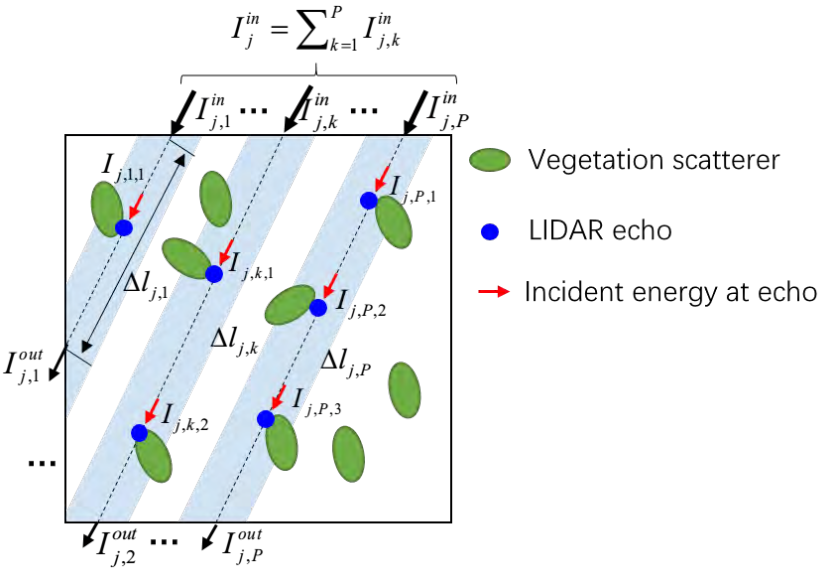


Figure 6.1: Voxel PVD inversion using pulse transmittance

6.3.2 Incident energy at each return

To determine the incident energy at the position of the first echo in each voxel, the LiDAR data is processed pulse by pulse. In the discrete LiDAR point cloud, the pulse information is not directly provided. However, it can be extracted from the GPS time, since echoes from the same pulse usually have the same GPS time, which is also unique in the whole point cloud. Then the whole LiDAR points can be grouped into different pulses. In forest area, there are usually three types of pulses, which are identified based on the interaction between the pulse and landscape

elements (leaves, branches and ground): vegetation-ground pulse, pure-ground pulse and pure-vegetation pulse. Vegetation-ground pulse is the pulse that contains both vegetation echoes and ground echo, while pure-ground pulse and pure-vegetation pulse only contain ground echo and vegetation echoes, respectively. Usually, the pure-ground pulse only has one echo.

In order to calculate the incident energy of a voxel, the incident energy at the position of all the echoes for all the pulses is first calculated. For a ground echo in a pulse, the incident energy I_g can be calculated by the backscattering energy Q_g (**Figure 6.2**),

$$LI_g = \frac{Q_g}{\pi f(\Omega_s, \Omega_v) \cos\theta_v} \tag{6-4}$$

where $f(\Omega_v, \Omega_v)$ is the bidirectional reflectance distribution function (BRDF) of the ground, Ω_s denotes the incident direction of the pulse and Ω_v denotes the sensor direction. In LiDAR measurements, Ω_s and $\Omega_v$is the same direction. For vegetation echoes, the backscattering is related to leaf surface reflectance and leaf angle distribution, which can be describe as a phase function,

$$\omega(\Omega_s \to \Omega_v) = \frac{\int_{2\pi} \frac{g(\Omega_L)}{2\pi} |\Omega_L \cdot \Omega_s| \cdot |\Omega_L \cdot \Omega_v| \frac{\rho_v}{\pi} d\Omega_L}{\int_{2\pi} \frac{g(\Omega_L)}{2\pi} |\Omega_L \cdot \Omega_s| d\Omega_L} \tag{6-5}$$

with ρ_v being the reflectance factor of leaf surface. $\omega(\Omega_s \to \Omega_v)$ is the proportion of the intercepted energy in direction Ω_s that is reflected into direction Ω_v per solid angle unit. Here, it is a constant because ρ_v is assumed to be the same for all leaves in a same voxel.

For echoes in a vegetation-ground pulse, the incident energy at the position of each echo can be written using an iterative expression:

$$I_i = \begin{cases} \frac{Q_i}{r_v} + I_{i+1}, & \text{if } i < N \\ \frac{Q_g}{r_g} & \text{if } i = N \end{cases} \tag{6-6}$$

where $r_g = \pi f(\Omega_s \to \Omega_v) \cos\theta_v$, and $r_v = \pi\omega(\Omega_s \to \Omega_v)$. For Eq. (6-6), we can derive

$$I_1 = \frac{1}{r_v}\sum_{k=1}^{N-1} Q_k + \frac{Q_g}{r_g}, \quad I_i = \frac{1}{r_v}\sum_{k=i}^{N-1} Q_k + \frac{Q_g}{r_g}, \quad \cdots, \quad I_N = \frac{Q_g}{r_g} \tag{6-7}$$

To determine the ground back scattering coefficient r_g, a nearest pure-ground pulse is used. If two pulses are not far from each other (spatial distribution of pure-ground pulses is dense enough, the incident angles (θ_v) should be very close, thus the back scattering coefficients are also the same,

no matter what the BRDF of ground is. In this study, we ignore the air attenuation from top of canopy (TOC) to ground for pure-ground pulse, then we have:

$$r_g = \frac{Q_g^*}{I_0} = \frac{Q_g^*}{I_1} \tag{6-8}$$

where Q_g^* is the intensity of the nearest pure-ground pulse. I_0 is the incident energy at TOC. For vegetation-ground and pure-ground pulses, $I_N = I_g$. For pure-ground pulse, $I_N = I_g = I_0$. Combining Eq. (6-8), then Eq. (6-7) can be rewritten as

$$I_1 = \frac{Q_g^* \sum_{k=1}^{N-1} Q_k}{r_v(Q_g^* - Q_g)}, \quad I_i = \frac{Q_g^* \sum_{k=i}^{N-1} Q_k + Q_g \sum_{k=1}^{i-1} Q_k}{r_v(Q_g^* - Q_g)}, \cdots, \quad I_N = \frac{Q_g \sum_{k=1}^{N-1} Q_k}{r_v(Q_g^* - Q_g)} \tag{6-9}$$

For pure-vegetation pulse, the incident energy at the position i can be simply written with

$$I_i = \frac{1}{r_v} \sum_{k=i}^{N} Q_k \tag{6-10}$$

Eq. (6-9) and (6-10) provide a way to calculating incident energy at the position of each vegetation echo. Q_g^* and Q_k is backscattered energy, which can be interpreted as the LiDAR intensity value. Leaf backscattering coefficient r_v is the only unknown parameter. However, it can be canceled out during the calculation of voxel transmittance, as described in section 6.3.3.

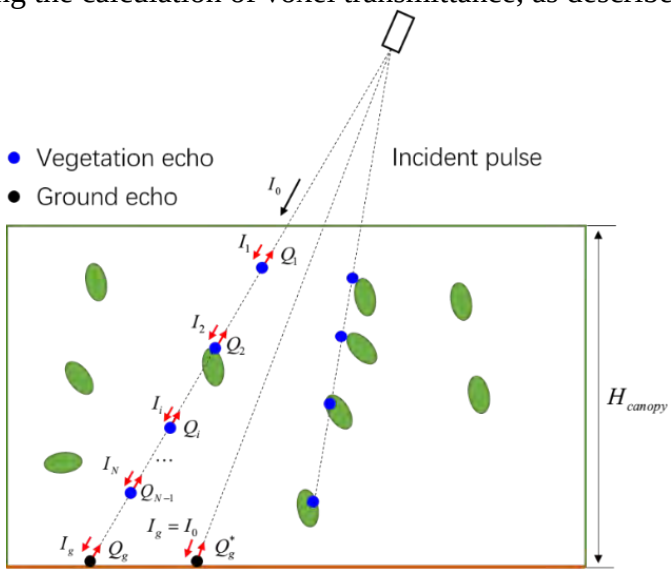


Figure 6.2: Incident energy calculation for all echoes in a pulse.

6.3.3 Ray traversal in voxels

A ray tracing approach is used to calculate voxel transmittance. It gives the voxel incident energy I_{cell}^{in} and outgoing energy I_{cell}^{out}. Here, the voxel is defined as a cube with a size of 1×1×1 m^3 (**Figure 6.3**). To trace a ray, the origin $\boldsymbol{o}$ and direction $\boldsymbol{d}$ must be known. Since the LiDAR

point cloud does not provide this information directly, we use the positions of the first echo $\boldsymbol{p_1}$ and last echo $\boldsymbol{p_N}$ to calculate $\boldsymbol{d}$, which gives $\boldsymbol{d} = (\boldsymbol{p_N} - \boldsymbol{p_1})/|\boldsymbol{p_N} - \boldsymbol{p_1}|$. Then any point on the ray can be expressed as $\boldsymbol{p} = \boldsymbol{o} + t\boldsymbol{d}$, where t is an arbitrary real number. The origin of the ray (sensor position) is achieved by moving the first echo in negative direction as long as it is above the canopy (e.g., t=1000). For pulses with only one echo, the direction of the nearest pulse with multiple echoes is used. For areas with large gaps, the nearest multiple-echo pulse may be very far, which causes the incident angle to be far from the real value. However, this area does not have vegetation and then is out of our interest in this study.

The ray tracing method used in this study is based on a voxel traversal algorithm(Amanatides et al., 1987). Given the origin and direction of a pulse, all the crossed voxels and the corresponding path lengths can be determined. Suppose all the crossed voxels are labeled as $1, 2, \ldots, \mathrm{m}, \ldots, M$ with M the total number of traversed voxels (**Figure 6.3**), the incident energy due to this pulse is $I_{m,k,\,1}$, and the outgoing energy due to this pulse is $I_{next(m,k),k,\,1}$, where $next(m,k)$ stands for the next non-empty (at least one echo from pulse k) voxel crossed by pulse k. Thus, the total incident energy and outgoing energy for a voxel can be calculated by accumulating all the pulse incident energy and pulse outgoing energy, respectively. The total transmittance is:

$$T_j = \frac{I_j^{out}}{I_j^{in}} = \frac{\sum_{k=1}^{P} I_{next(j,k),k,1}}{\sum_{k=1}^{P} I_{j,k,1}} \tag{6-11}$$

For voxels without echoes due to pulse k, the incident energy equals to outgoing energy, which is give as $I_{next(j,k),k,1}$.

During the voxelization process, an echo may lie in the corner of a voxel. This may produce very short path length (**Figure 6.3**) and produce a large PVD value. Basically, an echo stands for the interaction between a pulse and a cluster of vegetation elements, instead of a single point. Thus, we assume that a virtual voxel, which has the same size as the normal voxel, is centered around the echo. Its corresponding path length $\Delta l'$ can be computed using a ray-box intersection algorithm (Williams et al., 2005). Then the transmittance can be calibrated as:

$$ln\, T_{\Delta l} = ln\, T_{\Delta l'} \frac{\Delta l}{\Delta l'} \tag{6-12}$$

where $T_{\Delta l}$ and $T_{\Delta l}$ are the transmittance values along path lengths Δl and $\Delta l'$, respectively. Hence, $\Delta l'$ can be used as the corrected path length when deriving PVD from transmittance.

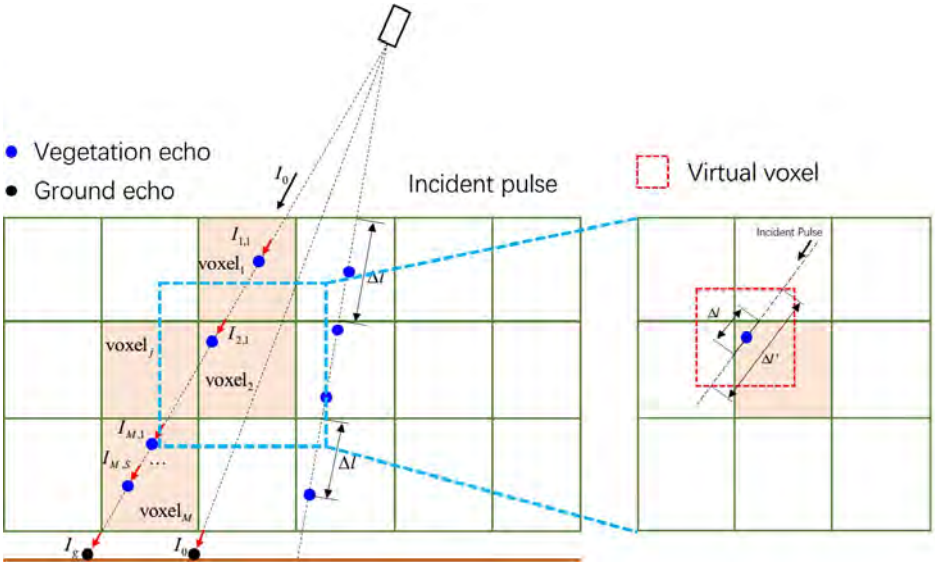


Figure 6.3: Ray traversal in voxels

6.3.4 Accuracy Validation

6.3.4.1 Validation with DART simulations

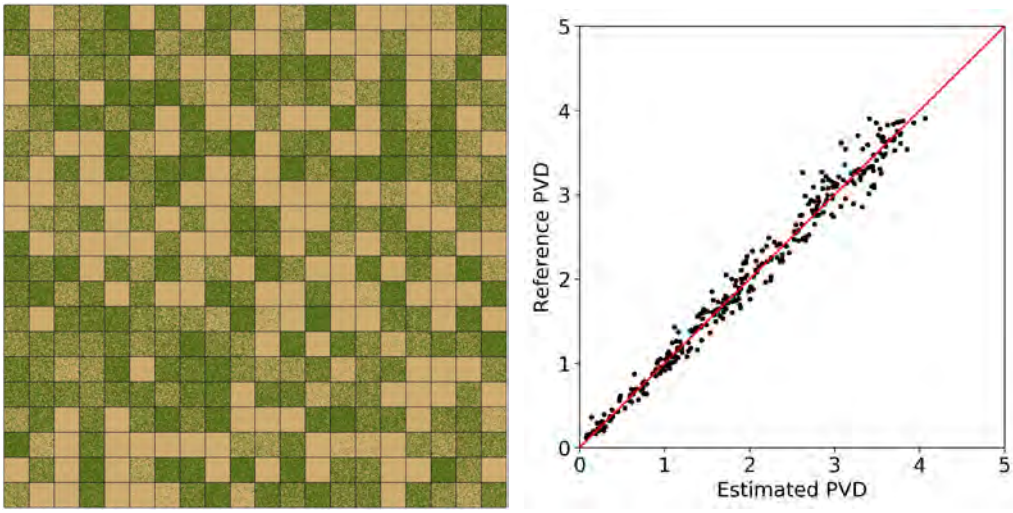
A simulated LiDAR dataset was used to validate the accuracy of the proposed PAD estimation method. LiDAR point cloud was generated using DART model (Gastellu-Etchegorry et al., 2015), which is a 3D radiative transfer model that can simulates full-waveform as well as discrete point cloud. The size of the virtual scene is 100 m×100 m×50 m. Voxel dimension is 5 m×5 m×5 m. We randomly selected some voxels from the grids, and filled them with randomly distributed leaves, using a spherical leaf angle distribution. Parameters (e.g., laser divergence) of the simulated LiDAR device are indicated in **Table 6.1**. The flight height is 500 m and the incident direction of the pulse is vertical. Based on the simulated point cloud, we calculated the PAD of each voxel and compared it with true values provided by LESS.

Table 6.1: LiDAR parameters for DART simulations

Parameter	Value	Parameter	Value
Wavelength	1550 nm	Cross-track FOV	±30°
Laser sampling interval	1 ns	Scanning Speed	100 lines/second
Laser beam divergence	0.3 mrad	Laser repetition rate	200 kHz
Flight height	500 m		

Figure 6.4 illustrates the estimated PVD and reference PVD that is provided by DART. It shows that the estimated PVD is consistent with reference PVD with R-squared value being 0.98 and RMSE being 0.14. This comparison is valid because the simulated scene is a highly abstracted

scene with spherical leaf angle distribution and randomly distributed leaves within voxel. Besides, we have removed the pulses that are partially intersected with a voxel. In real forests, other factors, such as branches and clumping of leaves, will also influence the estimated PVD. However, this validation experiment, at least, illustrates that the method proposed in this study is theoretically valid.



(a) (b)

Figure 6.4: Comparison between estimated PVD and reference PVD.
(a) Top view of the test scene; (b) Estimated PVD and reference PVD.

6.3.4.2 Cross validation with other PVD estimation method

The PVD distribution of the 500 m × 500 m study area is illustrated in **Figure 6.5**. It shows the PVD of each voxel in 3D space. The subplot (**Figure 6.5**b) gives a closer view of the 3D distribution. It successfully captures the canopy structures. This can be also seen from the 2D view (**Figure 6.5**c), which has similar spatial structures with its corresponding canopy height model (CHM).

In **Figure 6.6**, a 2D profile of the PVD and the original LiDAR measurements are shown. Overall, the spatial distribution of voxel reveals the distribution of LiDAR points, even the fine-scale structures. The ground is automatically filtered out during the processing of LiDAR data. It can also be noted that the undergrowth in this plot is also captured by our method.

To quantitatively evaluate the performance of the voxel based PVD. We project it onto a 2D map with 10 m resolution, and compare it with an echo number based method (She-Zhou et al., 2013), which can map PVD with total ground echoes divided by total echoes. **Figure 6.7** shows the comparison of these two methods. The 2D PVD of the voxel based method appears to be

consistent with the echo number based method. The R-squared value between them is 0.71. In this comparison, a slight difference is noted that the echo number based method has more spatially smooth LAI

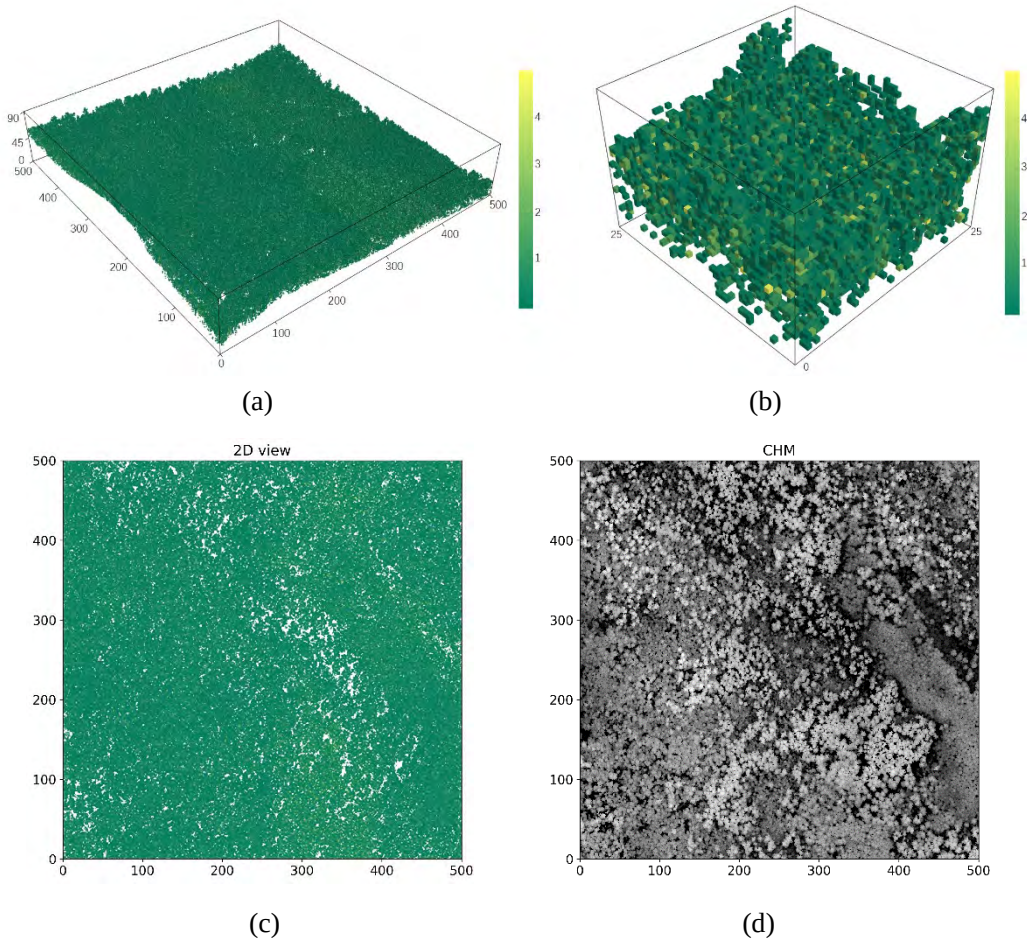


Figure 6.5: Voxel-based 3D distribution of PVD.

(a) 3D PVD of the 500 m × 500 m study area; (b) A subplot in the study area (25 × 25 m); (c) 2D view of the 3D PVD distribution; (c) Canopy height model (CHM) derived from LiDAR data.

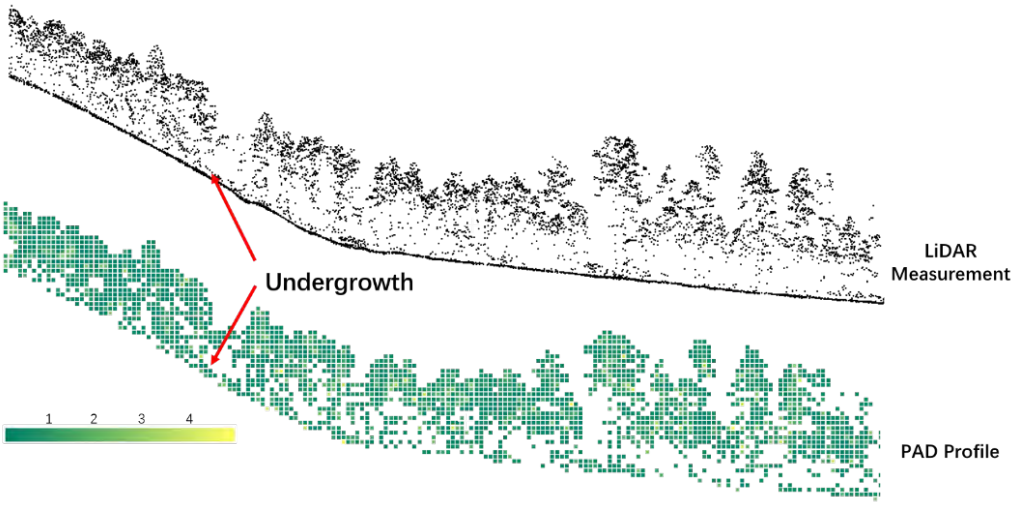
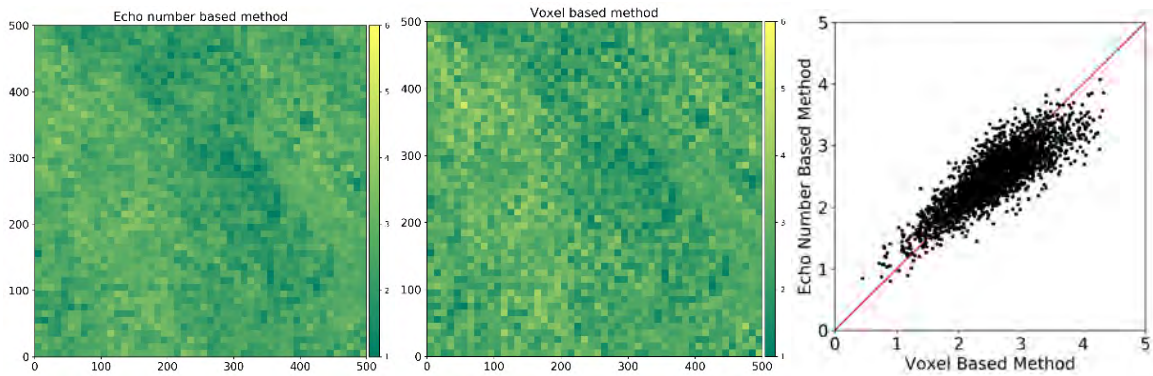


Figure 6.6: 2D profile of the PAD distribution and its corresponding LiDAR measurements.



(a) (b) (c)

Figure 6.7: Comparison of PAD map with established method.

(a) Echo number based method; (b) Voxel based method; (c) Scatter plot of this two methods.

6.3.4.3 Assessment of ALS-derived PVD with TLS-derived PVD

Compared to ALS measurements, TLS data usually has higher sampling density smaller beam divergence. This makes it possible to evaluate the ALS-derived PVD with TLS-derived PVD at a voxel level. However, a registration between ALS and TLS data must be performed before the comparison. This is achieved by a two-step procedure. First, a coarse registration was performed by using the GPS positions of the plot corners, which were measured by a differential GPS device in the field. Then a fine registration was done by using the iterative closest point (ICP) algorithm implemented in Cloudcompare software (Girardeau-Montaut, 2011). **Figure 6.8** illustrates an example of the aligned ALS and TLS data. Afterwards the retrieval of the voxel-level PAD from the TLS data was conducted by using a contact-frequency based method (Béland et al., 2011). The voxel size is also set to 1 m.

Figure 6.9 shows the 3D distribution of voxel-level PVD derived from ALS and TLS data for a subplot. A significant difference between them is that the distribution of TLS-derived PVD is much denser than ALS-derived, since some vegetation elements detected by TLS acquisitions are not detected by ALS system (Kükenbrink et al., 2017), especially for the lower layer of the canopy. This can also be confirmed by the top view of different layers, which is illustrated in **Figure 6.10**. It shows the horizontal distribution of plant area index (PAI), which is obtained by vertically accumulating PVD of each column, for bottom (0 m - 10 m), middle (10 m – 20 m) and top (20 m – 30 m) layers. It can be seen that the bottom layer of the canopy is not fully detected by ALS system, since pulses are incident from the top and can be blocked by top layers. For TLS measurements, it even detects the trunks of the trees, which shows some higher value pixels, because it is not distinguished from leaves in this study. The horizontal distribution of top layers shows more similarity for ALS and TLS derived PAI. **Figure 6.9**c illustrates the vertical profile of PAI for the subplot. It shows that the vegetation elements are mainly distributed around the height of 20 m, which is the center of the crown layer. Overall, the vertical profile of ALS-derived PAI matches well with the TLS-derived PAI, which indicates that the proposed voxel-based method has the ability to depict the vertical structure of canopies. At the bottom layers, TLS-derived PAI is larger than ALS-derived PAI, since TLS is more efficient to sample elements near ground. While for top layers, ALS system has more advantages to detect the tree tops.

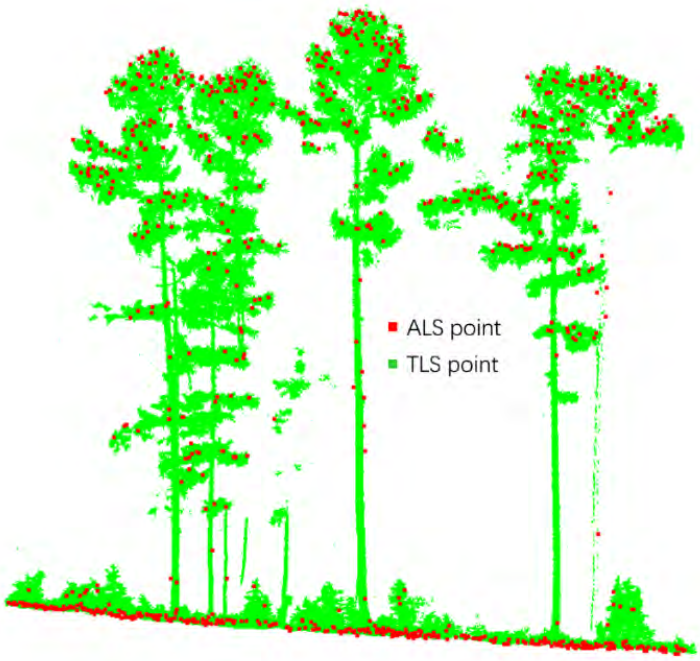
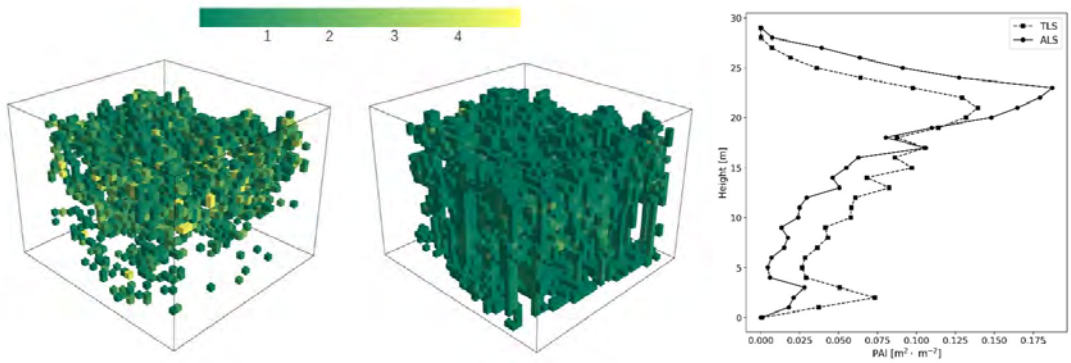


Figure 6.8: An illustration of the alignment between ALS and TLS data



(a) (b) (c)

Figure 6.9: 3D voxel of plot L9: (a) ALS; (b) TLS; (c) Vertical profile of the PVD

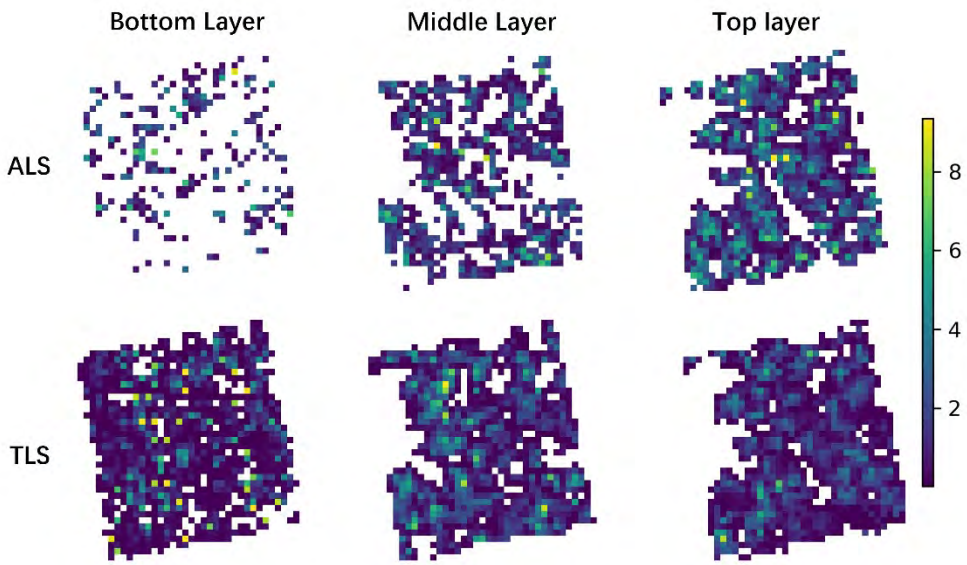


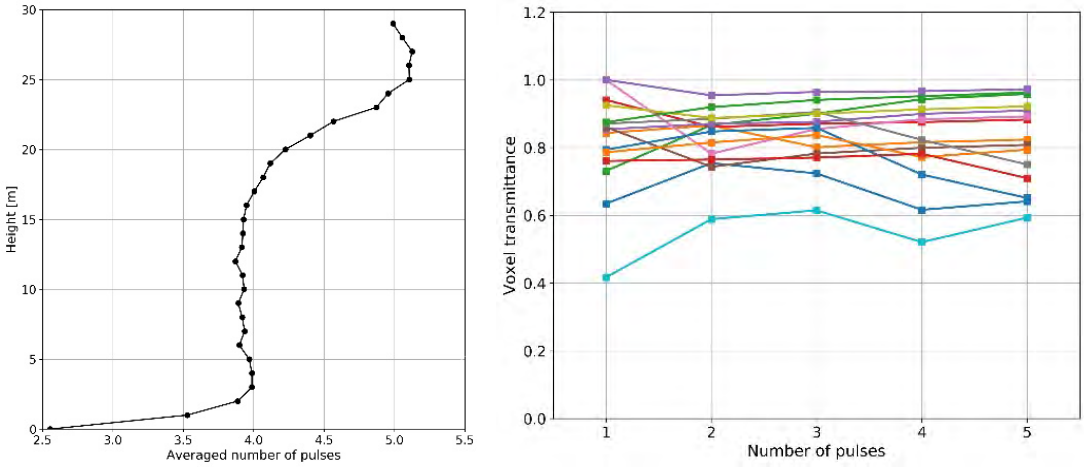
Figure 6.10: Top of view of the horizontal PAI distribution for bottom (0-10 m), middle (10–20 m) and top (20–30 m) layers. The PAI is integrated vertically in each layer for each column.

6.3.5 Discussion

6.3.5.1 Pulse number in each voxel

Number of pulses in a voxel is an important factor that influences the estimation of the total transmittance. For point based method, it needs a significant number of echoes to derive an accurate transmittance. However, for intensity based method, each pulse can be used to estimate transmittance separately. This makes it possible to estimate PVD under high resolution with low density point cloud. **Figure 6.11** illustrates the vertical profile of the average number of pulses per voxel of the subplot. It indicates that the top layer has higher density of pulses, and it drops rapidly around the height of 20 m, since this layer contains most of the vegetation elements of a canopy. From 3 m to 15 m, the number of pulses changes slowly, because these layers are mainly composed

of trunks, which do not block the propagation of pulses so rapidly since the distribution of trunks is usually sparser than leaves. At the bottom layers, the number of pulses drops dramatically, which is caused by the undergrowth. The influence of number of pulses on the estimation of transmittance is illustrated in **Figure 6.11**b. 15 voxels that have been traversed by 5 pulses are chosen, and the transmittance estimated by different number of pulses are plotted. It shows that the estimated transmittance is stable when increasing the number of pulses. The results are acceptable even with only one pulse.



(a) (b)

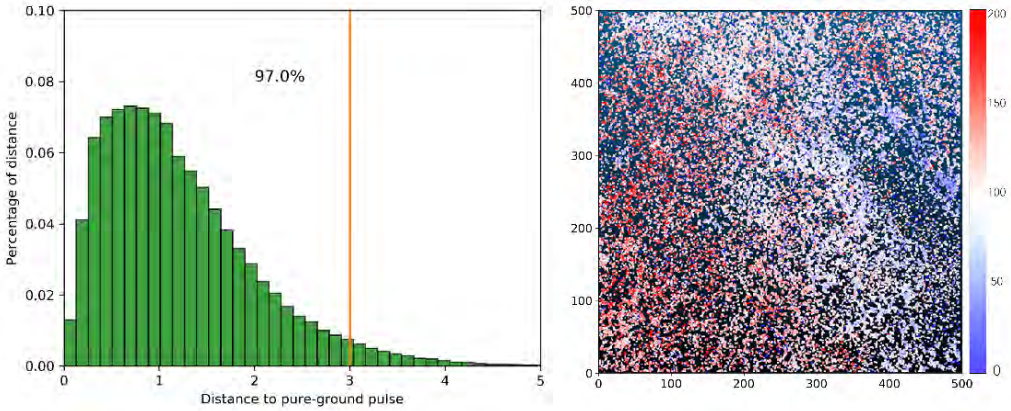
Figure 6.11: Number of pulses traverse the voxel.

(a) Vertical profile of average number of pulses per voxel; (b) Transmittance estimated by different number of pulses for 15 voxels.

6.3.5.2 Nearest pure-ground pulse

Another important factor that influences the accuracy of estimated PAD is the nearest pure-ground pulse found for each vegetation-ground pulse, since we assume that the reflectance of the ground at the positions of these two pulses is the same when they are not far from each other. **Figure 6.12**a illustrates the histogram of the distance between the vegetation-ground pulses and the corresponding pure-ground pulses. It shows that 97% of the distances are within a range of 3 m. This makes it possible to find a pure-ground pulse which has similar ground reflectance. **Figure 6.12**b also shows the intensity of all the pure-ground pulses from a single flight path. It can be seen that the ground intensity is varying spatially, which may be caused by the heterogeneity of the ground. However, at fine scale, the intensity is not changing rapidly, which provides the basis for the nearest pure-ground pulse approach. To quantitatively characterize the local properties of pure-ground pulse intensity values, all the ground points from pure-ground pulses are rasterized into pixels according to its coordinates. The standard deviation of the intensity within a pixel is

computed if the number of points is larger than 1. **Figure 6.13** shows the histogram of the standard deviation for pixel resolution 1, 3, 5 and 10 m. It can be seen that the standard deviation increases when resolution decreases. That means the intensity of two points is more similar when they are closer to each other.



(a) (b)

Figure 6.12: Distance between the vegetation-ground pulse to the nearest pure-ground pulse (a) and intensity distribution of pure-ground pulses (b).

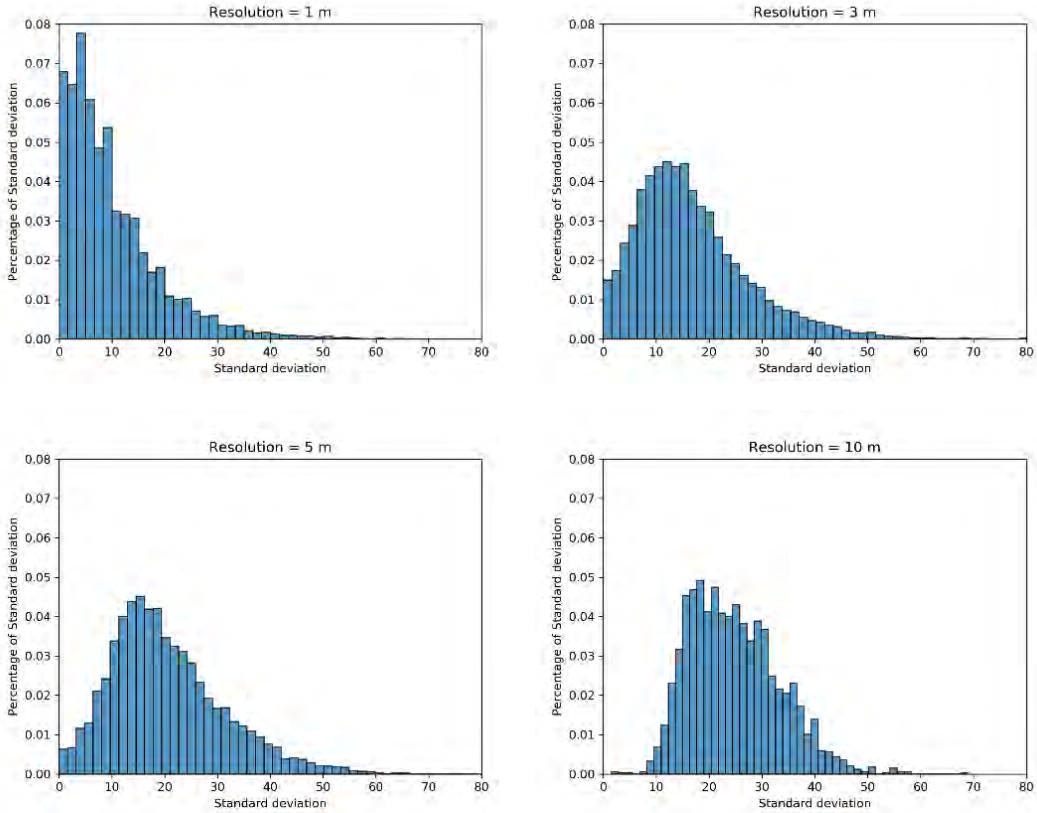


Figure 6.13: Standard deviation of intensity in each pixel with different resolutions. (a) Resolution=1 m; (b) Resolution=3 m; (c) Resolution= 5 m; (d) Resolution= 10 m.

6.4 3D forest scene reconstruction and simulation

6.4.1 3D landscape reconstruction

Reconstruction of 3D landscapes based on individual trees is sometimes difficult due to the difficulty of detecting individual trees from ALS data, especially in dense and mixed forests. In this study, the landscape was created using voxels with the size being 2 m × 2 m × 2 m. It should be noted that the tree species are not identified in this voxel-based approach due to the lack of enough information (e.g., the relative low density point cloud).

Since LESS currently does not support turbid representation of vegetation (like DART), we use facet-based leaf cloud to represent the voxel. Each voxel is filled with randomly distributed leaves within its volume. The LVD of each voxel estimated from the LiDAR data may be different from voxel to voxel. Generating an individual leaf cloud for each voxel in the scene is impractical for this 1 km scene due to the large number of facets. In LESS, the “instance” approach was applied to translate some base voxels to different positions, which only stores the transformation matrix, instead of copying the whole leaf cloud. The base voxels are a series of voxels with different LVD values, ranging from 0.1 to 6 with the step of 0.05 (**Figure 6.14**). The leaf in the base voxels is represented by a rectangle with an area of 0.1 m× 0.1 m. When constructing the scene, a base voxel with the nearest LVD value is chosen and translated to the desired location. **Figure 6.15** illustrates an example of a reconstructed 3D scene of a 100 m × 100 m subplot.

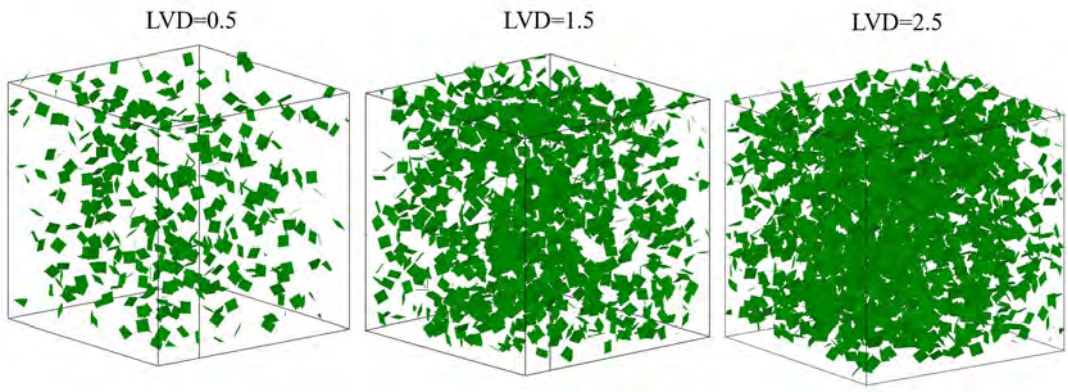


Figure 6.14: Base voxels with different LVD

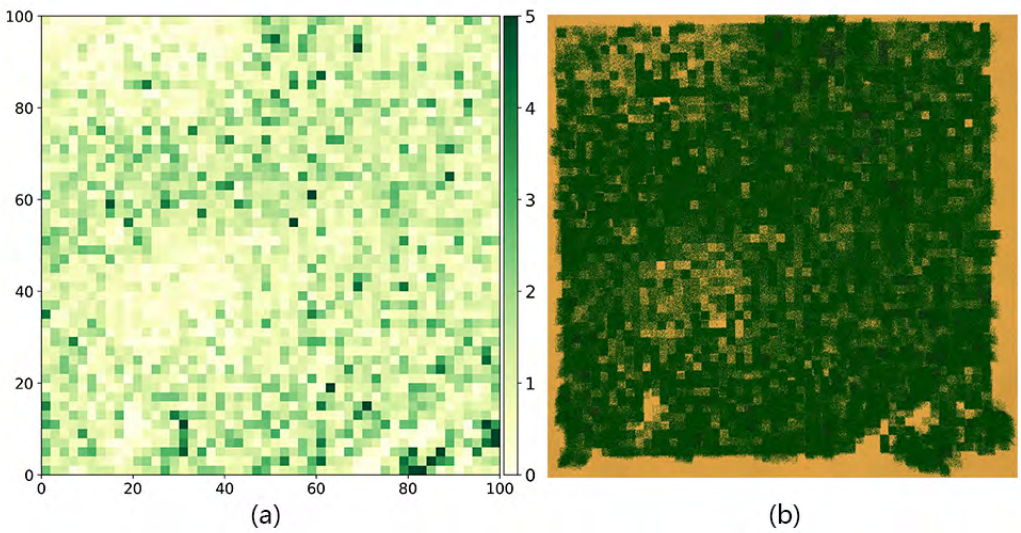


Figure 6.15: Reconstruction of a 100 m×100 m subplot scene from LiDAR point cloud. (a) 2D PAI distribution estimated from LiDAR point cloud; (b) Voxel-based representation of the reconstructed 3D scene (The leaf color is set to green).

6.4.2 Spectral image simulation and comparison

The simulation was carried out under 2 m spatial resolution with 2 spectral bands (NIR: 907 nm, RED: 656 nm). The reflectance/transmittance of leaf from the measurements is 0.08/0.08 and 0.43/0.43 for RED and NIR band. The ground reflectance of RED and NIR is 0.07 and 0.35, respectively. The sun zenith and azimuth angle (SZA = 42.3°, SAZ = 163°) were calculated by the Sun Position Calculator provided by LESS based on the flight date, time of and geographical position of the study area. The sensor was set with orthographic projection with view zenith angle equal to zero. A spectral BRF image was simulated and compared with the AISA hyperspectral image. This simulation took around 50 minutes on a laptop with 16 GB memory and 8 cores. The simulation result is illustrated in **Figure 6.16**, which compares the simulated NIR image with AISA image. Overall, the simulated image is visually similar to the actual image. At fine scale, the distribution of tree crowns and shadows in LESS simulated image (**Figure 6.16**c) also matches well with the AISA image (**Figure 6.16**d), which indicates that the voxel-based scene reconstruction approach can be used to parameterize canopy structures with fine scale accuracy.

To quantitatively assess the accuracy of the simulated image, a pixel-wise comparison between LESS simulated image and the AISA image was performed under a resolution of 25 m. **Figure 6.17** illustrates the differences images and scatter plots in RED and NIR band. The determination coefficient (R^2) is 0.56 and 0.5 for RED and NIR band, respectively. In the NIR band, there is an apparent underestimated region (the red rectangle in **Figure 6.17**a), which leads

to a bias around 0.04. The improper assigned optical properties of the leaves can explain this underestimation. In the LESS simulation, we assigned the same optical properties (i.e., leaf reflectance and transmittance) to all the leaves in the study area, which obviously does not match the real forest, which usually has spatially varying optical properties. Specifically, the real leaf reflectance is higher than the given one in this small region, which can also be seen from AISA image (**Figure 6.16**a). The overestimation is mainly lying in the right side of the study area, where low canopy cover is observed. Thus, the ground has more contribution to the simulated BRF. The input ground reflectance may not match well with the real ground situation, which leads to the inconsistency. Please note that most of the pixels at the last row of the image have been overestimated, since lateral boundary effect are not considered in this simulation. For the RED band, the major underestimation occurs in a region with low canopy cover (green rectangle in **Figure 6.17**b). In the simulation, the voxels used to represent the grass have the sample optical properties with other voxels for tree leaves, which may be significantly different from the real grass reflectance. The overestimation mainly exists in two regions (R1 and R2 in **Figure 6.17**b). R1 is a region from a different flight strip, because the AISA image is merged from several flight strips. This region has a more oblique viewing geometry, which may produce lower reflectance in the AISA image. R2 is nearly a pure forest area which is mainly composed of larch trees. The input optical properties for the leaves in the simulation may differ from the real ones of larch tree, which causes the systematical overestimation in this region. Besides, there may be other factors that can also explain the differences. The atmospheric correction and orthorectification (due to the rugged terrain) of the AISA image cannot be perfect. The branches are not specially handled, since they are involved in the total PAI. Thus they are considered as leaves.

In conclusion, the difference between LESS-simulated BRFs and actual BRFs is mainly caused by the insufficient parametrization of the 3D scene (structural and optical). The objective of the above simulation is mainly to emphasize the capability of LESS to simulate large-scale forest scene with reasonable computational resources (e.g., a laptop with 16 GB memory) in a reasonable time (e.g., 30 minutes). The simulated datasets can be used to parameterize some inversion models or train artificial neural network (ANN) based models (Malenovský et al., 2013) to retrieve biophysical parameters. Compared to field measurements, it can provide the “ground truth” to validate a model in a more controllable and repeatable environment.

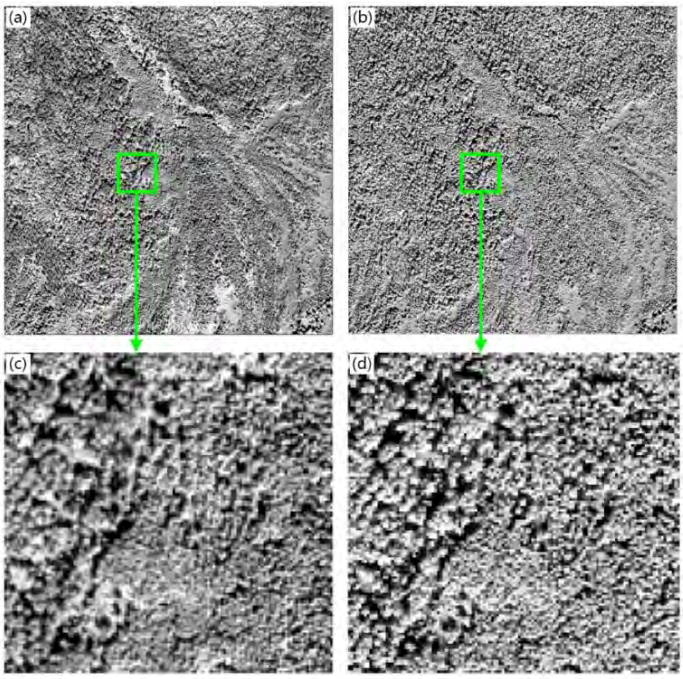


Figure 6.16: Comparison between LESS simulated image and the field-measured (AISA) hyperspectral image. (a) AISA image in NIR band; (b) LESS simulated image in NIR band; (c) Sub-region of AISA image; (d) Sub-region of simulated image.

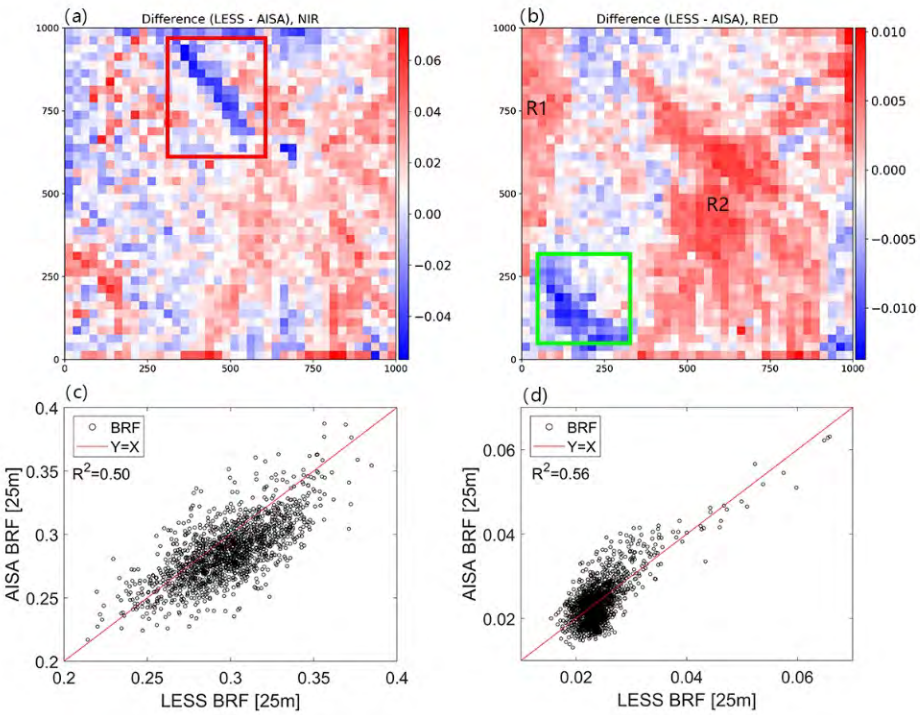


Figure 6.17: Differences between LESS simulated image and AISA image in a resolution of 25 m. (a) Difference image of NIR band; (b) Difference image of RED band; (c) Scatter plot of NIR BRFs; (d) Scatter plot of RED BRFs.

6.5 Concluding remarks

In this chapter, we proposed a voxel-based method to estimate the 3D distribution of PVD from relatively low density ALS data. The presented approach uses the nearest pure-ground pulse to calibrate the intensity of vegetation-ground pulse and then to calculate the incident energy at each echoes of a pulse. This eliminates the requirements of the priori information of ground reflectance and leaf surface reflectance. We use a ray-tracing algorithm to determine the transmittance of each voxel by reconstructing pulses from discrete point cloud. The results show that the 2D map of estimated PAI is comparable with other echo number based method under coarser resolutions. At fine scale, it is also inconsistency with TLS-derived PVD in terms of horizontal and vertical structures.

The proposed voxel-based method is a useful approach to estimate 3D structures of canopy from ALS data, which provides reliable parameters for 3D radiative transfer model. This chapter provides an example to link the 3D information extraction with the 3D radiative transfer model, which is seldom studies before, especially for complex forest canopies.

Chapter 7

Conclusions and perspectives

Summary

7.1 Major conclusions 118
7.2 Problems and perspectives 119

7.1 Major conclusions

The overall objective of this thesis is to contribute to two interconnected science issues: the development of 3D radiative transfer model and 3D parameterization of forest canopy.

1) A ray-tracing based 3D radiative transfer model LESS is proposed to simulate images and remote sensing data over large-scale and heterogeneous 3D canopies.

Current 3D physical radiative transfer models usually work on small realistic scene or abstract the canopy structures, which may cause biases for simulated data and make it difficult to validate physical models that is established with coarse resolution observations. To mitigate this problem, a ray-tracing based 3D radiative transfer model that takes full advantages of algorithms in computer graphics and acceleration techniques is proposed. According the characteristics of different simulation objectives, the LESS model uses different strategies, i.e., forward photon tracing to simulate BRF and energy-related data (e.g., FPAR) and backward path tracing to simulate images and atmosphere radiative transfer. Besides, a few improvements, such as on-the-fly computation of shaded components for thermal infrared radiation and virtual photon algorithm for simulating BRF, are also proposed in this thesis. In conclusion, the proposed LESS model is of great improvement compared to other models. Indeed, it can simulate various remote sensing products with considerable accuracy, and it has already been successfully validated with other models and field measurements through a series of comparison campaigns as described in chapter 2.

2) A hybrid acceleration scene structuring scheme is introduced into DART.

DART is 3D radiative transfer model that has been developed since 1992 and it has already become a mature model that can simulate various of remote sensing produces, e.g., LiDAR, thermal infared, atmosphere, fluorescence, etc. A major problem that faces DART is the uniform grid approach to organize landscape elements, which causes low efficiency for simulating complex and large-scale scene. Aimed at solving this problem, we have proposed a hybrid structuring scheme that removes empty grids from DART scene and organizes them with bounding volume hierarchy (BVH), which greatly improves computational efficiency and reduces memory usage, especially for high resolution voxels. Besides, the introduction of within-voxel BVH significantly accelerates DART simulations in presence of many triangles. Speedup can be as high as 200×. This acceleration has great importance for parameter sensitivity studies that usually require a large

number of simulations under different conditions. Without this improvement, the studies in chapter 4, which simulates LiDAR point cloud over scenes with a large number of triangles, would have been much more difficult to achieve.

3) Voxel-based inversion of leaf volume density and representation of forest canopies.

3D radiative transfer model usually needs high resolution 3D parameters to drive its simulations and retrieval of such parameters, e.g., LVD, from relatively low-density airborne LiDAR point cloud is an unsolved problem. This thesis first analyzes different LPI estimation approaches, which is the basis for retrieving LVD, by using DART simulated dataset. The study shows that intensity based method is the most appropriate one with no assumptions of the canopy structures through a nearest pure-ground pulse calibration mechanism. Taking the incident angle of LiDAR pulses into consideration, which has great influences on the inverted LVD, this thesis proposed a ray-tracing based method to estimate LPI and LVD for each voxel. The analysis also shows that the LPI is accurately estimated even with one pulse only, in contrast to dozens of penetrated pulses for point number based methods. The validation work stresses that the inversion approach presented in this thesis is consistent with other methods and TLS-derived LVD, i.e., it is possible to retrieve high resolution 3D LVD from relatively low-density LiDAR point cloud.

Besides, this thesis also demonstrates the possibility to represent forest canopies with 3D voxels. LESS simulated spectral images of the reconstructed voxel-based forest scene were compared to actual airborne hyperspectral images. This comparison stressed that LESS images are similar to actual images both in terms of spatial pattern and absolute BRF values.

7.2 Problems and perspectives

3D canopy radiative transfer modeling is a challenging task. This thesis brings solutions that mitigate major problems. However, a number of improvements are still necessary.

1) The proposed model LESS only implements the simulations of atmosphere radiative transfer and thermal inferred radiation in backward path tracing mode, instead of forward photon tracing mode. The major difficulty here is the intensive computational costs and high variance during the photon sampling, especially when a larger number of triangles are present. Thus, simulating landscape radiative budget with thermal infrared and atmosphere is difficult for the current LESS model. A solution for this problem is to use backward path tracing and compute incident radiation for each surface in the scene (Bailey and Ochoa, 2018). Another possibility is to

use GPU (Graphics Processing Unit) acceleration, which has already been paid much attention in scientific computation community (Malik et al., 2017).

2) When validating the LESS model with field measurements, there are a lot of uncertainties due to complexity of forest canopies, whose structures are usually difficult to obtain explicitly, such as the position of each leaf, even with the most precise TLS instrument. Another uncertainty is due to the spatially varying optical properties, which greatly influences the simulated remote sensing data and makes the simulated data different from actual field measurements.

3) The estimation of LVD from airborne point cloud is improved by still not perfect. Although the large gaps between crowns have been removed during the inversion through the voxel-based approach, the clumping of leaves is still difficult to consider when the low-density airborne point cloud. Besides, a pulse may partially hit a voxel and this information is too complex to be considered. All these problems need further studies either by more advanced problem or by more power data acquisition instruments.

Bibliography

Abdalati, W., Zwally, H.J., Bindschadler, R., Csatho, B., Farrell, S.L., Fricker, H.A., Harding, D., Kwok, R., Lefsky, M., Markus, T., others, 2010. The ICESat-2 laser altimetry mission. *Proceedings of the IEEE* 98, 735–751.

Ackermann, F., 1999. Airborne laser scanning—present status and future expectations. *ISPRS Journal of Photogrammetry and Remote Sensing* 54, 64–67.

Akenine-Möller, T., 2005. Fast 3D triangle-box overlap testing, in: Acm Siggraph 2005 Courses. ACM, p. 8.

Amanatides, J., Woo, A., others, 1987. A fast voxel traversal algorithm for ray tracing, in: Eurographics. pp. 3–10.

Anderson, K., Hancock, S., Disney, M., Gaston, K.J., 2016. Is waveform worth it? A comparison of LiDAR approaches for vegetation and landscape characterization. *Remote Sensing in Ecology and Conservation* 2, 5–15.

Antyufeev, V.S., Marshak, A.L., 1990. Monte Carlo method and transport equation in plant canopies. *Remote Sensing of Environment* 31, 183–191.

Arnold, M., Powell, B., Shanley, P., Sunderland, T.C.H., 2011. EDITORIAL: Forests, Biodiversity and Food Security. *ifre* 13, 259–265.

Asner, G.P., Knapp, D.E., Boardman, J., Green, R.O., Kennedy-Bowdoin, T., Eastwood, M., Martin, R.E., Anderson, C., Field, C.B., 2012. Carnegie Airborne Observatory-2: Increasing science data dimensionality via high-fidelity multi-sensor fusion. *Remote Sensing of Environment* 124, 454–465.

ASPRS, L., 2009. Specification. Version 1.3–R10. *Photogrammetric Engineering and Remote Sensing* 75, 1035–1042.

Auer, S., Bamler, R., Reinartz, P., 2016. RaySAR - 3D SAR simulator: Now open source, in: 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Presented at the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 6730–6733.

Bailey, B.N., 2018. A reverse ray-tracing method for modelling the net radiative flux in leaf-resolving plant canopy simulations. *Ecological Modelling* 368, 233–245.

Bailey, B.N., Mahaffee, W.F., 2017. Rapid measurement of the three-dimensional distribution of leaf orientation and the leaf angle probability density function using terrestrial LiDAR scanning. *Remote Sensing of Environment* 194, 63–76.

Bailey, B.N., Ochoa, M.H., 2018. Semi-direct tree reconstruction using terrestrial LiDAR point cloud data. *Remote Sensing of Environment* 208, 133–144.

Bailey, B.N., Overby, M., Willemsen, P., Pardyjak, E.R., Mahaffee, W.F., Stoll, R., 2014. A scalable plant-resolving radiative transfer model based on optimized GPU ray tracing. *Agricultural and Forest Meteorology* 198–199, 192–208.

Bailey, B.N., Stoll, R., Pardyjak, E.R., Miller, N.E., 2016. A new three-dimensional energy balance model for complex plant canopy geometries: Model development and improved validation strategies. *Agricultural and Forest Meteorology* 218–219, 146–160.

Banskota, A., Serbin, S.P., Wynne, R.H., Thomas, V.A., Falkowski, M.J., Kayastha, N., Gastellu-Etchegorry, J.P., Townsend, P.A., 2015. An LUT-Based Inversion of DART Model to Estimate Forest LAI from Hyperspectral Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8, 3147–3160.

Bass, S., Hughes, C., Hawthorne, W., 2001. Forests, biodiversity and livelihoods: linking policy and practice. *Living off biodiversity: exploring livelihoods and biodiversity issues in natural resources management* 23–74.

Beckers, B., Beckers, P., 2012. A general rule for disk and hemisphere partition into equal-area cells. *Computational Geometry* 45, 275–283.

Béland, M., Widlowski, J.-L., Fournier, R.A., Côté, J.-F., Verstraete, M.M., 2011. Estimating leaf area distribution in savanna trees from terrestrial LiDAR measurements. *Agricultural and Forest Meteorology* 151, 1252–1266.

Ben-Arie, J.R., Hay, G.J., Powers, R.P., Castilla, G., St-Onge, B., 2009. Development of a pit filling algorithm for LiDAR canopy height models. *Computers & Geosciences* 35, 1940–1949.

Bittner, S., Gayler, S., Biernath, C., Winkler, J.B., Seifert, S., Pretzsch, H., Priesack, E., 2012. Evaluation of a ray-tracing canopy light model based on terrestrial laser scans. *Canadian Journal of Remote Sensing* 38, 619–628.

Borel, C.C., Gerstl, S.A.W., Borel, C.C., Gerstl, S.A.W., 1994. Nonlinear spectral mixing models for vegetative and soil surfaces. *Remote Sensing of Environment* 47, 403–416.

Bremer, M., Wichmann, V., Rutzinger, M., 2017. Calibration and Validation of a Detailed Architectural Canopy Model Reconstruction for the Simulation of Synthetic Hemispherical Images and Airborne LiDAR Data. *Remote Sensing* 9, 220.

Calders, K., Origo, N., Burt, A., Disney, M., Nightingale, J., Raumonen, P., Åkerblom, M., Malhi, Y., Lewis, P., 2018. Realistic Forest Stand Reconstruction from Terrestrial LiDAR for Radiative Transfer Modelling. *Remote Sensing* 10, 933.

Chen, J. M., Blanken, P.D., Black, T.A., Guilbeault, M., Chen, S., 1997. Radiation regime and canopy architecture in a boreal aspen forest. *Agricultural and Forest Meteorology* 86, 107–125.

Chen, Jing M, Rich, P.M., Gower, S.T., Norman, J.M., Plummer, S., 1997. Leaf area index of boreal forests: Theory, techniques, and measurements. *Journal of Geophysical Research: Atmospheres* 102, 29429–29443.

Cleary, J.G., Wyvill, G., 1988. Analysis of an algorithm for fast ray tracing using uniform space subdivision. *The Visual Computer* 4, 65–83.

Cook, B., Nelson, R., Middleton, E., Morton, D., McCorkel, J., Masek, J., Ranson, K., Ly, V.,

Montesano, P., others, 2013. NASA Goddard's LiDAR, hyperspectral and thermal (G-LiHT) airborne imager. *Remote Sensing* 5, 4045–4066.

Côté, J.-F., Widlowski, J.-L., Fournier, R.A., Verstraete, M.M., 2009. The structural and radiative consistency of three-dimensional tree reconstructions from terrestrial lidar. *Remote Sensing of Environment* 113, 1067–1081.

Danson, F.M., Hetherington, D., Morsdorf, F., Koetz, B., Allgower, B., 2007. Forest Canopy Gap Fraction From Terrestrial Laser Scanning. *IEEE Geoscience & Remote Sensing Letters* 4, 157–160.

Disney, M.I., Lewis, P., North, P., 2000. Monte Carlo ray tracing in optical canopy reflectance modelling. *Remote Sensing Reviews* 18, 163–196.

Disney, M.I., Lewis, P.E., Bouvet, M., Prieto-Blanco, A., Hancock, S., 2009. Quantifying surface reflectivity for spaceborne lidar via two independent methods. *IEEE transactions on geoscience and remote sensing* 47, 3262–3271.

Dozier, J., Frew, J., 1990. Rapid calculation of terrain parameters for radiation modeling from digital elevation data. *IEEE Transactions on Geoscience and Remote Sensing* 28, 963–969.

Dubayah, R.O., Drake, J.B., 2000. Lidar Remote Sensing for Forestry. *j for* 98, 44–46.

Gastellu-Etchegorry, J.-P., Demarez, V., Pinel, V., Zagolski, F., 1996. Modeling radiative transfer in heterogeneous 3-D vegetation canopies. *Remote sensing of environment* 58, 131–156.

Gastellu-Etchegorry, J.P., Martin, E., Gascon, F., 2004. DART: a 3D model for simulating satellite images and studying surface radiation budget. *International Journal of Remote Sensing* 25, 73–96.

Gastellu-Etchegorry, J.-P., Yin, T., Lauret, N., Cajgfinger, T., Gregoire, T., Grau, E., Feret, J.-B., Lopes, M., Guilleux, J., Dedieu, G., Malenovský, Z., Cook, B.D., Morton, D., Rubio, J., Durrieu, S., Cazanave, G., Martin, E., Ristorcelli, T., 2015. Discrete Anisotropic Radiative Transfer (DART 5) for Modeling Airborne and Satellite Spectroradiometer and LIDAR Acquisitions of Natural and Urban Landscapes. *Remote Sensing* 7, 1667–1701.

Gastellu-Etchegorry, J.-P., Yin, T., Lauret, N., Grau, E., Rubio, J., Cook, B.D., Morton, D.C., Sun, G., 2016. Simulation of satellite, airborne and terrestrial LiDAR with DART (I): Waveform simulation with quasi-Monte Carlo ray tracing. *Remote Sensing of Environment* 184, 418–435.

Girardeau-Montaut, D., 2011. CloudCompare-Open Source project. *OpenSource Project*.

Glassner, A.S., 1984. Space subdivision for fast ray tracing. *IEEE Computer Graphics and applications* 4, 15–24.

Goel, N.S., Rozehnal, I., Thompson, R.L., 1991. A computer graphics based model for scattering from objects of arbitrary shapes in the optical region. *Remote Sensing of Environment* 36, 73–104.

Goodenough, A.A., Brown, S.D., 2017. DIRSIG5: Next-Generation Remote Sensing Data and Image Simulation Framework. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10, 4818–4833.

Goodenough, A.A., Brown, S.D., 2012. DIRSIG 5: Core design and implementation, in: SPIE Defense, Security, and Sensing. International Society for Optics and Photonics, pp. 83900H–

83900H.

Govaerts, Y.M., 1996. A model of light scattering in three-dimensional plant canopies: a Monte Carlo ray tracing approach. Office for Official Publ. of the Europ. Communities.

Govaerts, Y.M., Verstraete, M.M., 1998. Raytran: A Monte Carlo ray-tracing model to compute light scattering in three-dimensional heterogeneous media. *Geoscience and Remote Sensing, IEEE Transactions on* 36, 493–505.

Grau, E., Durrieu, S., Fournier, R., Gastellu-Etchegorry, J.-P., Yin, T., 2017. Estimation of 3D vegetation density with Terrestrial Laser Scanning data using voxels. A sensitivity analysis of influencing parameters. *Remote Sensing of Environment* 191, 373–388.

Grau, E., Gastellu-Etchegorry, J.-P., 2013. Radiative transfer modeling in the Earth–Atmosphere system with DART model. *Remote Sensing of Environment* 139, 149–170.

Hancock, S., Anderson, K., Disney, M., Gaston, K.J., 2017. Measurement of fine-spatial-resolution 3D vegetation structure with airborne waveform lidar: Calibration and validation with voxelised terrestrial lidar. *Remote Sensing of Environment* 188, 37–50.

Henyey, L.G., Greenstein, J.L., 1941. Diffuse radiation in the galaxy. *The Astrophysical Journal* 93, 70–83.

Hmida, S.B., Kallel, A., Gastellu-Etchegorry, J.-P., Roujean, J.-L., 2017. Crop Biophysical Properties Estimation Based on LiDAR Full-Waveform Inversion Using the DART RTM. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10, 4853–4868.

Hosoi, F., Omasa, K., 2006. Voxel-Based 3-D Modeling of Individual Trees for Estimating Leaf Area Density Using High-Resolution Portable Scanning Lidar. *IEEE Transactions on Geoscience and Remote Sensing* 44, 3610–3618.

Hu, R., Yan, G., Nerry, F., Liu, Y., Jiang, Y., Wang, S., Chen, Y., Mu, X., Zhang, W., Xie, D., 2018. Using Airborne Laser Scanner and Path Length Distribution Model to Quantify Clumping Effect and Estimate Leaf Area Index. *IEEE Transactions on Geoscience and Remote Sensing* 56, 3196–3209.

Huang, H., Qin, W., Liu, Q., 2013. RAPID: A Radiosity Applicable to Porous IndiviDual Objects for directional reflectance over complex vegetated scenes. *Remote Sensing of Environment* 132, 221–237.

Hughes, J.F., Foley, J.D., 2014. Computer graphics: principles and practice. Pearson Education.

Hyde, P., Dubayah, R., Walker, W., Blair, J.B., Hofton, M., Hunsaker, C., 2006. Mapping forest structure for wildlife habitat analysis using multi-sensor (LiDAR, SAR/InSAR, ETM+, Quickbird) synergy. *Remote Sensing of Environment* 102, 63–73.

Jacquemoud, S., Baret, F., Hanocq, J.F., 1992. Modeling spectral and bidirectional soil reflectance. *Remote Sensing of Environment* 41, 123–132.

Jakob, W., 2010. Mitsuba renderer. Mitsuba.

Kajiya, J.T., 1986. The rendering equation, in: ACM Siggraph Computer Graphics. ACM, pp. 143–150.

Kobayashi, H., Iwabuchi, H., 2008. A coupled 1-D atmosphere and 3-D canopy radiative transfer model for canopy reflectance, light environment, and photosynthesis simulation in a

heterogeneous landscape. *Remote Sensing of Environment* 112, 173–185.

Koetz, B., Morsdorf, F., Sun, G., Ranson, K.J., Itten, K., Allgower, B., 2006. Inversion of a lidar waveform model for forest biophysical parameter estimation. *IEEE Geoscience and Remote Sensing Letters* 3, 49–53.

Korhonen, L., Korpela, I., Heiskanen, J., Maltamo, M., 2011. Airborne discrete-return LIDAR data in the estimation of vertical canopy cover, angular canopy closure and leaf area index. *Remote Sensing of Environment* 115, 1065–1080.

Kükenbrink, D., Schneider, F.D., Leiterer, R., Schaepman, M.E., Morsdorf, F., 2017. Quantification of hidden canopy volume of airborne laser scanning data using a voxel traversal algorithm. *Remote Sensing of Environment* 194, 424–436.

Kuusk, A., 2018. Canopy Radiative Transfer Modeling, in: Comprehensive Remote Sensing. Elsevier, pp. 9–22.

Landier, L., Lauret, N., Yin, T., Al Bitar, A., Gastellu-Etchegorry, J., Feigenwinter, C., Parlow, E., Mitraka, Z., Chrysoulakis, N., 2016. Remote Sensing Studies of Urban Canopies: 3D Radiative Transfer Modeling, in: Sustainable Urbanization. InTech.

Lauterbach, C., Garland, M., Sengupta, S., Luebke, D., Manocha, D., 2009. Fast BVH construction on GPUs, in: Computer Graphics Forum. Wiley Online Library, pp. 375–384.

Leblanc, S.G., Bicheron, P., Chen, J.M., Leroy, M., Cihlar, J., 1999. Investigation of directional reflectance in boreal forests with an improved four-scale model and airborne POLDER data. *IEEE Transactions on Geoscience and Remote Sensing* 37, 1396–1414.

Lefsky, M.A., Cohen, W., Acker, S., Parker, G.G., Spies, T., Harding, D., 1999. Lidar remote sensing of the canopy structure and biophysical properties of Douglas-fir western hemlock forests. *Remote sensing of environment* 70, 339–361.

Lefsky, M.A., Cohen, W.B., Parker, G.G., Harding, D.J., 2002. Lidar remote sensing for ecosystem studies: Lidar, an emerging remote sensing technology that directly measures the three-dimensional distribution of plant canopies, can accurately estimate vegetation structural attributes and should be of particular interest to forest, landscape, and global ecologists. *AIBS Bulletin* 52, 19–30.

Lewis, P., 1999. Three-dimensional plant modelling for remote sensing simulation studies using the Botanical Plant Modelling System. *Agronomie* 19, 185–210.

Li, X., Cheng, G., Chen, X., Lu, L., 1999. Modification of solar radiation model over rugged terrain. *Chinese Science Bulletin* 44, 1345–1349.

Liang, S., Cheng, J., Jia, K., Jiang, B., Liu, Q., Liu, S., Xiao, Z., Xie, X., Yao, Y., Yuan, W., others, 2016. Recent progress in land surface quantitative remote sensing. *J. Remote Sens* 20, 875–898.

Lin, Y., West, G., 2016. Retrieval of effective leaf area index (LAIe) and leaf area density (LAD) profile at individual tree level using high density multi-return airborne LiDAR. *International Journal of Applied Earth Observation and Geoinformation* 50, 150–158.

Liu, J., Skidmore, A.K., Jones, S., Wang, T., Heurich, M., Zhu, X., Shi, Y., 2018. Large off-nadir scan angle of airborne LiDAR can severely affect the estimates of forest structure metrics. *ISPRS Journal of Photogrammetry and Remote Sensing* 136, 13–25.

Lovell, J., Jupp, D.L., Culvenor, D., Coops, N., 2003. Using airborne and ground-based ranging lidar to measure canopy structure in Australian forests. *Canadian Journal of Remote Sensing* 29, 607–622.

Malenovský, Z., Homolová, L., Zurita-Milla, R., Lukeš, P., Kaplan, V., Hanuš, J., Gastellu-Etchegorry, J.-P., Schaepman, M.E., 2013. Retrieval of spruce leaf chlorophyll content from airborne image data using continuum removal and radiative transfer. *Remote Sensing of Environment* 131, 85–102.

Malik, M., Grosheintz, L., Mendonça, J.M., Grimm, S.L., Lavie, B., Kitzmann, D., Tsai, S.-M., Burrows, A., Kreidberg, L., Bedell, M., others, 2017. HELIOS: An Open-source, GPU-accelerated Radiative Transfer Code For Self-consistent Exoplanetary Atmospheres. *The Astronomical Journal* 153, 56.

Mallet, C., Bretar, F., 2009. Full-waveform topographic lidar: State-of-the-art. *ISPRS Journal of photogrammetry and remote sensing* 64, 1–16.

Marchuk, G.I., Mikhailov, G.A., Nazareliev, M.A., Darbinjan, R.A., Kargin, B.A., Elepov, B.S., 1980. The Monte Carlo Methods in Atmospheric Optics, Springer Series in Optical Sciences. Springer-Verlag, Berlin Heidelberg.

McNeill, M., Shah, B., Hebert, M.-P., Lister, P.F., Grimsdale, R.L., 1992. Performance of space subdivision techniques in ray tracing, in: Computer Graphics Forum. Wiley Online Library, pp. 213–220.

Milenković, M., Wagner, W., Quast, R., Hollaus, M., Ressl, C., Pfeifer, N., 2017. Total canopy transmittance estimated from small-footprint, full-waveform airborne LiDAR. *ISPRS Journal of Photogrammetry and Remote Sensing* 128, 61–72.

Mitchard, E.T.A., 2018. The tropical forest carbon cycle and climate change. *Nature* 559, 527.

Mller, G., Fellner, D., 2000. Hybrid Scene Structuring with Application to Ray Tracing.

Morlet, J., 1983. Sampling theory and wave propagation, in: Issues in Acoustic Signal—Image Processing and Recognition. Springer, pp. 233–261.

Morsdorf, F., Kötz, B., Meier, E., Itten, K.I., Allgöwer, B., 2006. Estimation of LAI and fractional cover from small footprint airborne laser scanning data based on gap fraction. *Remote Sensing of Environment* 104, 50–61.

Müller-Linow, M., Pinto-Espinosa, F., Scharr, H., Rascher, U., 2015. The leaf angle distribution of natural plant populations: assessing the canopy with a novel software tool. *Plant Methods* 11, 11.

Myneni, R.B., Ramakrishna, R., Nemani, R., Running, S.W., 1997. Estimation of global leaf area index and absorbed PAR using radiative transfer models. *IEEE Transactions on Geoscience and remote sensing* 35, 1380–1393.

Myneni, R.B., Ross, J., 2012. Photon-Vegetation interactions: applications in optical remote sensing and plant ecology. Springer Science & Business Media.

Newville, M., Stensitzki, T., Allen, D.B., Rawlik, M., Ingargiola, A., Nelson, A., 2016. LMFIT: non-linear least-square minimization and curve-fitting for Python. *Astrophysics Source Code Library*.

North, P., Rosette, J., Suárez, J., Los, S., 2010. A Monte Carlo radiative transfer model of satellite

waveform LiDAR. *International Journal of Remote Sensing* 31, 1343–1358.

North, P.R.J., 1996. Three-dimensional forest light interaction model using a Monte Carlo method. *IEEE Transactions on Geoscience and Remote Sensing* 34, 946–956.

Pfennigbauer, M., Ullrich, A., 2010. Improving quality of laser scanning data acquisition through calibrated amplitude and pulse deviation measurement, in: Laser Radar Technology and Applications XV. International Society for Optics and Photonics, p. 76841F.

Pfennigbauer, M., Wolf, C., Ullrich, A., 2013. Enhancing online waveform processing by adding new point attributes, in: Turner, M.D., Kamerman, G.W. (Eds.), . p. 873104.

Pharr, M., Jakob, W., Humphreys, G., 2016. Physically based rendering: From theory to implementation. Morgan Kaufmann.

Plachetka, T., 1998. POV Ray: persistence of vision parallel raytracer, in: Proc. of Spring Conf. on Computer Graphics, Budmerice, Slovakia. pp. 123–129.

Purcell, T.J., Hanrahan, P.M., 2004. Ray tracing on a stream processor (PhD Thesis). Citeseer.

Qi, J., Xie, D., Guo, D., Yan, G., 2017. A Large-Scale Emulation System for Realistic Three-Dimensional (3-D) Forest Simulation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10, 4834–4843.

Qi, Jianbo, Xie, D., Guo, D., Yan, G., 2017. A Large-Scale Emulation System for Realistic Three-Dimensional (3-D) Forest Simulation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Qi, J., Xie, D., Yan, G., 2016. Realistic 3D-simulation of large-scale forest scene based on individual tree detection. IEEE, pp. 728–731.

Qi, J., Xie, D., Yin, T., Yan, G., Gastellu-Etchegorry, J.-P., Li, L., Zhang, W., Mu, X., Norford, L.K., 2019. LESS: LargE-Scale remote sensing data and image simulation framework over heterogeneous 3D scenes. *Remote Sensing of Environment* 221, 695–706.

Qin, W., Gerstl, S.A., 2000. 3-D scene modeling of semidesert vegetation cover and its radiation regime. *Remote Sensing of Environment* 74, 145–162.

Reitberger, J., Krzystek, P., Stilla, U., 2008. Analysis of full waveform LIDAR data for the classification of deciduous and coniferous trees. *International journal of remote sensing* 29, 1407–1431.

Reitberger, J., Schnörr, Cl., Krzystek, P., Stilla, U., 2009. 3D segmentation of single trees exploiting full waveform LIDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing* 64, 561–574.

Schaepman-Strub, G., Schaepman, M.E., Painter, T.H., Dangel, S., Martonchik, J.V., 2006. Reflectance quantities in optical remote sensing—definitions and case studies. *Remote Sensing of Environment* 103, 27–42.

Schlamadinger, B., Marland, G., 1996. The role of forest and bioenergy strategies in the global carbon cycle. *Biomass and Bioenergy* 10, 275–300.

Schneider, F.D., Leiterer, R., Morsdorf, F., Gastellu-Etchegorry, J.-P., Lauret, N., Pfeifer, N., Schaepman, M.E., 2014. Simulating imaging spectrometer data: 3D forest modeling based on LiDAR and in situ data. *Remote Sensing of Environment* 152, 235–250.

Sehgal, V.K., Chakraborty, D., Sahoo, R.N., 2016. Inversion of radiative transfer model for

retrieval of wheat biophysical parameters from broadband reflectance measurements. *Information Processing in Agriculture* 3, 107–118.

She-Zhou, L., Cheng, W., Gui-Bin, Z., Xiao-Huan, X., Gui-Cai, L., 2013. Forest Leaf Area Index (LAI) Estimation Using Airborne Discrete-Return Lidar Data. *Chinese Journal of Geophysics* 56, 233–242.

Solberg, S., 2010. Mapping gap fraction, LAI and defoliation using various ALS penetration variables. *International Journal of Remote Sensing* 31, 1227–1244.

Solberg, S., Brunner, A., Hanssen, K.H., Lange, H., Næsset, E., Rautiainen, M., Stenberg, P., 2009. Mapping LAI in a Norway spruce forest using airborne laser scanning. *Remote Sensing of Environment* 113, 2317–2327.

Solberg, S., Næsset, E., Hanssen, K.H., Christiansen, E., 2006. Mapping defoliation during a severe insect attack on Scots pine using airborne laser scanning. *Remote Sensing of Environment* 102, 364–376.

Song, Y., Maki, M., Imanishi, J., Morimoto, Y., 2011. Voxel-based estimation of plant area density from airborne laser scanner data, in: Proceedings of the ISPRS Workshop Laser Scanning, Calgary, Canada. p. W12.

Sun, G., Ranson, K.J., 2000. Modeling lidar returns from forest canopies. *IEEE Transactions on Geoscience and Remote Sensing* 38, 2617–2626.

Sung, K., Shirley, P., 1992. Ray tracing with the BSP tree, in: Graphics Gems III. Academic Press Professional, Inc., pp. 271–274.

Tevs, A., Ihrke, I., Seidel, H.-P., 2008. Maximum mipmaps for fast, accurate, and scalable dynamic height field rendering, in: Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games. ACM, pp. 183–190.

Thompson, R.L., Goel, N.S., 1998. Two models for rapidly calculating bidirectional reflectance of complex vegetation scenes: Photon spread (PS) model and statistical photon spread (SPS) model. *Remote Sensing Reviews* 16, 157–207.

Twery, M.J., Weiskittel, A.R., 2013. Forest-Management Modelling. *Environmental Modelling: Finding Simplicity in Complexity, Second Edition* 379–398.

Ullrich, A., Pfennigbauer, M., 2011. Categorisation of full waveform data provided by laser scanning devices, in: Kamerman, G.W., Steinvall, O., Bishop, G.J., Gonglewski, J.D., Lewis, K.L., Hollins, R.C., Merlet, T.J. (Eds.), . p. 818609.

Veach, E., 1997. Robust monte carlo methods for light transport simulation. Stanford University PhD thesis.

Verhoef, W., 1984. Light scattering by leaf layers with application to canopy reflectance modeling: the SAIL model. *Remote sensing of environment* 16, 125–141.

Verhoef, W., Jia, L., Su, Z., 2007. Optical-thermal canopy radiance directionality modelling by unified 4SAIL model.

Wagner, W., 2010. Radiometric calibration of small-footprint full-waveform airborne laser scanner measurements: Basic physical concepts. *ISPRS Journal of Photogrammetry and Remote Sensing, ISPRS Centenary Celebration Issue* 65, 505–513.

Wagner, W., Ullrich, A., Ducic, V., Melzer, T., Studnicka, N., 2006. Gaussian decomposition and

calibration of a novel small-footprint full-waveform digitising airborne laser scanner. *ISPRS Journal of Photogrammetry and Remote Sensing* 60, 100–112.

Wagner, W., Ullrich, A., Melzer, T., Briese, C., Kraus, K., 2004. From single-pulse to full-waveform airborne laser scanners: potential and practical challenges.

Wald, I., Woop, S., Benthin, C., Johnson, G.S., Ernst, M., 2014. Embree: A Kernel Framework for Efficient CPU Ray Tracing. *ACM Trans. Graph.* 33, 143:1–143:8.

Wang, K., Zhou, X., Liu, J., Sparrow, M., 2005. Estimating surface solar radiation over complex terrain using moderate-resolution satellite sensor data. *International Journal of Remote Sensing* 26, 47–58.

Wang, Z., Zhang, L., Fang, T., Mathiopoulos, P.T., Qu, H., Chen, D., Wang, Y., 2014. A Structure-Aware Global Optimization Method for Reconstructing 3-D Tree Models From Terrestrial Laser Scanning Data. *IEEE T. Geoscience and Remote Sensing* 52, 5653–5669.

Wen, J., Liu, Qiang, Xiao, Q., Liu, Qinhuo, You, D., Hao, D., Wu, S., Lin, X., 2018. Characterizing Land Surface Anisotropic Reflectance over Rugged Terrain: A Review of Concepts and Recent Developments. *Remote Sensing* 10, 370.

Widlowski, J.-L., Lavergne, T., Pinty, B., Verstraete, M., Gobron, N., 2006. Rayspread: A virtual laboratory for rapid BRF simulations over 3-D plant canopies. *Computational methods in transport* 211–231.

Widlowski, J.-L., Pinty, B., Lopatka, M., Atzberger, C., Buzica, D., Chelle, M., Disney, M., Gastellu-Etchegorry, J.-P., Gerboles, M., Gobron, N., Grau, E., Huang, H., Kallel, A., Kobayashi, H., Lewis, P.E., Qin, W., Schlerf, M., Stuckens, J., Xie, D., 2013. The fourth radiation transfer model intercomparison (RAMI-IV): Proficiency testing of canopy reflectance models with ISO-13528. *J. Geophys. Res. Atmos.* 118, 6869–6890.

Williams, A., Barrus, S., Morley, R.K., Shirley, P., 2005. An efficient and robust ray-box intersection algorithm, in: ACM SIGGRAPH 2005 Courses. ACM, p. 9.

Winjum, J.K., Dixon, R.K., Schroeder, P.E., 1992. Estimating the Global Potential of Forest and Agroforest Management Practices to Sequester Carbon, in: Wisniewski, J., Lugo, A.E. (Eds.), Natural Sinks of CO2. Springer Netherlands, pp. 213–227.

Xie, D., Wang, P., Liu, R., Zhu, Q., 2010. Research on PAR and FPAR of crop canopies based on RGM, in: 2010 IEEE International Geoscience and Remote Sensing Symposium. Presented at the 2010 IEEE International Geoscience and Remote Sensing Symposium, pp. 1493–1496.

Xie, Y., Sha, Z., Yu, M., 2008. Remote sensing imagery in vegetation mapping: a review. *J Plant Ecol* 1, 9–23.

Yan, G., Wang, T., Jiao, Z., Mu, X., Jing, Z., Ling, C., 2016. Topographic radiation modeling and spatial scaling of clear-sky land surface longwave radiation over rugged terrain. *Remote Sensing of Environment* 172, 15–27.

Yang, X., Miller, D.R., Montgomery, M.E., 1993. Vertical distributions of canopy foliage and biologically active radiation in a defoliated/refoliated hardwood forest. *Agricultural and Forest Meteorology* 67, 129–146.

Yin, T., Gastellu-Etchegorry, J.-P., Lauret, N., Grau, E., Rubio, J., 2013. A new approach of direction discretization and oversampling for 3D anisotropic radiative transfer modeling.

Remote Sensing of Environment 135, 213–223.

Yin, T., Lauret, N., Gastellu-Etchegorry, J.-P., 2016. Simulation of satellite, airborne and terrestrial LiDAR with DART (II): ALS and TLS multi-pulse acquisitions, photon counting, and solar noise. *Remote Sensing of Environment* 184, 454–468.

Yin, T., Lauret, N., Gastellu-Etchegorry, J.-P., 2015. Simulating images of passive sensors with finite field of view by coupling 3-D radiative transfer model and sensor perspective projection. *Remote Sensing of Environment* 162, 169–185.

Zhang, W., Qi, J., Wan, P., Wang, H., Xie, D., Wang, X., Yan, G., 2016. An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. *Remote Sensing* 8, 501.

Zhang, X., Du, P., He, Y., Fang, H., 2017. Review of Research and Application for Vegetation BRDF. *Guang Pu Xue Yu Guang Pu Fen Xi* 37, 829–835.

Zhao, F., Li, Y., Dai, X., Verhoef, W., Guo, Y., Shang, H., Gu, X., Huang, Y., Yu, T., Huang, J., 2015. Simulated impact of sensor field of view and distance on field measurements of bidirectional reflectance factors for row crops. *Remote Sensing of Environment* 156, 129–142.

Appendix

Appendix A: Importance Sampling

For a random variable $Y = f(X)$, which has a probability density function $g(x)$, the expectation is $E(Y) = \int f(x)g(x)dx$. In finite case, i.e. there are finite (k) outcomes of Y, the expected value can be calculated as

$$E(Y) = \sum_{i=1}^{k} f(x_i)g(x_i) \tag{1}$$

The probability $g(x_i)$ for each outcome can be approximated by counting the frequency of that outcome, which is $g(x_i) = \frac{n_i}{N}$, where N is the total number of outcomes and n_i is the number of outcome Y_i. Thus,

$$E(Y) = \frac{1}{N}\sum_{i=1}^{k} n_i f(x_i) \tag{2}$$

The above equation means, if we generate N samples from the distribution $g(x)$, the expected value can be estimated by $E(Y) = \frac{1}{N}\sum_{i=1}^{N} f(x_i)$. The importance here is that the samples must be generated from the density function $g(x)$.

This can be used to estimate the integral of a function $I = \int f(x)dx$. If we uniformly generate samples, the integral can be estimated by $I = \frac{1}{N} f(x_i)$. However, in most cases, this will produce large variance. For example, the function $f(x)$ can be zero in many places and with very high value in some regions. Alternatively, we can rewrite the integral estimator as

$$I = \int f(x)dx = \int \frac{f(x)}{p(x)} p(x)dx \tag{3}$$

We can generate samples from distribution function $p(x)$, instead of uniformly distribution,

to get $I = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)}$. The variance of this estimator depends on the density function $p(x)$. If the density function is chosen intelligently, i.e., proportional to $f(x)$, the variance can be reduced significantly. This is called *importance sampling*. $p(x)$ is called *importance density*. $w_i = \frac{f(x_i)}{p(x_i)}$ is called *importance weight*.